Package 'MSstatsLiP'

January 20, 2022

Type Package

Title LiP Significance Analysis in shotgun mass spectrometry-based proteomic experiments

Version 1.0.0

Date 2021-10-14

Description Tools for LiP peptide and protein significance analysis. Provides functions for summarization, estimation of LiP peptide abundance, and detection of changes across conditions. Utilizes functionality across the MSstats family of packages.

License Artistic-2.0

Depends R (>= 4.1)

Imports dplyr, gridExtra, stringr, ggplot2, grDevices, MSstats, MSstatsConvert, data.table, Biostrings, MSstatsPTM, Rcpp, checkmate, factoextra, ggpubr, purrr, tibble, tidyr, tidyverse, scales, stats

Suggests BiocStyle, knitr, rmarkdown, covr, tinytest

LinkingTo Rcpp

VignetteBuilder knitr

biocViews ImmunoOncology, MassSpectrometry, Proteomics, Software, DifferentialExpression, OneChannel, TwoChannel, Normalization, QualityControl

BugReports https://github.com/Vitek-Lab/MSstatsLiP/issues

Encoding UTF-8

LazyData TRUE

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.2

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

git_branch RELEASE_3_14

git_last_commit 93a0da8

git_last_commit_date 2021-10-26

Date/Publication 2022-01-20

Author Devon Kohler [aut, cre],
Tsung-Heng Tsai [aut],
Ting Huang [aut],
Mateusz Staniak [aut],
Meena Choi [aut],
Valentina Cappelletti [aut],
Liliana Malinovska [aut],
Olga Vitek [aut]

Maintainer Devon Kohler <kohler.d@northeastern.edu>

R topics documented:

annotSite 2
BarcodePlotLiP 3
calculateTrypticity 5
correlationPlotLiP 5
dataProcessPlotsLiP 6
dataSummarizationLiP 8
groupComparisonLiP 11
groupComparisonPlotsLiP 13
LiPRawData 15
locateMod 17
locatePTM 17
MSstatsLiP 18
MSstatsLiP_data 19
MSstatsLiP_model 20
MSstatsLiP_Summarized 21
PCAPlotLiP 22
SkylineTest 24
SkylinetoMSstatsLiPFormat 25
SpectronauttoMSstatsLiPFormat 26
tidyFasta 28
TrPRawData 29
trypticHistogramLiP 30

Index **32**

`annotSite` *Annotate modification site*

Description

`annotSite` annotates modified sites as their residues and locations.

Usage

```
annotSite(aaIndex, residue, lenIndex = NULL)
```

Arguments

aaIndex	An integer vector. Location of the sites.
residue	A string vector. Amino acid residue.
lenIndex	An integer. Default is NULL

Value

A string.

Examples

```
annotSite(10, "K")
annotSite(10, "K", 3L)
```

BarcodePlotLiP *Barcode plot. Shows protein coverge of LiP modified peptides.*

Description

Barcode plot. Shows protein coverge of LiP modified peptides.

Usage

```
BarcodePlotLiP(
  data,
  fasta,
  model_type = "Adjusted",
  which.prot = "all",
  which.comp = "all",
  adj.pvalue.cutoff = 0.05,
  FC.cutoff = 0,
  FT.only = FALSE,
  width = 12,
  height = 4,
  address = ""
)
```

Arguments

data	list of data.tables containing LiP and TrP data in MSstatsLiP format. Should be output of modeling function such as groupComparisonLiP.
fasta	A string of path to a FASTA file
model_type	A string of either "Adjusted" or "Unadjusted", indicating whether to plot the adjusted or unadjusted models. Default is "Adjusted".
which.prot	a list of peptides to be visualized. Default is "all" which will plot a separate barcode plot for each protein.
which.comp	a list of comparisons to be visualized. Default is "all" which will plot a separate barcode plot for each condition and protein.
adj.pvalue.cutoff	Defualt is .05. Alpha value for testing significance of model output.
FC.cutoff	Default is 0. Minimum absolute FC before a comparison will be considered significant.
FT.only	FALSE plots all FT and HT peptides, TRUE plots FT peptides only. Default is FALSE.
width	width of the saved file. Default is 10.
height	height of the saved file. Default is 10.
address	the name of folder that will store the results. Default folder is the current working directory. The other assigned folder has to be existed under the current working directory. An output pdf file is automatically created with the default name of "VolcanoPlot.pdf" or "Heatmap.pdf". The command address can help to specify where to store the file as well as how to modify the beginning of the file name. If address=FALSE, plot will be not saved as pdf file but showed in window

Value

plot or pdf

Examples

```
# Specify Fasta path
fasta_path <- system.file("extdata", "ExampleFastaFile.fasta", package="MSstatsLiP")

# Use model data to create Barcode Plot
BarcodePlotLiP(MSstatsLiP_model, fasta_path,
              model_type = "Adjusted",
              address=FALSE)
```

calculateTrypticity *Calculates level of trypticity for a list of LiP Peptides.*

Description

Takes as as input LiP data and a fasta file. These can be the outputs of MSstatsLiP functions.

Usage

```
calculateTrypticity(LiP_data, fasta_file)
```

Arguments

LiP_data	name of variable containing LiP data. Must contain at least two columns named 'PeptideSequence' and 'ProteinName'. The values in these column must match with what is in the corresponding FASTA file.
fasta_file	name of variable containing FASTA data. If FASTA file has not been processed please run the tidyFasta() function on it before inputting into this function.

Value

a data.frame including protein, peptide, and trypticity metrics.

Examples

```
fasta <- tidyFasta(system.file("extdata", "ExampleFastaFile.fasta", package="MSstatsLiP"))
calculateTrypticity(MSstatsLiP_data$LiP, fasta)
```

correlationPlotLiP *Plot run correlation for provided LiP and TrP experiment.*

Description

Plot run correlation for provided LiP and TrP experiment.

Usage

```
correlationPlotLiP(
  data,
  method = "pearson",
  value_columns = "INTENSITY",
  x.axis.size = 10,
  y.axis.size = 10,
  legend.size = 10,
  width = 10,
  height = 10,
  address = ""
)
```

Arguments

data	output of MSstatsLiP converter function. Must include at least ProteinName, Run, and Intensity columns
method	one of "pearson", "kendall", "spearman". Default is pearson.
value_columns	one of "INTENSITY" or "ABUNDANCE". INTENSITY is the raw data, whereas ABUNDANCE is the log transformed INTENSITY column. INTENSITY is default.
x.axis.size	size of axes labels, e.g. name of the comparisons in heatmap, and in comparison plot. Default is 10.
y.axis.size	size of axes labels, e.g. name of targeted proteins in heatmap. Default is 10.
legend.size	size of legend for color at the bottom of volcano plot. Default is 10.
width	width of the saved file. Default is 10.
height	height of the saved file. Default is 10.
address	the name of folder that will store the results. Default folder is the current working directory. The other assigned folder has to be existed under the current working directory. An output pdf file is automatically created with the default name of "VolcanoPlot.pdf" or "Heatmap.pdf". The command address can help to specify where to store the file as well as how to modify the beginning of the file name. If address=FALSE, plot will be not saved as pdf file but showed in window

Value

plot or pdf

Examples

```
## Use output of dataSummarizationLiP function
correlationPlotLiP(MSstatsLiP_Summarized, address = FALSE)
```

---

dataProcessPlotsLiP *Visualization for explanatory data analysis*

Description

To illustrate the quantitative data and quality control of MS runs, dataProcessPlotsLiP takes the quantitative data from MSstatsLiP converter functions as input and generate two types of figures in pdf files as output : (1) profile plot (specify "ProfilePlot" in option type), to identify the potential sources of variation for each protein; (2) quality control plot (specify "QCPlot" in option type), to evaluate the systematic bias between MS runs.

Usage

```
dataProcessPlotsLiP(
  data,
  type = "PROFILEPLOT",
  ylimUp = FALSE,
  ylimDown = FALSE,
  x.axis.size = 10,
  y.axis.size = 10,
  text.size = 4,
  text.angle = 90,
  legend.size = 7,
  dot.size.profile = 2,
  ncol.guide = 5,
  width = 10,
  height = 12,
  lip.title = "All Peptides",
  protein.title = "All Proteins",
  which.Peptide = "all",
  which.Protein = NULL,
  originalPlot = TRUE,
  summaryPlot = TRUE,
  address = ""
)
```

Arguments

`data`	name of the list with LiP and (optionally) Protein data, which can be the output of the MSstatsLiP. `dataSummarizationLiP` function.
`type`	choice of visualization. "ProfilePlot" represents profile plot of log intensities across MS runs. "QCPlot" represents box plots of log intensities across channels and MS runs.
`ylimUp`	upper limit for y-axis in the log scale. FALSE(Default) for Profile Plot and QC Plot uses the upper limit as rounded off maximum of log2(intensities) after normalization + 3..
`ylimDown`	lower limit for y-axis in the log scale. FALSE(Default) for Profile Plot and QC Plot uses 0..
`x.axis.size`	size of x-axis labeling for "Run" and "channel in Profile Plot and QC Plot.
`y.axis.size`	size of y-axis labels. Default is 10.
`text.size`	size of labels represented each condition at the top of Profile plot and QC plot. Default is 4.
`text.angle`	angle of labels represented each condition at the top of Profile plot and QC plot. Default is 0.
`legend.size`	size of legend above Profile plot. Default is 7.
`dot.size.profile`	size of dots in Profile plot. Default is 2.

ncol.guide	number of columns for legends at the top of plot. Default is 5.
width	width of the saved pdf file. Default is 10.
height	height of the saved pdf file. Default is 10.
lip.title	title of all LiP QC plot
protein.title	title of all Protein QC plot
which.Peptide	LiP peptide list to draw plots. List can be names of LiP peptides or order numbers of LiPs. Default is "all", which generates all plots for each protein. For QC plot, "allonly" will generate one QC plot with all proteins.
which.Protein	String of protein's to plot if the user would like to plot all Peptides associated with a given Protein. Default is NULL. Please do not include "all" or "allonly" here.
originalPlot	TRUE(default) draws original profile plots, without normalization.
summaryPlot	TRUE(default) draws profile plots with protein summarization for each channel and MS run.
address	the name of folder that will store the results. Default folder is the current working directory. The other assigned folder has to be existed under the current working directory. An output pdf file is automatically created with the default name of "ProfilePlot.pdf" or "QCplot.pdf". The command address can help to specify where to store the file as well as how to modify the beginning of the file name. If address=FALSE, plot will be not saved as pdf file but showed in window.

Value

plot or pdf

Examples

```
# Use the output of the MSstatsLiP_Summarized function
# Profile Plot
dataProcessPlotsLiP(MSstatsLiP_Summarized, type = "ProfilePlot")

# QCPlot Plot
dataProcessPlotsLiP(MSstatsLiP_Summarized, type = "QCPlot")
```

---

dataSummarizationLiP *Summarizes LiP and TrP datasets seperately using methods from MSstats.*

Description

Utilizes functionality from MSstats and MSstatsPTM to clean, summarize, and normalize LiP peptide and TrP global protein data. Imputes missing values, protein and LiP peptide level summarization from peptide level quantification. Applies global median normalization on peptide level data and normalizes between runs. Returns list of two summarized datasets.

Usage

```
dataSummarizationLiP(
  data,
  logTrans = 2,
  normalization = "equalizeMedians",
  normalization.LiP = "equalizeMedians",
  nameStandards = NULL,
  nameStandards.LiP = NULL,
  featureSubset = "all",
  featureSubset.LiP = "all",
  remove_uninformative_feature_outlier = FALSE,
  remove_uninformative_feature_outlier.LiP = FALSE,
  min_feature_count = 2,
  min_feature_count.LiP = 1,
  n_top_feature = 3,
  n_top_feature.LiP = 3,
  summaryMethod = "TMP",
  equalFeatureVar = TRUE,
  censoredInt = "NA",
  MBimpute = TRUE,
  MBimpute.LiP = FALSE,
  remove50missing = FALSE,
  fix_missing = NULL,
  maxQuantileforCensored = 0.999,
  use_log_file = FALSE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL,
  base = "MSstatsLiP_log_"
)
```

Arguments

data	name of the list with LiP and TrP data.tables, which can be the output of the MSstatsPTM converter functions
logTrans	logarithm transformation with base 2(default) or 10
normalization	normalization for the protein level dataset, to remove systematic bias between MS runs. There are three different normalizations supported. 'equalizeMedians'(default) represents constant normalization (equalizing the medians) based on reference signals is performed. 'quantile' represents quantile normalization based on reference signals is performed. 'globalStandards' represents normalization with global standards proteins. FALSE represents no normalization is performed
normalization.LiP	normalization for LiP level dataset. Default is 'equalizeMedians'. Can be adjusted to any of the options described above.
nameStandards	vector of global standard peptide names for protein dataset. only for normalization with global standard peptides.

`nameStandards.LiP`
: Same as above for LiP dataset.

`featureSubset`
: For protein dataset only. "all"(default) uses all features that the data set has. "top3" uses top 3 features which have highest average of log2(intensity) across runs. "topN" uses top N features which has highest average of log2(intensity) across runs. It needs the input for n_top_feature option. "highQuality" flags uninformative feature and outliers

`featureSubset.LiP`
: For LiP dataset only. Options same as above.

`remove_uninformative_feature_outlier`
: For protein dataset only. It only works after users used featureSubset="highQuality" in dataProcess. TRUE allows to remove 1) the features are flagged in the column, feature_quality="Uninformative" which are features with bad quality, 2) outliers that are flagged in the column, is_outlier=TRUE, for run-level summarization. FALSE (default) uses all features and intensities for run-level summarization.

`remove_uninformative_feature_outlier.LiP`
: For LiP dataset only. Options same as above.

`min_feature_count`
: optional. Only required if featureSubset = "highQuality". Defines a minimum number of informative features a protein needs to be considered in the feature selection algorithm.

`min_feature_count.LiP`
: For LiP dataset only. Options the same as above.

`n_top_feature`
: For protein dataset only. The number of top features for featureSubset='topN'. Default is 3, which means to use top 3 features.

`n_top_feature.LiP`
: For LiP dataset only. Options same as above.

`summaryMethod`
: "TMP"(default) means Tukey's median polish, which is robust estimation method. "linear" uses linear mixed model.

`equalFeatureVar`
: only for summaryMethod="linear". default is TRUE. Logical variable for whether the model should account for heterogeneous variation among intensities from different features. Default is TRUE, which assume equal variance among intensities from features. FALSE means that we cannot assume equal variance among intensities from features, then we will account for heterogeneous variation from different features.

`censoredInt`
: Missing values are censored or at random. 'NA' (default) assumes that all 'NA's in 'Intensity' column are censored. '0' uses zero intensities as censored intensity. In this case, NA intensities are missing at random. The output from Skyline should use '0'. Null assumes that all NA intensites are randomly missing.

`MBimpute`
: For protein dataset only. only for summaryMethod="TMP" and censoredInt='NA' or '0'. TRUE (default) imputes 'NA' or '0' (depending on censoredInt option) by Accelated failure model. FALSE uses the values assigned by cutoffCensored.

`MBimpute.LiP`
: For LiP dataset only. Options same as above. Default is FALSE.

remove50missing
: only for summaryMethod="TMP". TRUE removes the runs which have more than 50% missing values. FALSE is default.

fix_missing
: Default is Null. Optional, same as the 'fix_missing' parameter in MSstatsConvert::MSstatsBalancedDesign function

maxQuantileforCensored
: Maximum quantile for deciding censored missing values. default is 0.999

use_log_file
: logical. If TRUE, information about data processing will be saved to a file.

append
: logical. If TRUE, information about data processing will be added to an existing log file.

verbose
: logical. If TRUE, information about data processing will be printed to the console.

log_file_path
: character. Path to a file to which information about data processing will be saved. If not provided, such a file will be created automatically. If append = TRUE, has to be a valid path to a file.

base
: start of the file name.

Value

list of summarized LiP and TrP results. These results contain the reformatted input to the summarization function, as well as run-level summarization results.

Examples

```
# Use output of converter
head(MSstatsLiP_data[["LiP"]])
head(MSstatsLiP_data[["TrP"]])

# Run summarization
MSstatsLiP_model <- dataSummarizationLiP(MSstatsLiP_data)
```

---

groupComparisonLiP *Model LiP and TrP data and make adjustments if needed Returns list of three modeled datasets*

Description

Takes summarized LiP peptide and TrP protein data from dataSummarizationLiP If global protein data is unavailable, LiP data only can be passed into the function. Including protein data allows for adjusting LiP Fold Change by the change in global protein abundance..

Usage

```
groupComparisonLiP(
  data,
  contrast.matrix = "pairwise",
  fasta.path = NULL,
  log_base = 2,
  use_log_file = FALSE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL,
  base = "MSstatsLiP_log_"
)
```

Arguments

data	list of summarized datasets. Can be output of MSstatsLiP summarization function dataSummarizationLiP. Must include dataset named "LiP" as minimum.
contrast.matrix	comparison between conditions of interests. Default models full pairwise comparison between all conditions
fasta.path	a file path to a fasta file that includes the proteins listed in the data. Default is NULL. Include this parameter to determine trypticity of peptides in LiP models.
log_base	base of the logarithm used in dataProcess.
use_log_file	logical. If TRUE, information about data processing will be saved to a file.
append	logical. If TRUE, information about data processing will be added to an existing log file.
verbose	logical. If TRUE, information about data processing will be printed to the console.
log_file_path	character. Path to a file to which information about data processing will be saved. If not provided, such a file will be created automatically. If append = TRUE, has to be a valid path to a file.
base	start of the file name.

Value

list of modeling results. Includes LiP, PROTEIN, and ADJUSTED LiP data.tables with their corresponding model results.

Examples

```
## Use output of dataSummarizationLiP function
fasta <- system.file("extdata", "ExampleFastaFile.fasta", package="MSstatsLiP")

# Test for pairwise comparison
MSstatsLiP_model <- groupComparisonLiP(MSstatsLiP_Summarized,
                                       contrast.matrix = "pairwise",
                                       fasta.path = fasta)
```

```
# Returns list of three models
names(MSstatsLiP_model)
head(MSstatsLiP_model$LiP.Model)
head(MSstatsLiP_model$TrP.Model)
head(MSstatsLiP_model$Adjusted.LiP.Model)
```

---

groupComparisonPlotsLiP *Visualization for model-based analysis and summarization*

Description

To analyze the results of modeling changes in abundance of LiP peptides and overall protein, groupComparisonPlotsLiP takes as input the results of the groupComparisonLiP function. It asses the results of three models: unadjusted LiP, adjusted LiP, and overall protein. To asses the results of the model, the following visualizations can be created: (1) VolcanoPlot (specify "VolcanoPlot" in option type), to plot peptides or proteins and their significance for each model. (2) Heatmap (specify "Heatmap" in option type), to evaluate the fold change between conditions and peptides/proteins

Usage

```
groupComparisonPlotsLiP(
  data = data,
  type = type,
  sig = 0.05,
  FCcutoff = 1,
  logBase.pvalue = 10,
  ylimUp = FALSE,
  ylimDown = FALSE,
  xlimUp = FALSE,
  x.axis.size = 10,
  y.axis.size = 10,
  dot.size = 3,
  text.size = 4,
  text.angle = 0,
  legend.size = 13,
  ProteinName = TRUE,
  colorkey = TRUE,
  numProtein = 100,
  width = 10,
  height = 10,
  which.Comparison = "all",
  which.Peptide = "all",
  which.Protein = NULL,
  address = ""
)
```

Arguments

data	name of the list with models, which can be the output of the MSstatsLiP groupComparisonLiP function
type	choice of visualization, one of VolcanoPlot or Heatmap
sig	FDR cutoff for the adjusted p-values in heatmap and volcano plot. level of significance for comparison plot. 100(1-sig)% confidence interval will be drawn. sig=0.05 is default.
FCcutoff	or volcano plot or heatmap, whether involve fold change cutoff or not. FALSE (default) means no fold change cutoff is applied for significance analysis. FCcutoff = specific value means specific fold change cutoff is applied.
logBase.pvalue	for volcano plot or heatmap, (-) logarithm transformation of adjusted p-value with base 2 or 10(default).
ylimUp	for all three plots, upper limit for y-axis. FALSE (default) for volcano plot/heatmap use maximum of -log2 (adjusted p-value) or -log10 (adjusted p-value). FALSE (default) for comparison plot uses maximum of log-fold change + CI.
ylimDown	for all three plots, lower limit for y-axis. FALSE (default) for volcano plot/heatmap use minimum of -log2 (adjusted p-value) or -log10 (adjusted p-value). FALSE (default) for comparison plot uses minimum of log-fold change - CI.
xlimUp	for Volcano plot, the limit for x-axis. FALSE (default) for use maximum for absolute value of log-fold change or 3 as default if maximum for absolute value of log-fold change is less than 3.
x.axis.size	size of axes labels, e.g. name of the comparisons in heatmap, and in comparison plot. Default is 10.
y.axis.size	size of axes labels, e.g. name of targeted proteins in heatmap. Default is 10.
dot.size	size of dots in volcano plot and comparison plot. Default is 3.
text.size	size of ProteinName label in the graph for Volcano Plot. Default is 4.
text.angle	angle of x-axis labels represented each comparison at the bottom of graph in comparison plot. Default is 0.
legend.size	size of legend for color at the bottom of volcano plot. Default is 7.
ProteinName	for volcano plot only, whether display protein/peptide names or not. TRUE (default) means protein names, which are significant, are displayed next to the points. FALSE means no protein names are displayed.
colorkey	TRUE(default) shows colorkey.
numProtein	The number of proteins which will be presented in each heatmap. Default is 50.
width	width of the saved file. Default is 10.
height	height of the saved file. Default is 10.
which.Comparison	list of comparisons to draw plots. List can be labels of comparisons or order numbers of comparisons from levels(data$Label) , such as levels(testResultMultiComparisons$Comparisc Default is "all", which generates all plots for each protein.
which.Peptide	Peptide list to draw comparison plots. List can be names of Peptides or order numbers of Peptides from levels. Default is "all", which generates all comparison plots for each protein.

which.Protein	Protein list to draw comparison plots. Will draw all peptide plots for listed Proteins. List must be names of Proteins. Default is "all", which generates all comparison plots for each protein.
address	the name of folder that will store the results. Default folder is the current working directory. The other assigned folder has to be existed under the current working directory. An output pdf file is automatically created with the default name of "VolcanoPlot.pdf" or "Heatmap.pdf". The command address can help to specify where to store the file as well as how to modify the beginning of the file name. If address=FALSE, plot will be not saved as pdf file but showed in window

Value

plot or pdf

Examples

```
## Use output of the groupComparisonLiP function

# Volcano Plot
groupComparisonPlotsLiP(MSstatsLiP_model, type = "VOLCANOPLOT")

# Heatmap Plot
groupComparisonPlotsLiP(MSstatsLiP_model, type = "HEATMAP")
```

LiPRawData *LiPRawData*

Description

Example of input LiP dataset.

Usage

```
LiPRawData
```

Format

A data.table consisting of 546 rows and 29 columns. Raw LiP data for use in testing and examples.

Details

Input to MSstatsLiP converter SpectronauttoMSstatsLiPFormat. Contains the following columns:

- R.Condition : Label of conditions (EG Disease/Control)
- R.FileName : Name of spectral processing run
- R.Replicate : Name of biological replicate

- PG.ProteinAccessions : Protein name
- PG.ProteinGroups : Protein name, can be multiple
- PG.Quantity : Protein Quantity
- PEP.GroupingKey : Peptide grouping
- PEP.StrippedSequence : Peptide sequence
- PEP.Quantity : Peptide quantity
- EG.iRTPredicted : Predicted value
- EG.Library : Name of library
- EG.ModifiedSequence : Peptide sequence including any post-translational modifications
- EG.PrecursorId : Peptide sequence wiht modifications including charge
- EG.Qvalue : Qvalue
- FG.Charge : Identified Ion charge
- FG.Id : Peptide sequence with charge
- FG.PrecMz : Prec Mz reading
- FG.Quantity : Initial quantity reading
- F.Charge : F.Charge
- F.FrgIon : Fragment ion
- F.FrgLossType : Label for loss type
- F.FrgMz : Mz reading
- F.FrgNum : numeric Frg
- F.FrgType : character label for Frg
- F.ExcludedFromQuantification : True/False boolean for if to exclude
- F.NormalizedPeakArea : Normalized peak intensity
- F.NormalizedPeakHeight : Normalized peak height
- F.PeakArea : Unnormalized peak area
- F.PeakHeight : Unnormalized peak height

Examples

```
head(LiPRawData)
```

locateMod *Locate modified sites with a peptide*

Description

locateMod locates modified sites with a peptide.

Usage

```
locateMod(peptide, aaStart, residueSymbol)
```

Arguments

peptide	A string. Peptide sequence.
aaStart	An integer. Starting index of the peptide.
residueSymbol	A string. Modification residue and denoted symbol.

Value

A string.

Examples

```
locateMod("P*EP*TIDE", 3, "\\*")
```

locatePTM *Annotate modified sites with associated peptides*

Description

PTMlocate annotates modified sites with associated peptides.

Usage

```
locatePTM(peptide, uniprot, fasta, modResidue, modSymbol, rmConfound = FALSE)
```

Arguments

peptide	A string vector of peptide sequences. The peptide sequence does not include its preceding and following AAs.
uniprot	A string vector of Uniprot identifiers of the peptides' originating proteins. UniProtKB entry isoform sequence is used.
fasta	A tibble with FASTA information. Output of tidyFasta.
modResidue	A string. Modifiable amino acid residues.

modSymbol	A string. Symbol of a modified site.
rmConfound	A logical. TRUE removes confounded unmodified sites, FALSE otherwise. Default is FALSE.

Value

A data frame with three columns: uniprot_iso, peptide, site.

Examples

```
fasta <- tidyFasta(system.file("extdata", "O13297.fasta", package="MSstatsLiP"))
locatePTM("DRVSYIHNDSC*TR", "O13297", fasta, "C", "\\*")
```

MSstatsLiP *MSstatsLiP: A package for identifying and analyzing changes in protein structures caused by compound binding in cellur lysates.*

Description

A set of tools for detecting differentially abundant LiP peptides in shotgun mass spectrometry-based proteomic experiments. The package includes tools to convert raw data from different spectral processing tools, summarize feature intensities, and fit a linear mixed effects model. If overall protein abundance changes are included, the package will also adjust the LiP peptide fold change for changes in overall protein abundace. Additionally the package includes functionality to plot a variety of data visualizations.

functions

- SpectronauttoMSstatsLiPFormat : Generates MSstatsLiP required input format for Spectronaut outputs.
- trypticHistogramLiP : Histogram of Half vs Fully tryptic peptides. Calculates proteotypicity, and then uses calcualtions in histogram.
- correlationPlotLiP : Plot run correlation for provided LiP and TrP experiment.
- dataSummarizationLiP : Summarizes PSM level quantification to peptide (LiP) and protein level quantification.
- dataProcessPlotsLiP : Visualization for explanatory data analysis. Specifically gives ability to plot Profile and Quality Control plots.
- PCAPlotLiP :Visualize PCA analysis for LiP and TrP datasets. Specifically gives ability to plot explanined varaince per component, Protein/Peptide PCA, and Condition PCA.
- groupComparisonLiP : Tests for significant changes in LiP and protein abundance across conditions. Adjusts LiP fold change for changes in protein abundance.
- groupComparisonPlotsLiP : Visualization for model-based analysis and summarization.
- PCAPlotLiP : Runs PCA on the summarized data. Can visualize the PCA analysis in three different plots.
- BarcodePlotLiP : Shows protein coverage of LiP modified peptides. Shows significant, insignificant, and missing coverage.

MSstatsLiP_data *MSstatsLiP_data*

Description

Example output of MSstatsLiP converter functions.

Usage

```
MSstatsLiP_data
```

Format

A data.table consisting of 546 rows and 29 columns. Raw TrP data for use in testing and examples.

Details

Example output of MSstatsLiP converter functions. (Eg. SpectronauttoMSstatsLiPFormat). A list containing two data.tables named LiP and TrP corresponding to the processed LiP and TrP data now in MSstatsLiP format. The data.tables contain the following columns:

- ProteinName : Character column of protein names
- PeptideSequence : Character column of peptide sequence name
- PrecursorCharge : Numeric charge feature
- FragmentIon : Character fragment ion feature
- ProductCharge : Numeric charge of product
- IsotopeLabelType : Character label type
- Condition : Character label for condition (Eg. Disease/Control)
- BioReplicate : Name of biological replicate
- Run : Name of run
- Fraction : Fraction number if fractionation is present
- Intensity : Unnormalized feature intensity
- FULL_PEPTIDE(LiP data only) : Combined protein name and peptide sequence. Used for LiP data only because LiP is summarized to peptide level (not protein)

Examples

```
head(MSstatsLiP_data$LiP)
head(MSstatsLiP_data$TrP)
```

`MSstatsLiP_model` *MSstatsLiP_model*

Description

Example output of groupComparisonLiP converter functions.

Usage

```
MSstatsLiP_model
```

Format

A data.table consisting of 546 rows and 29 columns. Raw TrP data for use in testing and examples.

Details

Example output of MSstatsLiP groupComparisonLiP function. A list containing three data.tables corresponding to unadjusted LiP, TrP, and adjusted LiP models. The data.tables contain the following columns:

- ProteinName : Character column of protein names
- PeptideSequence : Character column of peptide sequence name
- Label : Condition comparison (Eg. Disease vs Control)
- log2FC : Fold Change output results of model
- SE : Standard error output of model
- Tvalue : Tvalue output of model
- DF : Degrees of Freedom output of model
- pvalue : Pvalue result of model (unadjusted)
- adj.pvalue : Adjusted Pvalue, generally BH adjustement is used
- issue : Issue in model if any is reported
- MissingPercentage : Percent of missing values in specific model
- ImputationPercentage : Percent of values that needed to be imputed
- fully_TRI: Boolean indicating if Peptide is fully tryptic
- NSEMI_TRI: Boolean indicating if Peptide is NSEMI tryptic
- CSEMI_TRI: Boolean indicating if Peptide is CSEMI tryptic
- CTERMINUS: Boolean indicating if Peptide is CTERMINUS tryptic
- NTERMINUS: Boolean indicating if Peptide is NTERMINUS tryptic
- StartPos: Start position of peptide sequence
- EndPos: End position of peptide sequence
- FULL_PEPTIDE(LiP data only) : Combined protein name and peptide sequence. Used for LiP data only because LiP is summarized to peptide level (not protein)

Examples

```
head(MSstatsLiP_model$LiP.Model)
head(MSstatsLiP_model$TrP.Model)
head(MSstatsLiP_model$Adjusted.LiP.Model)
```

MSstatsLiP_Summarized *MSstatsLiP_Summarized*

Description

Example output of MSstatsLiP summarization function dataSummarizationLiP.

Usage

```
MSstatsLiP_Summarized
```

Format

A list containing two lists of summarization information for LiP and TrP data.

Details

Example output of MSstatsLiP summarization function dataSummarizationLiP. A list containing two lists named LiP and TrP containing summarization information for LiP and TrP data. Each of LiP and TrP contain data named: FeatureLevelData, ProteinLevelData, SummaryMethod, ModelQC, PredictBySurvival. The two main data.tables (FeatureLevelData and ProteinLevelData are shown below):

- FeatureLevelData :
  - PROTEIN : Protein ID with modification site mapped in. Ex. Protein_1002_S836
  - FULL_PEPTIDE (LiP Only) : Combined name of protein and peptide sequence
  - PEPTIDE : Full peptide with charge
  - TRANSITION: Charge
  - FEATURE : Combination of Protien, Peptide, and Transition Columns
  - LABEL :
  - GROUP : Condition (ex. Healthy, Cancer, Time0)
  - RUN : Unique ID for technical replicate of one TMT mixture.
  - SUBJECT : Unique ID for biological subject.
  - FRACTION : Unique Fraction ID
  - originalRUN : Run name
  - censored :
  - INTENSITY : Original intensity value
  - ABUNDANCE : Log adjusted intensity value
  - newABUNDANCE : Normalized abundance column

- ProteinLevelData :
  - RUN : MS run ID
  - FULL_PEPTIDE (LiP Only) : Combined name of protein and peptide sequence
  - Protein : Protein ID with modification site mapped in. Ex. Protein_1002_S836
  - LogIntensities: Protein-level summarized abundance
  - originalRUN : Labeling information (126, ... 131)
  - GROUP : Condition (ex. Healthy, Cancer, Time0)
  - SUBJECT : Unique ID for biological subject.
  - TotalGroupMeasurements : Unique ID for technical replicate of one TMT mixture.
  - NumMeasuredFeature : Unique ID for TMT mixture.
  - MissingPercentage : Unique ID for TMT mixture.
  - more50missing : Unique ID for TMT mixture.
  - NumImputedFeature : Unique ID for TMT mixture.

Examples

```
head(MSstatsLiP_Summarized$LiP$FeatureLevelData)
head(MSstatsLiP_Summarized$LiP$ProteinLevelData)

head(MSstatsLiP_Summarized$TrP$FeatureLevelData)
head(MSstatsLiP_Summarized$TrP$ProteinLevelData)
```

PCAPlotLiP *Visualize PCA analysis for LiP and TrP datasets.*

Description

Takes as input LiP and TrP data from summarization function dataSummarizationLiP. Runs PCA on the summarized data. Can visualize the PCA analysis in three different plots: (1) BarPlot (specify "bar.plot=TRUE" in option bar.plot), to plot a bar plot showing the explained variance per PCA component (2) Peptide/Protein PCA (specify "protein.pca = TRUE" in option protein.pca), to create a dot plot with PCA component 1 and 2 on the axis, for different peptides and proteins. (3) Comparison PCA (specify "comparison.pca = TRUE" in option comparison.pca) , to create a arrow plot with PCA component 1 and 2 on the axis, for different comparisons

Usage

```
PCAPlotLiP(
  data,
  center.pca = TRUE,
  scale.pca = TRUE,
  n.components = 10,
  bar.plot = TRUE,
  protein.pca = TRUE,
  comparison.pca = FALSE,
```

```
  which.pep = "all",
  which.comparison = "all",
  width = 10,
  height = 10,
  address = ""
)
```

Arguments

data	data name of the list with LiP and (optionally) Protein data, which can be the output of the MSstatsLiP. dataSummarizationLiP function.
center.pca	a logical value indicating whether the variables should be shifted to be zero centered. Alternately, a vector of length equal the number of columns of x can be supplied. The value is passed to scale
scale.pca	a logical value indicating whether the variables should be scaled to have unit variance before the analysis takes place. The default is FALSE for consistency with S, but in general scaling is advisable. Alternatively, a vector of length equal the number of columns of x can be supplied. The value is passed to scale.
n.components	an integer of PCA components to be returned. Default is 10.
bar.plot	a logical value indicating if to visualize PCA bar plot
protein.pca	a logical value indicating if to visualize PCA peptide plot
comparison.pca	a logical value indicating if to visualize PCA comparison plot
which.pep	a list of peptides to be visualized. Default is "all". If too many peptides are plotted the names can overlap.
which.comparison	a list of comparisons to be visualized. Default is "all".
width	width of the saved file. Default is 10.
height	height of the saved file. Default is 10.
address	the name of folder that will store the results. Default folder is the current working directory. The other assigned folder has to be existed under the current working directory. An output pdf file is automatically created with the default name of "VolcanoPlot.pdf" or "Heatmap.pdf". The command address can help to specify where to store the file as well as how to modify the beginning of the file name. If address=FALSE, plot will be not saved as pdf file but showed in window

Value

plot or pdf

Examples

```
# Use output of dataSummarizationLiP function

# BarPlot
PCAPlotLiP(MSstatsLiP_Summarized, bar.plot = TRUE, protein.pca = FALSE)
```

```
# Protein/Peptide PCA Plot
PCAPlotLiP(MSstatsLiP_Summarized, bar.plot = FALSE, protein.pca = TRUE)

# Condition PCA Plot
PCAPlotLiP(MSstatsLiP_Summarized, bar.plot = FALSE, protein.pca = FALSE,
           comparison.pca = TRUE)
```

SkylineTest *SkylineTest*

Description

Example of input data from Skylinet.

Usage

```
SkylineTest
```

Format

A data.table consisting of 2115 rows and 13 columns. Raw data for use in testing and examples.

Details

Input to MSstatsLiP converter SkylinetoMSstatsLiPFormat Contains the following columns:

- Protein.Name : Name of Proteins identified by Skyline
- Peptide.Modified.Sequence : Peptide sequence
- Precursor.Charge : Charge of ion
- Fragment.Ion : Fragment ion
- Product.Charge : Identified Ion charge
- Isotope.Label.Type : Label Type
- Condition : Name of condition
- BioReplicate : name of bioreplicate annotated to data
- File.Name : Name of spectral processing run
- Area : Abudance area
- Standard.Type : Type name for row
- Truncated : Boolean if row was truncated

Examples

```
head(SkylineTest)
```

SkylinetoMSstatsLiPFormat *Converts raw LiP MS data from Skyline into the format needed for MSstatsLiP.*

Description

Takes as as input both raw LiP and Trp outputs from Skyline.

Usage

```
SkylinetoMSstatsLiPFormat(
  LiP.data,
  TrP.data = NULL,
  annotation = NULL,
  removeiRT = TRUE,
  filter_with_Qvalue = TRUE,
  qvalue_cutoff = 0.01,
  useUniquePeptide = TRUE,
  removeFewMeasurements = TRUE,
  removeOxidationMpeptides = FALSE,
  removeProtein_with1Feature = FALSE,
  use_log_file = FALSE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL
)
```

Arguments

LiP.data	name of LiP Skyline output, which is long-format.
TrP.data	name of TrP Skyline output, which is long-format.
annotation	name of 'annotation.txt' data which includes Condition, BioReplicate, Run. If annotation is already complete in Skyline, use annotation=NULL (default). It will use the annotation information from input.
removeiRT	TRUE (default) will remove the proteins or peptides which are labeld 'iRT' in 'StandardType' column. FALSE will keep them.
filter_with_Qvalue	TRUE(default) will filter out the intensities that have greater than qvalue_cutoff in DetectionQValue column. Those intensities will be replaced with zero and will be considered as censored missing values for imputation purpose.
qvalue_cutoff	Cutoff for DetectionQValue. default is 0.01.
useUniquePeptide	TRUE (default) removes peptides that are assigned for more than one proteins. We assume to use unique peptide for each protein.

`removeFewMeasurements`
: TRUE (default) will remove the features that have 1 or 2 measurements across runs.

`removeOxidationMpeptides`
: TRUE will remove the peptides including 'oxidation (M)' in modification. FALSE is default.

`removeProtein_with1Feature`
: TRUE will remove the proteins which have only 1 feature, which is the combination of peptide, precursor charge, fragment and charge. FALSE is default.

`use_log_file`
: logical. If TRUE, information about data processing will be saved to a file.

`append`
: logical. If TRUE, information about data processing will be saved to a file.

`verbose`
: logical. If TRUE, information about data processing wil be printed to the console.

`log_file_path`
: character. Path to a file to which information about data processing will be saved. If not provided, such a file will be created automatically. If 'append = TRUE', has to be a valid path to a file.

Value

a `list` of two data.frames in `MSstatsLiP` format

Examples

```
## Output will be in format
head(MSstatsLiP_data[["LiP"]])
head(MSstatsLiP_data[["TrP"]])
```

SpectronauttoMSstatsLiPFormat

Converts raw LiP MS data from Spectronautt into the format needed for MSstatsLiP.

Description

Takes as as input both raw LiP and Trp outputs from Spectronautt.

Usage

```
SpectronauttoMSstatsLiPFormat(
  LiP.data,
  fasta,
  Trp.data = NULL,
  annotation = NULL,
  intensity = "PeakArea",
  filter_with_Qvalue = TRUE,
  qvalue_cutoff = 0.01,
```

```
  useUniquePeptide = TRUE,
  removeFewMeasurements = TRUE,
  removeProtein_with1Feature = FALSE,
  removeNonUniqueProteins = TRUE,
  removeModifications = TRUE,
  removeiRT = TRUE,
  summaryforMultipleRows = max,
  which.Conditions = "all",
  use_log_file = FALSE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL,
  base = "MSstatsLiP_log_"
)
```

Arguments

LiP.data	name of LiP Spectronaut output, which is long-format.
fasta	A string of path to a FASTA file, used to match LiP peptides.
Trp.data	name of TrP Spectronaut output, which is long-format.
annotation	name of 'annotation.txt' data which includes Condition, BioReplicate, Run. If annotation is already complete in Spectronaut, use annotation=NULL (default). It will use the annotation information from input.
intensity	'PeakArea'(default) uses not normalized peak area. 'NormalizedPeakArea' uses peak area normalized by Spectronaut
filter_with_Qvalue	TRUE(default) will filter out the intensities that have greater than qvalue_cutoff in EG.Qvalue column. Those intensities will be replaced with zero and will be considered as censored missing values for imputation purpose.
qvalue_cutoff	Cutoff for EG.Qvalue. default is 0.01.
useUniquePeptide	TRUE(default) removes peptides that are assigned for more than one proteins. We assume to use unique peptide for each protein.
removeFewMeasurements	TRUE (default) will remove the features that have 1 or 2 measurements across runs.
removeProtein_with1Feature	TRUE will remove the proteins which have only 1 feature, which is the combination of peptide, precursor charge, fragment and charge. FALSE is default.
removeNonUniqueProteins	TRUE will remove proteins that were not uniquely identified. IE if the protein column contains multiple proteins seperated by ";". TRUE is default
removeModifications	TRUE will remove peptide that contain a modification. Modification must be indicated by "[". TRUE is default
removeiRT	TRUE will remove proteins that contain iRT. True is default

summaryforMultipleRows
: max(default) or sum - when there are multiple measurements for certain feature and certain run, use highest or sum of multiple intensities.

which.Conditions
: list of conditions to format into MSstatsLiP format. If "all" all conditions will be used. Default is "all".

use_log_file
: logical. If TRUE, information about data processing will be saved to a file.

append
: logical. If TRUE, information about data processing will be added to an existing log file.

verbose
: logical. If TRUE, information about data processing will be printed to the console.

log_file_path
: character. Path to a file to which information about data processing will be saved. If not provided, such a file will be created automatically. If append = TRUE, has to be a valid path to a file.

base
: start of the file name.

Value

a list of two data.frames in MSstatsLiP format

Examples

```
# Output datasets of Spectronaut
head(LiPRawData)
head(TrPRawData)

fasta_path <- system.file("extdata", "ExampleFastaFile.fasta", package="MSstatsLiP")

MSstatsLiP_data <- SpectronauttoMSstatsLiPFormat(LiPRawData,
                                                 fasta_path,
                                                 TrPRawData)
head(MSstatsLiP_data[["LiP"]])
head(MSstatsLiP_data[["TrP"]])
```

---

tidyFasta *Read and tidy a FASTA file*

Description

reads and tidys FASTA file.

Usage

```
tidyFasta(path)
```

Arguments

path
: a string of path pointing towards a fasta file

Value

a `tibble` of formatted FASTA information

Examples

```
tidyFasta(system.file("extdata", "O13297.fasta", package="MSstatsLiP"))
```

`TrPRawData` *TrPRawData*

Description

Example of input TrP dataset.

Usage

```
TrPRawData
```

Format

A data.table consisting of 4692 rows and 29 columns. Raw TrP data for use in testing and examples.

Details

Input to MSstatsLiP converter SpectronauttoMSstatsLiPFormat. Contains the following columns:

- R.Condition : Label of conditions (EG Disease/Control)
- R.FileName : Name of spectral processing run
- R.Replicate : Name of biological replicate
- PG.ProteinAccessions : Protein name
- PG.ProteinGroups : Protein name, can be multiple
- PG.Quantity : Protein Quantity
- PEP.GroupingKey : Peptide grouping
- PEP.StrippedSequence : Peptide sequence
- PEP.Quantity : Peptide quantity
- EG.iRTPredicted : Predicted value
- EG.Library : Name of library
- EG.ModifiedSequence : Peptide sequence including any post-translational modifications
- EG.PrecursorId : Peptide sequence wiht modifications including charge
- EG.Qvalue : Qvalue
- FG.Charge : Identified Ion charge
- FG.Id : Peptide sequence with charge
- FG.PrecMz : Prec Mz reading

- FG.Quantity : Initial quantity reading
- F.Charge : F.Charge
- F.FrgIon : Fragment ion
- F.FrgLossType : Label for loss type
- F.FrgMz : Mz reading
- F.FrgNum : numeric Frg
- F.FrgType : character label for Frg
- F.ExcludedFromQuantification : True/False boolean for if to exclude
- F.NormalizedPeakArea : Normalized peak intensity
- F.NormalizedPeakHeight : Normalized peak height
- F.PeakArea : Unnormalized peak area
- F.PeakHeight : Unnormalized peak height

Examples

```
head(TrPRawData)
```

`trypticHistogramLiP` *Histogram of Half vs Fully tryptic peptides. Calculates proteotypicity, and then uses calcualtions in histogram.*

Description

Histogram of Half vs Fully tryptic peptides. Calculates proteotypicity, and then uses calcualtions in histogram.

Usage

```
trypticHistogramLiP(
  data,
  fasta,
  x.axis.size = 10,
  y.axis.size = 10,
  legend.size = 10,
  width = 12,
  height = 4,
  color_scale = "bright",
  address = ""
)
```

Arguments

data	output of MSstatsLiP converter function. Must include at least ProteinName, PeptideSequence, BioReplicate, and Condition columns
fasta	A string of path to a FASTA file, used to match LiP peptides.
x.axis.size	size of x-axis labeling for plot. Default is 10.
y.axis.size	size of y-axis labeling for plot. Default is 10.
legend.size	size of feature legend for half vs fully tryptic peptides below graph. Default is 7.
width	Width of final pdf to be plotted
height	Height of final pdf to be plotted
color_scale	colors of bar chart. Must be one of "bright" or "grey". Default is "bright".
address	the name of folder that will store the results. Default folder is the current working directory. The other assigned folder has to be existed under the current working directory. An output pdf file is automatically created with the default name of "TyrpticPlot.pdf". If address=FALSE, plot will be not saved as pdf file but shown in window..

Value

plot or pdf

Examples

```
# Use output of summarization function
trypticHistogramLiP(MSstatsLiP_Summarized,
                 system.file("extdata", "ExampleFastaFile.fasta", package="MSstatsLiP"),
                    color_scale = "bright", address = FALSE)
```

Index

∗ **datasets**
LiPRawData, 15
MSstatsLiP_data, 19
MSstatsLiP_model, 20
MSstatsLiP_Summarized, 21
SkylineTest, 24
TrPRawData, 29

annotSite, 2

BarcodePlotLiP, 3, *18*

calculateTrypticity, 5
correlationPlotLiP, 5, *18*

dataProcessPlotsLiP, 6, *18*
dataSummarizationLiP, *7*, 8, *12*, *18*, *23*

groupComparisonLiP, *4*, 11, *14*, *18*
groupComparisonPlotsLiP, 13, *18*

LiPRawData, 15
locateMod, 17
locatePTM, 17

MSstatsLiP, 18
MSstatsLiP_data, 19
MSstatsLiP_model, 20
MSstatsLiP_Summarized, 21

PCAPlotLiP, *18*, 22

SkylineTest, 24
SkylinetoMSstatsLiPFormat, 25
SpectronauttoMSstatsLiPFormat, *18*, 26

tidyFasta, 28
TrPRawData, 29
trypticHistogramLiP, *18*, 30