Package 'mstherm'

April 27, 2017

Type Package

Title Analyze MS/MS Protein Melting Data

Version 0.4.7

Date 2017-04-21

Description Software to aid in modeling and analyzing mass-spectrometry-based proteome melting data. Quantitative data is imported and normalized and thermal behavior is modeled at the protein level. Methods exist for normalization, modeling, visualization, and export of results. For a general introduction to MS-based thermal profiling, see Savitski et al. (2014) <doi:10.1126/science.1255784>.

License GPL-3

Imports foreach, parallel, doParallel, nls2, RColorBrewer, plotrix

Suggests RSQLite, testthat, knitr, rmarkdown

Collate mstherm.R classes.R normalization.R modeling.R plot.R analysis.R export.R

RoxygenNote 6.0.1

VignetteBuilder knitr

NeedsCompilation no

Author Jeremy Volkening [aut, cre]

Maintainer Jeremy Volkening <jdv@base2bio.com>

Repository CRAN

Date/Publication 2017-04-27 17:22:16 UTC

R topics documented:

`as.data.frame.MSThermResultSet` *MSResultSet to data frame.*

Description

Populates a data frame with information from an MSResultSet, one row per protein/group

Usage

```
## S3 method for class 'MSThermResultSet'
as.data.frame(x, ...)
```

Arguments

x	an MSResultSet object
...	additional arguments passed to or from other functions

Value

A data frame populated with relevant information per result

Examples

```
control <- system.file("extdata", "demo_project/control.tsv", package="mstherm")
annots  <- system.file("extdata", "demo_project/annots.tsv",  package="mstherm")
expt    <- MSThermExperiment(control, annotations=annots)
expt    <- normalize_to_std(expt, "cRAP_ALBU_BOVIN", plot=FALSE)
res     <- model_experiment(expt, bootstrap=FALSE, np=2)

df <- as.data.frame(res)
```

model_experiment *Model MSThermExperiment.*

Description

Model multiple proteins from an MSThermExperiment object.

Usage

```
model_experiment(expt, proteins, np, ...)
```

Arguments

expt	An MSThermExperiment object
proteins	A vector of protein IDs to model (default is all proteins).
np	Number of parallel jobs to start (default = number of available processors)
...	Parameters passed to model_protein()

Value

MSThermResultSet object

Examples

```
control <- system.file("extdata", "demo_project/control.tsv", package="mstherm")
annots  <- system.file("extdata", "demo_project/annots.tsv",  package="mstherm")
expt    <- MSThermExperiment(control, annotations=annots)
expt    <- normalize_to_std(expt, "cRAP_ALBU_BOVIN", plot=FALSE)

res     <- model_experiment(expt, bootstrap=FALSE, np=2)
summary(res)
```

model_protein *Model single protein.*

Description

Model a single protein from an MSThermExperiment object.

Usage

```
model_protein(expt, protein, min_rep_psm = 0, min_smp_psm = 0,
  min_tot_psm = 0, max_inf = 1, min_score, max_score, smooth = 0,
  method = "sum", method.denom = "near", trim = 0, bootstrap = 0,
  min_bs_psms = 8, annot_sep = "|", max_slope = 0, min_r2 = 0,
  min_reps = 0, only_modeled = 0, check_missing = 0,
  missing_cutoff = 0.3)
```

Arguments

expt	An MSThermExperiment object
protein	ID of the protein to model
min_rep_psm	Minimum number of spectral matches required for each replicate to model protein
min_smp_psm	Minimum number of spectral matches required for each sample to model protein
min_tot_psm	Minimum number of spectral matches required across all replicates to model protein
max_inf	Maximum co-isolation interference level allowed to include a spectrum in protein-level quantification
min_score	minimum score allowed to include a spectrum in protein-level quantification
max_score	maximum score allowed to include a spectrum in protein-level quantification
smooth	(t/F) Perform loess smoothing on the data prior to modeling
method	Protein quantification method to use (see Details)
method.denom	Method used to calculate denominator of abundance (see Details)
trim	(t/F) Trim all lower data points less than the abundance maximum
bootstrap	(T/F) Perform bootstrap analysis to determine confidence intervals (slow)
min_bs_psms	Minimum number of spectral matches required to perform bootstrapping
annot_sep	Symbol used to separate protein group IDs (used for retrieval of annotations) (default: 'l')
max_slope	Maximum slope to consider model (implies "only_modeled")
min_r2	Minimum R2 value to consider model (implies "only_modeled")
min_reps	Minimum number of modeled replicates for each sample to return protein
only_modeled	(t/F) Only consider modeled proteins
check_missing	(t/F) Run simple test to filter out PSMs with missing quantification channels where values are expected
missing_cutoff	Minimum fraction relative to surrounding data points used in the check for missing channels

Details

Valid quantification methods include:

"sum" use the sum of the spectrum values for each channel

"median" use the median of the spectrum values for each channel

"ratio.median" Like "median", but values for each spectrum are first converted to ratios according to "method.denom" channel

"ratio.mean" Like "ratio.median" but using mean of ratios

Valid denominator methods include:

"first" Use the first value (lowest temperature point) (default)

"max" Use the maximum value

"top3" Use the mean of the three highest values

"near" Use the median of all values greater than 80 the first value

Value

MSThermResult object

Examples

```
control <- system.file("extdata", "demo_project/control.tsv", package="mstherm")
annots  <- system.file("extdata", "demo_project/annots.tsv",  package="mstherm")
expt    <- MSThermExperiment(control, annotations=annots)
expt    <- normalize_to_std(expt, "cRAP_ALBU_BOVIN", plot=FALSE)

model   <- model_protein(expt, "P38707", smooth=TRUE, bootstrap=FALSE)
summary(model)
```

---

mstherm *Model and analyze MS/MS-based protein melting data.*

Description

mstherm is a package for modeling and analysis of MS/MS-based thermal proteome profiling (TPP) experiments.

Author(s)

Jeremy Volkening <jdv@base2bio.com>

MSThermExperiment *Create a new MSThermExperiment.*

Description

MSThermExperiment creates a new experiment object from a set of filenames or data frames.

Usage

```
MSThermExperiment(control, annotations)
```

Arguments

control	data frame or filename of tab-delimited table describing the experimental setup and locations of data on disk (see Details)
annotations	data frame or filename to tab-delimited table containing protein names and annotations (usually functional descriptions but can be any text

Details

Both parameters can take either a data frame or a tab-delimited filename on disk (which will be read into a data frame). "control" should contain columns with the following headers (in any order):

"name" Unique identifier of a single replicate

"sample" Sample name that a replicate belongs to

"data_file" Path to file on disk containing the quantification data

"meta_file" Path to file on disk containing the labeling metadata

The "meta_file" should be tab-delimited text and contain two columns labeled "channel" and "temp". The "data_file" should be tab-delimited text and contain, at a minimum, the following columns:

"peptide" Sequence of the matched peptide in single-letter IUPAC

"protein" Protein or protein group to which the peptide belongs

"..." One column per isobaric channel, containing absolute quantification values. Column names must match those in the "channel" column of the meta file, with the exception that R will automatically convert any name not compatible with its syntax rules. To be safe, use only letters, digits, underscores, and periods in channel names and never start with a digit (e.g. use "TMT.126" rather than "126")

The following columns can also be utilized for filtering if included (all others will simply be ignored):

"coelute_inf" Calculated precursor co-isolation interference (0.0-1.0)

"score" Score assigned by the processing software to the PSM

"annotations" should contain two columns with the headers "name" and "annotation". "name" should match the protein names in the data files, and "annotation" can contain any text (generally a functional description)

Value

An MSThermExperiment object

Examples

```
control <- system.file("extdata", "demo_project/control.tsv", package="mstherm")
annots  <- system.file("extdata", "demo_project/annots.tsv",  package="mstherm")
expt    <- MSThermExperiment(control, annotations=annots)
```

normalize_to_profile *Normalize to a profile.*

Description

Normalizes an MSThermReplicate based on a pre-determined vector of relative abundances

Usage

```
normalize_to_profile(replicate, profile, model = T, plot = T)
```

Arguments

replicate	an MSThermReplicate object
profile	a vector of relative values
model	whether to fit scale factors to model
plot	(T/f) whether to display a summary plot

Value

An MsThermReplicate object with normalized data slots

Examples

```
control <- system.file("extdata", "demo_project/control.tsv", package="mstherm")
annots  <- system.file("extdata", "demo_project/annots.tsv",  package="mstherm")
expt    <- MSThermExperiment(control, annotations=annots)

profile <- c(50.0, 50.5, 47.5, 42.0, 37.0, 25.0, 16.0, 11.5, 10.5, 10.0)
expt$samples$Control$replicates$C1 <- normalize_to_profile(
   expt$samples$Control$replicates$C1, profile, plot=FALSE
)
```

normalize_to_std *Normalize to a spike-in standard.*

Description

Normalizes each replicate of an experiment based on a given spike-in protein standard (assumed to be present in equimolar amounts in each channel).

Usage

```
normalize_to_std(expt, protein, model = T, plot = T)
```

Arguments

expt	an MSThermExperiment object
protein	ID of a protein to normalize against
model	whether to fit scale factors to model
plot	(T/f) whether to display a summary plot

Value

An MsThermExperiment object with normalized data slots

Examples

```
control <- system.file("extdata", "demo_project/control.tsv", package="mstherm")
annots  <- system.file("extdata", "demo_project/annots.tsv",  package="mstherm")
expt    <- MSThermExperiment(control, annotations=annots)

expt    <- normalize_to_std(expt, "cRAP_ALBU_BOVIN", plot=FALSE)
```

---

normalize_to_tm *Re-normalize based on Tm.*

Description

Normalizes each replicate of an experiment based on linear regression of calculated Tm (corrects for remaining systematic error).

Usage

```
normalize_to_tm(expt, res)
```

Arguments

expt	An MSThermExperiment object
res	An MSThermResultSet object

Details

An assumption can be made in most TPP experiments using a single organism that the Tm of most proteins should not be changing. However, global shifts have been observed between replicates, presumably due to technical variance, which complicate data interpretation. This method attempts to remove this source of error by doing a bootstrap renormalization of the quantification values based on pairwise linear regression between calculated Tms of each replicate. A reference set of Tms is calculated based on all replicates, and each replicate is normalized to this based on the calculated slope and intercept of the input data.

Value

An MsThermExperiment object with re-normalized data slots

Examples

```
control <- system.file("extdata", "demo_project/control.tsv", package="mstherm")
annots  <- system.file("extdata", "demo_project/annots.tsv",  package="mstherm")
expt    <- MSThermExperiment(control, annotations=annots)
expt    <- normalize_to_std(expt, "cRAP_ALBU_BOVIN", plot=FALSE)
res     <- model_experiment(expt, smooth=TRUE, bootstrap=FALSE, np=2)

expt    <- normalize_to_tm(expt, res)
```

---

plot.MSThermResult *Plot MSThermResult object.*

Description

Generate a denaturation plot for an modeled protein/group.

Usage

```
## S3 method for class 'MSThermResult'
plot(x, table = T, col, CI.points = T, CI.Tm = T,
  ...)
```

Arguments

x	An MSThermResult object
table	(T/f) include table of per-replicate parameters
col	array of colors used to plot samples
CI.points	(T/F) plot temperature point confidence intervals
CI.Tm	(T/F) plot Tm confidence intervals
...	other parameters passed through to plot()

Value

Nothing

Examples

```
control <- system.file("extdata", "demo_project/control.tsv", package="mstherm")
annots  <- system.file("extdata", "demo_project/annots.tsv",  package="mstherm")
expt    <- MSThermExperiment(control, annotations=annots)
expt    <- normalize_to_std(expt, "cRAP_ALBU_BOVIN", plot=FALSE)
res     <- model_experiment(expt, bootstrap=FALSE, np=2)

# plot single MSThermResult
plot(res$P38707)

# plot all proteins (e.g. to pdf device, one-per-page)
plot(res)
```

---

plot.MSThermResultSet *Plot MSThermResultSet object.*

Description

Generate a series of denaturation plots for all results in an MSThermResultSet.

Usage

```
## S3 method for class 'MSThermResultSet'
plot(x, ...)
```

Arguments

x	an MSThermResultSet object
...	other parameters are passed through to plot.MSThermResult

Details

Since this function makes multiple sequential calls to plot.MSThermResult, it is usually used in conjunction with a multipage graphics device such as "pdf()". Otherwise each subsequent call will only overwrite the previous output.

Value

Nothing

Examples

```
# see plot.MSThermResult for an example
```

summary.MSThermResult *Summarize MSThermResult object.*

Description

Print a summary of an MSThermResult, including samples and parameters.

Usage

```
## S3 method for class 'MSThermResult'
summary(object, ...)
```

Arguments

object	an MSThermResult object
...	additional arguments passed to or from other functions

Value

Nothing

Examples

```
# see model_protein() for an example
```

summary.MSThermResultSet
Summarize MSThermResultSet object.

Description

Print a summary of an MSThermResultSet, including samples and parameters.

Usage

```
## S3 method for class 'MSThermResultSet'
summary(object, ...)
```

Arguments

object	an MSThermResultSet object
...	additional arguments passed to or from other functions

Value

Nothing

Examples

```
# see model_experiment() for an example
```

`write.sqlite` *Export MSThermResultSet to an SQLite database.*

Description

Exports and MSThermResultSet object to a new SQLite database file. Each model (specific to a given replicate and protein) is exported as an individual record. The schema used for the 'data' table can be seen in the code below.

Usage

```
write.sqlite(res, file)
```

Arguments

`res`	An MSThermResultSet object
`file`	Path to the output sqlite database to be created

Value

Nothing

Examples

```
control <- system.file("extdata", "demo_project/control.tsv", package="mstherm")
annots  <- system.file("extdata", "demo_project/annots.tsv",  package="mstherm")
expt    <- MSThermExperiment(control, annotations=annots)
expt    <- normalize_to_std(expt, "cRAP_ALBU_BOVIN", plot=FALSE)
res     <- model_experiment(expt, bootstrap=FALSE, np=2)

fn <- tempfile(fileext = ".sqlite")
write.sqlite(res, fn)
unlink(fn) # for example only
```

Index