Package 'PhosMap'

May 24, 2019

Type Package

Title A Comprehensive R Package For Analyzing Quantitative Phosphoproteomics Data

Version 0.99.33

Author Dongdong Zhan [aut, cre], Mengsha Tong [aut, cre]

Maintainer Dongdong Zhan <ecnuzdd@163.com>

Description PhosMap is a comprehensive R package for analyzing quantitative phosphoproteomics data. Modules in PhosMap were classified into two major categories: (1) data pre-processing and (2) data analysis and presentation. An intact data pre-processing procedure of phosphoproteomics data covered three main steps: merging input files after quality control, mapping phosphorylation sites (p-sites) to the corresponding protein sequence and data normalization. PhosMap incorporated four analysis modules, including clustering and differential expression analysis, time course analysis, kinase activity prediction to find activated/deactivated kinases and motif enrichment analysis.

License GPL (>= 2)

Encoding UTF-8

LazyData FALSE

RoxygenNote 6.1.1

Date 2019-05-06

biocViews Proteomics, DifferentialExpression, DataRepresentation, Visualization, Clustering, Normalization, QualityControl, TimeCourse

Depends R (>= 3.3)

Imports graphics, grDevices, stats, utils, stringr, ggseqlogo, samr, limma, e1071, ClueR, Rtsne, glmnet, yaml, impute, RCurl

Suggests knitr, rmarkdown, pheatmap

VignetteBuilder knitr

NeedsCompilation no

R topics documented:

`analysis_deps_anova` *Differential expression analysis using ANOVA*

Description

Differential expression analysis using ANOVA

Usage

```
analysis_deps_anova(expr_data_frame, group, log2_label = FALSE,
  return_padjust = TRUE, adjust_method = "BH")
```

Arguments

`expr_data_frame`	A data frame containing ID and quantification values.
`group`	A factor representing experimental groups.
`log2_label`	A boolean value for representing whether the value is logarithmic or not, the default is FALSE.
`return_padjust`	A boolean value for representing whether or not the p value is adjusted, the default is TRUE.
`adjust_method`	Method used to adjust the p-values for multiple testing. See p.adjust for the complete list of options, the default is "BH".

Value

A data frame containing ID, log2(FC) and p value.

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/analysis_deps_anova.RData"
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "analysis_deps_anova.RData")
load("analysis_deps_anova.RData")

anova_result <- analysis_deps_anova(
  expr_data_frame, group, log2_label = FALSE,
  return_padjust = TRUE, adjust_method = 'BH'
)
head(anova_result)

}
```

analysis_deps_limma *Differential expression analysis using limma.*

Description

Differential expression analysis using limma.

Usage

```
analysis_deps_limma(expr_data_frame, group, comparison_factor,
  log2_label = FALSE, adjust_method = "BH")
```

Arguments

expr_data_frame	A data frame containing ID and quantification values.
group	A factor for representing experimental groups.
comparison_factor	A vector for comparison groups.
log2_label	A boolean value for representing whether the value is logarithmic or not, the default is FALSE.
adjust_method	Method used to adjust the p-values for multiple testing. See p.adjust for the complete list of options, the default is "BH"

Value

A list containing results from limma analysis.

Author(s)

Dongdong Zhan and Mengsha Tong

References

Ritchie, M.E., Phipson, B., Wu, D., Hu, Y., Law, C.W., Shi, W., and Smyth, G.K. (2015). limma powers differential expression \ analyses for RNA-sequencing and microarray studies. Nucleic Acids Research 43(7), e47.

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/analysis_deps_limma.RData"
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "analysis_deps_limma.RData")
load("analysis_deps_limma.RData")

limma_results_df <- analysis_deps_limma(
  expr_data_frame, group, group_levels,
  log2_label = FALSE, adjust_method = 'none'
)
```

```
head(limma_results_df)

}
```

`analysis_deps_sam` *Differential expression analysis using SAM*

Description

Differential expression analysis using SAM

Usage

```
analysis_deps_sam(expr_data_frame, group, log2_label = FALSE, nperms = 100,
  rand = NULL, minFDR = 0.05, samr_plot = TRUE)
```

Arguments

`expr_data_frame`	A data frame containing ID and quantification values.
`group`	A factor representing experimental groups.
`log2_label`	A boolean value for representing whether the value is logarithmic or not, the default is FALSE.
`nperms`	Number of permutations used to estimate false discovery rates.
`rand`	if specified, the random number generator will be put in a reproducible state.
`minFDR`	A numeric value for filtering significant genes, the default is 0.05.
`samr_plot`	A boolean value for representing whether samr graph is plotted or not.

Value

A list containing results from sam analysis.

Author(s)

Dongdong Zhan and Mengsha Tong

References

R. Tibshirani, G. Chu, T. Hastie and Balasubramanian Narasimhan (2010). samr: SAM: Significance Analysis of Microarrays.\ Rpackage version 1.28. https://CRAN.R-project.org/package=samr

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/analysis_deps_sam.RData"
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "analysis_deps_sam.RData")
load("analysis_deps_sam.RData")
```

```
sam_results_list <- analysis_deps_sam(
  expr_data_frame, group, log2_label = FALSE,
  nperms = 100, rand = NULL, minFDR = 0.05,samr_plot = TRUE
)
head(sam_results_list)

}
```

---

check_mea_input *Check input for motif enrichment analysis (mea)*

Description

Check input for motif enrichment analysis (mea)

Usage

```
check_mea_input(foreground, background, center)
```

Arguments

foreground	A vector for AA sequences with fixed length as foreground input.
background	A vector for AA sequences with fixed length as background input.
center	A character for center of k-mer.

Value

A list passing check steps

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/check_mea_input.RData"
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "check_mea_input.RData")
load("check_mea_input.RData")

check_result_list <- check_mea_input(
  foreground[1:100],
  background[1:100],
  center
)
head(check_result_list)
}
```

compute_kses *computing kinase-substrate enrichment score*

Description

computing kinase-substrate enrichment score

Usage

```
compute_kses(substate_vector, regulons_of_kinase,
  substrates_of_kinase_in_exp_count)
```

Arguments

substate_vector	a vector for substrates with values indentified in current experiments.
regulons_of_kinase	a vector for substrates of a specific kinase, which with substrates identified in current experiments.
substrates_of_kinase_in_exp_count	a numeric for numbers in regulons_of_kinase vector.

Value

A numeric or NA for enrichment_score.

Author(s)

Dongdong Zhan and Mengsha Tong

References

Hernandez-Armenta C et al. Benchmarking substrate-based kinase activity inference using phosphoproteomic data[J]. Bioinformatics, 2017, 33(12): 1845-1851.

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/compute_kses.RData"
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "compute_kses.RData")
load("compute_kses.RData")

stochastic_enrichment_score_i <- compute_kses(
  substate_vector,
  regulons_of_kinase_i,
  substrates_of_kinase_in_exp_count
)
head(stochastic_enrichment_score_i)
}
```

construct_pwm *Construct position weight matrix*

Description

Construct position weight matrix

Usage

```
construct_pwm(sequences, width, frequency_flag = TRUE)
```

Arguments

sequences	A vector for aligned sequences with fixed length.
width	A numeric for specific k-mer.
frequency_flag	A boolean for showing real frequency or frequency probability, the default is TRUE for showing real frequency.

Value

A position weight matrix.

Author(s)

Dongdong Zhan and Mengsha Tong

References

Wagih O, Sugiyama N, Ishihama Y, et al. Uncovering phosphorylation-based specificities through functional interaction networks[J]. Molecular & Cellular Proteomics, 2016, 15(1): 236-245.

Examples

```
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/construct_pwm.RData"
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "construct_pwm.RData")
load("construct_pwm.RData")

foreground_pwm <- construct_pwm(
 foreground_sequence,
 width,
 frequency_flag = TRUE
)
head(foreground_pwm)
```

extract_psites_score *Create R codes to call python for parsering mascot xml.*

Description

Extract the confidence probability of phosphorylation sites (psites) from mascot xml. One file containing experiment codes and one folder containing mascot xml as input, the another folder is required as output. Python is required and the corresponding xml package is also required.

Usage

```
extract_psites_score(phosphorylation_exp_design_info_file_path, mascot_xml_dir,
  mascot_txt_dir)
```

Arguments

phosphorylation_exp_design_info_file_path	A string representing the file path of experiment code, for example: experiment_code.txt
mascot_xml_dir	A folder containing identification xml files searched by Mascot as input, for example: Exp020901_F1_R1.xml
mascot_txt_dir	A folder used for saving files which contains the confidence of phosphorylation sites, for example: Exp020901_F1_R1.txt

Value

A series of output file saved in the mascot_txt_dir

Examples

```
## Not run:
extract_psites_score(
  phosphorylation_exp_design_info_file_path,
  mascot_xml_dir,
  mascot_txt_dir
)

## End(Not run)
```

fore_seq_to_motif *Convert the list that consists of motifs and the corresponding sequences to data frame.*

Description

Convert the list that consists of motifs and the corresponding sequences to data frame.

Usage

```
fore_seq_to_motif(foreground_sequences_mapped_to_motifs)
```

Arguments

foreground_sequences_mapped_to_motifs
: A list that consists of motifs and the corresponding sequences.

Value

A data frame that consist of aligned sequences and the corresponding motifs.

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/fore_seq_to_motif.RData"
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "fore_seq_to_motif.RData")
load("fore_seq_to_motif.RData")

df <- fore_seq_to_motif(
  foreground_sequences_mapped_to_motifs
)
head(df)

}
```

---

formatted_output_mef_results

Output formatted sequences in foreground that are mapped to specific motifs.

Description

Output formatted sequences in foreground that are mapped to specific motifs.

Usage

```
formatted_output_mef_results(foreground_sequences_mapped_to_motifs)
```

Arguments

foreground_sequences_mapped_to_motifs
: A list that consists of motifs and their corresponding aligned sequences from foreground.

Value

A data frame that motifs and their corresponding aligned sequences from foreground.

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/formatted_output_mef_results.
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "formatted_output_mef_results.RData")
load("formatted_output_mef_results.RData")

formatted_output_df <- formatted_output_mef_results(
  foreground_sequences_mapped_to_motifs
)
head(formatted_output_df)
```

---

generate_psites_score_file

Generate peptide identification files with psites scores.

Description

Based on mascot txt files with psites and peptide identification files downloaded from Firmiana, the file with phosphorylation modifications is generated.

Usage

```
generate_psites_score_file(mascot_txt_dir, firmiana_peptide_dir,
  psites_score_dir)
```

Arguments

`mascot_txt_dir`	A folder containing identification xml files with psites scores as input.
`firmiana_peptide_dir`	A folder containing identification txt files downloaded from Firmiana as input.
`psites_score_dir`	A folder used for saving files of peptide identification files with psites scores

Value

A series of output files saved in the psites_score_dir

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
## Not run:
generate_psites_score_file(mascot_txt_dir, firmiana_peptide_dir, psites_score_dir)

## End(Not run)
```

get_aligned_seq_for_mea *Taking S/T/Y as the center, align sequence to fasta library by the specific length.*

Description

Taking S/T/Y as the center, align sequence to fasta library by the specific length.

Usage

```
get_aligned_seq_for_mea(ID, Sequence, AA_in_protein, fixed_length,
  species = "human", fasta_type = "refseq")
```

Arguments

ID	A vector for gi number of proteins.
Sequence	A vector for sequence of peptides.
AA_in_protein	A vector for the locations of S/T/Y in sequence of proteins.
fixed_length	Length of aligned sequence,the default is 15.
species	A string for the library of species, the options are human, mouse and rattus, the default is human.
fasta_type,	A string for fasta source, the options are refseq and uniprot, the default is refseq

Value

A data frame containing ID, Sequence, AA_in_protein, aligned_seq.

Author(s)

Dongdong Zhan and Mengsha Tong

References

Hadley Wickham (2018). stringr: Simple, Consistent Wrappers for Common String Operations. R package version 1.3.0.\ https://CRAN.R-project.org/package=stringr.

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/get_aligned_seq_for_mea.RData
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "get_aligned_seq_for_mea.RData")
load("get_aligned_seq_for_mea.RData")

foreground_df <- get_aligned_seq_for_mea(
  ID[1:100], Sequence[1:100], AA_in_protein[1:100],
  fixed_length, species = 'human',
  fasta_type = 'refseq'
```

```
)
head(foreground_df)

}
```

`get_colors_for_discrete_value` *Generate custom colors from discrete values for heatmaps.*

Description

Generate custom colors from discrete values for heatmaps.

Usage

```
get_colors_for_discrete_value(color_intervals_list, value_intervals_list)
```

Arguments

`color_intervals_list`	a list for building color intervals.
`value_intervals_list`	a list for building value intervals.

Value

A vectors containing color distributions.

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
value_intervals_list <- list(
seq(-4, -2, 0.2),
seq(-2, -1, 0.2),
seq(-1, 1, 0.2),
seq(1, 2, 0.2),
seq(2, 4, 0.2)
)
color_intervals_list <- list(
  c('blue', '#33CCFF'),
  c('#33CCFF', 'green'),
  c('green', 'white', '#FF6600'),
  c('#FF6600', 'red'),
  c('red', 'firebrick')
)
colors <- get_colors_for_discrete_value(
  color_intervals_list,
  value_intervals_list
)
head(colors)
```

get_combined_data_frame *Get a data frame mapped ID to Gene Symbol.*

Description

This is an intermediate file and a dataframe with Gene Symbol exported. Based on a library file consisting of mapping relationships about Gene Symbol, GeneID, RefSeq_Protein_GI, RefSeq_Protein_Accession and Uniprot_Protein_Accession, a new dataframe with Sequence, GI, Modification, Gene Symbol, Area and PSMs,is contructed.

Usage

```
get_combined_data_frame(merge_df_with_phospho_peptides, species = "human",
  id_type = "RefSeq_Protein_GI")
```

Arguments

merge_df_with_phospho_peptides	A dataframe consisting of IDs (Sequence_GI_Psite) and Area values.
species	A string, the options are human, mouse and rattus, the default is human.
id_type	A string, the options are 'GeneID', 'RefSeq_Protein_GI', 'RefSeq_Protein_Accession' and 'Uniprot_Protein_Accession', the default is RefSeq_Protein_GI.

Value

A dataframe with Sequence, GI, Modification, Gene Symbol, Area values and PSMs

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
   ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/get_combined_data_frame.RDa
    load_data <- load_data_with_ftp(ftp_url, 'RData')
    writeBin(load_data, "get_combined_data_frame.RData")
    load("get_combined_data_frame.RData")

    combined_df_with_mapped_gene_symbol <- get_combined_data_frame(
      merge_df_with_phospho_peptides[1:11,], species = 'human',
      id_type = 'RefSeq_Protein_GI'
    )
    head(combined_df_with_mapped_gene_symbol)
}
```

get_df_with_AAs_i *Get a data frame of amino acid sequences for proteins.*

Description

Get a data frame of amino acid sequences for proteins.

Usage

```
get_df_with_AAs_i(unique_proteins, i, id_data_only_peptide2gi,
  proteins_in_id_data_only_peptide2gi, sequences_in_id_data_only_peptide2gi,
  modification_index_in_protein_seq_list)
```

Arguments

unique_proteins	a vector for unique proteins.
i	the ith unique proteins.
id_data_only_peptide2gi	a data frame for peptides with protein gi.
proteins_in_id_data_only_peptide2gi	a vector for proteins with only protein gi.
sequences_in_id_data_only_peptide2gi	a vector for peptides with only protein gi.
modification_index_in_protein_seq_list	a list for the index of modifications in protein sequence.

Value

A data frame with sequences for proteins.

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/get_df_with_AAs_i.RData"
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "get_df_with_AAs_i.RData")
load("get_df_with_AAs_i.RData")

df_with_AAs_i <- get_df_with_AAs_i(unique_proteins, i,
  id_data_only_peptide2gi,
  proteins_in_id_data_only_peptide2gi,
  sequences_in_id_data_only_peptide2gi,
  modification_index_in_protein_seq_list
)
head(df_with_AAs_i)
}
```

get_file_info_from_dir

Get data lists from files and the corresponding file ids.

Description

Read batch files (.txt or .csv) from a specific directory.

Usage

```
get_file_info_from_dir(specific_dir, experiment_ID)
```

Arguments

specific_dir	A folder containing files as input.
experiment_ID	A vector containing experiment codes as input

Value

A list containing data from files and corresponding file ids

Examples

```
## Not run:
result_list <- get_file_info_from_dir(
  specific_dir,
  experiment_ID
)

## End(Not run)
```

get_file_suffix

Get sufffix of input file.

Description

Get sufffix of input file.

Usage

```
get_file_suffix(file_name)
```

Arguments

file_name	A string for file names.

Value

Return file suffix.

Examples

```
get_file_suffix("myfile.txt")
```

`get_filtered_df` *Get data frame filtered based on the Mascot and reference files.*

Description

Get data frame filtered based on the Mascot and reference files.

Usage

```
get_filtered_df(mascotfileName, refFileName)
```

Arguments

`mascotfileName`	a string for mascot names as input.
`refFileName`	a string for reference file names.

Value

A filtered data frame

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
## Not run:
df <- get_filtered_df(mascotfileName, refFileName)

## End(Not run)
```

`get_foreground_df_to_motifs` *Get filtered foreground data frame that its aligned sequences with specific motif.*

Description

Get filtered foreground data frame that its aligned sequences with specific motif.

Usage

```
get_foreground_df_to_motifs(foreground_sequences_mapped_to_motifs, foreground,
  foreground_df)
```

Arguments

foreground_sequences_mapped_to_motifs	A list that consists of motifs and its corresponding aligned sequences.
foreground	A vector for aligned sequences.
foreground_df	A data frame from the initial foreground data frame.

Value

A data frame that its aligned sequences with specific motif.

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
   ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/get_foreground_df_to_motifs
    load_data <- load_data_with_ftp(ftp_url, 'RData')
    writeBin(load_data, "get_foreground_df_to_motifs.RData")
    load("get_foreground_df_to_motifs.RData")

    foreground_df_mapped_to_motifs <- get_foreground_df_to_motifs(
      foreground_sequences_mapped_to_motifs,
      foreground, foreground_df
    )
   head(foreground_df_mapped_to_motifs)
}
```

---

get_foreground_seq_to_motifs

Get motifs and their corresponding aligned sequences form from foreground.

Description

Get motifs and their corresponding aligned sequences form from foreground.

Usage

```
get_foreground_seq_to_motifs(motifs_list, foreground)
```

Arguments

motifs_list	A list from motif enrichment analysis.
foreground	A vector for aligned sequences.

Value

A list containing motifs and the corresponding sequences from foreground.

Author(s)

Dongdong Zhan and Mengsha Tong

References

Hadley Wickham (2018). stringr: Simple, Consistent Wrappers for Common String Operations. R package version 1.3.0.\ https://CRAN.R-project.org/package=stringr.

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
   ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/get_foreground_seq_to_motif
    load_data <- load_data_with_ftp(ftp_url, 'RData')
    writeBin(load_data, "get_foreground_seq_to_motifs.RData")
    load("get_foreground_seq_to_motifs.RData")

    foreground_sequences_mapped_to_motifs <- get_foreground_seq_to_motifs(
      motifs_list,
      foreground
    )
    head(foreground_sequences_mapped_to_motifs)
    require(ggseqlogo)
    ggseqlogo(foreground_sequences_mapped_to_motifs[[15]])
}
```

---

get_global_background_df

Get background data frame (fasta library from Refseq).

Description

Get background data frame (fasta library from Refseq).

Usage

```
get_global_background_df(species = "human", fasta_type = "refseq")
```

Arguments

species	A string for that the alignment is based on which kinds of species, the options are human, mouse and rattus.
fasta_type,	A string for fasta source, the options are refseq and uniprot, the default is refseq

Value

A data frame of background

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server.
## It may take a few minutes.
if(FALSE){
    background_df <- get_global_background_df(species = 'human', fasta_type = 'refseq')
    head(background_df)
}
```

---

get_ka_by_mean_or_mlr *Computing kinase activity using mean value and multiple linear regression (ridge regression) except KSEA*

Description

Computing kinase activity using mean value and multiple linear regression (ridge regression) except KSEA

Usage

```
get_ka_by_mean_or_mlr(ptypes_data, species = "human", log2_label = FALSE,
  method = "mean")
```

Arguments

ptypes_data	A data frame of phosphorylation data after normalization.
species	A string representing the species of imported data, the options are human, mouse and rat.
log2_label	A boolean value representing whether data is logarithmetics, the default is FALSE.
method	A string for the method to compute kinase activity, the options are 'mean' and 'mlr' (multiple linear regression), the default is mean.

Value

A data frame that consists of kinase, psite, substrate, counting byond ratio_cutoff and corresponding original value.

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/get_ka_by_mean_or_mlr.RData"
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "get_ka_by_mean_or_mlr.RData")
load("get_ka_by_mean_or_mlr.RData")

kinase_activity_df <- get_ka_by_mean_or_mlr(
  cluster_df,
  species = 'human',
  log2_label = TRUE,
  method = 'mean'
)
head(kinase_activity_df)
```

---

get_ksea_regulons_info

Get informational data frame by combining results from all experiments

---

Description

Get informational data frame by combining results from all experiments

Usage

```
get_ksea_regulons_info(ksea_regulons, ksea_trans_list, ksea_x_list,
  ptypes_data_ratio_colnames)
```

Arguments

ksea_regulons	A kinase vector from all experiments.
ksea_trans_list	A list that consits of regulation direction of kinase from each experiment by ksea.
ksea_x_list	A list that consits of sepecific information from each experiment by ksea, like regulation direction, p value and activity etc..
ptypes_data_ratio_colnames	A vector that consists of column names from experiments.

Value

A data frame containing sepecific information of all experiments from ksea results, like regulation direction, pvalue and activity etc..

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/get_ksea_regulons_info.RData"
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "get_ksea_regulons_info.RData")
load("get_ksea_regulons_info.RData")

ksea_regulons_activity_df <- get_ksea_regulons_info(
  ksea_regulons,
  ksea_trans_list,
  ksea_activity_list,
  ptypes_data_ratio_colnames
)
ksea_regulons_activity_df
```

---

get_ksea_result_list *Kinase activity analysis based on known and predicted kinase-substrate relationships*

Description

Kinase activity analysis based on known and predicted kinase-substrate relationships

Usage

```
get_ksea_result_list(ptypes_data_ratio_in_single_exp, ID,
  kinase_substrate_regulation_relationship, ksea_activity_i_pvalue = 0.05)
```

Arguments

ptypes_data_ratio_in_single_exp	A quantification vector from a single experiment.
ID	A phosporylation ID vector like VIM_S56 (GeneSymbol_psite).
kinase_substrate_regulation_relationship	A data frame contanning kinase-substrate relationships that consists of "kinase", "substrate", "site", "sequence" and "predicted" columns.
ksea_activity_i_pvalue	A cutoff used for filtering significant activities computed from KSEA.

Value

A list containing results from ksea.

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/get_ksea_result_list.RData"
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "get_ksea_result_list.RData")
load("get_ksea_result_list.RData")

ksea_result_list_i <- get_ksea_result_list(
 ptypes_data_ratio_in_single_exp, ID,
 kinase_substrate_regulation_relationship,
 ksea_activity_i_pvalue = 0.05
)
head(ksea_result_list_i)
}
```

---

get_kses *computing kinase-substrate enrichment significance (pvalue)*

Description

computing kinase-substrate enrichment significance (pvalue)

Usage

```
get_kses(substate_vector, regulons_of_kinase, trial = 1000)
```

Arguments

substate_vector	a vector for substrates with values identified in current experiments.
regulons_of_kinase	a vector for substrates of a specific kinase, which identified in current experiments.
trial	a numeric for the number of random samples, the default is 1000.

Value

A list for expected enrichment scores and its significance

Author(s)

Dongdong Zhan and Mengsha Tong

References

Hernandez-Armenta C et al. Benchmarking substrate-based kinase activity inference using phosphoproteomic data[J]. Bioinformatics, 2017, 33(12): 1845-1851.

Examples

```
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/get_kses.RData"
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "get_kses.RData")
load("get_kses.RData")

ksea_result_i_l <- get_kses(
  ptypes_data_ratio_in_single_exp_desc,
  regulons_i_l,
  1000
)
head(ksea_result_i_l)
```

---

get_list_with_filtered_sites

Filter phosphorylation sites.

Description

Filter phosphorylation sites by extracting all peptides with ion score>=20 and FDR<0.01 from Firmiana and having psites scores.

Usage

```
get_list_with_filtered_sites(peptide_id, files, files_site_score, qc, min_score,
  min_FDR)
```

Arguments

peptide_id	A vector containing experiment ids as input.
files	A data list containing peptides identificaton from Firmiana as input.
files_site_score	A data list containing psites scores extracted from mascot xml. The default is NULL, which represents no QC file.
qc	A boolean value representing whether it has QC files. The default is True.
min_score	A numeric for the minimum score of credible peptides, the default is 20 for Mascot ion score.
min_FDR	A numeric for the minimum FDR of credible peptides, the default is 0.01.

Value

A list containing peptides dataframe with area values and psm, IDs with mergered sequences, gi and sites, new peptides dataframe combined previous peptides dataframe and IDs.

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
## Not run:
result_list_with_filtered_sites <- get_list_with_filted_sites(
  peptide.id,
  files,
  files_site_score
)

## End(Not run)
```

`get_merged_phospho_df` *Get merged data frame with phospho-peptides.*

Description

Get merged data frame with phospho-peptides.

Usage

```
get_merged_phospho_df(peptide_id, peptide_df_with_area_psm_list,
  ID_of_seq_gi_site_list, ID_DF_list)
```

Arguments

`peptide_id`	a vector for peptide ID.
`peptide_df_with_area_psm_list`	a list for peptides with areas and PSMs.
`ID_of_seq_gi_site_list`	a list for peptides ID with sequence, gi and site.
`ID_DF_list`	a list for ID and values.

Value

a merged data frame with phospho-peptides.

`get_modification_index` *Get indexes of modifications in protein sequences.*

Description

Get indexes of modifications in protein sequences.

Usage

```
get_modification_index(id_data_only_peptide2gi, fasta_data)
```

Arguments

id_data_only_peptide2gi	a data frame for peptides with protein gi.
fasta_data	a fasta data for a specific species.

Value

A vector for indexes of modifications in protein sequences.

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/get_modification_index.RData"
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "get_modification_index.RData")
load("get_modification_index.RData")

modification_index_in_protein_seq_list <- get_modification_index(
  id_data_only_peptide2gi[1:100, ],
  fasta_data
)
head(modification_index_in_protein_seq_list)
}
```

---

get_motifs_list *Motif enrichment using rmotifx.*

Description

Motif enrichment using rmotifx.

Usage

```
get_motifs_list(foreground, background, center_vector, motifx_pvalue)
```

Arguments

foreground	A vector for aligned sequences as the foreground input.
background	A vector for aligned sequences as the background input.
center_vector	A vector for aligned centers.
motifx_pvalue	A numeric value for selecting motifs that meets the minimum cutoff.

Value

A list for results of motif enrichment.

Author(s)

Dongdong Zhan and Mengsha Tong

References

Omar Wagih (2014). rmotifx: An iterative statistical approach to the discovery of biological sequence motifs. R package version 1.0.

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
   ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/get_motifs_list.RData"
    load_data <- load_data_with_ftp(ftp_url, 'RData')
    writeBin(load_data, "get_motifs_list.RData")
    load("get_motifs_list.RData")

   motifs_list <- get_motifs_list(foreground[1:100], background[1:100], center_vector, motifx_pvalue)
    head(motifs_list)
}
```

---

get_motif_analysis_summary

Get summary results of motif analysis for specific input

Description

Get summary results of motif analysis for specific input

Usage

```
get_motif_analysis_summary(foreground, background, center = "S",
  min_sequence_count = 1, min_pvalue = 0.01)
```

Arguments

foreground	A vector for AA sequences with fixed length as foreground input.
background	A vector for AA sequences with fixed length as background input.
center	A character for center of k-mer.
min_sequence_count	A numeric for the minimum sequence number assigned to a motif.
min_pvalue	A numeric for the minimum pvalue for found motif.

Value

A list for summary result of motif analysis

Author(s)

Dongdong Zhan and Mengsha Tong

References

Omar Wagih (2014). rmotifx: An iterative statistical approach to the discovery of biological sequence motifs. R package version 1.0.

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
   ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/get_motif_analysis_summary.
    load_data <- load_data_with_ftp(ftp_url, 'RData')
    writeBin(load_data, "get_motif_analysis_summary.RData")
    load("get_motif_analysis_summary.RData")

    motifs <- get_motif_analysis_summary(
      foreground[1:100], background[1:100],
      center = center,
      min_sequence_count = 1,
      min_pvalue = motifx_pvalue
    )
    head(motifs)
}
```

---

get_normalized_data_FOT5

Normailization on basis of sum

Description

Normailization on basis of sum

Usage

```
get_normalized_data_FOT5(data_frame, experiment_code_file_path)
```

Arguments

data_frame	A data frame containing IDs and values merged from multi-experiments as input.
experiment_code_file_path	A file path of storing experiment codes as input. The experiment codes are required to keep pace with column names of Values.

Value

A data frame after normalization

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
ftp_url1 <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/get_normalized_data_FOT5.RDa
ftp_url2 <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/profiling_exp_design_info.tx
load_data1 <- load_data_with_ftp(ftp_url1, 'Rdata')
writeBin(load_data1, "get_normalized_data_FOT5.RData")
load("get_normalized_data_FOT5.RData")

load_data2 <- load_data_with_ftp(ftp_url2, 'downloadtxt')
writeBin(load_data2, "profiling_exp_design_info.txt")
profiling_exp_design_info_file_path <- "./profiling_exp_design_info.txt"

profiling_data_normalized <- get_normalized_data_FOT5(profiling_data,
  profiling_exp_design_info_file_path
)
head(profiling_data_normalized)

}
```

---

get_normalized_data_of_psites

To normalize data and filter data only including phosphorylation sites.

Description

To normalize data and filter data only including phosphorylation sites.

Usage

```
get_normalized_data_of_psites(data_frame, experiment_code_file_path,
  topN = NA, mod_types = c("S", "T", "Y"))
```

Arguments

data_frame	A data frame containing IDs and quantification values merged from multi-experiments as input.
experiment_code_file_path	A file path of storing experiment codes as input. The experiment codes are required to keep pace with column names of Value.
topN,	A numeric value, selecting N p-sites with high intensity rank for normalization, the default is NA.
mod_types,	A vector for modification residues, the default is c('S', 'T', 'Y') for phosphorylation modifications.

Value

A list including data frame after filtering or normalization (x 1e5).

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
ftp_url1 <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/get_normalized_data_of_psite
ftp_url2 <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/phosphorylation_exp_design_i
load_data1 <- load_data_with_ftp(ftp_url1, 'Rdata')
writeBin(load_data1, "get_normalized_data_of_psites.RData")
load("get_normalized_data_of_psites.RData")

load_data2 <- load_data_with_ftp(ftp_url2, 'downloadtxt')
writeBin(load_data2, "phosphorylation_exp_design_info.txt")
phosphorylation_exp_design_info_file_path <- "./phosphorylation_exp_design_info.txt"

phospho_data_filtering_STY_and_normalization_list <- get_normalized_data_of_psites(
  summary_df_of_unique_proteins_with_sites,
  phosphorylation_exp_design_info_file_path,
  topN = NA, mod_types = c('S', 'T', 'Y')
)
head(phospho_data_filtering_STY_and_normalization_list)

}
```

---

get_substrate_expr_df *Get a data frame only containing kinase inferred by KSEA*

Description

Get a data frame only containing kinase inferred by KSEA

Usage

```
get_substrate_expr_df(ID, kinase_substrate_regulation_relationship,
  ksea_regulons, ptypes_data_ratio, ratio_cutoff = 3)
```

Arguments

ID	A phosporylation ID vector like VIM_S56 (GeneSymbol_psite).
kinase_substrate_regulation_relationship	A data frame contanning relationship of kinase-substrate that consists of "kinase", "substrate", "site", "sequence" and "predicted" columns.
ksea_regulons	A kinase vector from ksea
ptypes_data_ratio	A data frame that the ratio of phosphorylation and profiling data
ratio_cutoff	A cutoff that depicts quantification changes at phosphorylation level relative to profiling level, the default is 3.

Value

A data frame that consists of kinase, psite, substrate, counting byond ratio_cutoff and corresponding original value.

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/get_substrate_expr_df.RData"
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "get_substrate_expr_df.RData")
load("get_substrate_expr_df.RData")

kinase_site_substrate_original_ratio_df <- get_substrate_expr_df(
  ID,
  kinase_substrate_regulation_relationship,
  ksea_regulons,
  ptypes_data_ratio,
  ratio_cutoff = 3
)
head(kinase_site_substrate_original_ratio_df)
```

---

get_summary_from_ksea *Get a data frame only containing inforamtion of kinase inferred by KSEA*

Description

Get a data frame only containing inforamtion of kinase inferred by KSEA

Usage

```
get_summary_from_ksea(ptypes_data, species = "human", log2_label = TRUE,
  ratio_cutoff = 3)
```

Arguments

ptypes_data	A data frame of phosphorylation data after normalization.
species	A string representing the species of imported data, the options are human, mouse and rat.
log2_label	A boolean value representing whether data is logarithmetics, the default is FALSE.
ratio_cutoff	A cutoff that depicts quantification changes at phosphorylation level relative to profiling level, the default is 3.

Value

A data frame that consists of kinase, psite, substrate, counting byond ratio_cutoff and corresponding original value.

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
   ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/get_summary_from_ksea.RData
    load_data <- load_data_with_ftp(ftp_url, 'RData')
    writeBin(load_data, "get_summary_from_ksea.RData")
    load("get_summary_from_ksea.RData")

    summary_df_list_from_ksea_cluster <- get_summary_from_ksea(
      cluster_df, species = 'human',
      log2_label = FALSE, ratio_cutoff = 3
    )
    head(summary_df_list_from_ksea_cluster)
}
```

---

get_summary_with_unique_sites

Assign psites to protein sequence.

Description

Construct the data frame with unique phosphorylation site for each protein sequence and eliminate redundancy.

Usage

```
get_summary_with_unique_sites(combined_df_with_mapped_gene_symbol,
  species = "human", fasta_type = "refseq")
```

Arguments

combined_df_with_mapped_gene_symbol	A dataframe with Sequence, ID, Modification, Gene Symbol, Area and PSMs as input.
species	A string, the options are human, mouse and rattus, the default is human.
fasta_type,	A string for fasta source, the options are refseq and uniprot, the default is refseq

Value

A dataframe that all redundant psites are assigned to protein sequence.

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/get_summary_with_unique_sites
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "get_summary_with_unique_sites.RData")
load("get_summary_with_unique_sites.RData")

summary_df_of_unique_proteins_with_sites <- get_summary_with_unique_sites(
  combined_df_with_mapped_gene_symbol[1:100, ],
  species = 'human',
  fasta_type = 'refseq'
)
head(summary_df_of_unique_proteins_with_sites)

}
```

---

get_unique_AAs_i_df *Get data frame without redundancy.*

Description

Get data frame without redundancy.

Usage

```
get_unique_AAs_i_df(df_with_AAs_i)
```

Arguments

df_with_AAs_i	a data frame for peptides of the ith protein.

Value

A data frame with sites in unique protein.

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/get_unique_AAs_i_df.RData"
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "get_unique_AAs_i_df.RData")
load("get_unique_AAs_i_df.RData")

summary_df_of_unique_protein_with_sites <- get_unique_AAs_i_df(df_with_AAs_i)
head(summary_df_of_unique_protein_with_sites)
```

keep_psites_with_max_in_topX *Keep psites whose row maximum is top N (percentage).*

Description

Compute row maximum each psites, sort row maximum in decreasing order and keep top N (percentage).

Usage

```
keep_psites_with_max_in_topX(phospho_data, percent_of_kept_sites = 3/4)
```

Arguments

phospho_data	A data frame of phospho-data.
percent_of_kept_sites	A numeric value representing a cutoff used for filter psites. The default is 3/4.

Value

A data frame meeting specific cutoff.

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/keep_psites_with_max_in_topX.
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "keep_psites_with_max_in_topX.RData")
load("keep_psites_with_max_in_topX.RData")

phospho_data_topX <- keep_psites_with_max_in_topX(phospho_data,
  percent_of_kept_sites = 0.9
)
head(phospho_data_topX)
```

load_data_with_ftp *Load datasets from URL (ftp://111.198.139.72:4000/pub/PhosMap_datasets)*

Description

Some datasets with larger size need to be loaded for mapping ID or protein sequence when using PhosMap. These datasets could be ragarded as library and uploaded to ftp://111.198.139.72:4000/pub/PhosMap_datasets in advance. When first perfoming functions depending on these datasets, PhosMap will get them from specific URL and save them into local disk.

Usage

```
load_data_with_ftp(ftp_link, data_type)
```

Arguments

ftp_link	A string for URL of datasets.
data_type	A string for type of datasets (txt, csv, RData).

Value

A dataframe

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/profiling_exp_design_info.txt
load_data <- load_data_with_ftp(ftp_url, 'txt')
head(load_data)
```

---

load_data_with_http *Load datasets from URL (https://github.com/ecnuzdd/PhosMap_datasets)*

Description

Some datasets with larger size need to be loaded for mapping ID or protein sequence when using PhosMap. These datasets could be ragarded as library and uploaded to https://github.com/ecnuzdd/PhosMap_datasets in advance. When first perfoming functions depending on these datasets, PhosMap will get them from specific URL and save them into local disk.

Usage

```
load_data_with_http(http_link, data_type)
```

Arguments

http_link	A string for URL of datasets.
data_type	A string for type of datasets (txt of csv).

Value

A dataframe

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
http_link <- url('https://raw.githubusercontent.com/ecnuzdd/PhosMap_datasets/master/function_demo_data/prof
data_type <- 'txt'
load_data <- load_data_with_http(
  http_link, data_type
)
head(load_data)
```

---

mea_based_on_background *Motif enrichment based on global background (fasta library from Refseq).*

Description

Motif enrichment based on global background (fasta library from Refseq).

Usage

```
mea_based_on_background(foreground, AA_in_protein, background, motifx_pvalue)
```

Arguments

foreground	A vector for aligned sequence of foreground.
AA_in_protein	A vector for the location of S/T/Y in sequence of protein.
background	A vector for aligned sequence of background.
motifx_pvalue	A numeric value for selecting motifs that meets the minimum cutoff.

Value

A list containing motifs and the corresponding sequences

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
   ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/mea_based_on_background.RDa
    load_data <- load_data_with_ftp(ftp_url, 'RData')
    writeBin(load_data, "mea_based_on_background.RData")
    load("mea_based_on_background.RData")

    motifs_list <- mea_based_on_background(
      foreground[1:100],
      AA_in_protein[1:100],
      background[1:1000],
      motifx_pvalue
```

```
    )
}
```

`merge_profiling_file_from_Firmiana` *Merge profiling files downloaded from Firmiana.*

Description

Filter data based on US (Unique and Ionscore > 20) peptide counts then merge profiling files.

Usage

```
merge_profiling_file_from_Firmiana(firmiana_gene_dir, US_cutoff = 1,
  experiment_gene_file_path)
```

Arguments

`firmiana_gene_dir`	a folder containing gene identification results as input.
`US_cutoff`	a numerical value as a cutoff to filter data, the default is 1.
`experiment_gene_file_path`	a file path for storing experiemnt design of proteomics data.

Value

A merged data frame after filtering (US_cutoff) and replacing NAs to zeros.

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
## Not run:
merged_df <- merge_profiling_file_from_Firmiana(firmiana_gene_dir, US_cutoff = 1)

## End(Not run)
```

`motif_data_frame_to_sequence` *Convert data frame of motif to the sequence pattern*

Description

Convert data frame of motif to the sequence pattern

Usage

```
motif_data_frame_to_sequence(motif_data_frame, center, width)
```

Arguments

`motif_data_frame`	A data frame with two columns including amino acid and index on sequence with fixed length.
`center`	A character for center of k-mer.
`width`	A numeric for specific k-mer.

Value

A string for motif pattern

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/motif_data_frame_to_sequence.
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "motif_data_frame_to_sequence.RData")
load("motif_data_frame_to_sequence.RData")

motif_pattern <- motif_data_frame_to_sequence(
  motif_coordinate_data_frame,
  center, width
)
head(motif_pattern)
```

normalize_nopair_ctrl_by_col

For data without pairs but with control, normalize them to the control.

Description

For data without pairs but with control, normalize them to the control.

Usage

```
normalize_nopair_ctrl_by_col(data_frame, experiment_design_file, control_label)
```

Arguments

data_frame	a data frame as input.
experiment_design_file	a data frame for design of experiment.
control_label	a string for a control.

Value

A data frame after normalization.

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/normalize_nopair_ctrl_by_col.
load_data <- load_data_with_ftp(ftp_url, 'Rdata')
writeBin(load_data, "normalize_nopair_ctrl_by_col.RData")
load("normalize_nopair_ctrl_by_col.RData")

phospho_data_normalize_by_column <- normalize_nopair_ctrl_by_col(
  phospho_data_normalized,
  phosphorylation_experiment_design_file,
  control_label
)
head(phospho_data_normalize_by_column)
}
```

normalize_nopair_noctrl_by_colmed

For data without pairs and control, normalize them to the median.

Description

For data without pairs and control, normalize them to the median.

Usage

```
normalize_nopair_noctrl_by_colmed(data_frame)
```

Arguments

data_frame	a data frame as input.

Value

A data frame after normalization.

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/normalize_nopair_noctrl_by_co
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "normalize_nopair_noctrl_by_colmed.RData")
load("normalize_nopair_noctrl_by_colmed.RData")

phospho_data_normalize_by_column <- normalize_nopair_noctrl_by_colmed(
  phospho_data_normalized
)
head(phospho_data_normalize_by_column)
}
```

---

normalize_phos_data_to_profiling

Normalize phospho-data to profiling

Description

Normalize phospho-data to profiling

Usage

```
normalize_phos_data_to_profiling(phospho_data_normalized,
  profiling_data_normalized, phosphorylation_exp_design_info_file_path,
  profiling_exp_design_info_file_path, control_label = NA,
  pair_flag = FALSE)
```

Arguments

phospho_data_normalized	A data frame of phospho-data after normalization
profiling_data_normalized	A data frame of profiling after normalization
phosphorylation_exp_design_info_file_path	A file path about phosphorylation experiment design, it has 2 kinds of file configuration as follows: 1. experiment_design_noPair.txt must contain columns of Experiment_Code, Group. 2. experiment_design_Pair.txt must contain columns of Experiment_Code, Group, and Pair. (Pair: 1 -> case, -1 -> control)

profiling_exp_design_info_file_path
: A file path about profiling experiment design, it has 2 kinds of file configuration as same as phosphorylation_exp_design_info_file_path.

control_label
: A string represents label of control group. The default is NA which shows no control group.

pair_flag
: A boolean value represents whether experiments have pairs. The default is FALSE which shows no pairs.

Value

A data frame which comes from results that phospho-data is normalizated base on the abundance of proteins in the profiling experiments.

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
   ftp_url1 <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/normalize_phos_data_to_pro
   ftp_url2 <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/phosphorylation_exp_desig
   ftp_url3 <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/profiling_exp_design_info

    load_data1 <- load_data_with_ftp(ftp_url1, 'Rdata')
    writeBin(load_data1, "normalize_phos_data_to_profiling.RData")
    load("normalize_phos_data_to_profiling.RData")

    load_data2 <- load_data_with_ftp(ftp_url2, 'downloadtxt')
    writeBin(load_data2, "phosphorylation_exp_design_info.txt")
    phosphorylation_exp_design_info_file_path <- "./phosphorylation_exp_design_info.txt"

    load_data3 <- load_data_with_ftp(ftp_url3, 'downloadtxt')
    writeBin(load_data3, "profiling_exp_design_info.txt")
    profiling_exp_design_info_file_path <- "./profiling_exp_design_info.txt"

    data_frame_normalization_with_control_no_pair <- normalize_phos_data_to_profiling(
      phospho_data_topX, profiling_data_normalized,
      phosphorylation_exp_design_info_file_path,
      profiling_exp_design_info_file_path,
      control_label = '0',
      pair_flag = FALSE
    )
    head(data_frame_normalization_with_control_no_pair)
}
```

---

normalize_to_Pair *For data with pairs, normalize them to the sample with flag eaqul to -1.*

Description

For data with pairs, normalize them to the sample with flag eaqul to -1.

Usage

```
normalize_to_Pair(data_frame, experiment_design_file)
```

Arguments

data_frame	a data frame as input.
experiment_design_file	a data frame for design of experiment.

Value

A data frame after normalization.

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/normalize_to_Pair.RData"
load_data <- load_data_with_ftp(ftp_url, 'Rdata')
writeBin(load_data, "normalize_to_Pair.RData")
load("normalize_to_Pair.RData")

phospho_data_normalize_by_column <- normalize_to_Pair(
  phospho_data_normalized,
  pairing_phosphorylation_experiment_design_file
)
head(phospho_data_normalize_by_column)

}
```

---

plot_seqlogo *Plot sequence logo based on list that consist of motifs and sequences.*

Description

Plot sequence logo based on list that consist of motifs and sequences.

Usage

```
plot_seqlogo(base_dir, foreground_sequences_mapped_to_motifs,
  plot_min_seqs = 5)
```

Arguments

base_dir	A path used for saving plots.
foreground_sequences_mapped_to_motifs	A list that consist of motifs and sequences.
plot_min_seqs	A numeric value for cutoff, sequences of motifs greater than the cutoff are plotted, the default is 5.

Value

Plot sequence logo based on list that consist of motifs and sequences. The results will be saved in a folder named PhosMap_ggseqlogo in the BASE_DIR parameter specified directory.

Author(s)

Dongdong Zhan and Mengsha Tong

References

(1) Omar Wagih (2017). ggseqlogo: A 'ggplot2' Extension for Drawing Publication-Ready Sequence Logos. R package version 0.1.\ https://github.com/omarwagih/ggseqlogo; (2) Hadley Wickham (2018). stringr: Simple, Consistent Wrappers for Common String Operations. \ R package version 1.3.0. https://CRAN.R-project.org/package=stringr

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/plot_seqlogo.RData"
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "plot_seqlogo.RData")
load("plot_seqlogo.RData")

BASE_DIR = getwd() # current working directory
BASE_DIR = normalizePath(BASE_DIR)
plot_seqlogo(BASE_DIR, foreground_sequences_mapped_to_motifs[1:50], plot_min_seqs = 25)

}
```

---

pre_process_filter_psites

Get peptides data frame passed phosphorylation sites quality control.

Description

Filter phosphorylation sites by extracting all peptides with ion score>=20 and FDR<0.01 from Firmiana and having psites scores. Generate new IDs consisting of sequence, gi, psite. Quantification values containing area and psm.

Usage

```
pre_process_filter_psites(firmiana_peptide_dir, psites_score_dir,
  phospho_experiment_design_file_path, qc, min_score = 20, min_FDR = 0.01)
```

Arguments

`firmiana_peptide_dir`
: A folder containing peptide identification files from Firmiana as input.

`psites_score_dir`
: A folder containing psites scores files extracted from mascot xml as input.

phospho_experiment_design_file_path
: A string representing the path of phospho-experiment design file as input.

qc
: A boolean value representing whether it has QC files. The default is True.

min_score
: A numeric for the minimum score of credible peptides, the default is 20 for Mascot ion score.

min_FDR
: A numeric for the minimum FDR of credible peptides, the default is 0.01.

Value

A merged data frame containing sequence, gi, psite, area and psm.

Examples

```
## Not run:
merge_df_with_phospho_peptides <- pre_process_filter_psites(
  firmiana_peptide_dir,
  psites_score_dir
)

## End(Not run)
```

---

seach_motif_pattern *Convert data frame of motif to the sequence pattern*

Description

Convert data frame of motif to the sequence pattern

Usage

```
seach_motif_pattern(foreground_sequence, background_sequence,
  min_sequence_count = 1, min_pvalue = 0.01, center = "S", width)
```

Arguments

foreground_sequence
: A vector for AA sequences with fixed length as foreground input.

background_sequence
: A vector for AA sequences with fixed length as background input.

min_sequence_count
: A numeric for the minimum sequence number assigned to a motif.

min_pvalue
: A numeric for the minimum pvalue for found motif.

center
: A character for center of k-mer.

width
: A numeric for specific k-mer.

Value

A list for information summary of searching mortif

Author(s)

Dongdong Zhan and Mengsha Tong

References

Omar Wagih (2014). rmotifx: An iterative statistical approach to the discovery of biological sequence motifs. R package version 1.0.

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/seach_motif_pattern.RData"
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "seach_motif_pattern.RData")
load("seach_motif_pattern.RData")

motif_result_loop_i <- seach_motif_pattern(
  loop_foreground[1:100],
  loop_background[1:1000],
  min_sequence_count = min_sequence_count,
  min_pvalue = min_pvalue,
  center = center,
  width = check_result_list$width
)
head(motif_result_loop_i)

}
```

---

visualization_deps_with_scatter

Visualize differentially expressed results with scatter

Description

Visualize differentially expressed results with scatter

Usage

```
visualization_deps_with_scatter(deps_data, minFC = 2, minPvalue = 0.05,
  main = "Differentially expressed proteins", show_text = FALSE,
  min_up_text = 15, min_down_text = 15)
```

Arguments

deps_data	a data frame containing ID, logFC and pvalue.
minFC	a numeric for the minimum fold change.
minPvalue	a numeric for the significance cutoff.
main	an overall title for the plot.
show_text	a boolean value representing whether or not the text is showed, the default is FALSE.

min_up_text	cutoff value for showing up-IDs. Only IDs with lower than min_up_text are showed.
min_down_text	cutoff value for showing down-IDs. Only IDs with lower than min_down_text are showed.

Value

A scatter plot for showing differentially expressed results.

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/visualization_deps_with_scatt
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "visualization_deps_with_scatter.RData")
load("visualization_deps_with_scatter.RData")

visualization_deps_with_scatter(limma_results_df, minFC = 2,
  minPvalue = 0.05, main = 'Differentially expressed proteins  \n with limma',
  show_text = TRUE, min_up_text = 70, min_down_text = 70
)
```

---

visualization_fuzzycluster

Visualize results from fuzzy clusters with line chart

Description

Visualize results from fuzzy clusters with line chart

Usage

```
visualization_fuzzycluster(input_data, group, group_levels, k_cluster,
  iteration = 100, mfrow = c(3, 3), min_mem = 0.1, plot = TRUE)
```

Arguments

input_data	a data frame containing ID and expression profile.
group	a factor for representing groups.
group_levels	a factor levels for group.
k_cluster	number of clusters fuzzy cluster.
iteration	a numeric value for interation, the defualt is 100.
mfrow	a vector containing 2 elements for controling the subplots in graphic window, the default is mfrow = c(3,3)
min_mem	cutoff value for membership. Only results with greater than min_mem are showed.
plot	a boolean value for deciding whether ploting, the default is TRUE.

Value

A lines chart with fuzzy degree.

Author(s)

Dongdong Zhan and Mengsha Tong

References

(1) David Meyer, Evgenia Dimitriadou, Kurt Hornik, Andreas Weingessel and Friedrich Leisch (2017). e1071: Misc Functions of the \ Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien. R package version 1.6-8.https://CRAN.R-project.org/package=e1071 \ (2) Pengyi Yang (2018). ClueR: Cluster Evaluation. R package version 1.4. https://CRAN.R-project.org/package=ClueR

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/visualization_fuzzycluster.RD
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "visualization_fuzzycluster.RData")
load("visualization_fuzzycluster.RData")

fuzzy_clustObj <- visualization_fuzzycluster(
  fuzzy_input_df, group, group_levels,
  k_cluster=9, iteration = 100,
  mfrow = c(3,3), min_mem = 0.1
 )

}
```

---

visualization_with_simple_pca

A simple PCA plot.

Description

A simple PCA plot.

Usage

```
visualization_with_simple_pca(expr_data_frame, main = "Simple PCA",
  point_cex = 2, point_col = "firebrick", point_type = 20, text_cex = 1)
```

Arguments

expr_data_frame	A data frame containing ID and quantification value.
main	The main title of plot.
point_cex	a numerical value for point size.

point_col	a color code or name for point color.
point_type	point type, see points.
text_cex	a numerical value for text size.

Value

A simple PCA plot.

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/visualization_with_simple_pca
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "visualization_with_simple_pca.RData")
load("visualization_with_simple_pca.RData")

visualization_with_simple_pca(expr_df, main = 'Simple PCA',
  point_cex = 2, point_col = 'firebrick',
  point_type = 20, text_cex = 1
)
```

---

visualization_with_simple_tsne

A simple t-SNE plot.

Description

A simple t-SNE plot.

Usage

```
visualization_with_simple_tsne(expr_data_frame, group, main = "Simple t-SNE",
  perplexity = 10)
```

Arguments

expr_data_frame	A data frame containing ID and quantification value.
group	A factor for group information.
main	The main title of plot.
perplexity	A numerical value for perplexity, the default is 10.

Value

A simple t-SNE plot.

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
## The process needs to load data from PhosMap datasets stored into FTP server and perform large computation.
## It may take a few minutes.
if(FALSE){
ftp_url <- "ftp://111.198.139.72:4000/pub/PhosMap_datasets/function_demo_data/visualization_with_simple_tsn
load_data <- load_data_with_ftp(ftp_url, 'RData')
writeBin(load_data, "visualization_with_simple_tsne.RData")
load("visualization_with_simple_tsne.RData")

visualization_with_simple_tsne(
  expr_data_frame,
  group,
  main = 'Simple t-SNE',
  perplexity = 12
)

}
```

---

write_csv_pep_seq_conf

Write data to specific direction with CSV format.

Description

Write data to specific direction with CSV format.

Usage

```
write_csv_pep_seq_conf(expName, outputName, mascotfileNames, refFileName)
```

Arguments

expName	a string for experiment name as input.
outputName	a string for experiment name as output.
mascotfileNames	a vector for storing mascot file names.
refFileName	a string for reference file name.

Value

Write data to specific direction with CSV format.

Author(s)

Dongdong Zhan and Mengsha Tong

Examples

```
## Not run:
write_csv_pep_seq_conf(expName,
  outputName, mascot_txt_dir_path_expName_path,
  firmiana_peptide_dir_path_expName_path)

## End(Not run)
```

Index