Package ‘ChIPseeker’

April 12, 2022

Type Package

Title ChIPseeker for ChIP peak Annotation, Comparison, and Visualization

Version 1.30.3

Maintainer Guangchuang Yu <`guangchuangyu@gmail.com`>

Description This package implements functions to retrieve the nearest genes around the peak, annotate genomic region of the peak, statstical methods for estimate the significance of overlap among ChIP peak data sets, and incorporate GEO database for user to compare the own dataset with those deposited in database. The comparison can be used to infer cooperative regulation and thus can be used to generate hypotheses. Several visualization functions are implemented to summarize the coverage of the peak experiment, average profile and heatmap of peaks binding to TSS regions, genomic annotation, distance to TSS, and overlap of peaks or genes.

Depends R (>= 3.5.0)

Imports AnnotationDbi, BiocGenerics, boot, enrichplot, IRanges, GenomeInfoDb, GenomicRanges, GenomicFeatures, ggplot2, gplots, graphics, grDevices, gtools, methods, plotrix, dplyr, parallel, magrittr, RColorBrewer, rtracklayer, S4Vectors, stats, TxDb.Hsapiens.UCSC.hg19.knownGene, utils

Suggests clusterProfiler, ggimage, ggplotify, ggupset, ReactomePA, org.Hs.eg.db, knitr, rmarkdown, testthat, tibble

Remotes GuangchuangYu/enrichplot

URL `https://guangchuangyu.github.io/software/ChIPseeker`

BugReports `https://github.com/YuLab-SMU/ChIPseeker/issues`

Encoding UTF-8

VignetteBuilder knitr

ByteCompile true

License Artistic-2.0

biocViews Annotation, ChIPSeq, Software, Visualization, MultipleComparison

RoxygenNote 7.1.2

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

git_branch RELEASE_3_14

git_last_commit cce74f0

git_last_commit_date 2021-12-15

Date/Publication 2022-04-12

Author Guangchuang Yu [aut, cre] (<https://orcid.org/0000-0002-6485-8781>),
Ming Li [ctb],
Yun Yan [ctb],
Hervé Pagès [ctb],
Michael Kluge [ctb],
Thomas Schwarzl [ctb],
Zhougeng Xu [ctb]

R topics documented:

`ChIPseeker-package` *ChIP-SEQ Annotation, Visualization and Comparison*

Description

This package is designed for chip-seq data analysis

Details

Package:	ChIPseeker
Type:	Package
Version:	1.5.1
Date:	27-04-2015
biocViews:	ChIPSeq, Annotation, Software
Depends:	
Imports:	methods, ggplot2
Suggests:	clusterProfiler, GOSemSim
License:	Artistic-2.0

Author(s)

Guangchuang Yu

Maintainer: Guangchuang Yu <`guangchuangyu@gmail.com`>

`.` *.*

Description

capture name of variable

Usage

```
.(..., .env = parent.frame())
```

Arguments

`...`	expression
`.env`	environment

Value

expression

Examples

```
x <- 1
eval(.(x)[[1]])
```

`annotatePeak` *annotatePeak*

Description

Annotate peaks

Usage

```
annotatePeak(
  peak,
  tssRegion = c(-3000, 3000),
  TxDb = NULL,
  level = "transcript",
  assignGenomicAnnotation = TRUE,
  genomicAnnotationPriority = c("Promoter", "5UTR", "3UTR", "Exon", "Intron",
    "Downstream", "Intergenic"),
  annoDb = NULL,
  addFlankGeneInfo = FALSE,
  flankDistance = 5000,
  sameStrand = FALSE,
  ignoreOverlap = FALSE,
  ignoreUpstream = FALSE,
  ignoreDownstream = FALSE,
  overlap = "TSS",
  verbose = TRUE
)
```

Arguments

`peak`	peak file or GRanges object
`tssRegion`	Region Range of TSS
`TxDb`	TxDb or EnsDb annotation object
`level`	one of transcript and gene
`assignGenomicAnnotation`	logical, assign peak genomic annotation or not
`genomicAnnotationPriority`	genomic annotation priority
`annoDb`	annotation package
`addFlankGeneInfo`	logical, add flanking gene information from the peaks
`flankDistance`	distance of flanking sequence
`sameStrand`	logical, whether find nearest/overlap gene in the same strand
`ignoreOverlap`	logical, whether ignore overlap of TSS with peak
`ignoreUpstream`	logical, if True only annotate gene at the 3' of the peak.
`ignoreDownstream`	logical, if True only annotate gene at the 5' of the peak.
`overlap`	one of 'TSS' or 'all', if overlap="all", then gene overlap with peak will be reported as nearest gene, no matter the overlap is at TSS region or not.
`verbose`	print message or not

Value

data.frame or GRanges object with columns of:

all columns provided by input.

annotation: genomic feature of the peak, for instance if the peak is located in 5'UTR, it will annotated by 5'UTR. Possible annotation is Promoter-TSS, Exon, 5' UTR, 3' UTR, Intron, and Intergenic.

geneChr: Chromosome of the nearest gene

geneStart: gene start

geneEnd: gene end

geneLength: gene length

geneStrand: gene strand

geneId: entrezgene ID

distanceToTSS: distance from peak to gene TSS

if annoDb is provided, extra column will be included:

ENSEMBL: ensembl ID of the nearest gene

SYMBOL: gene symbol

GENENAME: full gene name

Author(s)

G Yu

See Also

`plotAnnoBar` `plotAnnoPie` `plotDistToTSS`

Examples

```
## Not run:
require(TxDb.Hsapiens.UCSC.hg19.knownGene)
txdb <- TxDb.Hsapiens.UCSC.hg19.knownGene
peakfile <- system.file("extdata", "sample_peaks.txt", package="ChIPseeker")
peakAnno <- annotatePeak(peakfile, tssRegion=c(-3000, 3000), TxDb=txdb)
peakAnno

## End(Not run)
```

`as.data.frame.csAnno` *as.data.frame.csAnno*

Description

convert csAnno object to data.frame

Usage

```
## S3 method for class 'csAnno'
as.data.frame(x, row.names = NULL, optional = FALSE, ...)
```

Arguments

`x`	csAnno object
`row.names`	row names
`optional`	should be omitted.
`...`	additional parameters

Value

data.frame

Author(s)

Guangchuang Yu `https://guangchuangyu.github.io`

`as.GRanges` *as.GRanges*

Description

convert csAnno object to GRanges

Usage

```
as.GRanges(x)
```

Arguments

`x`	csAnno object

Value

GRanges object

Author(s)

Guangchuang Yu `https://guangchuangyu.github.io`

`check_upstream_and_downstream` *check upstream and downstream parameter*

Description

check_upstream_and_downstream

Usage

```
check_upstream_and_downstream(upstream, downstream)
```

Arguments

`upstream`	upstream
`downstream`	downstream

`combine_csAnno` *combine_csAnno*

Description

Combine csAnno Object

Usage

```
combine_csAnno(x, ...)
```

Arguments

`x`	csAnno object
`...`	csAnno objects

Details

https://github.com/YuLab-SMU/ChIPseeker/issues/157

Value

csAnno object

`covplot` *covplot*

Description

plot peak coverage

Usage

```
covplot(
  peak,
  weightCol = NULL,
  xlab = "Chromosome Size (bp)",
  ylab = "",
  title = "ChIP Peaks over Chromosomes",
  chrs = NULL,
  xlim = NULL,
  lower = 1
)
```

Arguments

`peak`	peak file or GRanges object
`weightCol`	weight column of peak
`xlab`	xlab
`ylab`	ylab
`title`	title
`chrs`	selected chromosomes to plot, all chromosomes by default
`xlim`	ranges to plot, default is whole chromosome
`lower`	lower cutoff of coverage signal

Value

ggplot2 object

Author(s)

G Yu

`csAnno-class` *Class "csAnno" This class represents the output of ChIPseeker Annotation*

Description

Class "csAnno" This class represents the output of ChIPseeker Annotation

Slots

`anno`	annotation
`tssRegion`	TSS region
`level`	transcript or gene
`hasGenomicAnnotation`	logical
`detailGenomicAnnotation`	Genomic Annotation in detail
`annoStat`	annotation statistics
`peakNum`	number of peaks

Author(s)

Guangchuang Yu `https://guangchuangyu.github.io`

See Also

`annotatePeak`

`downloadGEObedFiles` *downloadGEObedFiles*

Description

download all BED files of a particular genome version

Usage

```
downloadGEObedFiles(genome, destDir = getwd())
```

Arguments

`genome`	genome version
`destDir`	destination folder

Author(s)

G Yu

`downloadGSMbedFiles` *downloadGSMbedFiles*

Description

download BED supplementary files of a list of GSM accession numbers

Usage

```
downloadGSMbedFiles(GSM, destDir = getwd())
```

Arguments

`GSM`	GSM accession numbers
`destDir`	destination folder

Author(s)

G Yu

`dropAnno` *dropAnno*

Description

dropAnno

Usage

```
dropAnno(csAnno, distanceToTSS_cutoff = 10000)
```

Arguments

`csAnno`	output of annotatePeak
`distanceToTSS_cutoff`	distance to TSS cutoff

Details

drop annotation exceeding distanceToTSS_cutoff

Value

csAnno object

Author(s)

Guangchuang Yu

`enrichAnnoOverlap` *enrichAnnoOverlap*

Description

calcuate overlap significant of ChIP experiments based on their nearest gene annotation

Usage

```
enrichAnnoOverlap(
  queryPeak,
  targetPeak,
  TxDb = NULL,
  pAdjustMethod = "BH",
  chainFile = NULL,
  distanceToTSS_cutoff = NULL
)
```

Arguments

`queryPeak`	query bed file
`targetPeak`	target bed file(s) or folder containing bed files
`TxDb`	TxDb
`pAdjustMethod`	pvalue adjustment method
`chainFile`	chain file for liftOver
`distanceToTSS_cutoff`	restrict nearest gene annotation by distance cutoff

Value

data.frame

Author(s)

G Yu

`enrichPeakOverlap` *enrichPeakOverlap*

Description

calculate overlap significant of ChIP experiments based on the genome coordinations

Usage

```
enrichPeakOverlap(
  queryPeak,
  targetPeak,
  TxDb = NULL,
  pAdjustMethod = "BH",
  nShuffle = 1000,
  chainFile = NULL,
  pool = TRUE,
  mc.cores = detectCores() - 1,
  verbose = TRUE
)
```

Arguments

`queryPeak`	query bed file or GRanges object
`targetPeak`	target bed file(s) or folder that containing bed files or a list of GRanges objects
`TxDb`	TxDb
`pAdjustMethod`	pvalue adjustment method
`nShuffle`	shuffle numbers
`chainFile`	chain file for liftOver
`pool`	logical, whether pool target peaks
`mc.cores`	number of cores, see mclapply
`verbose`	logical

Value

data.frame

Author(s)

G Yu

`getBioRegion` *getBioRegion*

Description

prepare a bioregion of selected feature

Usage

```
getBioRegion(
  TxDb = NULL,
  upstream = 1000,
  downstream = 1000,
  by = "gene",
  type = "start_site"
)
```

Arguments

`TxDb`	TxDb
`upstream`	upstream from start site
`downstream`	downstream from start site
`by`	one of 'gene', 'transcript', 'exon', 'intron' , '3UTR' , '5UTR'
`type`	one of "start_site", "end_site", "body"

Details

this function combined previous functions getPromoters(),getBioRegion(),getGeneBody() https://github.com/GuangchuangY
https://github.com/GuangchuangYu/ChIPseeker/issues/87

Value

GRanges object

Author(s)

Guangchuang Yu, Ming L

`getGeneAnno` *getGeneAnno*

Description

get gene annotation, symbol, gene name etc.

Usage

```
getGeneAnno(annoDb, geneID, type)
```

Arguments

`annoDb`	annotation package
`geneID`	query geneID
`type`	gene ID type

Value

data.frame

Author(s)

G Yu

`getGenomicAnnotation` *getGenomicAnnotation*

Description

get Genomic Annotation of peaks

Usage

```
getGenomicAnnotation(
  peaks,
  distance,
  tssRegion = c(-3000, 3000),
  TxDb,
  level,
  genomicAnnotationPriority,
  sameStrand = FALSE
)
```

Arguments

`peaks`	peaks in GRanges object
`distance`	distance of peak to TSS
`tssRegion`	tssRegion, default is -3kb to +3kb
`TxDb`	TxDb object
`level`	one of gene or transcript
`genomicAnnotationPriority`	genomic Annotation Priority
`sameStrand`	whether annotate gene in same strand

Value

character vector

Author(s)

G Yu

`getGEOgenomeVersion` *getGEOgenomeVersion*

Description

get genome version statistics collecting from GEO ChIPseq data

Usage

```
getGEOgenomeVersion()
```

Value

data.frame

Author(s)

G Yu

`getGEOInfo` *getGEOInfo*

Description

get subset of GEO information by genome version keyword

Usage

```
getGEOInfo(genome, simplify = TRUE)
```

Arguments

`genome`	genome version
`simplify`	simplify result or not

Value

data.frame

Author(s)

G Yu

`getGEOspecies` *getGEOspecies*

Description

accessing species statistics collecting from GEO database

Usage

```
getGEOspecies()
```

Value

data.frame

Author(s)

G Yu

`getNearestFeatureIndicesAndDistances`
getNearestFeatureIndicesAndDistances

Description

get index of features that closest to peak and calculate distance

Usage

```
getNearestFeatureIndicesAndDistances(
  peaks,
  features,
  sameStrand = FALSE,
  ignoreOverlap = FALSE,
  ignoreUpstream = FALSE,
  ignoreDownstream = FALSE,
  overlap = "TSS"
)
```

Arguments

`peaks`	peak in GRanges
`features`	features in GRanges
`sameStrand`	logical, whether find nearest gene in the same strand
`ignoreOverlap`	logical, whether ignore overlap of TSS with peak
`ignoreUpstream`	logical, if True only annotate gene at the 3' of the peak.
`ignoreDownstream`	logical, if True only annotate gene at the 5' of the peak.
`overlap`	one of "TSS" or "all"

Value

list

Author(s)

G Yu

`getPromoters` *getPromoters*

Description

prepare the promoter regions

Usage

```
getPromoters(TxDb = NULL, upstream = 1000, downstream = 1000, by = "gene")
```

Arguments

`TxDb`	TxDb
`upstream`	upstream from TSS site
`downstream`	downstream from TSS site
`by`	one of gene or transcript

Value

GRanges object

`getSampleFiles` *getSampleFiles*

Description

get filenames of sample files

Usage

```
getSampleFiles()
```

Value

list of file names

Author(s)

G Yu

`getTagMatrix` *getTagMatrix*

Description

calculate the tag matrix

Usage

```
getTagMatrix(
  peak,
  upstream,
  downstream,
  windows,
  type,
  by,
  TxDb = NULL,
  weightCol = NULL,
  nbin = NULL,
  verbose = TRUE,
  ignore_strand = FALSE
)
```

Arguments

`peak`	peak peak file or GRanges object
`upstream`	the distance of upstream extension
`downstream`	the distance of downstream extension
`windows`	a collection of region
`type`	one of "start_site", "end_site", "body"
`by`	one of 'gene', 'transcript', 'exon', 'intron' , '3UTR' , '5UTR'
`TxDb`	TxDb
`weightCol`	column name of weight, default is NULL
`nbin`	the amount of nbines
`verbose`	print message or not
`ignore_strand`	ignore the strand information or not

Value

tagMatrix

`getTagMatrix.binning.internal` *getTagMatrix.binning.internal*

Description

calculate the tagMatrix by binning the idea was derived from the function of deeptools https://deeptools.readthedocs.io/en/dev

Usage

```
getTagMatrix.binning.internal(
  peak,
  weightCol = NULL,
  windows,
  nbin = 800,
  upstream = NULL,
  downstream = NULL,
  ignore_strand = FALSE
)
```

Arguments

`peak`	peak peak file or GRanges object
`weightCol`	weightCol column name of weight, default is NULL
`windows`	windows a collection of region with equal or not equal size, eg. promoter region, gene region.
`nbin`	the amount of nbines needed to be splited and it should not be more than min_body_length
`upstream`	rel object, NULL or actual number
`downstream`	rel object, NULL or actual number
`ignore_strand`	ignore the strand information or not

Value

tagMatrix

`getTagMatrix.internal` *getTagMatrix.internal*

Description

calculate the tag matrix

Usage

```
getTagMatrix.internal(peak, weightCol = NULL, windows, ignore_strand = FALSE)
```

Arguments

`peak`	peak file or GRanges object
`weightCol`	column name of weight, default is NULL
`windows`	a collection of region with equal size, eg. promoter region.
`ignore_strand`	ignore the strand information or not

Value

tagMatrix

Author(s)

G Yu

`info` *Information Datasets*

Description

ucsc genome version, precalcuated data and gsm information

`overlap` *overlap*

Description

calculate the overlap matrix, which is useful for vennplot

Usage

```
overlap(Sets)
```

Arguments

`Sets`	a list of objects

Value

data.frame

Author(s)

G Yu

`peakHeatmap` *peakHeatmap*

Description

plot the heatmap of peaks align to flank sequences of TSS

Usage

```
peakHeatmap(
  peak,
  weightCol = NULL,
  TxDb = NULL,
  upstream = 1000,
  downstream = 1000,
  xlab = "",
  ylab = "",
  title = NULL,
  color = NULL,
  verbose = TRUE
)
```

Arguments

`peak`	peak file or GRanges object
`weightCol`	column name of weight
`TxDb`	TxDb object
`upstream`	upstream position
`downstream`	downstream position
`xlab`	xlab
`ylab`	ylab
`title`	title
`color`	color
`verbose`	print message or not

Value

figure

Author(s)

G Yu

`plotAnnoBar` *plotAnnoBar method generics*

Description

plotAnnoBar method for `csAnno` instance

Usage

```
plotAnnoBar(
  x,
  xlab = "",
  ylab = "Percentage(%)",
  title = "Feature Distribution",
  ...
)

## S4 method for signature 'list'
plotAnnoBar(
  x,
  xlab = "",
  ylab = "Percentage(%)",
  title = "Feature Distribution",
  ...
)

plotAnnoBar(x, xlab="", ylab='Percentage(%)',title="Feature Distribution", ...)
```

Arguments

`x`	csAnno instance
`xlab`	xlab
`ylab`	ylab
`title`	title
`...`	additional paramter

Value

plot

Author(s)

Guangchuang Yu `https://guangchuangyu.github.io`

`plotAnnoBar.data.frame` *plotAnnoBar.data.frame*

Description

plot feature distribution based on their chromosome region

Usage

```
plotAnnoBar.data.frame(
  anno.df,
  xlab = "",
  ylab = "Percentage(%)",
  title = "Feature Distribution",
  categoryColumn
)
```

Arguments

`anno.df`	annotation stats
`xlab`	xlab
`ylab`	ylab
`title`	plot title
`categoryColumn`	category column

Details

plot chromosome region features

Value

bar plot that summarize genomic features of peaks

Author(s)

Guangchuang Yu `https://yulab-smu.top`

See Also

`annotatePeak` `plotAnnoPie`

`plotAnnoPie` *plotAnnoPie method generics*

Description

plotAnnoPie method for `csAnno` instance

Usage

```
plotAnnoPie(
  x,
  ndigit = 2,
  cex = 0.9,
  col = NA,
  legend.position = "rightside",
  pie3D = FALSE,
  radius = 0.8,
  ...
)

plotAnnoPie(x,ndigit=2,cex=0.9,col=NA,legend.position="rightside",pie3D=FALSE,radius=0.8,...)
```

Arguments

`x`	`csAnno` instance
`ndigit`	number of digit to round
`cex`	label cex
`col`	color
`legend.position`	topright or other.
`pie3D`	plot in 3D or not
`radius`	radius of the pie
`...`	extra parameter

Value

plot

Author(s)

Guangchuang Yu `https://guangchuangyu.github.io`

`plotAnnoPie.csAnno` *plotAnnoPie*

Description

pieplot from peak genomic annotation

Usage

```
plotAnnoPie.csAnno(
  x,
  ndigit = 2,
  cex = 0.8,
  col = NA,
  legend.position = "rightside",
  pie3D = FALSE,
  radius = 0.8,
  ...
)
```

Arguments

`x`	csAnno object
`ndigit`	number of digit to round
`cex`	label cex
`col`	color
`legend.position`	topright or other.
`pie3D`	plot in 3D or not
`radius`	radius of Pie
`...`	extra parameter

Value

pie plot of peak genomic feature annotation

Author(s)

Guangchuang Yu `https://yulab-smu.top`

See Also

`annotatePeak` `plotAnnoBar`

Examples

```
## Not run:
require(TxDb.Hsapiens.UCSC.hg19.knownGene)
txdb <- TxDb.Hsapiens.UCSC.hg19.knownGene
peakfile <- system.file("extdata", "sample_peaks.txt", package="chipseeker")
peakAnno <- annotatePeak(peakfile, TxDb=txdb)
plotAnnoPie(peakAnno)

## End(Not run)
```

`plotAvgProf` *plotAvgProf*

Description

plot the profile of peaks

Usage

```
plotAvgProf(
  tagMatrix,
  xlim,
  xlab = "Genomic Region (5'->3')",
  ylab = "Peak Count Frequency",
  conf,
  facet = "none",
  free_y = TRUE,
  origin_label = "TSS",
  verbose = TRUE,
  ...
)
```

Arguments

`tagMatrix`	tagMatrix or a list of tagMatrix
`xlim`	xlim
`xlab`	x label
`ylab`	y label
`conf`	confidence interval
`facet`	one of 'none', 'row' and 'column'
`free_y`	if TRUE, y will be scaled by AvgProf
`origin_label`	label of the center
`verbose`	print message or not
`...`	additional parameter

Value

ggplot object

Author(s)

G Yu; Y Yan

`plotAvgProf.binning` *plotAvgProf.binning*

Description

plot the profile of peaks by binning

Usage

```
plotAvgProf.binning(
  tagMatrix,
  xlab = "Genomic Region (5'->3')",
  ylab = "Peak Count Frequency",
  conf,
  facet = "none",
  free_y = TRUE,
  upstream = NULL,
  downstream = NULL,
  label,
  ...
)
```

Arguments

`tagMatrix`	tagMatrix or a list of tagMatrix
`xlab`	x label
`ylab`	y label
`conf`	confidence interval
`facet`	one of 'none', 'row' and 'column'
`free_y`	if TRUE, y will be scaled
`upstream`	rel object reflects the percentage of flank extension, e.g rel(0.2) integer reflects the actual length of flank extension or TSS region NULL reflects the gene body with no extension
`downstream`	rel object reflects the percentage of flank extension, e.g rel(0.2) integer reflects the actual length of flank extension or TSS region NULL reflects the gene body with no extension
`label`	label
`...`	additional parameter

Value

ggplot object

`plotAvgProf2` *plotAvgProf*

Description

plot the profile of peaks that align to flank sequences of TSS

Usage

```
plotAvgProf2(
  peak,
  weightCol = NULL,
  TxDb = NULL,
  upstream = 1000,
  downstream = 1000,
  xlab = "Genomic Region (5'->3')",
  ylab = "Peak Count Frequency",
  conf,
  facet = "none",
  free_y = TRUE,
  verbose = TRUE,
  ignore_strand = FALSE,
  ...
)
```

Arguments

`peak`	peak file or GRanges object
`weightCol`	column name of weight
`TxDb`	TxDb object
`upstream`	upstream position
`downstream`	downstream position
`xlab`	xlab
`ylab`	ylab
`conf`	confidence interval
`facet`	one of 'none', 'row' and 'column'
`free_y`	if TRUE, y will be scaled by AvgProf
`verbose`	print message or not
`ignore_strand`	ignore the strand information or not
`...`	additional parameter

Value

ggplot object

Author(s)

G Yu, Ming L

`plotDistToTSS` *plotDistToTSS method generics*

Description

plotDistToTSS method for `csAnno` instance

Usage

```
plotDistToTSS(
  x,
  distanceColumn = "distanceToTSS",
  xlab = "",
  ylab = "Binding sites (%) (5'->3')",
  title = "Distribution of transcription factor-binding loci relative to TSS",
  ...
)

## S4 method for signature 'list'
plotDistToTSS(
  x,
  distanceColumn = "distanceToTSS",
  xlab = "",
  ylab = "Binding sites (%) (5'->3')",
  title = "Distribution of transcription factor-binding loci relative to TSS",
  ...
)

plotDistToTSS(x,distanceColumn="distanceToTSS", xlab="",
ylab="Binding sites (%) (5'->3')",
title="Distribution of transcription factor-binding loci relative to TSS",...)
```

Arguments

`x`	`csAnno` instance
`distanceColumn`	distance column name
`xlab`	xlab
`ylab`	ylab
`title`	title
`...`	additional parameter

Value

plot

Author(s)

Guangchuang Yu https://guangchuangyu.github.io

`plotDistToTSS.data.frame`

plotDistToTSS.data.frame

Description

plot feature distribution based on the distances to the TSS

Usage

```
plotDistToTSS.data.frame(
  peakDist,
  distanceColumn = "distanceToTSS",
  xlab = "",
  ylab = "Binding sites (%) (5'->3')",
  title = "Distribution of transcription factor-binding loci relative to TSS",
  categoryColumn
)
```

Arguments

`peakDist`	peak annotation
`distanceColumn`	column name of the distance from peak to nearest gene
`xlab`	x label
`ylab`	y lable
`title`	figure title
`categoryColumn`	category column

Value

bar plot that summarize distance from peak to TSS of the nearest gene.

Author(s)

Guangchuang Yu https://guangchuangyu.github.io

See Also

`annotatePeak`

Examples

```
## Not run:
require(TxDb.Hsapiens.UCSC.hg19.knownGene)
txdb <- TxDb.Hsapiens.UCSC.hg19.knownGene
peakfile <- system.file("extdata", "sample_peaks.txt", package="ChIPseeker")
peakAnno <- annotatePeak(peakfile, TxDb=txdb)
plotDistToTSS(peakAnno)

## End(Not run)
```

`plotPeakProf` *plotPeakProf*

Description

plot the profile of peaks

Usage

```
plotPeakProf(
  tagMatrix,
  conf,
  xlab = "Genomic Region (5'->3')",
  ylab = "Peak Count Frequency",
  facet = "none",
  free_y = TRUE,
  ...
)
```

Arguments

`tagMatrix`	tagMatrix or a list of tagMatrix
`conf`	confidence interval
`xlab`	x label
`ylab`	y label
`facet`	one of 'none', 'row' and 'column'
`free_y`	if TRUE, y will be scaled by AvgProf
`...`	additional parameter

Details

this function combined previous function plotAvgProf()

Value

ggplot object

`plotPeakProf2` *plotAvgProf2*

Description

plot the profile of peaks automatically

Usage

```
plotPeakProf2(
  peak,
  upstream,
  downstream,
  conf,
  by,
  type,
  weightCol = NULL,
  TxDb = NULL,
  xlab = "Genomic Region (5'->3')",
  ylab = "Peak Count Frequency",
  facet = "none",
  free_y = TRUE,
  verbose = TRUE,
  nbin = NULL,
  ignore_strand = FALSE,
  ...
)
```

Arguments

`peak`	peak file or GRanges object
`upstream`	upstream position
`downstream`	downstream position
`conf`	confidence interval
`by`	one of 'gene', 'transcript', 'exon', 'intron' , '3UTR' , '5UTR'
`type`	one of "start_site", "end_site", "body"
`weightCol`	column name of weight
`TxDb`	TxDb object
`xlab`	xlab
`ylab`	ylab
`facet`	one of 'none', 'row' and 'column'
`free_y`	if TRUE, y will be scaled by AvgProf
`verbose`	print message or not

`nbin`	the amount of nbines
`ignore_strand`	ignore the strand information or not
`...`	additional parameter

Value

ggplot object

Author(s)

G Yu, Ming Li

`readPeakFile` *readPeakFile*

Description

read peak file and store in data.frame or GRanges object

Usage

```
readPeakFile(peakfile, as = "GRanges", ...)
```

Arguments

`peakfile`	peak file
`as`	output format, one of GRanges or data.frame
`...`	additional parameter

Value

peak information, in GRanges or data.frame object

Author(s)

G Yu

Examples

```
peakfile <- system.file("extdata", "sample_peaks.txt", package="ChIPseeker")
peak.gr <- readPeakFile(peakfile, as="GRanges")
peak.gr
```

`seq2gene` *seq2gene*

Description

annotate genomic regions to genes in many-to-many mapping

Usage

```
seq2gene(seq, tssRegion, flankDistance, TxDb, sameStrand = FALSE)
```

Arguments

`seq`	genomic regions in GRanges object
`tssRegion`	TSS region
`flankDistance`	flanking search radius
`TxDb`	TranscriptDb object
`sameStrand`	logical whether find nearest/overlap gene in the same strand

Details

This funciton associates genomic regions with coding genes in a many-to-many mapping. It first maps genomic regions to host genes (either located in exon or intron), proximal genes (located in promoter regions) and flanking genes (located in upstream and downstream within user specify distance).

Value

gene vector

Author(s)

Guangchuang Yu

Examples

```
library(TxDb.Hsapiens.UCSC.hg19.knownGene)
TxDb <- TxDb.Hsapiens.UCSC.hg19.knownGene
file <- getSampleFiles()[[1]] # a bed file
gr <- readPeakFile(file)
genes <- seq2gene(gr, tssRegion=c(-1000, 1000), flankDistance = 3000, TxDb)
```

`show` *show method*

Description

show method for `csAnno` instance

Usage

```
show(object)
```

Arguments

`object`	A `csAnno` instance

Value

message

Author(s)

Guangchuang Yu `https://guangchuangyu.github.io`

`shuffle` *shuffle*

Description

shuffle the position of peak

Usage

```
shuffle(peak.gr, TxDb)
```

Arguments

`peak.gr`	GRanges object
`TxDb`	TxDb

Value

GRanges object

Author(s)

G Yu

`tagHeatmap` *tagHeatmap*

Description

plot the heatmap of tagMatrix

Usage

```
tagHeatmap(tagMatrix, xlim, xlab = "", ylab = "", title = NULL, color = "red")
```

Arguments

`tagMatrix`	tagMatrix or a list of tagMatrix
`xlim`	xlim
`xlab`	xlab
`ylab`	ylab
`title`	title
`color`	color

Value

figure

Author(s)

G Yu

`upsetplot` *upsetplot method*

Description

upsetplot method generics

Usage

```
upsetplot(x, ...)
```

Arguments

`x`	A `csAnno` instance
`...`	additional parameter

Value

plot

Author(s)

Guangchuang Yu `https://guangchuangyu.github.io`

`vennpie` *vennpie method generics*

Description

vennpie method generics

Usage

```
vennpie(x, r = 0.2, ...)

vennpie(x, r=0.2, ...)
```

Arguments

`x`	A `csAnno` instance
`r`	initial radius
`...`	additional parameter

Value

plot

Author(s)

Guangchuang Yu `https://guangchuangyu.github.io`

`vennplot` *vennplot*

Description

plot the overlap of a list of object

Usage

```
vennplot(Sets, by = "gplots")
```

Arguments

`Sets`	a list of object, can be vector or GRanges object
`by`	one of gplots or Vennerable

Value

venn plot that summarize the overlap of peaks from different experiments or gene annotation from different peak files.

Author(s)

G Yu

Examples

```
## example not run
## require(TxDb.Hsapiens.UCSC.hg19.knownGene)
## txdb <- TxDb.Hsapiens.UCSC.hg19.knownGene
## peakfiles <- getSampleFiles()
## peakAnnoList <- lapply(peakfiles, annotatePeak)
## names(peakAnnoList) <- names(peakfiles)
## genes= lapply(peakAnnoList, function(i) as.data.frame(i)$geneId)
## vennplot(genes)
```

`vennplot.peakfile` *vennplot.peakfile*

Description

vennplot for peak files

Usage

```
vennplot.peakfile(files, labels = NULL)
```

Arguments

`files`	peak files
`labels`	labels for peak files

Value

figure

Author(s)

G Yu

Index