Package 'SafeQuant'

April 8, 2016

Type Package

Title A Toolbox for the Analysis of Proteomics Data

Version 2.3

Date 2016-04-08

Author Erik Ahrne

Maintainer Erik Ahrne <erik.ahrne@unibas.ch>

Description Tools for the statistical analysis and visualization of (relative and absolute) quantitative (LFQ,TMT,HRM) Proteomics data.

Imports limma,
gplots,
seqinr,
corrplot,
optparse,
data.table,
epiR,
Biobase

License GPL-3

R topics documented:

`addIdQvalues` *Add identification leve q-values to ExpressionSet (calculated based on target-decoy score distribution)*

Description

Add identification leve q-values to ExpressionSet (calculated based on target-decoy score distribution)

Usage

```
addIdQvalues(eset = eset)
```

Arguments

`eset`	ExpressionSet

Details

if ptm column is part if the ExpressionSet q-values are calculated seperately for modified and non-modified features

Value

ExpressionSet object

Note

No note

See Also

`getIdLevelQvals`

Examples

```
print("No examples")
```

addScaffoldPTMFAnnotations *Add scaffold ptm annotaitons to tmt experiment*

Description

Add scaffold ptm annotaitons to tmt experiment

Usage

```
addScaffoldPTMFAnnotations(eset, file)
```

Arguments

eset	ExpressionSet
file	path to Scaffold file

Value

ExpressionSet object

Note

No note

References

No references

Examples

```
print("No examples")
```

barplotMSSignal *Barplot of ms-signal per column*

Description

Barplot of ms-signal per column

Usage

```
barplotMSSignal(eset,
  col = as.character(.getConditionColors(eset)[pData(eset)$condition, ]),
  method = c("sum", "sharedSignal"), cex.lab = 1.25, cex.axis = 1.25,
  cex.names = 0.9, labels = rownames(pData(eset)), ...)
```

Arguments

eset	expressionSet
col	default condition colors
method	c("median","sum","sharedSignal")
cex.lab	default 1.25
cex.axis	default 1.25
cex.names	default 0.9
labels	labels
...	see plot

Details

No details

Note

No note

References

NA

Examples

```
print("No examples")
```

---

calibrationCurve *S3 class object describing a calibration curve and storing some figures of merit*

Description

S3 class object describing a calibration curve and storing some figures of merit

Usage

```
calibrationCurve(eset, method = "blank")
```

Arguments

eset	ExpressionSet
method	to calculate Limit of Detection / Limit of Quantification. c("blank","low")

Details

No details

Value

calibrationCurve object

Note

No note

References

Statistical characterization of multiple-reaction monitoring mass spectrometry (MRM-MS) assays for quantitative proteomics, Mani et al. (2012), http://www.ncbi.nlm.nih.gov/pubmed/23176545

Examples

```
print("No examples")
```

COLORS *color vector*

Description

color vector

Usage

```
COLORS
```

Format

```
 chr [1:668] "red" "darkgreen" "blue" "darkmagenta" "darkorange" ...
```

---

createExpDesign *Create Experimental Design*

Description

Create Experimental Design

Usage

```
createExpDesign(tag, nbPlex)
```

Arguments

tag	user input tag e.g. 1,2,3:4,5,6 indicating two condition with 3 reps each
nbPlex	tmt 6 or 10 plex

Details

The first listed condition is always the control condition

Value

expDesign data.frame

Note

No note

References

NA

Examples

```
print("No examples")
```

`createExpressionDataset` *Create ExpressionSet object*

Description

Create ExpressionSet object

Usage

```
createExpressionDataset(expressionMatrix = expressionMatrix,
  expDesign = expDesign, featureAnnotations = featureAnnotations)
```

Arguments

`expressionMatrix`	matrix of expression signals per feature and sample
`expDesign`	experimental design data.frame
`featureAnnotations`	data.frame including e.g: Protein Description, Id score etc.

Details

No details

Value

ExpressionSet object

Note

No note

References

NA

See Also

`ExpressionSet`

Examples

```
print("No examples")
```

`createPairedExpDesign` *Create Paired Expdesign*

Description

Create Paired Expdesign

Usage

```
createPairedExpDesign(eset)
```

Arguments

`eset`	ExpressionSet

Details

Add subject colum to phenoData design data.frame

Value

ExpressionSet object

Note

No note

References

NA

See Also

`ExpressionSet`

Examples

```
print("No examples")
```

cvBoxplot *C.V. boxplot*

Description

C.V. boxplot

Usage

```
cvBoxplot(eset,
  col = as.character(.getConditionColors(eset)[unique(pData(eset)$condition),
  ]), cex.names = 0.9, cex.axis = 1.25, cex.lab = 1.25,
  ylab = "C.V. (%)", ...)
```

Arguments

eset	ExpressionSet
col	col
cex.names	default 0.9
cex.axis	default 1.25
cex.lab	default 1.25
ylab	C.V.
...	see plot

Details

No details

Note

No note

References

NA

Examples

```
print("No examples")
```

`expDesignTagToExpDesign` *Create experimental design data.frame from user input string*

Description

Create experimental design data.frame from user input string

Usage

```
expDesignTagToExpDesign(tag, expDesignDefault)
```

Arguments

`tag`	tag
`expDesignDefault`	data.frame

Details

tag: 1,2:3:4,5,6 condition isControl 1 Condition 1 TRUE 2 Condition 1 TRUE 3 Condition 1 TRUE 4 Condition 2 FALSE 5 Condition 2 FALSE 6 Condition 2 FALSE

Value

data.frame describing experimental design

Note

No note

References

NA

Examples

```
print("No examples")
```

export *Export content of safeQuantAnalysis object*

Description

Export content of safeQuantAnalysis object

Usage

```
export(sqa, nbRows = nrow(sqa$pValue), file = NA)
```

Arguments

sqa	safeQuantAnalysis object
nbRows	Number of rows to export. Features are ordred by increasing minimal p.value
file	file path

Details

NA

Note

No note

References

NA

See Also

safeQuantAnalysis

Examples

```
print("No examples")
```

getAAProteinCoordinates *Get amino acid coordinates on protein*

Description

Get amino acid coordinates on protein

Usage

```
getAAProteinCoordinates(peptideSeq, proteinSeq, aaRegExpr = "[STY]")
```

Arguments

peptideSeq	peptide sequence
proteinSeq	protein sequence
aaRegExpr	target AA reg exp

Details

NA

Value

vector of protein coordinates (mmodification residue number)

Note

No note

References

NA

Examples

```
print("No examples")
```

---

getAllCV *Calculate Coefficiant of Variance per feature (Relative standard Deviation) per Condition*

Description

Calculate Coefficiant of Variance per feature (Relative standard Deviation) per Condition

Usage

```
getAllCV(eset)
```

Arguments

eset	ExpressionSet

Details

CV = sd / mean

Value

data.frame of CVs per condition

Note

No note

References

NA

See Also

getCV

Examples

```
print("No examples")
```

`getAllEBayes` *Perform statistical test (mderated t-test), comparing all case to control*

Description

Perform statistical test (mderated t-test), comparing all case to control

Usage

```
getAllEBayes(eset = eset, adjust = F, log = T, method = "pairwise",
  adjustFilter = matrix(F, nrow = nrow(eset), ncol =
  length(levels(pData(eset)$condition)) - 1))
```

Arguments

`eset`	ExpressionSet
`adjust`	TRUE/FALSE adjust for multiple testing using Benjamini & Hochberg (1995) method
`log`	T/F log-transform expression values
`method`	c("all","pairwise")

Details

No details

Value

ExpressionSet object

Note

No note

References

Empirical Bayes method, Smyth (2004), http://www.ncbi.nlm.nih.gov/pubmed/16646809

See Also

eBayes

Examples

```
print("No examples")
```

`getBaselineIntensity` *Get signal at zscore x (x standard deviations below mean)*

Description

Get signal at zscore x (x standard deviations below mean)

Usage

```
getBaselineIntensity(intensities, promille = 5)
```

Arguments

`intensities`	refrence run signals
`promille`	baseline value set as specified promille

Value

baseline value

Note

No note

References

NA

Examples

```
print("No examples")
```

`getCV` *Calculate Coefficiant of Variance per feature (Relative standard Deviation)*

Description

Calculate Coefficiant of Variance per feature (Relative standard Deviation)

Usage

```
getCV(data)
```

Arguments

`data`	data.frame of replicate signals

Details

CV = sd / mean

Value

vector of CVs

Note

No note

References

NA

Examples

```
print("No examples")
```

---

getExpDesignProgenesisCsv *Parse Experimental Design from Progenesis Csv Export*

Description

Parse Experimental Design from Progenesis Csv Export

Usage

```
getExpDesignProgenesisCsv(file,
  expressionColIndices = .getProgenesisCsvExpressionColIndices(file))
```

Arguments

`file`	path to progenesis csv file
`expressionColIndices`	default .getProgenesisCsvExpressionColIndices(file)

Details

No details

Value

data.frame describing experimental design

Note

No note

References

NA

Examples

```
print("No examples")
```

`getGlobalNormFactors` *Get normalization factors. calculated as summed/median signal per run (column) over summed/median of first run.*

Description

Get normalization factors. calculated as summed/median signal per run (column) over summed/median of first run.

Usage

```
getGlobalNormFactors(eset, method = "sum")
```

Arguments

`eset`	ExpressionSet
`method`	c("sum","median)

Details

No details

Value

vector of normalization factors

Note

No note

References

NA

Examples

```
print("No examples")
```

`getIBAQEset` *Calculate intensity-based absolute-protein-quantification (iBAQ) metric per protein*

Description

Calculate intensity-based absolute-protein-quantification (iBAQ) metric per protein

Usage

```
getIBAQEset(eset, proteinDB = NA, peptideLength = c(5, 36),
  nbMiscleavages = 0, proteaseRegExp = .getProteaseRegExp("trypsin"))
```

Arguments

`eset`	protein level ExpressionSet
`proteinDB`	list protein sequneces
`peptideLength`	peptide length interval (to get number of peptides used for normalization)
`nbMiscleavages`	number of mis-cleavages allowed when digesting protein sequneces in silico (to get number of peptides used for normalization)
`proteaseRegExp`	protease Reg Exp cleavage rule

Details

No details

Value

ExpressionSet

Note

No note

References

Global quantification of mammalian gene expression control, Schwanhausser (2011), http://www.ncbi.nlm.nih.gov/pubmed/21593866, Critical assessment of proteome-wide label-free absolute abundance estimation strategies. Ahrne (2013), http://www.ncbi.nlm.nih.gov/pubmed/23794183

Examples

```
print("No examples")
```

`getIdLevelQvals` *Calculates identification level q-values based on target-decoy score distributions*

Description

Calculates identification level q-values based on target-decoy score distributions

Usage

```
getIdLevelQvals(scores, isDecoy)
```

Arguments

`scores`	peptide/protein identficationscore
`isDecoy`	vector of TRUE/FALSE

Details

q-value = (Nb. Decoy Entries at idScore Threshold S*) / (Nb. Target Entries at idScore Threshold S). (* idScore >= S)

Value

vector of q.values

Note

No note

References

NA

Examples

```
print("No examples")
```

`getImpuritiesMatrix` *Get Thermo TMT impurity matrix*

Description

Get Thermo TMT impurity matrix

Usage

```
getImpuritiesMatrix(plexNb = 6)
```

Arguments

plexNb	integer, 6 or 10 plex

Details

No details

Value

impurity matrix matrix

Note

No note

References

NA

Examples

```
print("No examples")
```

---

getIntSumPerProtein *Sum up raw intensities per protein and channel. keep track of number of summed spectra and unique peptides*

Description

Sum up raw intensities per protein and channel. keep track of number of summed spectra and unique peptides

Usage

```
getIntSumPerProtein(intData, proteinACs, peptides, minNbPeptPerProt = 1)
```

Arguments

intData	data.frame of intensities per channel
proteinACs	vector of protein accession numbers
peptides	vector of peptide sequneces
minNbPeptPerProt	minimal number of peptides per protein

Details

NA

Value

list containing 3 objects 1) data.frame of channel intensities per protein ac, 2) vector listing number of summed spectra per protein, 3) vector listing number of summed peptides per protein

Note

No note

References

NA

Examples

```
print("No examples")
```

`getLoocvFoldError` *Leave-One-Out Cross Validate Qunatification Model*

Description

Leave-One-Out Cross Validate Qunatification Model

Usage

```
getLoocvFoldError(df)
```

Arguments

`df`	data.frame of two columns 1) "signal" - ms metric 2) "cpc" absolute quantity

Details

No details

Value

data.frame of fold errors per (left-out) protein

Note

No note

References

NA

See Also

NA

Examples

```
print("No examples")
```

`getMaxIndex` *get index of max in vecotr of numeric values*

Description

get index of max in vecotr of numeric values

Usage

```
getMaxIndex(v)
```

Arguments

v	vector

`getMeanCenteredRange` *Get modification coordinates on protein*

Description

Get modification coordinates on protein

Usage

```
getMeanCenteredRange(d, nbSd = 4)
```

Arguments

d	numeric vector
nbSd	range spanning number of sd frmo mean

Details

NA

Value

vector range boundries

Note

No note

References

NA

Examples

```
print("No examples")
```

`getModifProteinCoordinates` *Get modification coordinates on protein*

Description

Get modification coordinates on protein

Usage

```
getModifProteinCoordinates(modifAnnot, peptideSeq, proteinSeq, format = 1)
```

Arguments

`modifAnnot`	modifcation as annotated by progenesis. E.g. '[15] Phospho (ST)|[30] Phospho (ST)'
`peptideSeq`	peptide sequence
`proteinSeq`	protein sequence
`format`	c(1,2) 1. progenesis 2. scaffold

Details

NA

Value

vector of protein coordinates (mmodification residue number)

Note

No note

References

NA

Examples

```
print("No examples")
```

`getMotifX` *Create motif-x peptide annotation*

Description

Create motif-x peptide annotation

Usage

```
getMotifX(modifPos, peptide, proteinSeq, motifLength = 4)
```

Arguments

`modifPos`	vector positions
`peptide`	peptide sequence
`proteinSeq`	protein sequence
`motifLength`	motif flanking sequence

Details

motif-x example PGDYS*TTPG

Value

vector of motifs

Note

No note

References

NA

Examples

```
print("No examples")
```

getNbDetectablePeptides *Get number peptides passing defined length criteria*

Description

Get number peptides passing defined length criteria

Usage

```
getNbDetectablePeptides(peptides, peptideLength = c(5, 36))
```

Arguments

peptides	list of peptides
peptideLength	vector of two integers defining peptide length range

Details

No details

Value

integer corresponding to number of detectable peptides

Note

No note

Examples

```
print("No examples")
```

getNbMisCleavages *Get number of mis-cleavages perp peptide*

Description

Get number of mis-cleavages perp peptide

Usage

```
getNbMisCleavages(peptide, protease = "trypsin")
```

Arguments

peptide	character vector
protease	regular expression

Details

NA

Value

vector ofintegers

Note

No note

References

NA

Examples

```
print("No examples")
```

`getNbPeptidesPerProtein` *Get number of peptides per protein*

Description

Get number of peptides per protein

Usage

```
getNbPeptidesPerProtein(eset)
```

Arguments

`eset`	ExpressionSet

Details

NA

Value

table

Note

No note

References

NA

Examples

```
print("No examples")
```

getNbSpectraPerProtein *Get number of spectra per protein*

Description

Get number of spectra per protein

Usage

```
getNbSpectraPerProtein(eset)
```

Arguments

eset	ExpressionSet

Details

NA

Value

table

Note

No note

References

NA

Examples

```
print("No examples")
```

getPeptides *Digest protein*

Description

Digest protein

Usage

```
getPeptides(proteinSeq, proteaseRegExp = .getProteaseRegExp("trypsin"),
  nbMiscleavages = 0)
```

Arguments

proteinSeq	protein sequence
proteaseRegExp	protease Regular Expression
nbMiscleavages	default 0

Details

No details

Value

vector of peptides

Note

No note

Examples

```
print("No examples")
```

---

getRatioCorrectionFactorModel

Get linear model explaning log2 ratio as a function of log2 tmt ratio

---

Description

Get linear model explaning log2 ratio as a function of log2 tmt ratio

Usage

```
getRatioCorrectionFactorModel(eset)
```

Arguments

eset	paired calibration mix

Details

Uses linear model of log tmt ratio vs log ref ratio

Value

linear model

Note

No note

References

NA

Examples

```
print("No examples")
```

`getRatios` *Calculate ratios, comparing all case to control*

Description

Calculate ratios, comparing all case to control

Usage

```
getRatios(eset, method = "median", log2 = T)
```

Arguments

`eset`	ExpressionSet
`method`	median, mean, paired
`log2`	transform

Details

No details

Value

ExpressionSet object

Note

No note

References

NA

Examples

```
print("No examples")
```

`getRTNormFactors` *Get retentiontime base normalization factors*

Description

Get retentiontime base normalization factors

Usage

```
getRTNormFactors(eset, minFeaturesPerBin = 100)
```

Arguments

`eset`	ExpressionSet
`minFeaturesPerBin`	minumum number of features per bin. If nb. features are < minFeaturesPerBin -> include neighbouring bins.

Details

No details

Value

data.frame normalization factors per retention time bin (minute)

Note

No note

References

In Silico Instrumental Response Correction Improves Precision of Label-free Proteomics and Accuracy of Proteomics-based Predictive Models, Lyutvinskiy et al. (2013), http://www.ncbi.nlm.nih.gov/pubmed/23589346

Examples

```
print("No examples")
```

getScoreCutOff *Get score cutoff for a given fdr cut-off*

Description

Get score cutoff for a given fdr cut-off

Usage

```
getScoreCutOff(scores, isDecoy, fdrCutOff = 0.01)
```

Arguments

scores	peptide/protein identficationscore
isDecoy	vector of TRUE/FALSE
fdrCutOff	[0,1]

Details

NA

Value

scoreCutoff

Note

No note

References

NA

Examples

```
print("No examples")
```

getSignalPerCondition *Summarize replicate signal per condition (min)*

Description

Summarize replicate signal per condition (min)

Usage

```
getSignalPerCondition(eset, method = "median")
```

Arguments

`eset`	ExpressionSet
`method`	median (default), mean, max, min, sd

Details

No details

Value

data.frame of per condition signals

Note

No note

References

NA

Examples

```
print("No examples")
```

`getTopX` *Calculate Mean of X most intense features*

Description

Calculate Mean of X most intense features

Usage

```
getTopX(entryData, topX = 3)
```

Arguments

`entryData`	data.frame listing feature intensities of one entry. Typically rows corresponds to Peptide entries of one protein
`topX`	best X flyers

Details

No details

Value

vector of topX intensities per column (sample)

Note

No note

References

Absolute quantification of proteins by LCMSE: A virtue of parallel MS acquisition, Silva (2006), http://www.ncbi.nlm.nih.gov/pubmed/16219938, Critical assessment of proteome-wide label-free absolute abundance estimation strategies. Ahrne (2013), http://www.ncbi.nlm.nih.gov/pubmed/23794183

Examples

```
print("No examples")
```

---

getUserOptions *Read User Specified Command Line Options*

Description

Read User Specified Command Line Options

Usage

```
getUserOptions(version = version)
```

Arguments

version	Safequant version number

Details

No details

Value

user options list

Note

No note

References

NA

Examples

```
print("No examples")
```

`globalNormalize` *Normalize, Norm factors calculated as median signal per run (column) over median of first run.*

Description

Normalize, Norm factors calculated as median signal per run (column) over median of first run.

Usage

```
globalNormalize(eset, globalNormFactors)
```

Arguments

`eset`	ExpressionSet
`globalNormFactors`	globalNormFactors

Details

No details

Value

eset ExpressionSet

Note

No note

References

NA

See Also

getGlobalNormFactors

Examples

```
print("No examples")
```

`hClustHeatMap` *Hierarchical clustering heat map, cluster by runs intensity, features by ratio and display log2 ratios to control median*

Description

Hierarchical clustering heat map, cluster by runs intensity, features by ratio and display log2 ratios to control median

Usage

```
hClustHeatMap(eset, conditionColors = .getConditionColors(eset),
  breaks = seq(-2, 2, length = 20), dendogram = "column",
  legendPos = "left", ...)
```

Arguments

`eset`	ExpressionSet
`conditionColors`	data.frame of colors per condition
`breaks`	default seq(-2,2,length=20)
`dendogram`	see heatmap.2 gplots
`legendPos`	see legend
`...`	see plot

Details

No details

Note

No note

References

NA

Examples

```
print("No examples")
```

isCon *Check if protein is a contaminant entry*

Description

Check if protein is a contaminant entry

Usage

```
isCon(ac)
```

Arguments

ac	vector of protein accession numbers

Details

contanminants proteins are typically annotated as: CON_P0000

Value

vector TRUE/FALSE

Note

No note

References

NA

Examples

```
print("No examples")
```

isDecoy *Check if protein is a decoy entry*

Description

Check if protein is a decoy entry

Usage

```
isDecoy(ac)
```

Arguments

ac	vector of protein accession numbers

Details

decoy proteins are typically annotated as: REV_P0000

Value

vector TRUE/FALSE

Note

No note

References

NA

Examples

```
print("No examples")
```

`isStrippedACs` *Check if ACs are in "non-stripped" uniprot format e.g. "sp|Q8CHJ2|AQP12_MOUSE"*

Description

Check if ACs are in "non-stripped" uniprot format e.g. "sp|Q8CHJ2|AQP12_MOUSE"

Usage

```
isStrippedACs(acs)
```

Arguments

`acs`	accession numbers

Details

TRUE if less than 10

Value

boolean TRUE/FALSE

Note

No note

References

NA

Examples

```
print("No examples")
```

missinValueBarplot *Plot Percentage of Features with with missing values*

Description

Plot Percentage of Features with with missing values

Usage

```
missinValueBarplot(eset,
  col = as.character(.getConditionColors(eset)[pData(eset)$condition, ]),
  cex.axis = 1.25, cex.lab = 1.25, ...)
```

Arguments

eset	ExpressionSet
col	col
cex.axis	cex.axis
cex.lab	cex.lab
...	see plot

Details

No details

Note

No note

References

NA

Examples

```
print("No examples")
```

option_list *Command Line Option List*

Description

Command Line Option List

Usage

```
option_list
```

Format

```
List of 27
 $ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
  .. ..@ short_flag: chr "-i"
  .. ..@ long_flag : chr "--inputFile"
  .. ..@ action    : chr "store"
  .. ..@ type      : chr "character"
  .. ..@ dest      : chr "inputFile"
  .. ..@ default   : chr ""
  .. ..@ help      : chr "I/O:  Input file: Progenesis (Feature,Protein or Peptide) .csv,\n\t\t\tor
  .. ..@ metavar   : chr "inputFile"
 $ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
  .. ..@ short_flag: chr "-o"
  .. ..@ long_flag : chr "--outputDir"
  .. ..@ action    : chr "store"
  .. ..@ type      : chr "character"
  .. ..@ dest      : chr "outputDir"
  .. ..@ default   : chr NA
  .. ..@ help      : chr "I/O:  Results Output Directory [default FOLDER OF INPUTFILE]"
  .. ..@ metavar   : chr "outputDir"
 $ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
  .. ..@ short_flag: chr "-l"
  .. ..@ long_flag : chr "--resultsFileLabel"
  .. ..@ action    : chr "store"
  .. ..@ type      : chr "character"
  .. ..@ dest      : chr "resultsFileLabel"
  .. ..@ default   : chr "SQ_Results"
  .. ..@ help      : chr "I/O: results file directory [default %default]"
  .. ..@ metavar   : chr "resultsFileLabel"
 $ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
  .. ..@ short_flag: chr "-f"
  .. ..@ long_flag : chr "--fastaFile"
  .. ..@ action    : chr "store"
  .. ..@ type      : chr "character"
  .. ..@ dest      : chr "fastaFile"
  .. ..@ default   : chr ""
  .. ..@ help      : chr "I/O:  Protein DB .fasta file [default ./]"
  .. ..@ metavar   : chr "fastaFile"
 $ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
  .. ..@ short_flag: chr "-p"
  .. ..@ long_flag : chr "--scaffoldPTMSpectrumReportFile"
  .. ..@ action    : chr "store"
  .. ..@ type      : chr "character"
  .. ..@ dest      : chr "scaffoldPTMSpectrumReportFile"
  .. ..@ default   : chr ""
  .. ..@ help      : chr "I/O:  Scaffold PTM Spectrum Report File [default ./]"
  .. ..@ metavar   : chr "scaffoldPTMSpectrumReportFile"
 $ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
  .. ..@ short_flag: chr NA
  .. ..@ long_flag : chr "--FProteinAccessionSelection"
  .. ..@ action    : chr "store"
  .. ..@ type      : chr "character"
```

```
 .. ..@ dest      : chr "FProteinAccessionSelection"
 .. ..@ default   : chr "."
 .. ..@ help      : chr "FILTER: --FP Filter features by Accession Regular Expression [default %def
 .. ..@ metavar   : chr "Protein Accession Reg. expr."
$ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
 .. ..@ short_flag: chr NA
 .. ..@ long_flag : chr "--FModificationSelection"
 .. ..@ action    : chr "store"
 .. ..@ type      : chr "character"
 .. ..@ dest      : chr "FModificationSelection"
 .. ..@ default   : chr ""
 .. ..@ help      : chr "FILTER (LFQ PEP ONLY): --FM Only keep Peptides with modifications matching
 .. ..@ metavar   : chr "modification name Reg. expr."
$ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
 .. ..@ short_flag: chr NA
 .. ..@ long_flag : chr "--FFdrCutoff"
 .. ..@ action    : chr "store"
 .. ..@ type      : chr "double"
 .. ..@ dest      : chr "FFdrCutoff"
 .. ..@ default   : num 0.01
 .. ..@ help      : chr "FILTER (LFQ ONLY): --FF Identification level False Discovery Rate Cutoff.
 .. ..@ metavar   : chr "Peptide/Protein FDR cutoff"
$ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
 .. ..@ short_flag: chr NA
 .. ..@ long_flag : chr "--FDeltaMassTolerancePrecursor"
 .. ..@ action    : chr "store"
 .. ..@ type      : chr "character"
 .. ..@ dest      : chr "FDeltaMassTolerancePrecursor"
 .. ..@ default   : chr "AUTO SET"
 .. ..@ help      : chr "FILTER (LFQ PEP ONLY): --FD Precursor mass Error Range filter (ppm) [defaul
 .. ..@ metavar   : chr "Mass Range [x,y]"
$ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
 .. ..@ short_flag: chr NA
 .. ..@ long_flag : chr "--FNumberOfPeptidesPerProteinMin"
 .. ..@ action    : chr "store"
 .. ..@ type      : chr "integer"
 .. ..@ dest      : chr "FNumberOfPeptidesPerProteinMin"
 .. ..@ default   : int 1
 .. ..@ help      : chr "FILTER: --FN Only include those proteins with at least x identified peptide
 .. ..@ metavar   : chr "Number of peptides"
$ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
 .. ..@ short_flag: chr NA
 .. ..@ long_flag : chr "--FSitesPerPeptide"
 .. ..@ action    : chr "store"
 .. ..@ type      : chr "integer"
 .. ..@ dest      : chr "FSitesPerPeptide"
 .. ..@ default   : int 99999
 .. ..@ help      : chr "FILTER: --FS Max Nb. Modifications Per Peptide [default Inf]\n\t\t\t\t\t\t
 .. ..@ metavar   : chr "Max Number of PTM sites Per Petptide"
$ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
 .. ..@ short_flag: chr NA
 .. ..@ long_flag : chr "--FLengthPeptide"
```

```
 .. ..@ action    : chr "store"
 .. ..@ type      : chr "integer"
 .. ..@ dest      : chr "FLengthPeptide"
 .. ..@ default   : int 1
 .. ..@ help      : chr "FILTER: --FL Min Peptide Length (Nb. AA's) [default Inf]\n\t\t\t\t\t\tPept
 .. ..@ metavar   : chr "Min Peptide Length (>=)"
$ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
 .. ..@ short_flag: chr NA
 .. ..@ long_flag : chr "--FUniquePeptides"
 .. ..@ action    : chr "store_true"
 .. ..@ type      : chr "logical"
 .. ..@ dest      : chr "FUniquePeptides"
 .. ..@ default   : logi FALSE
 .. ..@ help      : chr "FILTER: --FU Discard all peptides mapping to multiple protein entries [def
 .. ..@ metavar   : chr(0)
$ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
 .. ..@ short_flag: chr NA
 .. ..@ long_flag : chr "--FRatioCutOff"
 .. ..@ action    : chr "store"
 .. ..@ type      : chr "double"
 .. ..@ dest      : chr "FRatioCutOff"
 .. ..@ default   : num 1
 .. ..@ help      : chr "FILTER: --FR Intensity ratio (fold change) cut-off used for graphics expor
 .. ..@ metavar   : chr "Intensity ratio cutoff"
$ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
 .. ..@ short_flag: chr NA
 .. ..@ long_flag : chr "--TAdjustRatios"
 .. ..@ action    : chr "store_true"
 .. ..@ type      : chr "logical"
 .. ..@ dest      : chr "TAdjustRatios"
 .. ..@ default   : logi FALSE
 .. ..@ help      : chr "TMT: --TA Adjust TMT ratios using calibration mix proteins [default %defau
 .. ..@ metavar   : chr(0)
$ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
 .. ..@ short_flag: chr NA
 .. ..@ long_flag : chr "--SAnchorProtein"
 .. ..@ action    : chr "store"
 .. ..@ type      : chr "character"
 .. ..@ dest      : chr "SAnchorProtein"
 .. ..@ default   : chr "."
 .. ..@ help      : chr "STATISTICS: --SA Normalize Intensities by selected protein(s) Regular Expr
 .. ..@ metavar   : chr "Protein Accession Reg. expr."
$ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
 .. ..@ short_flag: chr NA
 .. ..@ long_flag : chr "--SPvalueInclude"
 .. ..@ action    : chr "store_true"
 .. ..@ type      : chr "logical"
 .. ..@ dest      : chr "SPvalueInclude"
 .. ..@ default   : logi FALSE
 .. ..@ help      : chr "STATISTICS: --SP output eBayes moderated t-statistic p-values [default %de
 .. ..@ metavar   : chr(0)
$ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
```

```
 .. ..@ short_flag: chr NA
 .. ..@ long_flag : chr "--SNonPairWiseStatTest"
 .. ..@ action    : chr "store_true"
 .. ..@ type      : chr "logical"
 .. ..@ dest      : chr "SNonPairWiseStatTest"
 .. ..@ default   : logi FALSE
 .. ..@ help      : chr "STATISTICS: --SN non pairwise eBayes moderated t-statistic p-values.\n\t\t
 .. ..@ metavar   : chr(0)
$ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
 .. ..@ short_flag: chr NA
 .. ..@ long_flag : chr "--EXperimentalDesign"
 .. ..@ action    : chr "store"
 .. ..@ type      : chr "character"
 .. ..@ dest      : chr "EXperimentalDesign"
 .. ..@ default   : chr NA
 .. ..@ help      : chr "EXPERIMENTAL DESIGN: --EX \",\" seperated samples, \":\" separated conditi
 .. ..@ metavar   : chr "EXperimentalDesign"
$ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
 .. ..@ short_flag: chr NA
 .. ..@ long_flag : chr "--EProteinQuantOff"
 .. ..@ action    : chr "store_false"
 .. ..@ type      : chr "logical"
 .. ..@ dest      : chr "EProteinQuantOff"
 .. ..@ default   : logi TRUE
 .. ..@ help      : chr "EXPERIMENTAL DESIGN: --EP Disable Protein Level Quantification [default %d
 .. ..@ metavar   : chr(0)
$ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
 .. ..@ short_flag: chr NA
 .. ..@ long_flag : chr "--ECorrelatedSamples "
 .. ..@ action    : chr "store_true"
 .. ..@ type      : chr "logical"
 .. ..@ dest      : chr "ECorrelatedSamples "
 .. ..@ default   : logi FALSE
 .. ..@ help      : chr "EXPERIMENTAL DESIGN: --EC Apply \"paired\" statistical tests [default %def
 .. ..@ metavar   : chr(0)
$ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
 .. ..@ short_flag: chr NA
 .. ..@ long_flag : chr "--PQvalueCutOff"
 .. ..@ action    : chr "store"
 .. ..@ type      : chr "double"
 .. ..@ dest      : chr "PQvalueCutOff"
 .. ..@ default   : num 0.01
 .. ..@ help      : chr "PDF-REPORT: --PQ Qvalue cut-off used for graphics. \n\t\t\tHigh-lighting f
 .. ..@ metavar   : chr "Differential expression qvalue cutOff"
$ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
 .. ..@ short_flag: chr NA
 .. ..@ long_flag : chr "--ARDataFile"
 .. ..@ action    : chr "store_true"
 .. ..@ type      : chr "logical"
 .. ..@ dest      : chr "ARDataFile"
 .. ..@ default   : logi FALSE
 .. ..@ help      : chr "ADDITIONAL-REPORTS: --AR Save R objects in 'label'.RData file [default %de
```

```
  .. ..@ metavar   : chr(0)
 $ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
  .. ..@ short_flag: chr NA
  .. ..@ long_flag : chr "--AIbaq"
  .. ..@ action    : chr "store_true"
  .. ..@ type      : chr "logical"
  .. ..@ dest      : chr "AIbaq"
  .. ..@ default   : logi FALSE
  .. ..@ help      : chr "ADDITIONAL-REPORTS (LFQ PROT): --AI creates .tsv output file\n\t\t\t\t\tir
  .. ..@ metavar   : chr(0)
 $ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
  .. ..@ short_flag: chr NA
  .. ..@ long_flag : chr "--ATop3"
  .. ..@ action    : chr "store_true"
  .. ..@ type      : chr "logical"
  .. ..@ dest      : chr "ATop3"
  .. ..@ default   : logi FALSE
  .. ..@ help      : chr "ADDITIONAL-REPORTS (LFQ PEP): --AT creates .tsv output file\n\t\t\t\t\tinc
  .. ..@ metavar   : chr(0)
 $ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
  .. ..@ short_flag: chr "-t"
  .. ..@ long_flag : chr "--test"
  .. ..@ action    : chr "store_true"
  .. ..@ type      : chr "logical"
  .. ..@ dest      : chr "test"
  .. ..@ default   : logi FALSE
  .. ..@ help      : chr "TEST: test option, include first 2000 entries only [default %default]\n\t\
  .. ..@ metavar   : chr(0)
 $ :Formal class 'OptionParserOption' [package "optparse"] with 8 slots
  .. ..@ short_flag: chr "-v"
  .. ..@ long_flag : chr "--verbose"
  .. ..@ action    : chr "store_true"
  .. ..@ type      : chr "logical"
  .. ..@ dest      : chr "verbose"
  .. ..@ default   : logi FALSE
  .. ..@ help      : chr "Print extra output [default %default]"
  .. ..@ metavar   : chr(0)
```

`pairsAnnot` *Plot lower triangle Pearsons R^2. Diagonal text, upper triangle all against all scatter plots with lm abline*

Description

Plot lower triangle Pearsons R^2. Diagonal text, upper triangle all against all scatter plots with lm abline

Usage

```
pairsAnnot(data, textCol = rep(1, ncol(data)), diagText = colnames(data),
  col = rgb(0, 100, 0, 50, maxColorValue = 255), isHeatCol = F,
  cexTxt = 2, ...)
```

Arguments

data	data.frame
textCol	text color
diagText	diagnoal text
col	dot col
isHeatCol	heat colors
cexTxt	cex txt
...	see plot

Details

No details

Note

No note

References

NA

Examples

```
print("No examples")
```

---

parseMaxQuantProteinGroupTxt

Parse MaxQuant Protein Group Txt

Description

Parse MaxQuant Protein Group Txt

Usage

```
parseMaxQuantProteinGroupTxt(file = file, expDesign = expDesign,
  method = "auc")
```

Arguments

file	path to MaxQuant Protein txt file
expDesign	experimental design data.frame
method	auc (area under curve) or spc (spectral count)

Details

No details

Value

ExpressionSet object

Note

No note

References

NA

See Also

ExpressionSet

Examples

```
print("No examples")
```

`parseProgenesisFeatureCsv`

Parse Progenesis Feature Csv Export

Description

Parse Progenesis Feature Csv Export

Usage

```
parseProgenesisFeatureCsv(file = file,
  expDesign = getExpDesignProgenesisCsv(file), method = "auc")
```

Arguments

`file`	path to Progenesis Feature csv file
`expDesign`	experimental design data.frame
`method`	auc (area under curve) or spc (spectral count)

Details

No details

Value

ExpressionSet object

Note

No note

References

NA

See Also

ExpressionSet

Examples

```
print("No examples")
```

`parseProgenesisPeptideMeasurementCsv`
Parse Progenesis Peptide Measurement Csv Export

Description

Parse Progenesis Peptide Measurement Csv Export

Usage

```
parseProgenesisPeptideMeasurementCsv(file, expDesign = expDesign,
  method = "auc",
  expressionColIndices = .getProgenesisCsvExpressionColIndices(file, method =
  method), uniqueProteins = F)
```

Arguments

`file`	path to Progenesis Peptide Measurement csv file
`expDesign`	experimental design data.frame
`method`	auc (area under curve) or spc (spectral count)
`expressionColIndices`	default .getProgenesisCsvExpressionColIndices()
`uniqueProteins`	T/F keep unique peptides only

Details

No details

Value

ExpressionSet object

Note

No note

References

NA

See Also

ExpressionSet

Examples

```
print("No examples")
```

`parseProgenesisProteinCsv` *Parse Progenesis Protein Csv*

Description

Parse Progenesis Protein Csv

Usage

```
parseProgenesisProteinCsv(file = file, expDesign = expDesign,
  method = "auc")
```

Arguments

`file`	path to Progenesis Protein csv file
`expDesign`	experimental design data.frame
`method`	auc (area under curve) or spc (spectral count)

Details

No details

Value

ExpressionSet object

Note

No note

References

NA

See Also

ExpressionSet

Examples

```
print("No examples")
```

parseScaffoldPTMReport *Parse scaffold PTM Spectrum Report*

Description

Parse scaffold PTM Spectrum Report

Usage

```
parseScaffoldPTMReport(file)
```

Arguments

file	path to Scaffold file

Details

No details

Value

data.frame

Note

No note

References

NA

Examples

```
print("No examples")
```

parseScaffoldRawFile *Parse scaffold output .xls file (RAW export)*

Description

Parse scaffold output .xls file (RAW export)

Usage

```
parseScaffoldRawFile(file, expDesign = expDesign, keepFirstAcOnly = FALSE,
  isPurityCorrect = T)
```

Arguments

file	path to Scaffold file
expDesign	experimental design data.frame
keepFirstAcOnly	TRUE/FALSE If multiple ACs in Accession.Numbers filed. Then keep the first one only
isPurityCorrect	do purity correction

Details

No details

Value

ExpressionSet object

Note

No note

References

NA

See Also

ExpressionSet

Examples

```
print("No examples")
```

---

perFeatureNormalization

Per Feature Normalization

Description

Per Feature Normalization

Usage

```
perFeatureNormalization(eset, normFactors)
```

Arguments

eset	ExpressionSet
normFactors	matrix normalization factors (logged) (row names are proteins)

Details

Example Usage: Normalize phospho peptide signals for Protein Changes

Value

ExpressionSet object

Note

No note

References

No references

Examples

```
print("No examples")
```

`plotAbsEstCalibrationCurve`
Plot absolut Estimation calibration Curve

Description

Plot absolut Estimation calibration Curve

Usage

```
plotAbsEstCalibrationCurve(fit, dispElements = c("formula", "lowess",
  "stats"), xlab = "Conc. (CPC) ", ylab = "Pred. Conc. (CPC) ",
  predictorName = paste("log10(", names(coef(fit))[2], ")", sep = ""),
  text = F, cex.lab = 1, cex.axis = 1, cex.text = 1, cex.dot = 1,
  main = "", ...)
```

Arguments

`fit`	simple log-linear model
`dispElements`	c("formula","lowess","stats")
`xlab`	xlab
`ylab`	ylab
`predictorName`	predictorName
`text`	add names beside each dot
`cex.lab`	expansion factor for axis labels
`cex.axis`	expansion factor for axis
`cex.text`	expansion factor for legend
`cex.dot`	expansion factor for plotted dots
`main`	main
`...`	see plot

Note

No note

References

NA

Examples

```
print("No examples")
```

`plotExpDesign` *Display experimental design, high-lighting the control condition*

Description

Display experimental design, high-lighting the control condition

Usage

```
plotExpDesign(eset, condColors = .getConditionColors(eset), version = "X")
```

Arguments

`eset`	ExpressionSet
`condColors`	condition colors
`version`	version number

Details

No details

Note

No note

References

NA

Examples

```
print("No examples")
```

plotIdScoreVsFDR *Plot FDR vs. identification score*

Description

Plot FDR vs. identification score

Usage

```
plotIdScoreVsFDR(idScore, qvals, qvalueThrs = 0.01,
  ylab = "False Discovery Rate", xlab = "Identification Score", ...)
```

Arguments

idScore	vector of identification scores
qvals	vector of q-valres
qvalueThrs	threshold indicated by horizontal line
ylab	default False Discovery Rate
xlab	default Identification Score
...	see plot

Details

No details

Note

No note

References

NA

Examples

```
print("No examples")
```

plotMSSignalDistributions *Plot ms.signal distributions*

Description

Plot ms.signal distributions

Usage

```
plotMSSignalDistributions(d, col = 1:100, ylab = "Frequnecy",
  xlab = "MS-Signal", ...)
```

Arguments

d	matrix of ms-signals
col	color
ylab	default "Frequnecy"
xlab	default "MS-Signal"
...	see plot

Details

No details

Note

No note

References

NA

Examples

```
print("No examples")
```

---

plotNbIdentificationsVsRT

Plot the number of identified Features per Reteintion Time minute.

Description

Plot the number of identified Features per Reteintion Time minute.

Usage

```
plotNbIdentificationsVsRT(eset, cex.axis = 1.25, cex.lab = 1.25,
  col = "blue", lwd = 2, ...)
```

Arguments

eset	ExpressionSet
cex.axis	default 1.25
cex.lab	default 1.25
col	default "blue"
lwd	default 2
...	see plot see plot

Note

No note

References

NA

Examples

```
print("No examples")
```

`plotNbValidDeFeaturesPerFDR`

Plot Total Number of diffrentially Abundant Features (applying ratio cutoff) vs. qValue/pValue for all conditions

---

Description

Plot Total Number of diffrentially Abundant Features (applying ratio cutoff) vs. qValue/pValue for all conditions

Usage

```
plotNbValidDeFeaturesPerFDR(sqa, upRegulated = T, log2RatioCufOff = log2(1),
  pvalRange = c(0, 0.3), pvalCutOff = 1, isLegend = T, isAdjusted = T,
  ylab = "Nb. Features", ...)
```

Arguments

`sqa`	SafeQuantAnalysis Object
`upRegulated`	TRUE/FALSE select for upregulated features
`log2RatioCufOff`	log2 ratio cut-off
`pvalRange`	pValue/qValue range
`pvalCutOff`	pValue/qValue cut-off
`isLegend`	TRUE/FALSE display legend
`isAdjusted`	TRUE/FALSE qValues/pValue on x-axis
`ylab`	default Nb. Features
`...`	see plot

Details

No details

Note

No note

References

NA

Examples

```
print("No examples")
```

`plotPrecMassErrorDistrib` *Plot Precursor Mass Error Distribution*

Description

Plot Precursor Mass Error Distribution

Usage

```
plotPrecMassErrorDistrib(eset, pMassTolWindow = c(-10, 10), ...)
```

Arguments

`eset`	ExpressionSet
`pMassTolWindow`	Precursor Mass Error Tolerance Window
`...`	see plot

Details

No details

Note

No note

References

NA

Examples

```
print("No examples")
```

`plotPrecMassErrorVsScore` *Plot precursorMass error v.s score highlighting decoy and displaying user specified user specified precursor mass filter*

Description

Plot precursorMass error v.s score highlighting decoy and displaying user specified user specified precursor mass filter

Usage

```
plotPrecMassErrorVsScore(eset, pMassTolWindow = c(-10, 10), ...)
```

Arguments

eset	ExpressionSet
pMassTolWindow	Precursor Mass Error Tolerance Window
...	see plot

Details

No details

Note

No note

References

NA

Examples

```
print("No examples")
```

---

plotQValueVsPValue *Plot qValue vs pValue*

Description

Plot qValue vs pValue

Usage

```
plotQValueVsPValue(sqa, lim = c(0, 1), ...)
```

Arguments

sqa	SafeQuantAnalysis Object
lim	x-axis and y-axis range
...	see plot

Details

No details

Note

No note

References

NA

Examples

```
print("No examples")
```

plotROC *Plot Number of Identifications vs. FDR*

Description

Plot Number of Identifications vs. FDR

Usage

```
plotROC(qvals, qvalueThrs = 0.01, xlab = "False Discovery Rate",
  ylab = "Nb. Valid Identifications", xlim = c(0, 0.1), breaks = 100,
  col = "blue", lwd = 1.5, ...)
```

Arguments

qvals	vector of q-values
qvalueThrs	threshold indicated by vertical line
xlab	default "False Discovery Rate"
ylab	default "Nb. Valid Identifications"
xlim	default c(0,0.1)
breaks	see breaks for hist function
col	default blue
lwd	default 1.5
...	see plot

Details

No details

Note

No note

References

NA

Examples

```
print("No examples")
```

`plotRTNorm` *Plot all retention time profile overalying ratios*

Description

Plot all retention time profile overalying ratios

Usage

```
plotRTNorm(rtNormFactors, eset, samples = 1:ncol(rtNormFactors), main = "",
  ...)
```

Arguments

`rtNormFactors`	data.frame of normalization factor per r.t bin and sample, obtained by getRTNormFactors
`eset`	ExprsssionSet
`samples`	specify samples (sample numbers) to be plotted
`main`	main
`...`	see plot see plot

Details

No details

Note

No note

References

In Silico Instrumental Response Correction Improves Precision of Label-free Proteomics and Accuracy of Proteomics-based Predictive Models, Lyutvinskiy et al. (2013), http://www.ncbi.nlm.nih.gov/pubmed/23589346

See Also

`getRTNormFactors`

Examples

```
print("No examples")
```

`plotRTNormSummary` *Plot all retention time normalization profiles*

Description

Plot all retention time normalization profiles

Usage

```
plotRTNormSummary(eset,
  col = as.character(.getConditionColors(eset)[pData(eset)$condition, 1]),
  ...)
```

Arguments

`eset`	ExpressionSet
`col`	condition colors
`...`	see plot

Details

No details

Note

No note

References

In Silico Instrumental Response Correction Improves Precision of Label-free Proteomics and Accuracy of Proteomics-based Predictive Models, Lyutvinskiy et al. (2013), http://www.ncbi.nlm.nih.gov/pubmed/23589346

See Also

`getRTNormFactors`

Examples

```
print("No examples")
```

plotScoreDistrib — *Plot identifications target decoy distribution*

Description

Plot identifications target decoy distribution

Usage

```
plotScoreDistrib(targetScores, decoyScores, xlab = "Identification Score",
  ylab = "Counts", ...)
```

Arguments

targetScores	target Scores
decoyScores	decoy Scores
xlab	default "Identification Score"
ylab	default "Counts"
...	see plot

Details

No details

Note

No note

References

NA

Examples

```
print("No examples")
```

plotVolcano — *Plots volcano, data points colored by max cv of the 2 compared conditions*

Description

Plots volcano, data points colored by max cv of the 2 compared conditions

Usage

```
plotVolcano(obj, ratioThrs = 1, pValueThreshold = 0.01, adjusted = T, ...)
```

Arguments

obj	safeQuantAnalysis object or data.frame
ratioThrs	default 1
pValueThreshold	default 0.01
adjusted	TRUE/FALSE plot qValues or pValues on y-axis
...	see plot

Details

data.frame input object should contain 3 columns (ratio,qValue,cv)

Note

No note

References

NA

Examples

```
print("No examples")
```

---

plotXYDensity *Scatter plot with density coloring*

Description

Scatter plot with density coloring

Usage

```
plotXYDensity(x, y, isFitLm = T, legendPos = "bottomright",
  disp = c("abline", "R", "Rc"), ...)
```

Arguments

x	number vector
y	number vector
isFitLm	fit linear model
legendPos	see legend
disp	c("abline","R","Rc") display options
...	see plot

Note

No note

References

NA

Examples

```
print("No examples")
```

`purityCorrectTMT` *Correct channel intensities based on Reporter ion Isotopic Distributions*

Description

Correct channel intensities based on Reporter ion Isotopic Distributions

Usage

```
purityCorrectTMT(tmtData, impurityMatrix = impurityMatrix)
```

Arguments

`tmtData`	data.frame containing tmt channel intensities
`impurityMatrix`	correction matrix

Details

Same method as MSnbase, and described in Breitwieser et al. 2012 (Book Chapter)

Value

data.frame of corrected tmt intensities

Note

No note

References

NA

Examples

```
print("No examples")
```

`removeOutliers` *Set value to NA if it deviatves with more than 1.5 * IQR from lower/upper quantile*

Description

Set value to NA if it deviatves with more than 1.5 * IQR from lower/upper quantile

Usage

```
removeOutliers(x, na.rm = TRUE, ...)
```

Arguments

`x`	vector numeric
`na.rm`	logical indicating whether missing values should be removed.
`...`	qunatile args

Details

No details

Note

No note

References

NA

See Also

NA

Examples

```
print("No examples")
```

`rollUp` *Roll up feature intensites per unique colum combination*

Description

Roll up feature intensites per unique colum combination

Usage

```
rollUp(eset, method = "sum", featureDataColumnName = c("proteinName"))
```

Arguments

eset	ExpressionSet
method	"sum", "mean" or "top3"
featureDataColumnName	vector of column names e.g. peptide or proteinName

Details

featureDataColumnName = c("peptide","charge","ptm"), method= c("sum"), sums up intensities per peptie modification charge state

Value

ExpressionSet object

Note

No note

References

No references

Examples

```
print("No examples")
```

---

rtNormalize *Normalization data per retention time bin*

Description

Normalization data per retention time bin

Usage

```
rtNormalize(eset, rtNormFactors)
```

Arguments

eset	ExpressionSet
rtNormFactors	obtained using getRTNormFactors

Details

Normalize for variations in elelctrospray ionization current.

Value

data.frame normalization factors per retention time bin (minute)

Note

No note

References

In Silico Instrumental Response Correction Improves Precision of Label-free Proteomics and Accuracy of Proteomics-based Predictive Models, Lyutvinskiy et al. (2013), http://www.ncbi.nlm.nih.gov/pubmed/23589346

See Also

getRTNormFactors

Examples

```
print("No examples")
```

`safeQuantAnalysis` *safeQunat s3 class*

Description

safeQunat s3 class

Usage

```
safeQuantAnalysis(eset = eset, method = c("global", "naRep", "pairwise"),
  ratioCorrectionModel = NA, fcThrs = 1)
```

Arguments

`eset`	ExpressionSet
`method`	c("global","naRep","rt","quantile","pairwise","all)
`ratioCorrectionModel`	lm (linear model object)
`fcThrs`	fold change threshold

setNbPeptidesPerProtein *Set nbPeptides coulmn of featureData*

Description

Set nbPeptides coulmn of featureData

Usage

```
setNbPeptidesPerProtein(eset)
```

Arguments

eset	ExpressionSet

Details

NA

Value

eset

Note

No note

References

NA

Examples

```
print("No examples")
```

setNbSpectraPerProtein *Set nbPeptides coulmn of featureData*

Description

Set nbPeptides coulmn of featureData

Usage

```
setNbSpectraPerProtein(eset)
```

Arguments

eset	ExpressionSet

Details

NA

Value

eset

Note

No note

References

NA

Examples

```
print("No examples")
```

`sqNormalize` *Normalize*

Description

Normalize

Usage

```
sqNormalize(eset, method = "global")
```

Arguments

`eset`	ExpressionSet
`method`	c("global","rt","quantile")

Details

No details

Value

eset ExpressionSet

Note

No note

References

NA

See Also

getGlobalNormFactors, getRTNormFactors

Examples

```
print("No examples")
```

`standardise` *Standardise data*

Description

Standardise data

Usage

```
standardise(d)
```

Arguments

`d`	vector or data.frame or matrix

Details

No details

Value

vector or data.frame or matrix

Note

No note

Examples

```
print("No examples")
```

`stripACs` *strip uniprot format e.g. "sp|Q8CHJ2|AQP12_MOUSE" -> Q8CHJ2*

Description

strip uniprot format e.g. "sp|Q8CHJ2|AQP12_MOUSE" -> Q8CHJ2

Usage

```
stripACs(acs)
```

Arguments

`acs`	accession numbers

Details

TRUE if less than 10

Value

vector character

Note

No note

References

NA

Examples

```
print("No examples")
```

Index