Package 'cliProfiler'

February 27, 2026

Type Package

Title A package for the CLIP data visualization

Version 1.16.0

Date 2021-09-30

Description An easy and fast way to visualize and profile the high-throughput IP data. This package generates the meta gene profile and other profiles. These profiles could provide valuable information for understanding the IP experiment results.

License Artistic-2.0

VignetteBuilder knitr

Encoding UTF-8

LazyData TRUE

URL https://github.com/Codezy99/cliProfiler

BugReports https://github.com/Codezy99/cliProfiler/issues

biocViews Sequencing, ChIPSeq, Visualization, Epigenetics, Genetics

RoxygenNote 7.1.1

Imports dplyr, rtracklayer, GenomicRanges, ggplot2, BSgenome, Biostrings, utils

Depends S4Vectors, methods, R (>= 4.1)

Suggests knitr, rmarkdown, bookdown, testthat, BiocStyle, BSgenome.Mmusculus.UCSC.mm10

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

git_branch RELEASE_3_22

git_last_commit 3ed7cb4

git_last_commit_date 2025-10-29

Repository Bioconductor 3.22

Date/Publication 2026-02-27

Author You Zhou [aut, cre] (ORCID: <https://orcid.org/0000-0003-1755-9073>), Kathi Zarnack [aut] (ORCID: <https://orcid.org/0000-0003-3527-3378>)

Maintainer You Zhou <youzhoulearning@gmail.com>

Contents

`exonProfile` *exonProfile for the GRanges objects*

Description

An function to check the position of peaks in the exonic region.

Usage

```
exonProfile(
    object,
    annotation,
    title = "Exon Profile",
    group = NA,
    exlevel = NA,
    extranscript_support_level = NA,
    maxLength = NA,
    minLength = NA,
    nomap = FALSE
)
```

Arguments

`object`	A GRanges object which should contains all the peaks that you want to check
`annotation`	A path way to the annotation file. The format of the annotation file should be gff3 and downloaded from https://www.gencodegenes.org/
`title`	The main title for the output meta gene profile plot.
`group`	The column name which contains the information of grouping for making the comparison plot. NA means all the peaks belongs to the same catagory.
`exlevel`	A parameter for the annotation filtering. exlevel represents the level that you would like to exclude. NA means no level filtering for the annotation file. The level from the annotations refers to how reliable this annotation is. For more information about level please check https://www.gencodegenes.org/pages/data_format.html.
`extranscript_support_level`	A parameter for the annotation filtering. extranscript_support_level represents the transcript_support_level that you would like to exclude (e.g. 4 and 5). NA means no transcript_support_level filtering for the annotation file. Transcripts are scored according to how well mRNA and EST alignments match over its full length. Here the number 6 means the transcript_support_level NA. For more information about level please check https://www.gencodegenes.org/pages/data_format.html.

maxLength	A numeric value which indicate the maximum value of exon length for the annotation filtering. Or a NA which will turn off the max length annotation filtering.
minLength	A numeric value which indicate the minimum value of exon length for the annotation filtering. Or a NA which will turn off the min length annotation filtering.
nomap	A logical vector (TRUE or FALSE). It indicates whether you would like to exclude peaks that cannot assign to annotations in the plot.

Details

Here is an explanation of output meta data in the `list 1`:

`center`: The center position of each peaks. This center position is used for calculating the position of peaks within the genomic regions.

- `exon_S` and `exon_E`: The location of 5' and 3' splice sites (SS) of the exon.
- `exon_length`: The length of the exon that peak assigned.
- `exon_transcript_id`: The transcript ID for the exon
- `exon_map`: The relative position of each peak. This value close to 0 means this peak located close to the 3' SS. The position value close to one means the peak close to the 5' SS. Value 3 means this peaks can not map to any annotation.

Value

A list object, the list 1 contains the information of the assignment of the peaks and their position value within the exon. The value close to 1 means the peak close to the 5' splice site. The list 2 includes the plot of exonProfile.

Author(s)

You Zhou, Kathi Zarnack

Examples

```
## Load the test data and get the path to the test gff3 file
testpath <- system.file("extdata", package = "cliProfiler")
test <- readRDS(file.path(testpath, "test.rds"))
test_gff3 <- file.path(testpath, "annotation_test.gff3")

output <- exonProfile(test, test_gff3)
```

---

geneTypeProfile *geneTypeProfile for the GRanges objects*

Description

An function to check the gene type belonging for the peaks.

Usage

```
geneTypeProfile(
    object,
    annotation,
    title = "Gene Type Profile",
    exlevel = NA,
    extranscript_support_level = NA
)
```

Arguments

object	A GRanges object which should contains all the peaks that you want to check
annotation	A path way to the annotation file. The format of the annotation file should be gff3 and downloaded from https://www.gencodegenes.org/
title	The main title for the output meta gene profile plot.
exlevel	A parameter for the annotation filtering. exlevel represents the level that you would like to exclude. NA means no level filtering for the annotation file. The level from the annotations refers to how reliable this annotation is. For more information about level please check https://www.gencodegenes.org/pages/data_format.html.
extranscript_support_level	A parameter for the annotation filtering. extranscript_support_level represents the transcript_support_level that you would like to exclude (e.g. 4 and 5). NA means no transcript_support_level filtering for the annotation file. Transcripts are scored according to how well mRNA and EST alignments match over its full length. Here the number 6 means the transcript_support_level NA. For more information about level please check https://www.gencodegenes.org/pages/data_format.html.

Details

Here is an explanation of output meta data in the `list 1`:

`center`: The center position of each peaks. This center position is used for calculating the position of peaks within the genomic regions.

- `geneType`: The gene type of the gene that input peak belongs to.
- `Gene_ID`: The gene ID of the gene that input peak belongs to.

Value

A list object, the list 1 contains the information of the assignment of the peaks and the gene type of their located genes. The list 2 includes the plot of geneTypeProfile

Author(s)

You Zhou, Kathi Zarnack

Examples

```
## Load the test data and get the path to the test gff3 file
testpath <- system.file("extdata", package = "cliProfiler")
test <- readRDS(file.path(testpath, "test.rds"))
test_gff3 <- file.path(testpath, "annotation_test.gff3")

output <- geneTypeProfile(test, test_gff3)
```

`intronProfile` *intronProfile for the GRanges objects*

Description

An function to check the position of peaks in the intronic region.

Usage

```
intronProfile(
    object,
    annotation,
    title = "Intron Profile",
    group = NA,
    exlevel = NA,
    extranscript_support_level = NA,
    maxLength = NA,
    minLength = NA,
    nomap = FALSE
)
```

Arguments

`object`	A GRanges object which should contains all the peaks that you want to check
`annotation`	A path way to the annotation file. The format of the annotation file should be gff3 and downloaded from https://www.gencodegenes.org/
`title`	The main title for the output meta gene profile plot.
`group`	The column name which contains the information of grouping for making the comparison plot. NA means all the peaks belongs to the same catagory.
`exlevel`	A parameter for the annotation filtering. exlevel represents the level that you would like to exclude. NA means no level filtering for the annotation file. The level from the annotations refers to how reliable this annotation is. For more information about level please check https://www.gencodegenes.org/pages/data_format.html.
`extranscript_support_level`	A parameter for the annotation filtering. extranscript_support_level represents the transcript_support_level that you would like to exclude (e.g. 4 and 5). NA means no transcript_support_level filtering for the annotation file. Transcripts are scored according to how well mRNA and EST alignments match over its full length. Here the number 6 means the transcript_support_level NA. For more information about level please check https://www.gencodegenes.org/pages/data_format.html.
`maxLength`	A numeric value which indicate the maximum value of exon length for the annotation filtering. Or a NA which will turn off the max length annotation filtering.
`minLength`	A numeric value which indicate the minimum value of exon length for the annotation filtering. Or a NA which will turn off the min length annotation filtering.
`nomap`	A logical vector (TRUE or FALSE). It indicates whether you would like to exclude peaks that cannot assign to annotations in the plot.

Details

Here is an explanation of output meta data in the `list 1`:

`center`: The center position of each peaks. This center position is used for calculating the position of peaks within the genomic regions.

- `Intron_S` and `Intron_E`: The location of 5' and 3' splice sites (SS) of the intron.
- `Intron_length`: The length of the intron that peak assigned.
- `Intron_transcript_id`: The transcript ID for the intron.
- `Intron_map`: The relative position of each peak. This value close to 0 means this peak located close to the 5' SS. The position value close to one means the peak close to the 3' SS. Value 3 means this peaks can not map to any annotation.

Value

A list object, the list 1 contains the information of the assignment of the peaks and their position value within the intron. The value close to 1 means the peak close to the 3' splice site. The list 2 includes the plot of intronProfile.

Author(s)

You Zhou, Kathi Zarnack

Examples

```
## Load the test data and get the path to the test gff3 file
testpath <- system.file("extdata", package = "cliProfiler")
test <- readRDS(file.path(testpath, "test.rds"))
test_gff3 <- file.path(testpath, "annotation_test.gff3")

output <- intronProfile(test, test_gff3)
```

---

metaGeneProfile *metaGeneProfile for the GRanges objects*

Description

An function for calculating the genomic position and generate the meta gene profile plot of the input peaks.

Usage

```
metaGeneProfile(
    object,
    annotation,
    include_intron = FALSE,
    title = "Meta Gene Profile",
    group = NA,
    split = FALSE,
    exlevel = NA,
    extranscript_support_level = NA,
    adjust = 1,
    nomap = FALSE
)
```

Arguments

object	A GRanges object which should contains all the peaks that you want to check
annotation	A path way to the annotation file. The format of the annotation file should be gff3 and downloaded from https://www.gencodegenes.org/
include_intron	A logical vector TRUE or FALSE that define whether the intronic region should be included in the position calculation or not.
title	The main title for the output meta gene profile plot.
group	The column name which contains the information of grouping for making the comparison plot. NA means all the peaks belongs to the same catagory.
split	A logical vector which indicates whether the plot should show the density curve for 3'UTR, CDS 5'UTR, respectively.
exlevel	A parameter for the annotation filtering. exlevel represents the level that you would like to exclude. NA means no level filtering for the annotation file. The level from the annotations refers to how reliable this annotation is. For more information about level please check https://www.gencodegenes.org/pages/data_format.html.
extranscript_support_level	A parameter for the annotation filtering. extranscript_support_level represents the transcript_support_level that you would like to exclude (e.g. 4 and 5). NA means no transcript_support_level filtering for the annotation file. Transcripts are scored according to how well mRNA and EST alignments match over its full length. Here the number 6 means the transcript_support_level NA. For more information about level please check https://www.gencodegenes.org/pages/data_format.html.
adjust	A parameter inherit from ggplot2. A multiplicate bandwidth adjustment. This makes it possible to adjust the bandwidth while still using the a bandwidth estimator. For example, adjust = 1/2 means use half of the default bandwidth.
nomap	A logical vector. It indicates whether you would like to exclude peaks that cannot assign to annotations in the plot.

Details

Here is an explanation of output meta data in the `list 1`:

`center`: The center position of each peaks. This center position is used for calculating the position of peaks within the genomic regions.

- `location`: Which genomic region this peak belongs to.
- `Gene ID`: Which gene this peak belongs to.
- `Position`: The relative position of each peak. This value close to 0 means this peak located close to the 5' end of the genomic feature. The position value close to one means the peak close to the 3' end of the genomic feature. Value 5 means this peaks can not map to any annotation.

Value

A list object, the list 1 contains the information of the assignment of the peaks and their position value. The position value between 0 to 1 means it located at the 5' UTR, the value close to the 1 means the position of this peak close to the 3' end of the 5' UTR. Peaks located at CDS would have a number between 1 and 2. Postion value between 2 to 3 means this peak assigned to the 3' UTR. For the peaks which can not be assignment to any annotations, they have the value 5. The list 2 includes the plot of meta gene profile.

Author(s)

You Zhou, Kathi Zarnack

Examples

```
## Load the test data and get the path to the test gff3 file
testpath <- system.file("extdata", package = "cliProfiler")
test <- readRDS(file.path(testpath, "test.rds"))
test_gff3 <- file.path(testpath, "annotation_test.gff3")

output <- metaGeneProfile(
    object = test, annotation = test_gff3,
    include_intron = FALSE
)
```

---

motifProfile *motifProfile for the GRanges objects*

Description

An function to plot the frequency or fraction of the interested motif around the center of input peaks.

Usage

```
motifProfile(
    object,
    motif = NA,
    genome = NA,
    fraction = TRUE,
    title = "Motif Profile",
    flanking = 10
)
```

Arguments

object	A GRanges object which should contains all the peaks that you want to check
motif	A character string which use the IUPAC nucleotide code, e.g. DRACH, TTAGGG.
genome	The name of the full genome sequences package in the Bioconductor, e.g. "BSgenome.Mmusculus.UC You should install the package before running this function.
fraction	A logical vector (TRUE or FALSE) that the result should be presented in fraction or number.
title	The main title for the output meta gene profile plot.
flanking	The size of the flanking windows that you would like to check. Flanking=5 will give you the result of the 10+1nt windows around the center of peaks.

Value

A list object, the list 1 contains the information of the frequency of specified motif around the center of peaks. The list 2 includes the plot of motifProfile.

Author(s)

You Zhou, Kathi Zarnack

Examples

```
## Load the test data and get the path to the test gff3 file
testpath <- system.file("extdata", package = "cliProfiler")
test <- readRDS(file.path(testpath, "test.rds"))
test_gff3 <- file.path(testpath, "annotation_test.gff3")

## Please make sure that the correct BSgenome package have installed before
## running motifProfile. For example,library("BSgenome.Mmusculus.UCSC.mm10")
## would be required for the mouse data.

output <- motifProfile(test,
    motif = "DRACH",
    genome = "BSgenome.Mmusculus.UCSC.mm10",
    flanking = 20
)
```

---

spliceSiteProfile *spliceSiteProfile for the GRanges objects*

Description

An function to check the enrichment of peaks around the splice sites in a absolute distance.

Usage

```
spliceSiteProfile(
    object,
    annotation,
    title = "Splice Site Profile",
    exlevel = NA,
    extranscript_support_level = NA,
    exon_length_filtering = TRUE,
    intron_length_filtering = TRUE,
    flanking = 150,
    bin = 30
)
```

Arguments

object	A GRanges object which should contains all the peaks that you want to check
annotation	A path way to the annotation file. The format of the annotation file should be gff3 and downloaded from https://www.gencodegenes.org/
title	The main title for the output meta gene profile plot.
exlevel	A parameter for the annotation filtering. exlevel represents the level that you would like to exclude. NA means no level filtering for the annotation file. The level from the annotations refers to how reliable this annotation is. For more information about level please check https://www.gencodegenes.org/pages/data_format.html.

`extranscript_support_level`
: A parameter for the annotation filtering. extranscript_support_level represents the transcript_support_level that you would like to exclude (e.g. 4 and 5). NA means no transcript_support_level filtering for the annotation file. Transcripts are scored according to how well mRNA and EST alignments match over its full length. Here the number 6 means the transcript_support_level NA. For more information about level please check https://www.gencodegenes.org/pages/data_format.html.

`exon_length_filtering`
: The exon_length_filtering should be a logical value which indicated whether user would like to exclude the exons that have a length less than flanking value. Set this parameter to TRUE to turn on this filtering step.

`intron_length_filtering`
: The intron_length_filtering should be a logical value which indicated whether user would like to exclude the introns that have a length less than flanking value. Set this parameter to TRUE to turn on this filtering step.

`flanking`
: The size of the flanking windows that you would like to check. Flanking=5 will give you the result of the 10+1nt windows around the center of peaks.

`bin`
: A number that indicates how many bins would you like to use in the histogram.

Value

A list object, the list 1 contains the information of the position of peaks around 5' or 3' splice sites. The list 2 includes the plot of spliceSiteProfile

Author(s)

You Zhou, Kathi Zarnack

Examples

```
## Load the test data and get the path to the test gff3 file
testpath <- system.file("extdata", package = "cliProfiler")
test <- readRDS(file.path(testpath, "test.rds"))
test_gff3 <- file.path(testpath, "annotation_test.gff3")

output <- spliceSiteProfile(test, test_gff3,
    flanking = 200, bin = 40
)
```

---

windowProfile *windowProfile for the GRanges objects*

Description

An function to check the position of peaks within the given GRanges windows.

Usage

```
windowProfile(
    object,
    annotation,
    title = "Window Profile",
    group = NA,
    nomap = FALSE
)
```

Arguments

object	A GRanges object which contains all the peaks that you want to check
annotation	A GRanges object that includes the customised genomic region.
title	The main title for the output meta gene profile plot.
group	The column name which contains the information of grouping for making the comparison plot. NA means all the peaks belongs to the same catagory.
nomap	A logical vector (TRUE or FALSE). It indicates whether you would like to exclude peaks that cannot assign to annotations in the plot.

Details

Here is an explanation of output meta data in the list 1:

center: The center position of each peaks. This center position is used for calculating the position of peaks within the genomic regions.

- window_S and window_E: The boundary of the annotation that peaks are assigned.
- window_length: The length of the annotation feature that peak assigned.
- window_map: The relative position of each peak. This value close to 0 means this peak located close to the 5' end of the annotation. The position value close to one means the peak close to the 3' end. Value 3 means this peaks can not map to any annotation.

Value

A list object, the list 1 contains the information of the assignment of the peaks and their position value within the given region. The value close to 1 means the peak close to the end of region in 3' end direction. The list 2 includes the ggplot of windowProfile.

Author(s)

You Zhou, Kathi Zarnack

Examples

```
## Load the test data and get the path to the test gff3 file
testpath <- system.file("extdata", package = "cliProfiler")
test <- readRDS(file.path(testpath, "test.rds"))
test_gff3 <- file.path(testpath, "annotation_test.gff3")
test_gff3 <- rtracklayer::import.gff3(test_gff3)

output <- windowProfile(test, test_gff3)
```

Index