Package ‘SpectroX’

January 11, 2018

Type Package

Title Proteotypic peptide selection and SRM/PRM/HRM panel creation

Version 2.0

Author Erik Ahrne

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

Description Select proteptypic peptides based on full proteom analysis MaxQuant 1.6 search results or create SRM/PRM/HRM panel/library from MaxQuant 1.6 search results.

License GPL-3

Encoding UTF-8

LazyData true

Suggests testthat

RoxygenNote 6.0.1

R topics documented:

barplotPeptidesPerProtein

plot ranking metric vs peptide (per protein) barplot

Description

plot ranking metric vs peptide (per protein) barplot

Usage

```
barplotPeptidesPerProtein(df, pepLenTrunc = 12, pepLabCex = 0.7,
  rankingMetric = "rankingMetric", ptmCol = getPTMColors(levels(df$ptm)),
  ...)
```

Arguments

`df`	data.frame
`pepLenTrunc`	integer AFADAMEVIPSTLAENAGLNPISTVTELR -> AFADAMEVIPSTLAE..
`pepLabCex`	default 0.7
`rankingMetric`	character
`ptmCol`	data.frame row.names: ptm col:col

Note

No note

References

NA

Examples

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

barplotPetideCountPerProtein

plot peptide count per protein

Description

plot peptide count per protein

Usage

```
barplotPetideCountPerProtein(spectralLibrary, protLabCex = 0.9,
  acLenTrunc = 12, col = "blue", ...)
```

Arguments

`spectralLibrary`	data.frame
`protLabCex`	default 0.9
`acLenTrunc`	integer default 12 "SOMEVERYLONGAC" -> "SOMEVERY.."

Note

No note

References

NA

Examples

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

`createComplementaryIsotopeLibrary`

Create complementary isotope (Arg, Lys H/L) spectral library

Description

Create complementary isotope (Arg, Lys H/L) spectral library

Usage

```
createComplementaryIsotopeLibrary(sl)
```

Arguments

`sl`	tibble or data frame

Details

No details

Value

data.frame

Note

No note

References

NA

Examples

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

`createLibrarySpectrum` *Parse spectrum framgment match information listed in MaxQuant 'Identifications' results*

Description

Parse spectrum framgment match information listed in MaxQuant 'Identifications' results

Usage

```
createLibrarySpectrum(psm)
```

Arguments

`psm`	a row of MaxQuant 'Identifications' data.frame

Details

No details

Value

list with attributes intensity, ionType, charge, mz, isNL

Note

No note

References

NA

Examples

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

`createSpectralLibrary` *Create Spectral Library*

Description

Create Spectral Library

Usage

```
createSpectralLibrary(tb, minFragNb = 3, minNbTransitions = 5,
  maxNbTransitions = 5, minBasePeakFraction = 0, ionTypeFilter = c("a",
  "b", "x", "y"), includeNL = T, rankingMetric = "adjustedIntensitySum")
```

Arguments

`tb` tibble maxQuant spectrum level search results

`minFragNb` min frag number default 3 (i.e. b3 and y3 will be kept)

`minNbTransitions`
minimum number of fragments

`minBasePeakFraction`
minimum intnsity fraction of base peak (most intense kept fragment)

`ionTypeFilter` selected ion type default a,b,x,y

`includeNL` include neutral loss peaks defualt TRUE

`rankingMetric` character. column name, used for ranking peptide, default "adjustedIntensitySum", c("adjustedIntensitySum","psmScore","precApexIntensity")

Details

No details

Value

data.frame

Note

No note

References

NA

Examples

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

`digestProteome` *Get peptide candidates per protein*

Description

Get peptide candidates per protein

Usage

```
digestProteome(proteins, proteaseRegExp = "[KR](?!P)", dispProgressBar = T,
  peptideLengthRange = c(6, 21), nbMiscleavages = 0,
  exclusivePeptides = T, trimAC = T)
```

Arguments

`proteins`	list
`proteaseRegExp`	character '[KR](?!P)' - trypsin
`dispProgressBar`	T/F
`peptideLengthRange`	integer vector default c(6,21)
`nbMiscleavages`	0
`exclusivePeptides`	TRUE/FALSE get exclusive peptides only i.e. peptides mapping to a single protein. default T
`trimAC`	TRUE/FALSE default TRUE sp|P62258|1433E_HUMAN -> P62258

Value

data.frame peptide,protein

Note

No note

References

NA

Examples

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

`getCMDLineOptions` *Define and User Specified Command Line Options*

Description

Define and User Specified Command Line Options

Usage

```
getCMDLineOptions(version = version)
```

Arguments

`version`	SpectroX version number

Details

No details

Value

list() cmd line options

Note

No note

References

NA

Examples

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

`getEmpiricalIRT` *Add iRT metric to data.frame*

Description

Add iRT metric to data.frame

Usage

```
getEmpiricalIRT(tb, fit)
```

Arguments

`tb`	tibble containing colum labelled "Retention time"
`fit`	lm object

Details

No details

Value

vector of normalised rt (empirical irt)

Note

No note

References

Using iRT, a normalized retention time for more targeted measurement of peptides, Escher et al. (2012), `http://www.ncbi.nlm.nih.gov/pubmed/22577012`

Examples

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

getFragmentSequence *Get Fragment Sub Sequnece*

Description

Get Fragment Sub Sequnece

Usage

```
getFragmentSequence(peptide = peptide, ionType = ionType,
  fragmentNb = fragmentNb)
```

Arguments

peptide	character string
ionType	c("a","b","x","y")
fragmentNb	integer

Details

No details

Value

character string

Note

No note

References

NA

Examples

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

getIRTModel *Get linear model predicting iRT as a function retention time (column name "rt")*

Description

Get linear model predicting iRT as a function retention time (column name "rt")

Usage

```
getIRTModel(tb)
```

Arguments

`tb`	tibble including column "Retention time" and "Sequence"

Details

Uses Robust Linear Regression. Retention times of at least 3 iRT unique peptides needs to be listed in the input tibble

Value

list including fit rlm object, data.frame peptide,rt,irtRef

Note

No note

References

Using iRT, a normalized retention time for more targeted measurement of peptides, Escher et al. (2012), http://www.ncbi.nlm.nih.gov/pubmed/22577012

Examples

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

`getPeptides` *Get Mz Shift of complementary peptide/fragment ion*

Description

Get Mz Shift of complementary peptide/fragment ion

Usage

```
getPeptides(proteinSeq, proteaseRegExp = "[KR](?!P)", nbMiscleavages = 0)
```

Arguments

`proteinSeq`	protein sequence
`proteaseRegExp`	protease Regular Expression
`nbMiscleavages`	default 0
`aaSeq`	character string
`charge`	default 1
`isHeavy`	c(T,F)
`annotSpectrum`	Annotated Spectrum object

Details

Calculate mass shift of complimentary spectrum.

- PETIDEK (light) -> 8.014199
- PETIDEK (heavy) -> -8.014199
- PETIDER (light) -> 10.008269
- PETIDER (heavy) -> -10.008269

No details

No details

Value

numeric

annotSpectrum Annotated Spectrum object

vector of peptides

Note

No note

No note

No note

References

NA

NA

Examples

```
print("No examples")
Created complimentary Heavy/Light annotated spectrum with updates precursor and fragment m/z values.
print("No examples")
Digest protein
print("No examples")
```

`getPTMColors` *assign colors to ptms*

Description

assign colors to ptms

Usage

```
getPTMColors(ptms)
```

Arguments

`ptms`	vector

Value

data.frame

Note

No note

References

NA

Examples

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

`getSearchedModifications` *Get list of variable modifiections condsidered by MaxQuant*

Description

Get list of variable modifiections condsidered by MaxQuant

Usage

```
getSearchedModifications(tb, ignoreArgLysIsoLabel = T)
```

Arguments

`ignoreArgLysIsoLabel`	default T Do not consider Arg Lys C,N heavy isotope label
`df`	tible or data.frame

Details

No details

Value

character vector of modification names (does not return "Arg10","Lys8" labels)

Note

No note

References

NA

Examples

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

`getUserOptions` *check and return User Specified Command Line Options*

Description

check and return User Specified Command Line Options

Usage

```
getUserOptions(cmdlineOptions)
```

Arguments

`list()`	command line options

Details

No details

Value

list() user options

Note

No note

References

NA

Examples

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

`parseMaxQuantMSMS` *Parse MaxQuant msms.txt*

Description

Parse MaxQuant msms.txt

Usage

```
parseMaxQuantMSMS(file, pepCutoff = 0.05, targetPeptides = NA,
  targetProteins = NA, filterContaminants = T,
  contaminantRegExp = "^CON_", selectedPTMRegExp = NA,
  filterNonExclusivePeptides = T, pepLength = c(1, Inf),
  chargeState = c(1, Inf), label = NA, maxMissedCleavages = 0,
  keepBestSpectrumOnly = T, requiredPepSeqRegExp = ".", ...)
```

Arguments

`file` path

`pepCutoff` numeric default 0.05

`targetPeptides` character default NA

`targetProteins` character default NA

`filterContaminants`
TRUE

`contaminantRegExp`
'^CON_'

`selectedPTMRegExp`
NA

`filterNonExclusivePeptides`
default TRUE

`pepLength` default [1,Inf)

`chargeState` default [1,Inf)

`label` (Arg10 and Lys8 only filter) options NA (deafault,no filter), 'light','heavy'

`keepBestSpectrumOnly`
keep top-scoring spectrum only, per peptidoform, default TRUE

`requiredSequenceRegExp`
dafualt '.' (no filter)

Details

No details

Value

data.frame of maxQuant psm level results

Note

No note

References

NA

Examples

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

`parseRange` *Parse range in format 2:4 and return integer vector. Inthis case 2,3,4*

Description

Parse range in format 2:4 and return integer vector. Inthis case 2,3,4

Usage

```
parseRange(rangeStr)
```

Arguments

`rangeStr`	e.g. '2:4'

Details

No details

Value

vectot of integers

Note

No note

References

NA

Examples

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

`parseTargetsFile` *Parse list of target peptides/proteins*

Description

Parse list of target peptides/proteins

Usage

```
parseTargetsFile(file)
```

Arguments

`file`	path

Details

No details

Value

list() peptides, proteins

Note

No note

References

NA

Examples

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

`plotIRTCalibration` *Plot IRT calibration curve*

Description

Plot IRT calibration curve

Usage

```
plotIRTCalibration(irtModel, cex = 1.5, col = "blue", pch = 19, ...)
```

Arguments

`irtModel`	list

Details

No details

Note

No note

References

NA

Examples

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

proteotypicPeptideExport

# Export top X peptide per protein (ordered by 'adjustedIntensitySum') and add theoretical peptides ranked by length if fewer than X peptides were identified by MaxQuant

Description

# Export top X peptide per protein (ordered by 'adjustedIntensitySum') and add theoretical peptides ranked by length if fewer than X peptides were identified by MaxQuant

Usage

```
proteotypicPeptideExport(spectralLibrary, theoPeptides = NA,
  targetProteins = unique(spectralLibrary$protein),
  nbPeptidesPerProtein = 5, outFile = paste0(tempdir(), "/tmp.xls"))
```

Arguments

`spectralLibrary`
tibble

`theoPeptides` data.frame 'peptide' 'protein' 'length'

`targetProteins` list of protein accesion numbers e.g. P62258. default all proteins in spectralLibrary

`nbPeptidesPerProtein`
number of peptides per protein to be exported

`outFile` path to output file defult tmp.xls in tmp dir

Note

No note

References

NA

Examples

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

`skylineExport` *Write skyline compatible xls file*

Description

Write skyline compatible xls file

Usage

```
skylineExport(sl, file)
```

Arguments

`sl`	tibble or data frame

Details

No details

Note

No note

References

NA

Examples

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

`spectroDiveExport` *Write SpectroDive compatible xls file*

Description

Write SpectroDive compatible xls file

Usage

```
spectroDiveExport(sl, file)
```

Arguments

`sl`	tibble or data frame

Details

No details

Note

No note

References

NA

Examples

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

`spectronautExport` *Write Spectronaut compatible xls file*

Description

Write Spectronaut compatible xls file

Usage

```
spectronautExport(sl, file)
```

Arguments

`sl`	tibble or data frame

Details

No details

Note

No note

References

NA

Examples

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

Index