Package 'MSstats'

April 5, 2014

Title Protein Significance Analysis in LC-MS, SRM and DIA for Label-free or Label-based Proteomics Experiments

Version 2.0.1

Date 2013-09-20

Description
A set of tools for protein significance analysis in label-free or LC-MS, SRM and DIA experiments.

Author Meena Choi <choi67@purdue.edu>, Ching-Yun Chang <chang54@purdue.edu>, Olga Vitek <ovitek@purdue.edu>

Maintainer Meena Choi <choi67@purdue.edu>

License Artistic-2.0

Depends R (>= 3.0), Rcpp, MSnbase

Imports lme4,marray,limma,gplots,ggplot2

bioViews Bioinformatics, MassSpectrometry, Proteomics, Software

LazyData yes

R topics documented:

`MSstats-package` *Tools for protein significance analysis in LC-MS,SRM and DIA for label-free or label-based proteomic experiments*

Description

A set of tools for protein significance analysis in label-free and labeled synthetic peptides LC-MS, SRM and DIA experiments.

Details

Package:	MSstats
Version:	1.99.0
Date:	2013-09-20
License:	Artistic-2.0
LazyLoad:	yes

The package includes four main sections: I. explanatory data analysis (data pre-processing and quality control of MS runs), II. model-based analysis (finding differentially abundant proteins), III. statistical design of future experiments (sample size calculations), and IV. protein quantification (estimation of protein abundance). Section I contains functions for (1) data pre-processing and quality control of MS runs (see `dataProcess`) and (2) visualizing for explanatory data analysis (see `dataProcessPlots`). Section II contains functions for (1) finding differentially abundant proteins (see `groupComparison`) and (2) visualizing for the testing results (see `groupComparisonPlots`) and for the model-based quality control (see `modelBasedQCPlots`). Section III contains functions for (1) calculating sample size (see `designSampleSize`) and (2) visualizing for the sample size calculations (see `designSampleSizePlots`). Section IV contains functions for (1) per-protein group quantification and patient quantification (see `quantification`)

Examples of data or results in MSstats are (1) example of required input data format `RawData`; (2) example of raw data after data pre-processing `QuantData`; (3) results of significance testing of a single comparison `testResultOneComparison`; (4) results of significance testing of multiple comparisons `testResultMultiComparisons`.

Author(s)

Ching-Yun Chang, Meena Choi, Olga Vitek.

Maintainer: Meena Choi (<choi67@purdue.edu>)

References

Ching-Yun Chang, Paola Picotti, Ruth Huttenhain, Viola Heinzelmann-Schwarz, Marko Jovanovic, Ruedi Aebersold, Olga Vitek. "Protein significance analysis in selected reaction monitoring (SRM) measurements." *Molecular & Cellular Proteomics*, 11:M111.014662, 2012.

Timothy Clough, Safia Thaminy, Susanne Ragg, Ruedi Aebersold, Olga Vitek. "Statistical protein quantification and significance analysis in label-free LC-M experiments with complex designs" *BMC Bioinformatics*, 13:S16, 2012.

`dataProcess` *Data pre-processing and quality control of MS runs of raw data*

Description

Data pre-processing and quality control of MS runs of the original raw data into quantitative data for model fitting and group comparison. Log transformation is automatically applied and additional variables are created in columns for model fitting and group comparison process. Three options of data pre-processing and quality control of MS runs in dataProcess are (1) Transformation: logarithm transformation with base 2 or 10; (2) Normalization: to remove systematic bias between MS runs.

Usage

```
dataProcess(raw, logTrans=2, normalization=TRUE,betweenRunInterferenceScore=FALSE,address="")
```

Arguments

`raw`	name of the raw (input) data set.
`logTrans`	logarithm transformation with base 2(default) or 10.
`normalization`	normalization to remove systematic bias between MS runs. TRUE(default) represents constant normalization based on reference signals is performed. FALSE represents no normalization is performed.
`betweenRunInterferenceScore`	interference is detected by a between-run-interference score. TRUE means the scores are generated automatically and stored in a .csv file. FALSE(default) means no scores are generated.
`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 csv file is automatically created with the default name of "BetweenRunInterferenceFile.csv". The command address can help to specify where to store the file as well as how to modify the beginning of the file name.

Details

- raw : See RawData for the required data structure of raw (input) data.
- logTrans : if logTrans=2, the measurement of Variable ABUNDANCE is log-transformed with base 2. Same apply to logTrans=10.
- normalization : if normalization=TRUE and logTrans=2, the measurement of Variable ABUNDANCE is log-transformed with base 2 and normalized. Same as for logTrans=10.

Warning

When a transition is missing completely in a condition or a MS run, a warning message is sent to the console notifying the user of the missing transitions.

The types of experiment that MSstats can analyze are LC-MS, SRM, DIA(SWATH) with label-free or labeled synthetic peptides. MSstats does not support for metabolic labeling or iTRAQ experiments.

Author(s)

Ching-Yun Chang, Meena Choi, Olga Vitek.

Maintainer: Meena Choi (<choi67@purdue.edu>)

References

Ching-Yun Chang, Paola Picotti, Ruth Huttenhain, Viola Heinzelmann-Schwarz, Marko Jovanovic, Ruedi Aebersold, Olga Vitek. "Protein significance analysis in selected reaction monitoring (SRM) measurements" *Molecular & Cellular Proteomics*, 11:M111.014662, 2012.

Timothy Clough, Safia Thaminy, Susanne Ragg, Ruedi Aebersold, Olga Vitek. "Statistical protein quantification and significance analysis in label-free LC-M experiments with complex designs" *BMC Bioinformatics*, 13:S16, 2012.

Examples

```
# Consider a raw data (i.e. RawData) for a label-based SRM experiment from a yeast study with ten time points (T1-T10
# It is a time course experiment. The goal is to detect protein abundance changes across time points.

head(RawData)

# Log2 transformation and normalization are applied (default)
QuantData<-dataProcess(RawData)
head(QuantData)

# Log10 transformation and normalization are applied
QuantData1<-dataProcess(RawData, logTrans=10)
head(QuantData1)

# Log2 transformation and no normalization are applied
QuantData2<-dataProcess(RawData,normalization=FALSE)
head(QuantData2)
```

dataProcessPlots *Visualization for explanatory data analysis*

Description

To illustrate the quantitative data after data-preprocessing and quality control of MS runs, dataProcessPlots takes the quantitative data from function (dataProcess) as input and automatically generate three 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; (3) mean plot for conditions (specify "ConditionPlot" in option type), to illustrate mean and variability of each condition per protein.

Usage

```
dataProcessPlots(data=data,type=type,featureName="Transition",ylimUp=FALSE,ylimDown=FALSE,scale=FAL
```

Arguments

data	name of the (processed) data set.
type	choice of visualization. "ProfilePlot" represents profile plot of log intensities across MS runs. "QCPlot" represents quality control plot of log intensities across MS runs. "ConditionPlot" represents mean plot of log ratios (Light/Heavy) across conditions.
featureName	for "ProfilePlot" only, "Transition" (default) means printing feature legend in transition-level; "Peptide" means printing feature legend in peptide-level; "NA" means no feature legend printing.
ylimUp	upper limit for y-axis in the log scale. FALSE(Default) for Profile Plot and QC Plot is 30. FALSE(Default) for Condition Plot is maximum of log ratio + SD or SE.
ylimDown	lower limit for y-axis in the log scale. FALSE(Default) for Profile Plot and QC Plot is 0. FALSE(Default) for Condition Plot is minumum of log ratio - SD or SE.
scale	for "ConditionPlot" only, FALSE(default) means each conditional level is not scaled at x-axis according to its actual value (equal space at x-axis). TRUE means each conditional level is scaled at x-axis according to its actual value (unequal space at x-axis).
interval	for "ConditionPlot" only, "SE"(default) uses standard error for the width of error bar. "SD"(default) uses standard deviation for the width of error bar.
axis.size	size of x-axis labeling for "Run" in Profile Plot and QC Plot, and "Condition" in Condition Plot. Default is 10.
text.size	size of labels represented each condition at the top of graph in Profile Plot and QC plot. Default is 4.
text.angle	angle of labels represented each condition at the top of graph in Profile Plot and QC plot or x-axis labeling in Condition plot. Default is 0.

legend.size	size of feature legend (transition-level or peptide-level) above graph in Profile Plot. Default is 7.
width	width of the saved file. Default is 10.
height	height of the saved file. Default is 10.
which.Protein	Protein list to draw plots. List can be names of Proteins or order numbers of Proteins from levels(data$PROTEIN). Default is "all", which generates all 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 "ProfilePlot.pdf" or "QCplot.pdf" or "ComparisonPlot.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.

Details

- Profile Plot : identify the potential sources of variation of each protein. X-axis is run. Y-axis is log-intensities of transitions. Reference/endogenous signals are in the left/right panel. Line colors indicate peptides and line types indicate transitions.
- QC Plot : illustrate the systematic bias between MS runs. After normalization, the reference signals for all proteins should be stable across MS runs. X-axis is run. Y-axis is log-intensities of transition. Reference/endogenous signals are in the left/right panel. The pdf file contains (1) QC plot for all proteins and (2) QC plots for each protein separately.
- Condition Plot : illustrate the systematic difference between conditions. X-axis is condition. Y-axis is log ratio of endogenous over reference. For label-free, Y-axis is log intensity of endogenous. If scale is TRUE, the levels of conditions is scaled according to its actual values at x-axis. Red points indicate the mean of log ratio for each condition. If interval is "SE", blue error bars indicate the confidence interval with 0.95 significant level for each condition. If interval is "SD", blue error bars indicate the standard deviation for each condition.The interval is not related with model-based analysis

The input of this function is the quantitative data from function (dataProcess). The example data is QuantData.

Author(s)

Ching-Yun Chang, Meena Choi, Olga Vitek.

Maintainer: Meena Choi (<choi67@purdue.edu>)

References

Ching-Yun Chang, Paola Picotti, Ruth Huttenhain, Viola Heinzelmann-Schwarz, Marko Jovanovic, Ruedi Aebersold, Olga Vitek. "Protein significance analysis in selected reaction monitoring (SRM) measurements." *Molecular & Cellular Proteomics*, 11:M111.014662, 2012.

Timothy Clough, Safia Thaminy, Susanne Ragg, Ruedi Aebersold, Olga Vitek. "Statistical protein quantification and significance analysis in label-free LC-M experiments with complex designs" *BMC Bioinformatics*, 13:S16, 2012.

Examples

```
#Consider quantitative data (i.e. QuantData) from a yeast study with ten time points of interests, three biological
#The goal is to provide pre-analysis visualization by automatically generate two types of figures in two separate pd
#Protein IDHC (gene name IDP2) is differentially expressed in time point 1 and time point 7, whereas, Protein PMG2 (g

head(QuantData)

# Profile plot

dataProcessPlots(data=QuantData,type="ProfilePlot")

# Quality control plot

dataProcessPlots(data=QuantData,type="QCPlot")

# Quantification plot for conditions

dataProcessPlots(data=QuantData,type="ConditionPlot", address="Ex1_")

# Quantification plot for conditions, interval is based on standard deviation, and fixed limit of y axis.

dataProcessPlots(data=QuantData,type="ConditionPlot", ylimUp=3,ylimDown=-8,interval="SD",address="Ex2_")
```

---

designSampleSize *Planning future experimental designs of label-free and label-based LC-MS, SRM, DIA experiments in sample size calculation*

Description

Calculate sample size for future experiments of a label-free and label-based LC-MS, SRM and DIA experiment based on intensity-based linear model. Four options of the calculation: (1) number of biological replicates per condition, (2) number of peptides per protein, and (3) number of transitions per peptide, (4) power.

Usage

```
designSampleSize(data=data,numSample=numSample,numPep=numPep,numTran=numTran,desiredFC=desiredFC,FD
```

Arguments

data	name of the data set.
numSample	minimal number of biological replicates per condition. TRUE represents you require to calculate the sample size for this category, else you should input the exact number of biological replicates.
numPep	minimal number of peptides per protein. TRUE represents you require to calculate the sample size for this category, else you should input the exact number of peptides.

numTran	minimal number of transitions per peptide. TRUE represents you require to calculate the sample size for this category, else you should input the exact number of transitions.
desiredFC	the range of a desired fold change which includes the lower and upper values of the desired fold change.
FDR	a pre-specified false discovery ratio (FDR) to control the overall false positive.
power	a pre-specified statistical power which defined as the probability of detecting a true fold change. TRUE represent you require to calculate the power for this category, else you should input the average of power you expect.
scopeOfBioReplication	choice of scope of biological replication. "restricted"(default) represents restricted scope of biological replication to the selected individuals. "expanded" represents expanded scope of biological replication to the whole population.
interference	choice of interference data. TRUE(default) means data contain interference transitions and need additional model interaction to address the interference. FALSE means data contain no interference transitions and no need additional model interaction to address the interference.

Details

The function fits the model and uses variance components to calculate sample size. The underlying model fitting with intensity-based linear model with technical MS run replication. Data in the example is QuantData. Estimated sample size is rounded to 0 decimal.

Value

A list of the sample size calculation results including Variable desiredFC, numSample, numPep, numTran, FDR, and power.

Warning

It can only obtain either one of the categories of the sample size calculation (numSample, numPep, numTran, power) at the same time.

Author(s)

Ching-Yun Chang, Meena Choi, Olga Vitek.

Maintainer: Meena Choi (<choi67@purdue.edu>)

References

Ching-Yun Chang, Paola Picotti, Ruth Huttenhain, Viola Heinzelmann-Schwarz, Marko Jovanovic, Ruedi Aebersold, Olga Vitek. "Protein significance analysis in selected reaction monitoring (SRM) measurements." *Molecular & Cellular Proteomics*, 11:M111.014662, 2012.

Timothy Clough, Safia Thaminy, Susanne Ragg, Ruedi Aebersold, Olga Vitek. "Statistical protein quantification and significance analysis in label-free LC-M experiments with complex designs" *BMC Bioinformatics*, 13:S16, 2012.

Examples

```
# Consider quantitative data (i.e. QuantData) from yeast study.
# A time course study with ten time points of interests and three biological replicates.

head(QuantData)

## Calculate sample size for future experiments:

#(1) Minimal number of biological replicates per condition

designSampleSize(data=QuantData,numSample=TRUE,numPep=3,numTran=4,
desiredFC=c(1.25,1.75),FDR=0.05,power=0.8)

#(2) Minimal number of peptides per protein

designSampleSize(data=QuantData,numSample=2,numPep=TRUE,numTran=4,
desiredFC=c(1.25,1.75),FDR=0.05,power=0.8)

#(3) Minimal number of transitions per peptide

designSampleSize(data=QuantData,numSample=2,numPep=3,numTran=TRUE,
desiredFC=c(1.25,1.75),FDR=0.05,power=0.8)

#(4) Power calculation

designSampleSize(data=QuantData,numSample=2,numPep=3,numTran=4,
desiredFC=c(1.25,1.75),FDR=0.05,power=TRUE)
```

---

designSampleSizePlots *Visualization for sample size calculation*

Description

To illustrate the relationship of desired fold change and the calculated minimal number sample size which are (1) number of biological replicates per condition, (2) number of peptides per protein, (3) number of transitions per peptide, and (4) power. The input is the result from function (designSampleSize.

Usage

```
designSampleSizePlots(data=data)
```

Arguments

data	output from function designSampleSize.

Details

Data in the example is based on the results of sample size calculation from function designSampleSize.

Author(s)

Ching-Yun Chang, Meena Choi, Olga Vitek.

Maintainer: Meena Choi (<choi67@purdue.edu>)

References

Ching-Yun Chang, Paola Picotti, Ruth Huttenhain, Viola Heinzelmann-Schwarz, Marko Jovanovic, Ruedi Aebersold, Olga Vitek. "Protein significance analysis in selected reaction monitoring (SRM) measurements." *Molecular & Cellular Proteomics*, 11:M111.014662, 2012.

Timothy Clough, Safia Thaminy, Susanne Ragg, Ruedi Aebersold, Olga Vitek. "Statistical protein quantification and significance analysis in label-free LC-M experiments with complex designs" *BMC Bioinformatics*, 13:S16, 2012.

Examples

```
# Based on the results of sample size calculation from function designSampleSize, we generate a series of sample siz

head(QuantData)

## plot the calculated sample sizes for future experiments:

#(1) Minimal number of biological replicates per condition

result.sample<-designSampleSize(data=QuantData,numSample=TRUE,numPep=3,numTran=4,desiredFC=c(1.25,1.75),FDR=0.
designSampleSizePlots(data=result.sample)

#(2) Minimal number of peptides per protein

result.peptide<-designSampleSize(data=QuantData,numSample=2,numPep=TRUE,numTran=4,desiredFC=c(1.25,1.75),FDR=0
designSampleSizePlots(data=result.peptide)

#(3) Minimal number of transitions per peptide

result.tran<-designSampleSize(data=QuantData,numSample=2,numPep=3,numTran=TRUE,desiredFC=c(1.25,1.75),FDR=0.05
designSampleSizePlots(data=result.tran)

#(4) Power

result.power<-designSampleSize(data=QuantData,numSample=2,numPep=3,numTran=4,desiredFC=c(1.25,1.75),FDR=0.05,p
designSampleSizePlots(data=result.power)
```

`groupComparison` *Finding differentially abundant proteins across conditions in LC-MS, SRM and DIA experiment*

Description

Tests for significant changes in protein abundance across conditions based on a family of linear mixed-effects models in LC-MS, SRM, DIA experiment. Experimental design of case-control study (patients are not repeatedly measured) or time course study (patients are repeatedly measured) is automatically determined based on proper statistical model. Other choices of model specification include (1) labeling technique: label-based or label-free; (2) scope of inference: restricted scope (Variable RUN is fixed effect) or expanded scope (Variable RUN is random effect) of technical MS run replication; (3) interference: including or excluding additional model interaction to account interference; (4) unequal variance between features: whether the model consider hetergenous variation among intensities between features.

Usage

```
groupComparison(contrast.matrix=contrast.matrix, data=data, labeled=TRUE, scopeOfBioReplication="res
```

Arguments

`contrast.matrix`	comparison between conditions of interests.
`data`	name of the (processed) data set.
`labeled`	choice of labeling technique. TRUE(default) represents the label-based study. FALSE represents label-free study.
`scopeOfBioReplication`	choice of scope of biological replication. "restricted" represents restricted scope of biological replication by specifying subject term as fixed effect in the model. "expanded" (default) represents expanded scope of biological replication by specifying subject term as random effect in the model.
`scopeOfTechReplication`	choice of scope of technical MS run replication. "restricted" represents restricted scope of technical MS run replication by specifying run term as fixed effect in the model. "expanded" (default) represents expanded scope of technical MS run replication by specifying run term as random effect in the model.
`interference`	choice of interference data. TRUE(default) means data contain interference transitions and need additional model interaction to address the interference. FALSE means data contain no interference transitions and no need additional model interaction to address the interference.
`featureVar`	logical variable for whether the model should account for heterogeneous variation among intensities from different features. Default is FALSE, which assume equal variance among intensities from features.
`missing.action`	specifies the action to take in presence of extreme missing values; must be one of 'nointeraction','impute', or 'remove'. Default is 'nointeraction'.

Details

- contrast.matrix : comparison of interest. Based on the levels of conditions, specify 1 or -1 to the conditions of interests and 0 otherwise. The levels of conditions are sorted alphabetically. Command levels(QuantData$GROUP_ORIGINAL) can illustrate the actual order of the levels of conditions.

- labeled : choices of labeling technique. In label-based study (labeled=TRUE), scopeOfTechReplication, scopeOfBioReplication and interference work as described above. In label-free study (labeled=FALSE), no need to specify scopeOfTechReplication because biological replicates and technical MS runs are confounding. interference works as described above.
- interference : the model can be specified with interaction model terms that reflect interferences in the quantified transitions.
- featureVar : If the unequal variation of error for different peptide features is detected, then a possible solution is to account for the unequal error variation by means of a procedure called iteratively re-weighted least squares. featureVar=TRUE performs an iterative fitting procedure, in which features are weighted inversely proportionaly to the variation in their intensities, so that feature with large variation are given less importance in the estimation of parameters in the model.
- missing.action : When peak intensities from all replicates in a condition are missing for at least one feature, there are three possible actions; (1) remove interaction (missing.action="nointeraction"), which means to assume feature demonstrate no interference across runs, (2) impute with the average minimum intensity across run (missing.action="impute"), or (3) remove the features from the dataset (missing.action="remove")

The underlying model fitting functions are `lm` and `lmer` for the fixed effects model and mixed effects model, respectively.

The input of this function is the quantitative data from function (`dataProcess`). The example data is `QuantData`.

Warning

When a feature is missing completely in a condition or a MS run, a warning message is sent to the console notifying the user of the missing feature. Additional filtering or imputing process is required before model fitting.

Author(s)

Ching-Yun Chang, Meena Choi, Olga Vitek.

Maintainer: Meena Choi (<choi67@purdue.edu>)

References

Ching-Yun Chang, Paola Picotti, Ruth Huttenhain, Viola Heinzelmann-Schwarz, Marko Jovanovic, Ruedi Aebersold, Olga Vitek. "Protein significance analysis in selected reaction monitoring (SRM) measurements." *Molecular & Cellular Proteomics*, 11:M111.014662, 2012.

Timothy Clough, Safia Thaminy, Susanne Ragg, Ruedi Aebersold, Olga Vitek. "Statistical protein quantification and significance analysis in label-free LC-M experiments with complex designs" *BMC Bioinformatics*, 13:S16, 2012.

Examples

```
#Consider quantitative data (i.e. QuantData) from yeast study with ten time points of interests, three biological re
#It is a time-course experiment and we attempt to compare differential abundance between time 1 and 7 in a set of tar
#In this label-based SRM experiment, we recommend the fitted model with expanded scope of technical replication and
```

```
head(QuantData)

levels(QuantData$GROUP_ORIGINAL)
comparison<-matrix(c(-1,0,0,0,0,0,1,0,0,0),nrow=1)
row.names(comparison)<-"T7-T1"

# Tests for differentially abundant proteins with models:

#(1) label-based SRM experiment with restricted scope of biological replication and expanded scope of technical MS r
testResultOneComparison<-groupComparison(contrast.matrix=comparison, data=QuantData)
testResultOneComparison$ComparisonResult

testresult2<-groupComparison(contrast.matrix=comparison, data=QuantData, interference=FALSE)
testresult2$ComparisonResult

#(2) label-based SRM experiment with restricted scope of technical MS run replication and restricted scope of biolog
testresult3<-groupComparison(contrast.matrix=comparison, data=QuantData, scopeOfTechReplication="restricted")
testresult3$ComparisonResult

#(3) label-based SRM experiment with expanded scope of technical MS run replication and expanded scope of biological
testresult4<-groupComparison(contrast.matrix=comparison, data=QuantData, scopeOfBioReplication="expanded")
testresult4$ComparisonResult

#(4) label-free SRM experiment with expanded scope of biological replication and interference
testresult5<-groupComparison(contrast.matrix=comparison, data=QuantData, labeled=FALSE)
testresult5$ComparisonResult
```

`groupComparisonPlots` *Visualization for model-based analysis and summarizing differentially abundant proteins*

Description

To summarize the results of log-fold changes and adjusted p-values for differentially abundant proteins, groupComparisonPlots takes testing results from function (`groupComparison`) as input and automatically generate three types of figures in pdf files as output : (1) volcano plot (specify "VolcanoPlot" in option type) for each comparison separately; (2) heatmap (specify "Heatmap" in option type) for multiple comparisons ; (3) comparison plot (specify "ComparisonPlot" in option type) for multiple comparisons per protein.

Usage

```
groupComparisonPlots(data=data,type=type,sig=0.05,FCcutoff=FALSE,ylimUp=FALSE,ylimDown=FALSE,xlimUp
```

Arguments

`data`	'ComparisonResult' in testing output from function groupComparison.

type	choice of visualization. "VolcanoPlot" represents volcano plot of log fold changes and adjusted p-values for each comparison separately. "Heatmap" represents heatmap of adjusted p-values for multiple comparisons. "ComparisonPlot" represents comparison plot of log fold changes for multiple comparisons per protein.
sig	FDR cutoff for the adjusted p-values. sig=0.05 is default.
FCcutoff	for 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.
ylimUp	for all three plots, upper limit for y-axis. FALSE (default) for volcano plot/heatmap use maximum of -log2 (adjusted p-value). FALSE (default) for comparison plot uses maximum of log-fold change + SE.
ylimDown	for all three plots, lower limit for y-axis. FALSE (default) for volcano plot/heatmap use minimum of -log2 (adjusted p-value). FALSE (default) for comparison plot uses minimum of log-fold change - SE.
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.
axis.size	size of axes labels, e.g. name of the comparisons and of targeted proteins in heatmap, and name of the comparisons in comparison plot. Default is 10.
text.size	size of ProteinName label in the graph for Volcano Plot. Default is 4.
ProteinName	for volcano plot only, whether display protein names or not. TRUE (default) means protein names are displayed next to the testing results. FALSE means no protein names are displayed.
ProteinNameLoc	for volcano plot only, assign the distance between the point and the displayed protein name. Default is 1.
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$Label), such as levels(testResultMultiComparisons$Compariso Default is "all", which generates all 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" or "ComparisonPlot.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.

Details

- Volcano plot : illustrate actual log-fold changes and adjusted p-values for each comparison separately with all proteins. The x-axis is the log fold change. The base of logarithm transformation is the same as specified in "logTrans" from dataProcess. The y-axis is the negative

log2 adjusted p-values. The horizontal dashed line represents the FDR cutoff. The points below the FDR cutoff line are non-significantly abundant proteins (colored in black). The points above the FDR cutoff line are significantly abundant proteins (colored in red/blue for up-/down-regulated). If fold change cutoff is specified (FCcutoff = specific value), the points above the FDR cutoff line but within the FC cutoff line are non-significantly abundant proteins (colored in black)/

- Heatmap : illustrate up-/down-regulated proteins for multiple comparisons with all proteins. Each column represents each comparison of interest. Each row represents each protein. Color red/blue represents proteins in that specific comparison are significantly up-regulated/down-regulated proteins with FDR cutoff and/or FC cutoff. The color scheme shows the evidences of significance. The darker color it is, the stronger evidence of significance it has. Color gold represents proteins are not significantly different in abundance.
- Comparison plot : illustrate log-fold change and its variation of multiple comparisons for single protein. X-axis is comparison of interest. Y-axis is the log fold change. The red points are the estimated log fold change from the model. The blue error bars are the confidence interval with 0.95 significant level for log fold change. This interval is only based on the standard error, which is estimated from the model.

The input of this function is "ComparisonResult" in the testing results from function (groupComparison). The example result is testResultMultiComparisons.

Author(s)

Ching-Yun Chang, Meena Choi, Olga Vitek.

Maintainer: Meena Choi (<choi67@purdue.edu>)

References

Ching-Yun Chang, Paola Picotti, Ruth Huttenhain, Viola Heinzelmann-Schwarz, Marko Jovanovic, Ruedi Aebersold, Olga Vitek. "Protein significance analysis in selected reaction monitoring (SRM) measurements." *Molecular & Cellular Proteomics*, 11:M111.014662, 2012.

Timothy Clough, Safia Thaminy, Susanne Ragg, Ruedi Aebersold, Olga Vitek. "Statistical protein quantification and significance analysis in label-free LC-M experiments with complex designs" *BMC Bioinformatics*, 13:S16, 2012.

Examples

```
## based on multiple comparisons  (T1 vs T3; T1 vs T7; T1 vs T9)
comparison1<-matrix(c(-1,0,1,0,0,0,0,0,0,0),nrow=1)
comparison2<-matrix(c(-1,0,0,0,0,0,1,0,0,0),nrow=1)
comparison3<-matrix(c(-1,0,0,0,0,0,0,0,1,0),nrow=1)
comparison<-rbind(comparison1,comparison2, comparison3)
row.names(comparison)<-c("T3-T1","T7-T1","T9-T1")

testResultMultiComparisons<-groupComparison(contrast.matrix=comparison,data=QuantData)

testResultMultiComparisons$ComparisonResult

# Volcano plot with FDR cutoff = 0.05 and no FC cutoff
```

```
groupComparisonPlots(data=testResultMultiComparisons$ComparisonResult,type="VolcanoPlot",address="Ex1_")

# Volcano plot with FDR cutoff = 0.05, FC cutoff = 70, upper y-axis limit = 100, and no protein name displayed
# FCcutoff=70 is for demonstration purpose
groupComparisonPlots(data=testResultMultiComparisons$ComparisonResult,type="VolcanoPlot",FCcutoff=70, ylimUp=10

# Heatmap with FDR cutoff = 0.05
groupComparisonPlots(data=testResultMultiComparisons$ComparisonResult,type="Heatmap",axis.size=1.2,address="Ex

# Heatmap with FDR cutoff = 0.05 and FC cutoff = 70
# FCcutoff=70 is for demonstration purpose
groupComparisonPlots(data=testResultMultiComparisons$ComparisonResult,type="Heatmap",FCcutoff=70,axis.size=1.2

# Comparison Plot
groupComparisonPlots(data=testResultMultiComparisons$ComparisonResult,type="ComparisonPlot",address="Ex1_")

# Comparison Plot
groupComparisonPlots(data=testResultMultiComparisons$ComparisonResult,type="ComparisonPlot",ylimUp=8,ylimDown=
```

`modelBasedQCPlots` *Visualization for model-based quality control in fitting model*

Description

To check the assumption of model, modelBasedQCPlots takes the results after fitting models from function (`groupComparison`) as input and automatically generate two types of figures in pdf files as output : (1) normal quantile-quantile plot (specify "QQPlot" in option type) for checking normally distributed errors.; (2) residual plot (specify "ResidualPlot" in option type) for checking constant variance among different features.

Usage

```
modelBasedQCPlots(data,type,axis.size=10,point.size=3,text.size=7,width=10, height=10,featureName=TR
```

Arguments

`data`	'ModelQC' in testing output from function groupComparison.
`type`	choice of visualization. "QQPlots" represents normal quantile-quantile plot for each protein after fitting models. "ResidualPlots" represents a plot of residuals versus fitted values for each protein in the dataset.
`axis.size`	size of axes labels, e.g. name of the comparisons and of targeted proteins in heatmap, and name of the comparisons in comparison plot. Default is 10.
`point.size`	size of points in the graph for residual plots. Default is 3.
`text.size`	size of labeling for feature names only in normal quantile-quantile plots separately for each feature. Default is 7.

featureName	for "ResidualPlot" only, TRUE show feature labeling and FALSE means no feature legend printing.
feature.QQPlot	"all"(Default) means that one normal quantile-quantile plot will be generated with across features of a protein. "byFeature" will generate normal quantile-quantile plots separately for each feature of a protein.
width	width of the saved file. Default is 10.
height	height of the saved file. Default is 10.
which.Protein	Protein list to draw plots. List can be names of Proteins or order numbers of Proteins from levels(data$PROTEIN). Default is "all", which generates all 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. If type="residualPlots" or "QQPlots", "ResidualPlots.pdf" or "QQPlots.plf" will be generated. 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.

Details

Results based on statistical models are accurate as long as the assumptions of the model are met. The model assumes that the measurement errors are normally distributed with mean 0 and constant variance. The assumption of a constant variance can be checked by examining the residuals from the model.

- QQPlots : a normal quantile-quantile plot for each protein is generated in order to check whether the errors are well approximated by a normal distribution. If points fall approximately along a straight line, then the assumption is appropriate for that protein. Only large deviations from the line are problematic.
- ResidualPlots : The plots of residuals against predicted(fitted) values. If it shows a random scatter, then the assumption is appropriate.

The input of this function is "ModelQC" in the testing results from function (groupComparison). The example result is testResultMultiComparisons.

Author(s)

Ching-Yun Chang, Meena Choi, Olga Vitek.

Maintainer: Meena Choi (<choi67@purdue.edu>)

References

Ching-Yun Chang, Paola Picotti, Ruth Huttenhain, Viola Heinzelmann-Schwarz, Marko Jovanovic, Ruedi Aebersold, Olga Vitek. "Protein significance analysis in selected reaction monitoring (SRM) measurements." *Molecular & Cellular Proteomics*, 11:M111.014662, 2012.

Timothy Clough, Safia Thaminy, Susanne Ragg, Ruedi Aebersold, Olga Vitek. "Statistical protein quantification and significance analysis in label-free LC-M experiments with complex designs" *BMC Bioinformatics*, 13:S16, 2012.

Examples

```
## based on multiple comparisons  (T1 vs T3; T1 vs T7; T1 vs T9)
comparison1<-matrix(c(-1,0,1,0,0,0,0,0,0,0),nrow=1)
comparison2<-matrix(c(-1,0,0,0,0,0,1,0,0,0),nrow=1)
comparison3<-matrix(c(-1,0,0,0,0,0,0,0,1,0),nrow=1)
comparison<-rbind(comparison1,comparison2, comparison3)
row.names(comparison)<-c("T3-T1","T7-T1","T9-T1")

testResultMultiComparisons<-groupComparison(contrast.matrix=comparison,data=QuantData)

head(testResultMultiComparisons$ModelQC)

# normal quantile-quantile plots
modelBasedQCPlots(data=testResultMultiComparisons$ModelQC,type="QQPlots",address="")

# residual plots
modelBasedQCPlots(data=testResultMultiComparisons$ModelQC,type="ResidualPlots",address="")
```

`QuantData` *Quantitative data after data pre-processing and quality control of MS runs*

Description

This is an example of quantitative data after data pre-processing and quality control of MS runs (i.e., transformation, normalization, and filtering of the original intensities measurements) and can be obtained directly from `dataProcess`.

Usage

```
QuantData
```

Format

data.frame

Details

The quantitative data after data pre-processing and quality control of MS runs not only contain the same variable from the raw data, but with addition variables for statistical model fitting and group comparison. For examples, Variable ABUNDANCE represents the final measurement, which could be normalized or not depending on the options you specified in `dataProcess`. Default option in `dataProcess` is with log2 transformation and normalization.

Author(s)

Ching-Yun Chang, Meena Choi, Olga Vitek.

Maintainer: Meena Choi (<choi67@purdue.edu>)

References

Ching-Yun Chang, Paola Picotti, Ruth Huttenhain, Viola Heinzelmann-Schwarz, Marko Jovanovic, Ruedi Aebersold, Olga Vitek. Protein significance analysis in selected reaction monitoring (SRM) measurements. *Molecular & Cellular Proteomics*, 11:M111.014662, 2012.

Examples

```
QuantData<-dataProcess(RawData)

head(QuantData)
```

`quantification` *Protein sample quantification or group quantification*

Description

Model-based quantification for each condition or for each biological samples per protein in a label-free and label-based LC-MS, SRM and DIA experiment. Quantification takes the processed data set by `dataProcess` as input and automatically generate the quantification results (data.frame) with long or matrix format.

Usage

```
quantification(data, type="Sample", format="matrix")
```

Arguments

`data`	name of the (processed) data set.
`type`	choice of quantification. "Sample" or "Group" for protein sample quantification or group quantification.
`format`	choice of returned format. "long" for long format which has the columns named Protein, Condition, LonIntensities (and BioReplicate if it is subject quantification), NumFeature for number of transitions for a protein, and NumPeaks for number of observed peak intensities for a protein. "matrix" for data matrix format which has the rows for Protein and the columns, which are Groups(or Conditions) for group quantification or the combinations of BioReplicate and Condition (labeled by "BioReplicate"_"Condition") for sample quantification. Default is "matrix"

Details

- Sample quantification : model-based individual biological sample quantification for each protein. The label of each biological sample is a combination of the corresponding group and the sample ID. The same model with groupComparison will be used. However, if there is only one transition in a certain protein, the estimate of variation is NA. Therefore, the result may be unreliable.
- Group quantification : model-based quantification for individual group or individual condition per protein. The same model with groupComparison will be used. The quantification for reference is the average among all reference intensities.
- The quantification for endogenous samples is based on the log-intensities of model-based averaging of all endogenous transitions within a specific sample. The quantification for reference sample is based on the log-intensities of the model-based averaging among all reference transitions. The quantification of log-ratios of specific endogenous sample over reference sample can be obtained by the quantification of that endogenous sample minus the quantification of the reference sample.

  The input of this function is the quantitative data from function (dataProcess). The example data is QuantData.

Author(s)

Ching-Yun Chang, Meena Choi, Olga Vitek.

Maintainer: Meena Choi (<choi67@purdue.edu>)

References

Ching-Yun Chang, Paola Picotti, Ruth Huttenhain, Viola Heinzelmann-Schwarz, Marko Jovanovic, Ruedi Aebersold, Olga Vitek. "Protein significance analysis in selected reaction monitoring (SRM) measurements." *Molecular & Cellular Proteomics*, 11:M111.014662, 2012.

Timothy Clough, Safia Thaminy, Susanne Ragg, Ruedi Aebersold, Olga Vitek. "Statistical protein quantification and significance analysis in label-free LC-M experiments with complex designs" *BMC Bioinformatics*, 13:S16, 2012.

Examples

```
#Consider quantitative data (i.e. QuantData) from a yeast study with ten time points of interests, three biological
#Sample quantification shows model-based estimation of protein abundance in each biological replicate within each t
#Group quantification shows model-based estimation of protein abundance in each time point.

head(QuantData)

# Sample quantification

quantification(QuantData)

# Group quantification

quantification(QuantData, type="Group")
```

RawData *Example dataset from a label-based SRM experiment of a time course yeast study*

Description

This is a partial data set obtained from a published study (Picotti, et. al, 2009). The experiment targeted 45 proteins in the glycolysis/gluconeogenesis/TCA cycle/glyoxylate cycle network, which spans the range of protein abundance from less than 128 to 10E6 copies per cell. Three biological replicates were analyzed at ten time points (T1-T10), while yeasts transited through exponential growth in a glucose-rich medium (T1-T4), diauxic shift (T5-T6), post-diauxic phase (T7-T9), and stationary phase (T10). Prior to trypsinization, the samples were mixed with an equal amount of proteins from the same N15-labeled yeast sample, which was used as a reference. Each sample was profiled in a single mass spectrometry run, where each protein was represented by up to two peptides and each peptide by up to three transitions. The goal of this study is to detect significantly change in protein abundance across time points. Transcriptional activity under the same experimental conditions has been previously investigated by (DeRisi et. al., 1997). Genes coding for 29 of the proteins are differentially expressed between conditions similar to those represented by T7 and T1 and could be treated as external sources to validate the proteomics analysis. In this exampled data set, two of the targeted proteins are selected and validated with gene expression study: Protein IDHC (gene name IDP2) is differentially expressed in time point 1 and time point 7, whereas, Protein PMG2 (gene name GPM2) is not. The protein names are based on Swiss Prot Name.

Usage

```
RawData
```

Format

data.frame

Details

The raw data (input data for SRMstats) is required to contain variable of ProteinName, PeptideSequence, PrecursorCharge, FragmentIon, ProductCharge, IsotopeLabelType, Condition, BioReplicate, Run, Intensity. The variable names should be fixed.

If the information of one or more columns is not available for the original raw data, please retain the column variables and type in fixed value. For example, the original raw data does not contain the information of ProductCharge, we retain the column ProductCharge and type in NA for all transitions in RawData.

The column BioReplicate should label with unique patient ID (i.e., same patients should label with the same ID).

Variable Intensity is required to be original signal without any log transformation and can be specified as the peak of height or the peak of area under curve.

Author(s)

Ching-Yun Chang, Meena Choi, Olga Vitek.

Maintainer: Meena Choi (<choi67@purdue.edu>)

References

Ching-Yun Chang, Paola Picotti, Ruth Huttenhain, Viola Heinzelmann-Schwarz, Marko Jovanovic, Ruedi Aebersold, Olga Vitek. Protein significance analysis in selected reaction monitoring (SRM) measurements. *Molecular & Cellular Proteomics*, 11:M111.014662, 2012.

Examples

```
head(RawData)
```

---

testResultMultiComparisons

Significance testing result of a set of comparisons between two groups

---

Description

Consider quantitative data (i.e. QuantData) from yeast study with ten time points of interests, three biological replicates, and no technical replicates. It is a time-course experiment. In this label-based SRM experiment, we recommend the fitted model with expanded scope of technical replication (i.e. labeled=TRUE, scopeOfTechReplication="expanded" as default). This is a testing result of multiple comparisons of QuantData based on the intensity-based linear model with expanded scope. The comparison is time 1 vs time 3 (T3-T1), time 1 vs time 7 (T7-T1), time 1 vs time 9 (T9-T1).

Usage

```
testResultMultiComparisons
```

Details

The testing result contains variable of Protein, Comparison(Label), log2 fold change(logFC), standard error (SE), T values (Tvalue), degree of freedom (DF), raw p-values (pvalue), adjusted p-values based on Benjamini and Hochberg method to collect multiple testing issue and further control false discovery rate (adj.pvalue). There are multiple lines for the same proteins.

Author(s)

Ching-Yun Chang, Meena Choi, Olga Vitek.

Maintainer: Meena Choi (<choi67@purdue.edu>)

References

Ching-Yun Chang, Paola Picotti, Ruth Huttenhain, Viola Heinzelmann-Schwarz, Marko Jovanovic, Ruedi Aebersold, Olga Vitek. Protein significance analysis in selected reaction monitoring (SRM) measurements. *Molecular & Cellular Proteomics*, 11:M111.014662, 2012.

Examples

```
comparison1<-matrix(c(-1,0,1,0,0,0,0,0,0,0),nrow=1)
comparison2<-matrix(c(-1,0,0,0,0,0,1,0,0,0),nrow=1)
comparison3<-matrix(c(-1,0,0,0,0,0,0,0,1,0),nrow=1)
comparison<-rbind(comparison1,comparison2, comparison3)
row.names(comparison)<-c("T3-T1","T7-T1","T9-T1")

testResultMultiComparisons<-groupComparison(contrast.matrix=comparison,data=QuantData)

testResultMultiComparisons$ComparisonResult
```

`testResultOneComparison`

Significance testing result of one comparison between two groups

---

Description

Consider quantitative data (i.e. `QuantData`) from yeast study with ten time points of interests, three biological replicates, and no technical replicates. It is a time-course experiment, and we attempt to compare differential abundance between time 1 and 7 in a set of targeted proteins. In this label-based SRM experiment, we recommend the fitted model with expanded scope of technical replication (i.e. labeled=TRUE, scopeOfTechReplication="expanded" as default).

Usage

```
testResultOneComparison
```

Details

The testing result contains variable of Protein, Comparison(Label), log2 fold change(logFC), standard error (SE), T values (Tvalue), degree of freedom (DF), raw p-values (pvalue), adjusted p-values based on Benjamini and Hochberg method to collect multiple testing issue and further control false discovery rate (adj.pvalue).

Author(s)

Ching-Yun Chang, Meena Choi, Olga Vitek.

Maintainer: Meena Choi (<choi67@purdue.edu>)

References

Ching-Yun Chang, Paola Picotti, Ruth Huttenhain, Viola Heinzelmann-Schwarz, Marko Jovanovic, Ruedi Aebersold, Olga Vitek. Protein significance analysis in selected reaction monitoring (SRM) measurements. *Molecular & Cellular Proteomics*, 11:M111.014662, 2012.

Examples

```
comparison<-matrix(c(-1,0,0,0,0,0,1,0,0,0),nrow=1)
row.names(comparison)<-"T7-T1"

testResultOneComparison<-groupComparison(contrast.matrix=comparison,data=QuantData)

testResultOneComparison$ComparisonResult
```

`transformMSnSetToMSstats` *Transforms a MSnSet class dataset into a required input for MSstats*

Description

Convert MSnSet class into the required input format for MSstats

Usage

```
transformMSnSetToMSstats(ProteinName,PeptideSequence, PrecursorCharge, FragmentIon, ProductCharge ,Is
```

Arguments

`data`	name of dataset with MSnSet class
`ProteinName`	name of column in the MSnSet that contains protein information. If not assigned, "ProteinAccession" column will be used.
`PeptideSequence`	name of column in the MSnSet that contains information of peptide sequence. If not assigned, "PeptideSequence" column will be used.
`PrecursorCharge`	name of column in the MSnSet that contains information of peptide charge. If not assigned, "charge" will be used.
`FragmentIon`	name of column in the MSnSet that contains information of transition. If not assigned, value of "NA" will be used.
`ProductCharge`	name of column in the MSnSet that contains information of transition charge. If not assigned, value of "NA" will be used.
`IsotopeLabelType`	name of the column in `phenoData` component of MSnSet that contains labeling information. If not assigned, "mz" column will be used.
`Bioreplicate`	name of the column in `phenoData` component of MSnSet that contains unique ids of biological replicates of the corresponding samples. If not assigned, rownames of pData(data) will be used.
`Run`	name of the column in MSnSet that contains information of experimental MS runs. If not assigned, "file" column will be used.
`Condition`	names of the columns in `phenoData` that correspond to the group variables of interest. If more than one variable is listed, a concatentated variable is created based on the variables.

Details

raw : See MSnSet for the general format on the proteomics. Condition must be specified. Intensity should not be specified, as this information is extracted automatically from the assayData component of the MSnSet.

Warning

The types of experiment that MSstats can analyze are LC-MS, SRM, DIA(SWATH) with label-free or labeled synthetic peptides. MSstats does not support for metabolic labeling or iTRAQ experiments.

Author(s)

Ching-Yun Chang, Meena Choi, Olga Vitek.

Maintainer: Meena Choi (<choi67@purdue.edu>)

References

Ching-Yun Chang, Paola Picotti, Ruth Huttenhain, Viola Heinzelmann-Schwarz, Marko Jovanovic, Ruedi Aebersold, Olga Vitek. "Protein significance analysis in selected reaction monitoring (SRM) measurements." *Molecular & Cellular Proteomics*, 11:M111.014662, 2012.

Timothy Clough, Safia Thaminy, Susanne Ragg, Ruedi Aebersold, Olga Vitek. "Statistical protein quantification and significance analysis in label-free LC-M experiments with complex designs" *BMC Bioinformatics*, 13:S16, 2012.

Gatto, L. and Lilly, K.S. (2012). MSnbase-an R Bioconductor package for isobaric tagged mass spectrometry data visualization, processing and quantitation. *Bioinformatics*, 28, 288-289.

Examples

```
library("MSnbase")
data(itraqdata)
class(itraqdata)

msnset <- quantify(itraqdata[10:15], method = "trap", reporters = iTRAQ4, verbose = FALSE)
msnset

pData(msnset)$group<-c("control","disease","control","disease")

transformMSnSetToMSstats(data=msnset,Condition="group")
```

`transformMSstatsToMSnSet` *Transformation input format for MSstats to MSnSet class*

Description

Convert the required input format for MSstats into general format (MSnSet class in MSnbase package) on the proteomics.

Usage

```
transformMSstatsToMSnSet(data)
```

Arguments

`data`	name of the raw (input) data set with required column for MSstats.

Details

- raw : See `RawData` for the required data structure of raw (input) data.
- output : After transformation, assayData includes value of Intensity. phenoData has variables of IsotopeLabelType, Condition, BioReplicate,Run. featureData has variables of ProteinName, PeptideSequence, PrecursorCharge, FragmentIon, ProductCharge.

Author(s)

Ching-Yun Chang, Meena Choi, Olga Vitek.

Maintainer: Meena Choi (<choi67@purdue.edu>)

References

Ching-Yun Chang, Paola Picotti, Ruth Huttenhain, Viola Heinzelmann-Schwarz, Marko Jovanovic, Ruedi Aebersold, Olga Vitek. "Protein significance analysis in selected reaction monitoring (SRM) measurements." *Molecular & Cellular Proteomics*, 11:M111.014662, 2012.

Timothy Clough, Safia Thaminy, Susanne Ragg, Ruedi Aebersold, Olga Vitek. "Statistical protein quantification and significance analysis in label-free LC-M experiments with complex designs" *BMC Bioinformatics*, 13:S16, 2012.

Gatto, L. and Lilly, K.S. (2012). MSnbase-an R Bioconductor package for isobaric tagged mass spectrometry data visualization, processing and quantitation. *Bioinformatics*, 28, 288-289.

Examples

```
library(MSnbase)
quant.msnset<-transformMSstatsToMSnSet(RawData)
```

Index