Package 'TFregulomeR'

March 6, 2020

Type Package

Title TFregulomeR reveals transcription factors' context-specific features and functions

Version 2.0.0

Description An R-package linked to a large timely-updated compendium of cistrome and methylome datasets, implemented with functionalities that facilitate the manipulation and analysis of TFBS and methylome meta-data. In particular, TFregulomeR permits the characterisation of TF binding partners and cell-specific TFBSs, along with the study of TF's functions in the context of different partners' combinations and DNA methylation levels.

Depends R (>= 3.6.0), methods

Imports graphics,
grDevices,
utils,
jsonlite (>= 1.5),
ggplot2 (>= 3.0.0),
ggseqlogo (>= 0.1),
gridExtra (>= 2.3),
grid (>= 3.6.0),
IRanges (>= 2.14.12),
S4Vectors (>= 0.18.3),
GenomicRanges (>= 1.32.7),
curl (>= 3.2),
gplots (>= 3.0.1.1)

License GPL-3.0

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

Suggests knitr,
rmarkdown,
rGREAT (>= 1.16.1),
rtracklayer (>= 1.42.1),
rbokeh (>= 0.5.0),
TxDb.Hsapiens.UCSC.hg38.knownGene (>= 3.4.0),
TxDb.Hsapiens.UCSC.hg19.knownGene (>= 3.2.2),
TxDb.Mmusculus.UCSC.mm10.knownGene (>= 3.4.4),
TxDb.Mmusculus.UCSC.mm9.knownGene (>= 3.2.2),
TFBSTools (>= 1.20.0),
GenomeInfoDb (>= 1.18.1),

BiocGenerics (>= 0.28.0),
RUnit,
GenomicFeatures (>= 1.34.2)

biocViews MotifAnnotation, Epigenetics, ChIPSeq, MethylSeq, Transcription, DNAMethylation

VignetteBuilder knitr

BugReports https://github.com/benoukraflab/TFregulomeR/issues

R topics documented:

`cofactorReport` *cofactorReport*

Description

This function allows you to get a PDF report of top cofactors along with DNA methylation for a TF.

Usage

```
cofactorReport(intersectPeakMatrix, top_num = 10,
  cobinding_threshold = 0.05)
```

Arguments

`intersectPeakMatrix`	Output of function 'intersectPeakMatrix()'.
`top_num`	Number of highest co-binding factors to report for each TF (up to 10). By default the number is 10.
`cobinding_threshold`	Only the co-factors with co-binding percentages more than this threshold value will be reported. By default the threshold is 0.05.

Value

A PDF file

Examples

```
peak_id_x <- c("MM1_HSA_K562_CEBPB")
peak_id_y <- c("MM1_HSA_K562_CEBPD", "MM1_HSA_K562_ATF4")
intersect_output <- intersectPeakMatrix(peak_id_x=peak_id_x,
                                        motif_only_for_id_x=TRUE,
                                        peak_id_y=peak_id_y)
cofactorReport(intersectPeakMatrix = intersect_output)
```

---

commonPeakResult *commonPeaks result*

Description

This function allows you to get the reuslts from the commonPeaks() output, including a list of common peak sets, (Meth)Motif logos, methylation profile in common peaks and common peak summary.

Usage

```
commonPeakResult(commonPeaks, return_common_peak_sites = FALSE,
  save_MethMotif_logo = FALSE, return_methylation_profile = FALSE,
  return_summary = FALSE, logo_type = "entropy", meth_level = "all")
```

Arguments

commonPeaks	Required. commonPeaks() output, a matrix of CommonPeaksMM class objects.
return_common_peak_sites	Either TRUE of FALSE (default). If TRUE, a list of data.frames containing common peak sets.
save_MethMotif_logo	Either TRUE of FALSE (default). If TRUE, (Meth)Motif logos for the common peak sets will be saved.
return_methylation_profile	Either TRUE of FALSE (default). If TRUE, the methylation profiles in 200bp window around common peak summits will be returned.
return_summary	Either TRUE of FALSE (default). If TRUE, a numeric matrix containing the percentage of peaks as common will be returned.
logo_type	Logo type for the (Meth)Motif logo to be saved, either "entropy" (default) or "frequency",
meth_level	Methylation level to be plotted for the (Meth)Motif logo, and it should be one of the values, "all" (default), "methylated", and "unmethylated".

Value

a list of data.frames, a numeric matrix or (Meth)Motif logo PDF files depending on the options.

Examples

```
target_id <- c("MM1_HSA_K562_CEBPB")
compared_id <- c("MM1_HSA_HepG2_CEBPB")
commonPeaks_output <- commonPeaks(target_peak_id=target_id,
                                  motif_only_for_target_peak=TRUE,
                                  compared_peak_id=compared_id,
                                  motif_only_for_compared_peak=TRUE,
                                  methylation_profile_in_narrow_region=TRUE)
commonPeaks_result <- commonPeakResult(commonPeaks=commonPeaks_output,
                                       return_common_peak_sites=TRUE,
                                       save_MethMotif_logo=TRUE,
                                       return_methylation_profile=TRUE,
                                       return_summary=TRUE)
```

---

commonPeaks *commonPeaks*

Description

This function allows you to obtain a list of common peak subsets along with the DNA methylation profiles.

Usage

```
commonPeaks(target_peak_id, motif_only_for_target_peak = FALSE,
  user_target_peak_list, user_target_peak_id, compared_peak_id,
  motif_only_for_compared_peak = FALSE, user_compared_peak_list,
  user_compared_peak_id, methylation_profile_in_narrow_region = TRUE,
  motif_type = "MEME", server = "sg", TFregulome_url)
```

Arguments

`target_peak_id` Character of vector, each of which is a TFregulomeR ID. Each of target peak will be compared with all "compared peaks" to get its common subset.

`motif_only_for_target_peak`
Either TRUE of FALSE (default). If TRUE, only peaks with motif will be loaded for each TFregulomeR ID in target_peak_id.

`user_target_peak_list`
A list of data.frames, each of which contains user's own bed-format target peak regions.

`user_target_peak_id`
Character of vector, each of which is a unique ID corresponding to each peak set in the list user_target_peak_list. If the IDs are not provided or not unique, the function will automatically generate the IDs of its own. If any of the peak sets is derived from TFregulomeR, its TFregulomeR ID should be used here correspondingly.

`compared_peak_id`
Character of vector, each of which is a TFregulomeR ID.

`motif_only_for_compared_peak`
Either TRUE of FALSE (default). If TRUE, only peaks with motif will be loaded for each TFregulomeR ID in compared_peak_id.

user_compared_peak_list
: A list of data.frames, each of which contains user's own bed-format compared peak regions.

user_compared_peak_id
: Character of vector, each of which is a unique ID corresponding to each peak set in the list user_compared_peak_list. If the IDs are not provided or not unique, the function will automatically generate the IDs of its own. If any of the peak sets is derived from TFregulomeR, its TFregulomeR ID should be used here correspondingly.

methylation_profile_in_narrow_region
: Either TRUE (default) of FALSE. If TRUE, methylation states in 200bp window surrounding peak summits for each common peak from target_peak_id and user_target_peak_list with TFregulomeR ID.

motif_type
: Motif PFM format, either in MEME by default or TRANSFAC.

server
: server localtion to be linked, either 'sg' or 'ca'.

TFregulome_url
: TFregulomeR server is implemented in MethMotif server. If the MethMotif url is NO more "http://bioinfo-csi.nus.edu.sg/methmotif/" or "http://methmotif.org", please use a new url.

Value

matrix of CommonPeaksMM class objects

Examples

```
target_id <- c("MM1_HSA_K562_CEBPB")
compared_id <- c("MM1_HSA_HepG2_CEBPB")
commonPeaks_output <- commonPeaks(target_peak_id=target_id,
                                  motif_only_for_target_peak=TRUE,
                                  compared_peak_id=compared_id,
                                  motif_only_for_compared_peak=TRUE,
                                  methylation_profile_in_narrow_region=TRUE)
```

---

dataBrowser *Browse the current data available in TFregulomeR*

Description

This function allows you to get the current data in TFregulomeR

Usage

```
dataBrowser(species, organ, sample_type, cell_tissue_name, tf,
  disease_state, source, server = "sg", TFregulome_url)
```

Arguments

species
: The species of interset

organ
: The organ of interset

sample_type
: The sample type of interset

`cell_tissue_name`
: The name of tissue or cell of interset

`tf`
: The TF of interset

`disease_state`
: The disease state of interset

`source`
: The source of interset

`server`
: server localtion to be linked, either 'sg' or 'ca'.

`TFregulome_url`
: TFregulomeR server is implemented in MethMotif server. If the MethMotif url is NO more "http://bioinfo-csi.nus.edu.sg/methmotif/" or "http://methmotif.org", please use a new url.

Value

data.frame containing the information of the queried TFBSs in TFregulomeR

Examples

```
TFBS_brain <- dataBrowser(organ = "brain")
```

---

exclusivePeakResult *exclusivePeaks result*

Description

This function allows you to get the reuslts from the exclusivePeaks() output, including a list of exclusive peak sets, (Meth)Motif logo, methylation profile in exclsive peaks and exclsive peak summary.

Usage

```
exclusivePeakResult(exclusivePeaks, return_exclusive_peak_sites = FALSE,
  save_MethMotif_logo = FALSE, return_methylation_profile = FALSE,
  return_summary = FALSE, logo_type = "entropy", meth_level = "all")
```

Arguments

`exclusivePeaks`
: Required. exclusivePeaks() output, a matrix of ExclusivePeaksMM class objects.

`return_exclusive_peak_sites`
: Either TRUE of FALSE (default). If TRUE, a list of data.frames containing exclusive peak sets will be returned.

`save_MethMotif_logo`
: Either TRUE of FALSE (default). If TRUE, (Meth)Motif logos for the exclusive peak sets will be saved.

`return_methylation_profile`
: Either TRUE of FALSE (default). If TRUE, the methylation profiles in 200bp window around exclusive peak summits will be returned.

`return_summary`
: Either TRUE of FALSE (default). If TRUE, a numeric matrix containing the percentage of peaks as exclusive will be returned.

`logo_type`
: Logo type for the (Meth)Motif logo to be saved, either "entropy" (default) or "frequency".

`meth_level`
: Methylation level to be plotted for the (Meth)Motif logo, and it should be one of the values, "all" (default), "methylated", and "unmethylated".

Value

a list of data.frames, a numeric matrix or (Meth)Motif logo PDF files depending on the options.

Examples

```
target_id <- "MM1_HSA_K562_CEBPB"
excluded_id <- c("MM1_HSA_HepG2_CEBPB", "MM1_HSA_HCT116_CEBPB")
excluPeak_output <- exclusivePeaks(target_peak_id=target_id,
                                   motif_only_for_target_peak=TRUE,
                                   excluded_peak_id=excluded_id,
                                   motif_only_for_excluded_peak=TRUE,
                                   methylation_profile_in_narrow_region=TRUE)
exclusivePeaks_result <- exclusivePeakResult(exclusivePeaks=excluPeak_output,
                                             return_exclusive_peak_sites=TRUE,
                                             save_MethMotif_logo=TRUE,
                                             return_summary=TRUE)
```

`exclusivePeaks` *exclusivePeaks*

Description

This function allows you to obtain a list of exclusive peak subsets along with the DNA methylation profiles.

Usage

```
exclusivePeaks(target_peak_id, motif_only_for_target_peak = FALSE,
  user_target_peak_list, user_target_peak_id, excluded_peak_id,
  motif_only_for_excluded_peak = FALSE, user_excluded_peak_list,
  user_excluded_peak_id, methylation_profile_in_narrow_region = TRUE,
  motif_type = "MEME", server = "sg", TFregulome_url)
```

Arguments

`target_peak_id` Character of vector, each of which is a TFregulomeR ID. Each of target peak will be compared with all "excluded peaks" to get its exclusive subset.

`motif_only_for_target_peak`
Either TRUE of FALSE (default). If TRUE, only peaks with motif will be loaded for each TFregulomeR ID in target_peak_id.

`user_target_peak_list`
A list of data.frames, each of which contains user's own bed-format target peak regions.

`user_target_peak_id`
Character of vector, each of which is a unique ID corresponding to each peak set in the list user_target_peak_list. If the IDs are not provided or not unique, the function will automatically generate the IDs of its own. If any of the peak sets is derived from TFregulomeR database, its TFregulomeR ID should be used here correspondingly.

`excluded_peak_id`
Character of vector, each of which is a TFregulomeR ID.

motif_only_for_excluded_peak
: Either TRUE of FALSE (default). If TRUE, only peaks with motif will be loaded for each TFregulomeR ID in excluded_peak_id.

user_excluded_peak_list
: A list of data.frames, each of which contains user's own bed-format excluded peak regions.

user_excluded_peak_id
: Character of vector, each of which is a unique ID corresponding to each peak set in the list user_excluded_peak_list. If the IDs are not provided or not unique, the function will automatically generate the IDs of its own. If any of the peak sets is derived from TFregulomeR database, its TFregulomeR ID should be used here correspondingly.

methylation_profile_in_narrow_region
: Either TRUE (default) of FALSE. If TRUE, methylation states in 200bp window surrounding peak summits for each exclusive peak from target_peak_id and user_target_peak_list (with TFregulomeR ID).

motif_type
: Motif PFM format, either in MEME by default or TRANSFAC.

server
: server localtion to be linked, either 'sg' or 'ca'.

TFregulome_url
: TFregulomeR server is implemented in MethMotif server. If the MethMotif url is NO more "http://bioinfo-csi.nus.edu.sg/methmotif/" or "http://methmotif.org", please use a new url.

Value

matrix of ExclusivePeaksMM class objects

Examples

```
target_id <- "MM1_HSA_K562_CEBPB"
excluded_id <- c("MM1_HSA_HepG2_CEBPB", "MM1_HSA_HCT116_CEBPB")
excluPeak_output <- exclusivePeaks(target_peak_id=target_id,
                                   motif_only_for_target_peak=TRUE,
                                   excluded_peak_id=excluded_id,
                                   motif_only_for_excluded_peak=TRUE,
                                   methylation_profile_in_narrow_region=TRUE)
```

---

exportMMPFM *export motif position frequency matrix and beta score matrix*

Description

This function allows you to export motif position frequency matrix and beta score matrix from the output of " searchMotif", "commonPeaks", "exclusivePeaks" or "intersectPeakMatrix".

Usage

```
exportMMPFM(fun_output, fun, save_motif_PFM = FALSE,
  save_betaScore_matrix = FALSE,
  angle_of_matrix_for_intersectPeakMatrix = "x",
  saving_id_x_for_intersectPeakMatrix = "all",
  saving_id_y_for_intersectPeakMatrix = "all")
```

Arguments

fun_output	Required. Output from "searchMotif", "commonPeaks", "exclusivePeaks" or "intersectPeakMatrix".
fun	Required. The function that was used to get the output and should be one of the options, 'searchMotif', 'commonPeaks', 'exclusivePeaks' and 'intersectPeakMatrix'.
save_motif_PFM	Either TRUE or FALSE (default). If "TRUE", the motif position frequency matrix will be saved.
save_betaScore_matrix	Either TRUE or FALSE (default). If "TRUE", the beta score matrix will be saved.
angle_of_matrix_for_intersectPeakMatrix	Only applicable when "fun='intersectPeakMatrix'". Either "x" (default) or "y". If "x", motif PFM for the peak sets in "peak_list_x" intersected with "peak_list_y" will be saved; if "y", motif PFM for the peak sets in "peak_list_y" intersected with "peak_list_x" will be saved.
saving_id_x_for_intersectPeakMatrix	Only applicable when "fun='intersectPeakMatrix'". Either "all" (default) or a subset of "peak_id_x". If a subset of "peak_id_x" is provided, only the MethMotif logos for them will be saved.
saving_id_y_for_intersectPeakMatrix	Only applicable when "fun='intersectPeakMatrix'". Either "all" (default) or a subset of "peak_id_y". If a subset of "peak_id_y" is provided, only the MethMotif logos for them will be saved.

Value

motif position frequency matrix file and beta score matrix file

Examples

```
methmotif_cebpb <- searchMotif(id = "MM1_HSA_K562_CEBPB")
exportMMPFM(fun_output = methmotif_cebpb, fun = "searchMotif",
            save_motif_PFM = TRUE, save_betaScore_matrix = TRUE)
```

---

genomeAnnotate *genomeAnnotate*

Description

This function allows you to annotate genomic locations of cis-regulatory regions.

Usage

```
genomeAnnotate(peaks, assembly = "hg38", return_annotation = FALSE,
  return_html_report = FALSE, promoter_range = c(-1000, 100),
  TTS_range = c(-100, 1000), server = "sg", TFregulome_url)
```

Arguments

peaks	Required. A bed-format genomic regions in data frame.
assembly	The genome assembly of the input regions, currently supporting 'hg19', 'hg38' (default), 'mm9' and 'mm10'.
return_annotation	Either TRUE of FALSE (default). If TRUE, a data.frame containing annotation results will be returned.
return_html_report	Either TRUE of FALSE (default). If TRUE, a dynamic HTML report will be saved.
promoter_range	A numeric vector to define promoter range. By default, c(-1000, 100) defines promoters as 1000bp upstream and 100bp downstream of TSS.
TTS_range	A numeric vector to define TTS range. By default, c(-100, 1000) defines promoters as 100bp upstream and 1000bp downstream of real TTS.
server	server localtion to be linked, either 'sg' or 'ca'.
TFregulome_url	TFregulomeR server is implemented in MethMotif server. If the MethMotif url is NO more "http://bioinfo-csi.nus.edu.sg/methmotif/" or "http://methmotif.org", please use a new url.

Value

a data.frame, or an HTML report depending on the options.

Examples

```
require(TxDb.Hsapiens.UCSC.hg38.knownGene)
HCT116_CEBPB_regions <- loadPeaks(id = "MM1_HSA_HCT116_CEBPB", includeMotifOnly=TRUE)
HCT116_CEBPB_regions_annotation <- genomeAnnotate(peaks = HCT116_CEBPB_regions[1:10,],
                              return_annotation = TRUE, return_html_report = TRUE)
```

---

greatAnnotate *greatAnnotate*

Description

This function allows you to analyse the gene ontologies of targeting genes by cis-regulatory regions.

Usage

```
greatAnnotate(peaks, assembly = "hg38", return_annotation = FALSE,
  return_html_report = FALSE, pvalue = 0.01, test = "binomial",
  great_rule = "basalPlusExt", great_adv_upstream = 5,
  great_adv_downstream = 1, great_adv_span = 1000,
  great_adv_twoDistance = 1000, great_adv_oneDistance = 1000,
  request_interval = 60, great_version = "4.0")
```

Arguments

peaks	Required. A bed-format genomic regions in data frame.
assembly	The genome assembly of the input regions, currently supporting 'hg19', 'hg38' (default), 'mm9' and 'mm10'.
return_annotation	Either TRUE of FALSE (default). If TRUE, a data.frame containing annotation results will be returned.
return_html_report	Either TRUE of FALSE (default). If TRUE, a dynamic HTML report will be saved.
pvalue	The adjusted p-value which is applied to filter the results, by default 0.01.
test	The statistical test used in GREAT analysis, either 'binomial' (default) or 'hypergeometric'.
great_rule	Equivalent to the rGREAT input 'rule', 'basalPlusExt' (default, basal plus extension), 'twoClosest' (two nearest genes), or 'oneClosest' (single nearest gene).
great_adv_upstream	Equivalent to the rGREAT input 'adv_upstream' (upstream extension, kb). Only applicable when 'great_rule' is 'basalPlusExt', by default 5.0 (kb).
great_adv_downstream	Equivalent to the rGREAT input 'adv_downstream' (downstream extension, kb). Only applicable when 'great_rule' is 'basalPlusExt', by default 1.0 (kb).
great_adv_span	Equivalent to the rGREAT input 'adv_span' (maximal distal region). Only applicable when 'great_rule' is 'basalPlusExt', by default 1000.0 (kb).
great_adv_twoDistance	Equivalent to the rGREAT input 'adv_twoDistance' (region range to be considered). Only applicable when 'great_rule' is 'twoClosest', by default 1000.0 (kb).
great_adv_oneDistance	Equivalent to the rGREAT input 'adv_oneDistance' (region range to be considered). Only applicable when 'great_rule' is 'oneClosest', by default 1000.0 (kb).
request_interval	The minimal gap time between two requests using greatAnnotate, by default 60 (s).
great_version	Equivalent to the rGREAT input 'version', by default 4.0.

Value

a data.frame, or an HTML report depending on the options.

Examples

```
require(rGREAT)
require(rbokeh)
K562_CEBPB_regions <- loadPeaks(id = "MM1_HSA_K562_CEBPB")
K562_CEBPB_regions_annotation <- greatAnnotate(peaks = K562_CEBPB_regions[1:500,],
                            return_annotation = TRUE, return_html_report = TRUE)
```

interactome3D *interactome3D*

Description

This function allows you to get a html report of a 3D dynamic TF interactome with CpG methylation and external source signal.

Usage

```
interactome3D(intersectPeakMatrix, return_interactome_with_mCpG = FALSE,
  mCpG_threshold = 0.8,
  return_interactome_with_external_source = FALSE,
  external_source_value = "median", angle_of_matrix = "x")
```

Arguments

intersectPeakMatrix	Output of function 'intersectPeakMatrix()'.
return_interactome_with_mCpG	Either TRUE of FALSE (default). If TRUE, html report of TF interactome with mCpG portion will be saved.
mCpG_threshold	A mininum beta score to determine CpG methylation. Should be 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8 (default), or 0.9
return_interactome_with_external_source	Either TRUE of FALSE (default). If TRUE, html report of TF interactome with external source signal will be saved.
external_source_value	The value of external source signal in the intersected peaks. It should be one of the following values: "median" (default),"mean","SD","quartile_25","quartile_75".
angle_of_matrix	Either "x" (default) or "y". If "x", will focus on the peak sets in "peak_list_x" intersected with "peak_list_y"; if "y", will focus on peak sets in "peak_list_y" intersected with "peak_list_x".

Value

An html file

intersectPeakMatrix *intersectPeakMatrix*

Description

This function allows you to obtain the pair-wise intersected regions, along with the DNA methylation profiles, between two lists of peak sets, as well as (Meth)Motif logos

Usage

```
intersectPeakMatrix(peak_id_x, motif_only_for_id_x = FALSE,
  user_peak_list_x, user_peak_x_id, peak_id_y,
  motif_only_for_id_y = FALSE, user_peak_list_y, user_peak_y_id,
  methylation_profile_in_narrow_region = FALSE, external_source,
  motif_type = "MEME", server = "sg", TFregulome_url)
```

Arguments

peak_id_x	Character of vector, each of which is a unique TFregulomeR ID.
motif_only_for_id_x	Either TRUE of FALSE (default). If TRUE, only peaks with motif will be loaded for each TFregulomeR ID in peak_id_x.
user_peak_list_x	A list of data.frames, each of which contains user's own bed-format peak regions for peak list x.
user_peak_x_id	Character of vector, each of which is a unique ID corresponding to each peak set in the list user_peak_list_x. If the IDs are not provided or not unique, the function will automatically generate the IDs of its own. If any of the peak sets is derived from TFregulomeR, its TFregulomeR ID should be used here correspondingly.
peak_id_y	Character of vector, each of which is a unique TFregulomeR ID.
motif_only_for_id_y	Either TRUE of FALSE (default). If TRUE, only peaks with motif will be loaded for each TFregulomeR ID in peak_id_y.
user_peak_list_y	A list of data.frames, each of which contains user's own bed-format peak regions for peak list y.
user_peak_y_id	Character of vector, each of which is a unique ID corresponding to each peak set in the list user_peak_list_y. If the IDs are not provided or not unique, the function will automatically generate the IDs of its own. If any of the peak sets is derived from TFregulomeR, its TFregulomeR ID should be used here correspondingly.
methylation_profile_in_narrow_region	Either TRUE (default) of FALSE. If TRUE, methylation states in 200bp window surrounding peak summits for each intersected peak pair from peak_id_x (peak_id_y) and user_peak_list_x (user_peak_list_y) with TFregulomeR ID.
external_source	a bed-like data.frame files with the fourth column as the score to be profiled in pairwise comparison regions.
motif_type	Motif PFM format, either in MEME by default or TRANSFAC.
server	server localtion to be linked, either 'sg' or 'ca'.
TFregulome_url	TFregulomeR server is implemented in MethMotif server. If the MethMotif url is NO more "http://bioinfo-csi.nus.edu.sg/methmotif/" or "http://methmotif.org", please use a new url.

Value

matrix of IntersectPeakMatrix class objects

Examples

```
peak_id_x <- c("MM1_HSA_K562_CEBPB", "MM1_HSA_HCT116_CEBPB")
peak_id_y <- c("MM1_HSA_HepG2_CEBPB", "MM1_HSA_HCT116_CEBPB")
intersectPeakMatrix_output <- intersectPeakMatrix(peak_id_x=peak_id_x,
                                                  motif_only_for_id_x=TRUE,
                                                  peak_id_y=peak_id_y,
                                                  motif_only_for_id_y=TRUE)
```

---

intersectPeakMatrixResult

intersectPeakMatrix result

Description

This function allows you to get the reuslts from the intersectPeakMatrix() output, including a matrix of the pair-wise intersecting percentages between two lists of peak sets, DNA methylation profiles in the intersected regions (for peaks from TFregulomeR) and (Meth)Motif logos for each pair of intersections (for peaks from TFregulomeR).

Usage

```
intersectPeakMatrixResult(intersectPeakMatrix,
  return_intersection_matrix = FALSE, angle_of_matrix = "x",
  return_tag_density = FALSE, angle_of_tag_density = "x",
  tag_density_value = "median", return_external_source = FALSE,
  angle_of_external_source = "x", external_source_value = "median",
  return_methylation_profile = FALSE,
  angle_of_methylation_profile = "x", save_MethMotif_logo = FALSE,
  angle_of_logo = "x", logo_type = "entropy", meth_level = "all",
  saving_MethMotif_logo_x_id = "all",
  saving_MethMotif_logo_y_id = "all")
```

Arguments

`intersectPeakMatrix`
: Required. intersectPeakMatrix() output, a marix of IntersectPeakMatrix class objects.

`return_intersection_matrix`
: Either TRUE of FALSE (default). If TRUE, a matrix of the pair-wise intersecting percentages between two lists of peak sets will be returned.

`angle_of_matrix`
: Either "x" (default) or "y". If "x", a matrix denoting the percentages of peak sets in "peak_list_x" intersected with "peak_list_y" will be returned; if "y", a matrix denoting the percentages of peak sets in "peak_list_y" intersected with "peak_list_x" will be returned.

`return_tag_density`
: Either TRUE of FALSE (default). If TRUE, a matrix of tag density values in intersected peaks between "peak_list_x" and "peak_list_y" will be returned.

`angle_of_tag_density`
: Either "x" (default) or "y". If "x", a matrix denoting tag density values in "peak_list_x" intersected with "peak_list_y" will be returned; if "y", a matrix denoting tag density values in "peak_list_y" intersected with "peak_list_x" will be returned.

`tag_density_value`
: The value of tag density in the intersected peaks. It should be one of the following values: "median" (default),"mean","SD","quartile_25","quartile_75".

`return_external_source`
: Either TRUE of FALSE (default). If TRUE, a matrix of external source values in intersected peaks between "peak_list_x" and "peak_list_y" will be returned.

`angle_of_external_source`
: Either "x" (default) or "y". If "x", a matrix denoting external source values in "peak_list_x" intersected with "peak_list_y" will be returned; if "y", a matrix denoting tag density values in "peak_list_y" intersected with "peak_list_x" will be returned.

`external_source_value`
: The value of external source signal in the intersected peaks. It should be one of the following values: "median" (default),"mean","SD","quartile_25","quartile_75".

`return_methylation_profile`
: Either TRUE of FALSE (default). If TRUE, a matrix of DNA methylation state data in the intersected regions will be returned.

`angle_of_methylation_profile`
: Either "x" (default) or "y". If "x", a matrix denoting DNA methylation state data in "peak_list_x" intersected with "peak_list_y" will be returned; if "y", a matrix denoting DNA methylation state data in "peak_list_y" intersected with "peak_list_x" will be returned.

`save_MethMotif_logo`
: Either TRUE of FALSE (default). If TRUE, (Meth)Motif logos for the intersected peaks will be saved.

`angle_of_logo`
: Either "x" (default) or "y". If "x", (Meth)Motif logos for the peak sets in "peak_list_x" intersected with "peak_list_y" will be saved; if "y", (Meth)Motif logos for the peak sets in "peak_list_y" intersected with "peak_list_x" will be saved.

`logo_type`
: Logo type for the (Meth)Motif logo to be saved, either "entropy" (default) or "frequency".

`meth_level`
: Methylation level to be plot for the (Meth)Motif logo, and it should be one of the values, "all" (default), "methylated", and "unmethylated".

`saving_MethMotif_logo_x_id`
: Either "all" (default) or a subset of "peak_id_x". If a subset of "peak_id_x" is provided, only the (Meth)Motif logos for them will be saved.

`saving_MethMotif_logo_y_id`
: Either "all" (default) or a subset of "peak_id_y". If a subset of "peak_id_y" is provided, only the (Meth)Motif logos for them will be saved.

Value

a matrix of pair-wise intersecting percantages between two lists of peak sets, a matrix of DNA methylation data in the intersected regions or (Meth)Motif PDF files depending on the options

Examples

```
peak_id_x <- c("MM1_HSA_K562_CEBPB", "MM1_HSA_HCT116_CEBPB")
peak_id_y <- c("MM1_HSA_HepG2_CEBPB", "MM1_HSA_HCT116_CEBPB")
intersect_output <- intersectPeakMatrix(peak_id_x=peak_id_x,
                                                    motif_only_for_id_x=TRUE,
                                                    peak_id_y=peak_id_y,
                                                    motif_only_for_id_y=TRUE)
intersect_matrix <- intersectPeakMatrixResult(intersectPeakMatrix=intersect_output,
                                              return_intersection_matrix=TRUE,
                                              save_MethMotif_logo=TRUE,
                            saving_MethMotif_logo_x_id=c("MM1_HSA_K562_CEBPB"))
```

`loadPeaks` *load peaks from TFregulomeR*

Description

This function allows you to obtain the peaks from TFregulomeR using TFregulomeR ID.

Usage

```
loadPeaks(id, includeMotifOnly = FALSE, server = "sg", TFregulome_url)
```

Arguments

`id`	Required. TFregulomeR ID
`includeMotifOnly`	Either TRUE or FALSE (default). If TRUE, only peaks with motif will be returned
`server`	server localtion to be linked, either 'sg' or 'ca'.
`TFregulome_url`	TFregulomeR server is implemented in MethMotif server. If the MethMotif url is NO more "http://bioinfo-csi.nus.edu.sg/methmotif/" or "http://methmotif.org", please use a new url.

Value

a data.frame containing peak coordinates

Examples

```
CEBPB_peaks <- loadPeaks(id = "MM1_HSA_K562_CEBPB")
```

`motifDistrib` *profile TFBS distribution*

Description

This function allows you to profile TFBS distributions in a given list of peak sets.

Usage

```
motifDistrib(id, peak_list, peak_id, plot_at_each_side = 100,
  server = "sg", TFregulome_url)
```

Arguments

`id`	Required. TFregulomeR ID. The TFBS of interest to be profiled.
`peak_list`	Required. List of data.frames, each of which contains bed-format peak regions. They are the peak sets in which you want to profile the TFBS distributions, and can be loaded from TFregulomeR database or self-provided.
`peak_id`	Required. Character of vector, each of which is a unique ID corresponding to the element in "peak_list". If a peak set is orignally from TFregulomeR Database, its TFregulomeR ID should be used here.
`plot_at_each_side`	By default 100bp, and motif occurrences in a window of +/- 100bp around peak centres will be returned.
`server`	server localtion to be linked, either 'sg' or 'ca'.
`TFregulome_url`	TFregulomeR server is implemented in MethMotif server. If the MethMotif url is NO more "http://bioinfo-csi.nus.edu.sg/methmotif/" or "http://methmotif.org", please use a new url.

Value

a list containing the numbers of input peaks and peaks with motif, as well as motif occurrences in the plot window.

Examples

```
CEBPB_peaks <- loadPeaks(id = "MM1_HSA_K562_CEBPB")
motifDistrib_output <- motifDistrib(id = "MM1_HSA_K562_CEBPB",
                                    peak_list = list(CEBPB_peaks[1:100,]),
                                    peak_id = "MM1_HSA_K562_CEBPB")
```

plotDistrib *plot TFBS distribution from motifDistrib() output*

Description

This function allows you to plot TFBS distributions in a given list of peak sets from the output of motifDistrib().

Usage

```
plotDistrib(motifDistrib)
```

Arguments

motifDistrib	Required. motifDistrib() output.

Value

TFBS distribution PDF file.

Examples

```
CEBPB_peaks <- loadPeaks(id = "MM1_HSA_K562_CEBPB")
motifDistrib_output <- motifDistrib(id = "MM1_HSA_K562_CEBPB",
                                    peak_list = list(CEBPB_peaks[1:100,]),
                                    peak_id = c("MM1_HSA_K562_CEBPB"))
plotDistrib(motifDistrib = motifDistrib_output)
```

plotLogo *plot (Meth)Motif logo*

Description

This function allows you to plot (Meth)Motif logo.

Usage

```
plotLogo(MM_object, logo_type = "entropy", meth_level = "all")
```

Arguments

MM_object	Required. MethMotif class object
logo_type	Logo type for the (Meth)Motif logo to be saved, either "entropy" (default) or "frequency".
meth_level	Methylation level to be plot for the (Meth)Motif logo, and it should be one of the values, "all" (default), "methylated", and "unmethylated".

Value

(Meth)Motif logo pdf file. If the TFregulomeR peak source is from MethMotif, MethMotif logo will be saved; if the source is GTRD, only motif logo will be saved.

Examples

```
K562_CEBPB <- searchMotif(id = "MM1_HSA_K562_CEBPB")
plotLogo(MM_object = K562_CEBPB)
```

`searchMotif` *Search motif PFM and beta score matrix (if source is MethMotif) for a given TFregulomeR ID in TFregulomeR*

Description

This function allows you to obtain motif PFM matrix and beta score matrix (if source is MethMotif) for a given TFregulomeR ID in TFregulomeR

Usage

```
searchMotif(id, motif_format = "MEME", server = "sg", TFregulome_url)
```

Arguments

`id`	Required. TFregulomeR ID.
`motif_format`	Motif PFM format, either in MEME by default or TRANSFAC.
`server`	server localtion to be linked, either 'sg' or 'ca'.
`TFregulome_url`	TFregulomeR server is implemented in MethMotif server. If the MethMotif url is NO more "http://bioinfo-csi.nus.edu.sg/methmotif/" or "http://methmotif.org", please use a new url.

Value

MethMotif class object

Examples

```
K562_CEBPB <- searchMotif(id = "MM1_HSA_K562_CEBPB")
K562_CEBPB_transfac <- searchMotif(id = "MM1_HSA_K562_CEBPB",
                                   motif_format = "TRANSFAC")
```

`toTFBSTools` *convert motif PFM in TFregulomeR into PFMatrix class object in TFBSTools package*

Description

This function allows you to retrieve and convert motif PFM in TFregulomeR database into PFMatrix class object, which can be used in TFBSTools package.

Usage

```
toTFBSTools(id, server = "sg", TFregulome_url)
```

Arguments

id	Required. TFregulomeR ID.
server	server localtion to be linked, either 'sg' or 'ca'.
TFregulome_url	TFregulomeR server is implemented in MethMotif server. If the MethMotif url is NO more "http://bioinfo-csi.nus.edu.sg/methmotif/" or "http://methmotif.org", please use a new url.

Value

An object of class PFMatrix

Examples

```
require(TFBSTools)
CEBPB_pfm <- toTFBSTools(id = "MM1_HSA_K562_CEBPB")
```

Index