

Package: MSstatsConvert (via r-universe)

September 30, 2026

Title Import Data from Various Mass Spectrometry Signal Processing
Tools to MSstats Format

Version 1.23.5

Description MSstatsConvert provides tools for importing reports of
Mass Spectrometry data processing tools into R format suitable
for statistical analysis using the MSstats and MSstatsTMT
packages.

License Artistic-2.0

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

biocViews MassSpectrometry, Proteomics, Software, DataImport,
QualityControl

Depends R (>= 4.0)

Imports data.table, methods, checkmate, utils, stringi, Rcpp, parallel

Suggests tinytest, covr, knitr, arrow, rmarkdown

LinkingTo Rcpp

Collate 'clean_MZMine.R' 'clean_ProteinProspector.R'
'clean_Metamorpheus.R' 'clean_DIANN.R' 'clean_Philosopher.R'
'clean_Spectronaut.R' 'clean_SpectroMine.R' 'clean_Skyline.R'
'clean_ProteomeDiscoverer.R' 'clean_Progenesis.R'
'clean_OpenSWATH.R' 'clean_OpenMS.R' 'clean_MaxQuant.R'
'clean_DIAUmpire.R' 'MSstatsConvert_core_functions.R'
'RcppExports.R' 'converters_DIANNtoMSstatsFormat.R'
'converters_DIAUmpiretoMSstatsFormat.R'
'converters_FragPipeToMSstatsFormat.R'
'converters_MZMinetoMSstatsFormat.R'
'converters_MaxQtoMSstatsFormat.R'
'converters_MaxQtoMSstatsTMTFormat.R'
'converters_MetamorpheusToMSstatsFormat.R'
'converters_OpenMStoMSstatsFormat.R'
'converters_OpenMStoMSstatsTMTFormat.R'

```
'converters_OpenSWATHtoMSstatsFormat.R'
'converters_PDtoMSstatsFormat.R'
'converters_PDtoMSstatsTMTFormat.R'
'converters_PhilosophertoMSstatsTMTFormat.R'
'converters_ProgenesistoMSstatsFormat.R'
'converters_ProteinProspectortoMSstatsTMTFormat.R'
'converters_SkylinetoMSstatsFormat.R'
'converters_SpectroMinetoMSstatsTMTFormat.R'
'converters_SpectronauttoMSstatsFormat.R'
'utils_MSstatsConvert.R' 'utils_annotation.R'
'utils_anomaly_score.R' 'utils_balanced_design.R'
'utils_checks.R' 'utils_classes.R' 'utils_clean_features.R'
'utils_data_health.R' 'utils_documentation.R'
'utils_dt_operations.R' 'utils_filtering.R' 'utils_fractions.R'
'utils_logging.R' 'utils_shared_peptides.R'
```

VignetteBuilder knitr

Config/roxygen2/version 8.0.0

Config/pak/sysreqs libicu-dev

Repository <https://bioc.r-universe.dev>

Date/Publication 2026-08-31 13:45:11 UTC

RemoteUrl <https://github.com/bioc/MSstatsConvert>

RemoteRef HEAD

RemoteSha 4d93a961ba39beed3eae9c3a8900ba504543fc7

Contents

.cleanRawPD	3
as.data.frame.MSstatsValidated	4
as.data.table.MSstatsValidated	4
CheckDataHealth	5
DIANNtoMSstatsFormat	5
DIAUmpiretoMSstatsFormat	9
FragPipetoMSstatsFormat	11
getInputFile	12
MaxQtoMSstatsFormat	13
MaxQtoMSstatsTMTFormat	15
MetamorpheusToMSstatsFormat	17
MSstatsAnomalyScores	19
MSstatsBalancedDesign	20
MSstatsClean	21
MSstatsImport	26
MSstatsLogsSettings	27
MSstatsMakeAnnotation	28
MSstatsPreprocess	29
MSstatsSaveSessionInfo	31

MZMinetoMSstatsFormat	32
OpenMStoMSstatsFormat	34
OpenMStoMSstatsTMTFormat	36
OpenSWATHtoMSstatsFormat	37
PDtoMSstatsFormat	39
PDtoMSstatsTMTFormat	41
PhilosophertoMSstatsTMTFormat	42
ProgenesistoMSstatsFormat	44
ProteinProspectortoMSstatsTMTFormat	46
SkylinetoMSstatsFormat	47
SpectroMinetoMSstatsTMTFormat	49
SpectronauttoMSstatsFormat	51

Index	55
--------------	-----------

<code>.cleanRawPD</code>	<i>Clean raw Proteome Discoverer data</i>
--------------------------	---

Description

Clean raw Proteome Discoverer data

Usage

```
.cleanRawPD(
  msstats_object,
  quantification_column,
  protein_id_column,
  sequence_column,
  remove_shared,
  remove_protein_groups = TRUE,
  intensity_columns_regexp = "Abundance"
)
```

Arguments

`msstats_object` an object of class `MSstatsSpectroMineFiles`.

`quantification_column`
chr, name of a column used for quantification.

`protein_id_column`
chr, name of a column with protein IDs.

`sequence_column`
chr, name of a column with peptide sequences.

`remove_shared` lgl, if TRUE, shared peptides will be removed.

`remove_protein_groups`
if TRUE, proteins with `numProteins > 1` will be removed.

intensity_columns_regexp

regular expressions that defines intensity columns. Defaults to "Abundance", which means that columns that contain the word "Abundance" will be treated as corresponding to intensities for different channels.

Value

data.table

as.data.frame.MSstatsValidated

Convert output of converters to data.frame

Description

Convert output of converters to data.frame

Usage

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

Arguments

x object of class MSstatsValidated
... Additional arguments to be passed to or from other methods.

Value

data.frame

as.data.table.MSstatsValidated

Convert output of converters to data.table

Description

Convert output of converters to data.table

Usage

```
## S3 method for class 'MSstatsValidated'  
as.data.table(x, ...)
```

Arguments

- x object of class MSstatsValidated
- ... Additional arguments to be passed to or from other methods.

Value

data.tables

CheckDataHealth	<i>Takes as input the output of the SpectronautoMSstatsFormat function and calculates various quality metrics to assess the health of the data. Requires Anomaly Detection model to be fit.</i>
-----------------	---

Description

Takes as input the output of the SpectronautoMSstatsFormat function and calculates various quality metrics to assess the health of the data. Requires Anomaly Detection model to be fit.

Usage

CheckDataHealth(input)

Arguments

- input MSstats input which is the output of Spectronaut converter

Value

list of two data.tables

DIANNtoMSstatsFormat	<i>Import Diann files</i>
----------------------	---------------------------

Description

Import Diann files

Usage

```

DIANNtoMSstatsFormat(
  input,
  annotation = NULL,
  global_qvalue_cutoff = 0.01,
  qvalue_cutoff = 0.01,
  pg_qvalue_cutoff = 0.01,
  useUniquePeptide = TRUE,
  removeFewMeasurements = TRUE,
  removeOxidationMpeptides = TRUE,
  removeProtein_with1Feature = TRUE,
  MBR = TRUE,
  labeledAminoAcids = NULL,
  quantificationColumn = "FragmentQuantCorrected",
  calculateAnomalyScores = FALSE,
  anomalyModelFeatures = c(),
  anomalyModelFeatureTemporal = c(),
  removeMissingFeatures = 0.5,
  anomalyModelFeatureCount = 100,
  runOrder = NULL,
  n_trees = 100,
  max_depth = "auto",
  numberOfCores = 1,
  use_log_file = TRUE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL,
  ...
)

```

Arguments

<code>input</code>	name of MSstats input report from Diann, which includes fragment-level data. Output fragment data with <code>-export-quant</code> flag in DIA-NN 2.0
<code>annotation</code>	name of 'annotation.txt' data which includes Condition, BioReplicate, Run.
<code>global_qvalue_cutoff</code>	The qvalue cutoff for the Q.Value column, i.e. the run-specific precursor q-value. Default is 0.01.
<code>qvalue_cutoff</code>	If MBR is false, the qvalue cutoff for the Global.Q.Value column, i.e. global precursor q-value. If MBR is true, the qvalue cutoff for the Lib.Q.Value column, i.e. the q-value for the library created after the first MBR pass. Default is 0.01.
<code>pg_qvalue_cutoff</code>	If MBR is false, the qvalue cutoff for the Global.PG.Q.Value column, i.e. the global q-value for the protein group. If MBR is true, the qvalue cutoff for the Lib.PG.Q.Value column, i.e. the protein group q-value for the library created after the first MBR pass. Default is 0.01.

useUniquePeptide	should unique peptides be removed
removeFewMeasurements	should proteins with few measurements be removed
removeOxidationMpeptides	should peptides with oxidation be removed
removeProtein_with1Feature	should proteins with a single feature be removed
MBR	True if analysis was done with match between runs
labeledAminoAcids	Character vector of single-letter codes for the amino acids carrying the SILAC label, e.g. c("K") or c("K", "R"). Supplying it turns on protein-turnover mode. Defaults to NULL, which marks every peptide IsotopeLabelType = "Light". How the label is read depends on the export: DIA-NN 2.x, with a Channel column. Channel gives the label directly: "H" and "L" pass through, anything else becomes NA. The amino acid codes only switch the mode on here; sequences are not inspected. DIA-NN 1.x, without a Channel column. The label is read from ModifiedSequence, where it follows the labeled residue in parentheses as (<label>-<aminoAcid>-H) or (<label>-<aminoAcid>-L); for example K(label-K-H) is a heavy-labeled lysine. <label> is the label name DIA-NN reported (e.g. SILAC). Untagged sequences become NA, and the tags are stripped from PeptideSequence afterwards. A peptide with two or more labelable residues (counted across all of labeledAminoAcids) may be only partially labeled, which the two-state turnover model cannot represent, so it is dropped and the number removed is reported.
quantificationColumn	Use 'FragmentQuantCorrected'(default) column for quantified intensities for DIANN 1.8.x. Use 'FragmentQuantRaw' for quantified intensities for DIANN 1.9.x. Use 'auto' for quantified intensities for DIANN 2.x where each fragment intensity is a separate column, e.g. Fr0Quantity.
calculateAnomalyScores	Default is FALSE. If TRUE, will run anomaly detection model and calculate anomaly scores for each feature. Used downstream to weigh measurements in differential analysis.
anomalyModelFeatures	character vector of quality metric column names to be used as features in the anomaly detection model. List must not be empty if calculateAnomalyScores=TRUE.
anomalyModelFeatureTemporal	character vector of temporal direction corresponding to columns passed to anomalyModelFeatures. Values must be one of: mean_decrease, mean_increase, dispersion_increase, or NULL (to perform no temporal feature engineering). Default is empty vector. If calculateAnomalyScores=TRUE, vector must have as many values as anomalyModelFeatures (even if all NULL).
removeMissingFeatures	Remove features with missing values in more than this fraction of runs. Default is 0.5. Only used if calculateAnomalyScores=TRUE.


```
# For DIANN 2.0, set quantificationColumn = 'auto'
input_file_path_2_0 = system.file("tinytest/raw_data/DIANN/diann_2.0.parquet",
                                  package="MSstatsConvert")
annotation_file_path_2_0 = system.file("tinytest/raw_data/DIANN/annotation_diann_2.0.csv",
                                       package = "MSstatsConvert")
input_2_0 = arrow::read_parquet(input_file_path_2_0)
annot_2_0 = data.table::fread(annotation_file_path_2_0)
output_2_0 = DIANNtoMSstatsFormat(input_2_0, annotation = annot_2_0, MBR = FALSE,
                                  use_log_file = FALSE, quantificationColumn = 'auto')

head(output_2_0)
```

DIAUmpiretoMSstatsFormat

Import DIA-Umpire files

Description

Import DIA-Umpire files

Usage

```
DIAUmpiretoMSstatsFormat(
  raw.frag,
  raw.pep,
  raw.pro,
  annotation,
  useSelectedFrag = TRUE,
  useSelectedPep = TRUE,
  removeFewMeasurements = TRUE,
  removeProtein_with1Feature = FALSE,
  summaryforMultipleRows = max,
  use_log_file = TRUE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL,
  ...
)
```

Arguments

raw.frag	name of FragSummary_date.xls data, which includes feature-level data.
raw.pep	name of PeptideSummary_date.xls data, which includes selected fragments information.
raw.pro	name of ProteinSummary_date.xls data, which includes selected peptides information.

annotation	name of annotation data which includes Condition, BioReplicate, Run information.
useSelectedFrag	TRUE will use the selected fragment for each peptide. 'Selected_fragments' column is required.
useSelectedPep	TRUE will use the selected peptide for each protein. 'Selected_peptides' column is required.
removeFewMeasurements	TRUE (default) will remove the features that have 1 or 2 measurements across runs.
removeProtein_with1Feature	TRUE will remove the proteins which have only 1 feature, which is the combination of peptide, precursor charge, fragment and charge. FALSE is default.
summaryforMultipleRows	max or sum - when multiple PSMs identify the same feature within a single MS run (duplicate PSMs), use the highest (max) or sum of the duplicate intensities. Default is max for label-free converters and sum for TMT converters. Note that this parameter does NOT control collapsing across fractions of the same biological mixture.
use_log_file	logical. If TRUE, information about data processing will be saved to a file.
append	logical. If TRUE, information about data processing will be added to an existing log file.
verbose	logical. If TRUE, information about data processing will be printed to the console.
log_file_path	character. Path to a file to which information about data processing will be saved. If not provided, such a file will be created automatically. If append = TRUE, has to be a valid path to a file.
...	additional parameters to <code>data.table::fread</code> .

Value

data.frame in the MSstats required format.

Author(s)

Meena Choi, Olga Vitek

Examples

```
diau_frag = system.file("tinytest/raw_data/DIAUmpire/dia_frag.csv",
                        package = "MSstatsConvert")
diau_pept = system.file("tinytest/raw_data/DIAUmpire/dia_pept.csv",
                        package = "MSstatsConvert")
diau_prot = system.file("tinytest/raw_data/DIAUmpire/dia_prot.csv",
                        package = "MSstatsConvert")
annot = system.file("tinytest/raw_data/DIAUmpire/annot_diau.csv",
                    package = "MSstatsConvert")
diau_frag = data.table::fread(diau_frag)
```

```

diau_pept = data.table::fread(diau_pept)
diau_prot = data.table::fread(diau_prot)
annot = data.table::fread(annot)
diau_frag = diau_frag[, lapply(.SD, function(x) if (is.integer(x)) as.numeric(x) else x)]
# In case numeric columns are not interpreted correctly

diau_imported = DIAUmpireToMSstatsFormat(diau_frag, diau_pept, diau_prot,
                                          annot, use_log_file = FALSE)

head(diau_imported)

```

FragPipeToMSstatsFormat

Import FragPipe files

Description

Import FragPipe files

Usage

```

FragPipeToMSstatsFormat(
  input,
  useUniquePeptide = TRUE,
  removeFewMeasurements = TRUE,
  removeProtein_with1Feature = FALSE,
  summaryforMultipleRows = max,
  use_log_file = TRUE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL,
  ...
)

```

Arguments

input	name of FragPipe msstats.csv export. ProteinName, PeptideSequence, PrecursorCharge, FragmentIon, ProductCharge, IsotopeLabelType, Condition, BioReplicate, Run, Intensity are required.
useUniquePeptide	TRUE (default) removes peptides that are assigned for more than one proteins. We assume to use unique peptide for each protein.
removeFewMeasurements	TRUE (default) will remove the features that have 1 or 2 measurements across runs.
removeProtein_with1Feature	TRUE will remove the proteins which have only 1 feature, which is the combination of peptide, precursor charge, fragment and charge. FALSE is default.

```
summaryforMultipleRows
    max or sum - when multiple PSMs identify the same feature within a single MS
    run (duplicate PSMs), use the highest (max) or sum of the duplicate intensities.
    Default is max for label-free converters and sum for TMT converters. Note
    that this parameter does NOT control collapsing across fractions of the same
    biological mixture.
use_log_file    logical. If TRUE, information about data processing will be saved to a file.
append         logical. If TRUE, information about data processing will be added to an existing
log file.
verbose        logical. If TRUE, information about data processing will be printed to the con-
sole.
log_file_path  character. Path to a file to which information about data processing will be
saved. If not provided, such a file will be created automatically. If append =
TRUE, has to be a valid path to a file.
...           additional parameters to data.table::fread.
```

Value

data.frame in the MSstats required format.

Author(s)

Devon Kohler

Examples

```
fragpipe_raw = system.file("tinytest/raw_data/FragPipe/fragpipe_input.csv",
                           package = "MSstatsConvert")
fragpipe_raw = data.table::fread(fragpipe_raw)
fragpipe_imported = FragPipeToMSstatsFormat(fragpipe_raw, use_log_file = FALSE)
head(fragpipe_imported)
```

getInputFile

Get one of files contained in an instance of MSstatsInputFiles class.

Description

Get one of files contained in an instance of MSstatsInputFiles class.

Usage

```
getInputFile(msstats_object, file_type)

## S4 method for signature 'MSstatsInputFiles'
getInputFile(msstats_object, file_type = "input")

## S4 method for signature 'MSstatsPhilosopherFiles'
getInputFile(msstats_object, file_type = "input")
```

Arguments

`msstats_object` object that inherits from `MSstatsPhilosopherFiles` class.
`file_type` character name of a type file. Usually equal to "input".

Value

`data.table`
`data.table`
`data.table`

Examples

```
evidence_path = system.file("tinytest/raw_data/MaxQuant/mq_ev.csv",
                             package = "MSstatsConvert")
pg_path = system.file("tinytest/raw_data/MaxQuant/mq_pg.csv",
                      package = "MSstatsConvert")
evidence = read.csv(evidence_path)
pg = read.csv(pg_path)
imported = MSstatsImport(list(evidence = evidence, protein_groups = pg),
                          "MSstats", "MaxQuant")

class(imported)
head(getInputFile(imported, "evidence"))
```

MaxQtoMSstatsFormat	<i>Import MaxQuant files</i>
---------------------	------------------------------

Description

Import MaxQuant files

Usage

```
MaxQtoMSstatsFormat(
  evidence,
  annotation,
  proteinGroups,
  proteinID = "Proteins",
  useUniquePeptide = TRUE,
  summaryforMultipleRows = max,
  removeFewMeasurements = TRUE,
  removeMpeptides = FALSE,
  removeOxidationMpeptides = FALSE,
  removeProtein_with1Peptide = FALSE,
  use_log_file = TRUE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL,
```

```
    ...  
  )
```

Arguments

evidence	name of 'evidence.txt' data, which includes feature-level data.
annotation	name of 'annotation.txt' data which includes Raw.file, Condition, BioReplicate, Run, IsotopeLabelType information.
proteinGroups	name of 'proteinGroups.txt' data. It needs to matching protein group ID. If proteinGroups=NULL, use 'Proteins' column in 'evidence.txt'.
proteinID	'Proteins'(default) or 'Leading.razor.protein' for Protein ID.
useUniquePeptide	TRUE (default) removes peptides that are assigned for more than one proteins. We assume to use unique peptide for each protein.
summaryforMultipleRows	max or sum - when multiple PSMs identify the same feature within a single MS run (duplicate PSMs), use the highest (max) or sum of the duplicate intensities. Default is max for label-free converters and sum for TMT converters. Note that this parameter does NOT control collapsing across fractions of the same biological mixture.
removeFewMeasurements	TRUE (default) will remove the features that have 1 or 2 measurements across runs.
removeMpeptides	TRUE will remove the peptides including 'M' sequence. FALSE is default.
removeOxidationMpeptides	TRUE will remove the peptides including 'oxidation (M)' in modification. FALSE is default.
removeProtein_with1Peptide	TRUE will remove the proteins which have only 1 peptide and charge. FALSE is default.
use_log_file	logical. If TRUE, information about data processing will be saved to a file.
append	logical. If TRUE, information about data processing will be added to an existing log file.
verbose	logical. If TRUE, information about data processing will be printed to the console.
log_file_path	character. Path to a file to which information about data processing will be saved. If not provided, such a file will be created automatically. If append = TRUE, has to be a valid path to a file.
...	additional parameters to <code>data.table::fread</code> .

Value

data.frame in the MSstats required format.

Note

Warning: MSstats does not support for metabolic labeling or iTRAQ experiments.

Author(s)

Meena Choi, Olga Vitek.

Examples

```
mq_ev = data.table::fread(system.file("tinytest/raw_data/MaxQuant/mq_ev.csv",
                                     package = "MSstatsConvert"))
mq_pg = data.table::fread(system.file("tinytest/raw_data/MaxQuant/mq_pg.csv",
                                     package = "MSstatsConvert"))
annot = data.table::fread(system.file("tinytest/raw_data/MaxQuant/annotation.csv",
                                     package = "MSstatsConvert"))
maxq_imported = MaxQtoMSstatsFormat(mq_ev, annot, mq_pg, use_log_file = FALSE)
head(maxq_imported)
```

MaxQtoMSstatsTMTFormat

Generate MSstatsTMT required input format from MaxQuant output

Description

Generate MSstatsTMT required input format from MaxQuant output

Usage

```
MaxQtoMSstatsTMTFormat(
  evidence,
  proteinGroups,
  annotation,
  which.proteinid = "Proteins",
  rmProt_Only.identified.by.site = FALSE,
  useUniquePeptide = TRUE,
  rmPSM_withfewMea_withinRun = TRUE,
  rmProtein_with1Feature = FALSE,
  summaryforMultipleRows = sum,
  use_log_file = TRUE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL,
  ...
)
```

Arguments

evidence	name of 'evidence.txt' data, which includes feature-level data.
proteinGroups	name of 'proteinGroups.txt' data.
annotation	data frame which contains column Run, Fraction, TechRepMixture, Mixture, Channel, BioReplicate, Condition. Refer to the example 'annotation.mq' for the meaning of each column.
which.proteinid	Use 'Proteins' (default) column for protein name. 'Leading.proteins' or 'Leading.razor.proteins' or 'Gene.names' can be used instead to get the protein ID with single protein. However, those can potentially have the shared peptides.
rmProt_Only.identified.by.site	TRUE will remove proteins with '+' in 'Only.identified.by.site' column from proteinGroups.txt, which was identified only by a modification site. FALSE is the default.
useUniquePeptide	TRUE (default) removes peptides that are assigned for more than one proteins. We assume to use unique peptide for each protein.
rmPSM_withfewMea_withinRun	TRUE (default) will remove the features that have 1 or 2 measurements within each Run.
rmProtein_with1Feature	TRUE will remove the proteins which have only 1 peptide and charge. Default is FALSE.
summaryforMultipleRows	max or sum - when multiple PSMs identify the same feature within a single MS run (duplicate PSMs), use the highest (max) or sum of the duplicate intensities. Default is max for label-free converters and sum for TMT converters. Note that this parameter does NOT control collapsing across fractions of the same biological mixture.
use_log_file	logical. If TRUE, information about data processing will be saved to a file.
append	logical. If TRUE, information about data processing will be added to an existing log file.
verbose	logical. If TRUE, information about data processing will be printed to the console.
log_file_path	character. Path to a file to which information about data processing will be saved. If not provided, such a file will be created automatically. If append = TRUE, has to be a valid path to a file.
...	additional parameters to <code>data.table::fread</code> .

Value

data.frame of class "MSstatsTMT"

Examples

```
evidence = data.table::fread(system.file("tinytest/raw_data/MaxQuantTMT/mq_ev.csv",
                                         package = "MSstatsConvert"))
proteinGroups = data.table::fread(system.file("tinytest/raw_data/MaxQuantTMT/mq_pg.csv",
                                              package = "MSstatsConvert"))
annotation.mq = data.table::fread(system.file("tinytest/raw_data/MaxQuantTMT/mq_annotation.csv",
                                              package = "MSstatsConvert"))
input.mq <- MaxQtoMSstatsTMTFormat(evidence, proteinGroups, annotation.mq)
head(input.mq)
```

MetamorpheusToMSstatsFormat

Import Metamorpheus files

Description

Import Metamorpheus files

Usage

```
MetamorpheusToMSstatsFormat(
  input,
  annotation = NULL,
  MBR = TRUE,
  qvalue_cutoff = 0.05,
  useUniquePeptide = TRUE,
  removeFewMeasurements = TRUE,
  removeProtein_with1Feature = FALSE,
  summaryforMultipleRows = max,
  use_log_file = TRUE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL,
  ...
)
```

Arguments

input	name of Metamorpheus output file, which is tabular format. Use the AllQuantifiedPeaks.tsv file from the Metamorpheus output.
annotation	name of 'annotation.txt' data which includes Condition, BioReplicate.
MBR	If TRUE, the function will include peaks detected by MBR
qvalue_cutoff	The q-value cutoff for filtering peaks detected by MBR
useUniquePeptide	TRUE (default) removes peptides that are assigned for more than one proteins. We assume to use unique peptide for each protein.

<code>removeFewMeasurements</code>	TRUE (default) will remove the features that have 1 or 2 measurements across runs.
<code>removeProtein_with1Feature</code>	TRUE will remove the proteins which have only 1 feature, which is the combination of peptide, precursor charge, fragment and charge. FALSE is default.
<code>summaryforMultipleRows</code>	max or sum - when multiple PSMs identify the same feature within a single MS run (duplicate PSMs), use the highest (max) or sum of the duplicate intensities. Default is max for label-free converters and sum for TMT converters. Note that this parameter does NOT control collapsing across fractions of the same biological mixture.
<code>use_log_file</code>	logical. If TRUE, information about data processing will be saved to a file.
<code>append</code>	logical. If TRUE, information about data processing will be added to an existing log file.
<code>verbose</code>	logical. If TRUE, information about data processing will be printed to the console.
<code>log_file_path</code>	character. Path to a file to which information about data processing will be saved. If not provided, such a file will be created automatically. If <code>append = TRUE</code> , has to be a valid path to a file.
<code>...</code>	additional parameters to <code>data.table::fread</code> .

Value

data.frame in the MSstats required format.

Author(s)

Anthony Wu

Examples

```
input = system.file("tinytest/raw_data/Metamorpheus/QuantifiedPeaks.tsv",
                    package = "MSstatsConvert")
input = data.table::fread(input)
annot = system.file("tinytest/raw_data/Metamorpheus/annotation.csv",
                    package = "MSstatsConvert")
annot = data.table::fread(annot)
metamorpheus_imported = MSstatsConvert::MetamorpheusToMSstatsFormat(input, annotation = annot)
head(metamorpheus_imported)
```

MSstatsAnomalyScores *Run Anomaly Model*

Description

Detects anomalous measurements in mass spectrometry data using an isolation forest algorithm. This function identifies unusual precursor measurements based on quality metrics and their temporal patterns. For features with insufficient quality metric data, it assigns anomaly scores based on the median score of similar features (same peptide and charge combination). The model supports parallel processing for improved performance on large datasets.

Usage

```
MSstatsAnomalyScores(  
  input,  
  quality_metrics,  
  temporal_direction,  
  missing_run_count,  
  n_feat,  
  run_order,  
  n_trees,  
  max_depth,  
  cores  
)
```

Arguments

input	data.table preprocessed by the MSstatsBalancedDesign function
quality_metrics	character vector of quality metrics to use in the model
temporal_direction	character vector of same length as quality_metrics indicating temporal feature to create.
missing_run_count	numeric, maximum allowed fraction of missing runs per feature.
n_feat	numeric, maximum number of features per protein to use in the model.
run_order	data.frame with two columns: Run and Order. Order should be numeric and indicate the order of runs.
n_trees	numeric, number of trees to use in the isolation forest model. Default is 100.
max_depth	numeric or "auto", maximum depth of each tree. Default is "auto" which sets depth to $\log_2(N)$ where N is the number of runs.
cores	numeric, number of cores to use for parallel processing. Default is 1.

Value

data.table

MSstatsBalancedDesign *Creates balanced design by removing overlapping fractions and filling incomplete rows*

Description

Creates balanced design by removing overlapping fractions and filling incomplete rows

Usage

```
MSstatsBalancedDesign(
  input,
  feature_columns,
  fill_incomplete = TRUE,
  handle_fractions = TRUE,
  fix_missing = NULL,
  remove_few = TRUE,
  anomaly_metrics = c()
)
```

Arguments

input	data.table processed by the MSstatsPreprocess function
feature_columns	str, names of columns that define spectral features
fill_incomplete	if TRUE (default), ensures that rows with missing data for specific features are added as NA. For example, if the y10 ion of peptideA is measured in the "disease" samples but entirely missing for the "healthy" samples, rows with NA values will be created for the y10 ion of peptideA in the "healthy" group. This process increases the number of rows to account for all possible feature-sample combinations.
handle_fractions	if TRUE (default), overlapping fractions will be resolved
fix_missing	str, optional. Defaults to NULL, which means no action. If not NULL, must be one of the options: "zero_to_na" or "na_to_zero". If "zero_to_na", Intensity values equal exactly to 0 will be converted to NA. If "na_to_zero", missing values will be replaced by zeros.
remove_few	lgl, if TRUE, features with one or two measurements across runs will be removed.
anomaly_metrics	character vector of names of columns with quality metrics

Value

data.frame of class MSstatsValidated

Examples

```

unbalanced_data = system.file("tinytest/raw_data/unbalanced_data.csv",
                              package = "MSstatsConvert")
unbalanced_data = data.table::as.data.table(read.csv(unbalanced_data))
balanced = MSstatsBalancedDesign(unbalanced_data,
                                 c("PeptideSequence", "PrecursorCharge",
                                   "FragmentIon", "ProductCharge"))
dim(balanced) # Now balanced has additional rows (with Intensity = NA)
# for runs that were not included in the unbalanced_data table

```

MSstatsClean

*Clean files generated by a signal processing tools.***Description**

Clean files generated by a signal processing tools.

Clean DIAUmpire files

Clean MaxQuant files

Clean OpenMS files

Clean OpenSWATH files

Clean Progenesis files

Clean ProteomeDiscoverer files

Clean Skyline files

Clean SpectroMine files

Clean Spectronaut files

Clean Philosopher files

Clean DIA-NN files

Clean Metamorpheus files

Clean Protein Prospector files

Clean MZMine files

Usage

```
MSstatsClean(msstats_object, ...)
```

```
## S4 method for signature 'MSstatsDIAUmpireFiles'
```

```
MSstatsClean(msstats_object, use_frag, use_pept)
```

```
## S4 method for signature 'MSstatsMaxQuantFiles'
```

```
MSstatsClean(
  msstats_object,
  protein_id_col,
```

```
    remove_by_site = FALSE,
    channel_columns = "Reporterintensitycorrected"
)

## S4 method for signature 'MSstatsOpenMSFiles'
MSstatsClean(msstats_object)

## S4 method for signature 'MSstatsOpenSWATHFiles'
MSstatsClean(msstats_object)

## S4 method for signature 'MSstatsProgenesisFiles'
MSstatsClean(msstats_object, runs, fix_colnames = TRUE)

## S4 method for signature 'MSstatsProteomeDiscovererFiles'
MSstatsClean(
  msstats_object,
  quantification_column,
  protein_id_column,
  sequence_column,
  remove_shared,
  remove_protein_groups = TRUE,
  intensity_columns_regexp = "Abundance"
)

## S4 method for signature 'MSstatsSkylineFiles'
MSstatsClean(msstats_object)

## S4 method for signature 'MSstatsSpectroMineFiles'
MSstatsClean(msstats_object)

## S4 method for signature 'MSstatsSpectronautFiles'
MSstatsClean(
  msstats_object,
  intensity,
  calculateAnomalyScores,
  anomalyModelFeatures,
  peptideSequenceColumn = "EG.ModifiedSequence",
  heavyLabels = NULL
)

## S4 method for signature 'MSstatsPhilosopherFiles'
MSstatsClean(
  msstats_object,
  protein_id_col,
  peptide_id_col,
  channels,
  remove_shared_peptides
)
```

```

## S4 method for signature 'MSstatsDIANNFiles'
MSstatsClean(
  msstats_object,
  MBR = TRUE,
  quantificationColumn = "FragmentQuantCorrected",
  global_qvalue_cutoff = 0.01,
  qvalue_cutoff = 0.01,
  pg_qvalue_cutoff = 0.01,
  calculateAnomalyScores = FALSE,
  anomalyModelFeatures = c(),
  labeledAminoAcids = NULL
)

## S4 method for signature 'MSstatsMetamorpheusFiles'
MSstatsClean(msstats_object, MBR = TRUE, qvalue_cutoff = 0.05)

## S4 method for signature 'MSstatsProteinProspectorFiles'
MSstatsClean(msstats_object)

## S4 method for signature 'MSstatsMZMineFiles'
MSstatsClean(msstats_object, mzmine_annotations, sirius_annotations = NULL)

```

Arguments

<code>msstats_object</code>	object that inherits from <code>MSstatsInputFiles</code> class.
<code>...</code>	additional parameter to specific cleaning functions.
<code>use_frag</code>	<code>TRUE</code> will use the selected fragment for each peptide. 'Selected_fragments' column is required.
<code>use_pept</code>	<code>TRUE</code> will use the selected fragment for each protein 'Selected_peptides' column is required.
<code>protein_id_col</code>	character, name of a column with names of proteins.
<code>remove_by_site</code>	logical, if <code>TRUE</code> , proteins only identified by site will be removed.
<code>channel_columns</code>	character, regular expression that identifies channel columns in TMT data.
<code>runs</code>	chr, vector of Run labels.
<code>fix_colnames</code>	lgl, if <code>TRUE</code> , one of the rows will be used as colnames.
<code>quantification_column</code>	chr, name of a column used for quantification.
<code>protein_id_column</code>	chr, name of a column with protein IDs.
<code>sequence_column</code>	chr, name of a column with peptide sequences.
<code>remove_shared</code>	lgl, if <code>TRUE</code> , shared peptides will be removed.
<code>remove_protein_groups</code>	if <code>TRUE</code> , proteins with <code>numProteins > 1</code> will be removed.

intensity_columns_regexp	regular expressions that defines intensity columns. Defaults to "Abundance", which means that columns that contain the word "Abundance" will be treated as corresponding to intensities for different channels.
intensity	Intensity column to use. Accepts legacy enum values 'PeakArea' (default, uses F.PeakArea), 'NormalizedPeakArea' (uses F.NormalizedPeakArea). Can also be any raw Spectronaut column name passed as a string (e.g. "FG.MS1Quantity"); the column name is standardized internally. For protein turnover workflows the recommended default is "FG.MS1Quantity".
calculateAnomalyScores	Default is FALSE. If TRUE, will run anomaly detection model and calculate anomaly scores for each feature. Used downstream to weigh measurements in differential analysis.
anomalyModelFeatures	character vector of quality metric column names to be used as features in the anomaly detection model. List must not be empty if calculateAnomalyScores=TRUE.
peptideSequenceColumn	Name of the Spectronaut column that contains the peptide sequence. Defaults to "EG.ModifiedSequence". The value is standardized internally (dots and spaces removed) before column lookup.
heavyLabels	Character vector naming each labeled residue and its heavy label as they appear in the peptide sequence column: the single-letter amino acid code, then the label in square brackets. For example "K[Lys6]", or c("K[Lys6]", "R[Arg10]") for a double-label experiment. Any label name Spectronaut reports is accepted. The residue letter is required, because it is what tells MSstats which peptides could have carried a label. Peptides carrying the label are marked heavy (IsotopeLabelType = "H"), those that could carry it but do not are light ("L"), and those with no labelable residue are NA. A peptide with two or more labelable residues (counted across all labels supplied) may be only partially labeled, which the two-state turnover model cannot represent, so it is dropped and the number removed is reported. Defaults to NULL: turnover mode off, every peptide marked light.
peptide_id_col	character name of a column that identifies peptides
channels	character vector of channel labels
remove_shared_peptides	logical, if TRUE, shared peptides will be removed based on the IsUnique column from Philosopher output
MBR	True if analysis was done with match between runs
quantificationColumn	Use 'FragmentQuantCorrected'(default) column for quantified intensities for DIANN 1.8.x. Use 'FragmentQuantRaw' for quantified intensities for DIANN 1.9.x. Use 'auto' for quantified intensities for DIANN 2.x where each fragment intensity is a separate column, e.g. Fr0Quantity.
global_qvalue_cutoff	The qvalue cutoff for the Q.Value column, i.e. the run-specific precursor q-value. Default is 0.01.

- qvalue_cutoff** If MBR is false, the qvalue cutoff for the Global.Q.Value column, i.e. global precursor q-value. If MBR is true, the qvalue cutoff for the Lib.Q.Value column, i.e. the q-value for the library created after the first MBR pass. Default is 0.01.
- pg_qvalue_cutoff** If MBR is false, the qvalue cutoff for the Global.PG.Q.Value column, i.e. the global q-value for the protein group. If MBR is true, the qvalue cutoff for the Lib.PG.Q.Value column, i.e. the protein group q-value for the library created after the first MBR pass. Default is 0.01.
- labeledAminoAcids** Character vector of single-letter codes for the amino acids carrying the SILAC label, e.g. c("K") or c("K", "R"). Supplying it turns on protein-turnover mode. Defaults to NULL, which marks every peptide IsotopeLabelType = "Light". How the label is read depends on the export:
DIA-NN 2.x, with a Channel column. Channel gives the label directly: "H" and "L" pass through, anything else becomes NA. The amino acid codes only switch the mode on here; sequences are not inspected.
DIA-NN 1.x, without a Channel column. The label is read from ModifiedSequence, where it follows the labeled residue in parentheses as (<label>-<aminoAcid>-H) or (<label>-<aminoAcid>-L); for example K(label-K-H) is a heavy-labeled lysine. <label> is the label name DIA-NN reported (e.g. SILAC). Untagged sequences become NA, and the tags are stripped from PeptideSequence afterwards. A peptide with two or more labelable residues (counted across all of labeledAminoAcids) may be only partially labeled, which the two-state turnover model cannot represent, so it is dropped and the number removed is reported.
- mzmine_annotations** data.frame of MZMine spectral-library annotations with columns id, compound_name, score. Required; passing NULL raises an error. The highest-scoring compound_name per feature (MSI Level 2 putative identification via MS/MS spectral matching) is used as ProteinName.
- sirius_annotations** Optional data.frame of SIRIUS structure_identifications.tsv output, or NULL. Only the mappingFeatureId and name columns are read; score columns (ConfidenceScoreExact, ConfidenceScoreApproximate, SiriusScore) are ignored in this release. When supplied, the SIRIUS name (MSI Level 3, in-silico structure prediction) fills ProteinName for features that received no MZMine compound name. The schema is validated against SIRIUS 6 output; users on other versions can rename columns to match. Pass NULL to disable.

Value

data.table
 data.table
 data.table
 data.table
 data.table

data.table
data.table
data.table
data.table
data.table
data.table
data.table
data.table
data.table

Examples

```
evidence_path = system.file("tinytest/raw_data/MaxQuant/mq_ev.csv",
                             package = "MSstatsConvert")
pg_path = system.file("tinytest/raw_data/MaxQuant/mq_pg.csv",
                      package = "MSstatsConvert")
evidence = read.csv(evidence_path)
pg = read.csv(pg_path)
imported = MSstatsImport(list(evidence = evidence, protein_groups = pg),
                          "MSstats", "MaxQuant")
cleaned_data = MSstatsClean(imported, protein_id_col = "Proteins")
head(cleaned_data)
```

MSstatsImport	<i>Import files from signal processing tools.</i>
---------------	---

Description

Import files from signal processing tools.

Usage

```
MSstatsImport(input_files, type, tool, tool_version = NULL, ...)
```

Arguments

- | | |
|--------------|---|
| input_files | list of paths to input files or data.frame objects. Interpretation of this parameter depends on values of parameters type and tool. |
| type | chr, "MSstats" or "MSstatsTMT". |
| tool | chr, name of a signal processing tool that generated input files. |
| tool_version | not implemented yet. In the future, this parameter will allow handling different versions of each signal processing tools. |
| ... | optional additional parameters to data.table::fread. |

Value

an object of class MSstatsInputFiles.

Examples

```
evidence_path = system.file("tinytest/raw_data/MaxQuant/mq_ev.csv",
                             package = "MSstatsConvert")
pg_path = system.file("tinytest/raw_data/MaxQuant/mq_pg.csv",
                       package = "MSstatsConvert")
evidence = read.csv(evidence_path)
pg = read.csv(pg_path)
imported = MSstatsImport(list(evidence = evidence, protein_groups = pg),
                          "MSstats", "MaxQuant")

class(imported)
head(getInputFile(imported, "evidence"))
```

MSstatsLogsSettings *Set how MSstats will log information from data processing*

Description

Set how MSstats will log information from data processing

Usage

```
MSstatsLogsSettings(
  use_log_file = TRUE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL,
  base = "MSstats_log_",
  pkg_name = "MSstats"
)
```

Arguments

use_log_file	logical. If TRUE, information about data processing will be saved to a file.
append	logical. If TRUE, information about data processing will be added to an existing log file.
verbose	logical. If TRUE, information about data processing will be printed to the console.
log_file_path	character. Path to a file to which information about data processing will be saved. If not provided, such a file will be created automatically. If append = TRUE, has to be a valid path to a file.
base	start of the file name.
pkg_name	currently "MSstats", "MSstatsPTM" or "MSstatsTMT". Each package can use its own separate log settings.

Value

TRUE invisibly in case of successful logging setup.

Examples

```
# No logging and no messages
MSstatsLogsSettings(FALSE, FALSE, FALSE)
# Log, but do not display messages
MSstatsLogsSettings(TRUE, FALSE, FALSE)
# Log to an existing file
file.create("new_log.log")
MSstatsLogsSettings(TRUE, TRUE, log_file_path = "new_log.log")
# Do not log, but display messages
MSstatsLogsSettings(FALSE)
```

MSstatsMakeAnnotation *Create annotation*

Description

Create annotation

Usage

```
MSstatsMakeAnnotation(input, annotation, ...)
```

Arguments

input	data.table preprocessed by the MSstatsClean function
annotation	data.table
...	key-value pairs, where keys are names of columns of annotation

Value

data.table

Examples

```
evidence_path = system.file("tinytest/raw_data/MaxQuant/mq_ev.csv",
                             package = "MSstatsConvert")
pg_path = system.file("tinytest/raw_data/MaxQuant/mq_pg.csv",
                      package = "MSstatsConvert")
evidence = read.csv(evidence_path)
pg = read.csv(pg_path)
imported = MSstatsImport(list(evidence = evidence, protein_groups = pg),
                          "MSstats", "MaxQuant")
cleaned_data = MSstatsClean(imported, protein_id_col = "Proteins")
```

```

annot_path = system.file("tinytest/raw_data/MaxQuant/annotation.csv",
                          package = "MSstatsConvert")
mq_annot = MSstatsMakeAnnotation(cleaned_data, read.csv(annot_path),
                                Run = "Rawfile")
head(mq_annot)

```

MSstatsPreprocess	<i>Preprocess outputs from MS signal processing tools for analysis with MSstats</i>
-------------------	---

Description

Preprocess outputs from MS signal processing tools for analysis with MSstats

Usage

```

MSstatsPreprocess(
  input,
  annotation,
  feature_columns,
  remove_shared_peptides = TRUE,
  remove_single_feature_proteins = TRUE,
  feature_cleaning = list(remove_features_with_few_measurements = TRUE,
    summarize_multiple_psms = max),
  score_filtering = list(),
  exact_filtering = list(),
  pattern_filtering = list(),
  columns_to_fill = list(),
  aggregate_isotopic = FALSE,
  anomaly_metrics = c(),
  ...
)

```

Arguments

input	data.table processed by the MSstatsClean function.
annotation	annotation file generated by a signal processing tool.
feature_columns	character vector of names of columns that define spectral features.
remove_shared_peptides	logical, if TRUE shared peptides will be removed.
remove_single_feature_proteins	logical, if TRUE, proteins that only have one feature will be removed.

<code>feature_cleaning</code>	named list with maximum two (for MSstats converters) or three (for MSstatsTMT converter) elements. If <code>handle_few_measurements</code> is set to "remove", feature with less than three measurements will be removed (otherwise it should be equal to "keep"). <code>summarize_multiple_psms</code> is a function that will be used to aggregate multiple feature measurements in a run. It should return a scalar and accept an <code>na.rm</code> parameter. For MSstatsTMT converters, setting <code>remove_psms_with_any_missing</code> will remove features which have missing values in a run from that run.
<code>score_filtering</code>	a list of named lists that specify filtering options. Details are provided in the vignette.
<code>exact_filtering</code>	a list of named lists that specify filtering options. Details are provided in the vignette.
<code>pattern_filtering</code>	a list of named lists that specify filtering options. Details are provided in the vignette.
<code>columns_to_fill</code>	a named list of scalars. If provided, columns with names defined by the names of this list and values corresponding to its elements will be added to the output <code>data.frame</code> .
<code>aggregate_isotopic</code>	logical. If TRUE, isotopic peaks will be summed.
<code>anomaly_metrics</code>	character vector of names of columns with quality metrics. Default is missing and is not required if anomaly model not run.
<code>...</code>	additional parameters to <code>data.table::fread</code> .

Value

`data.table`

Examples

```
evidence_path = system.file("tinytest/raw_data/MaxQuant/mq_ev.csv",
                             package = "MSstatsConvert")
pg_path = system.file("tinytest/raw_data/MaxQuant/mq_pg.csv",
                       package = "MSstatsConvert")
evidence = read.csv(evidence_path)
pg = read.csv(pg_path)
imported = MSstatsImport(list(evidence = evidence, protein_groups = pg),
                          "MSstats", "MaxQuant")
cleaned_data = MSstatsClean(imported, protein_id_col = "Proteins")
annot_path = system.file("tinytest/raw_data/MaxQuant/annotation.csv",
                          package = "MSstatsConvert")
mq_annot = MSstatsMakeAnnotation(cleaned_data, read.csv(annot_path),
                                 Run = "Rawfile")
```

```
# To filter M-peptides and oxidatin peptides
m_filter = list(col_name = "PeptideSequence", pattern = "M",
                filter = TRUE, drop_column = FALSE)
oxidation_filter = list(col_name = "Modifications", pattern = "Oxidation",
                       filter = TRUE, drop_column = TRUE)
msstats_format = MSstatsPreprocess(
  cleaned_data, mq_annot,
  feature_columns = c("PeptideSequence", "PrecursorCharge"),
  columns_to_fill = list(FragmentIon = NA, ProductCharge = NA),
  pattern_filtering = list(oxidation = oxidation_filter, m = m_filter)
)
# Output in the standard MSstats format
head(msstats_format)
```

MSstatsSaveSessionInfo

Save session information

Description

Save session information

Usage

```
MSstatsSaveSessionInfo(
  path = NULL,
  append = TRUE,
  base = "MSstats_session_info_"
)
```

Arguments

path	optional path to output file. If not provided, "MSstats_session_info" and current timestamp will be used as a file name
append	if TRUE and file given by the path parameter already exists, session info will be appended to the file
base	beginning of a file name

Value

TRUE invisibly after session info was saved

Examples

```
MSstatsSaveSessionInfo("session_info.txt")
MSstatsSaveSessionInfo("session_info.txt", base = "MSstatsTMT_session_info_")
```

MZMinetoMSstatsFormat *Import MZMine files*

Description

Import MZMine files

Usage

```
MZMinetoMSstatsFormat(
  input,
  annotation = NULL,
  mzmine_annotations,
  sirius_annotations = NULL,
  summaryforMultipleRows = max,
  use_log_file = TRUE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL,
  ...
)
```

Arguments

input	MZMine feature-quantification table (wide format; one row per feature). Must include the metadata columns row ID, row m/z, row retention time, and per-sample peak-area columns named "<run> Peak area" (e.g. "sampleA.mzML Peak area").
annotation	data.frame with columns Run, Condition, BioReplicate. Run values must match MSstatsConvert-standardized sample names (after column-name normalization removes spaces and dots) with the trailing "Peakarea" suffix removed. For example, a quant-file column "sampleA.mzML Peak area" becomes "sampleAmzML" after standardization, so the corresponding Run value must be sampleAmzML.
mzmine_annotations	data.frame of MZMine spectral-library annotations with columns id, compound_name, score. Required; passing NULL raises an error. The highest-scoring compound_name per feature (MSI Level 2 putative identification via MS/MS spectral matching) is used as ProteinName.
sirius_annotations	Optional data.frame of SIRIUS structure_identifications.tsv output, or NULL. Only the mappingFeatureId and name columns are read; score columns (ConfidenceScoreExact, ConfidenceScoreApproximate, SiriusScore) are ignored in this release. When supplied, the SIRIUS name (MSI Level 3, in-silico structure prediction) fills ProteinName for features that received no MZMine compound name. The schema is validated against SIRIUS 6 output; users on other versions can rename columns to match. Pass NULL to disable.

<code>summaryforMultipleRows</code>	max or sum - when multiple PSMs identify the same feature within a single MS run (duplicate PSMs), use the highest (max) or sum of the duplicate intensities. Default is max for label-free converters and sum for TMT converters. Note that this parameter does NOT control collapsing across fractions of the same biological mixture.
<code>use_log_file</code>	logical. If TRUE, information about data processing will be saved to a file.
<code>append</code>	logical. If TRUE, information about data processing will be added to an existing log file.
<code>verbose</code>	logical. If TRUE, information about data processing will be printed to the console.
<code>log_file_path</code>	character. Path to a file to which information about data processing will be saved. If not provided, such a file will be created automatically. If <code>append = TRUE</code> , has to be a valid path to a file.
<code>...</code>	additional parameters to <code>data.table::fread</code> .

Details

`ProteinName` is assigned from one of three sources, in priority order: the MZMine compound name (mandatory), the SIRIUS name (optional), and an m/z-RT fallback (always available).

The **MZMine compound name** is the highest-scoring `compound_name` from `mzmine_annot` for each feature. This corresponds to MSI Level 2 (Sumner et al. 2007, PMID 27624161): a putative identification by MS/MS spectral matching to a reference library.

The **SIRIUS name** comes from SIRIUS's `structure_identifications.tsv` and corresponds to MSI Level 3: an in-silico structure prediction. When `sirius_annot` is non-NULL, the SIRIUS name fills `ProteinName` only for features the MZMine library missed – the MZMine compound name takes precedence.

The **m/z-RT fallback** is an identifier built from the feature's m/z and retention time (for example, `455.282_0.65`). Features that receive no MZMine or SIRIUS annotation are retained, not dropped, and assigned an m/z-RT identifier as their `ProteinName`.

Retaining every feature is a deliberate trade-off. A fuller feature set gives more stable medians and a more reliable empirical distribution for global normalization. SIRIUS extends discovery coverage to features that level-2 spectral matching misses. The cost is an increase in the number of hypotheses tested downstream (in `MSstats::groupComparison`), which weakens multiple-testing correction. Users running confirmatory analyses should restrict to the MZMine-annotated features post-conversion; users running discovery analyses benefit from the additional sources despite the FDR burden.

Value

`data.table` in the MSstats required format.

Examples

```
input_path = system.file("tinytest/raw_data/MZMine/mzmine_input.csv",
                          package = "MSstatsConvert")
annot_path = system.file("tinytest/raw_data/MZMine/annotation.csv",
```

```

                                package = "MSstatsConvert")
lib_path  = system.file("tinytest/raw_data/MZMine/mzmine_annotations.csv",
                                package = "MSstatsConvert")
input = data.table::fread(input_path)
annot = data.table::fread(annot_path)
lib    = data.table::fread(lib_path)
output = MZMinetoMSstatsFormat(input, annotation = annot,
                                mzmine_annotations = lib,
                                use_log_file = FALSE)

head(output)

# With SIRIUS annotations:
sirius_path = system.file(
  "tinytest/raw_data/MZMine/structure_identifications.tsv",
  package = "MSstatsConvert")
sirius = data.table::fread(sirius_path)
output_with_sirius = MZMinetoMSstatsFormat(
  input, annotation = annot,
  mzmine_annotations = lib,
  sirius_annotations = sirius,
  use_log_file = FALSE)
head(output_with_sirius)

```

OpenMStoMSstatsFormat *Import OpenMS files*

Description

Import OpenMS files

Usage

```

OpenMStoMSstatsFormat(
  input,
  annotation = NULL,
  useUniquePeptide = TRUE,
  removeFewMeasurements = TRUE,
  removeProtein_with1Feature = FALSE,
  summaryforMultipleRows = max,
  use_log_file = TRUE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL,
  ...
)

```

Arguments

input	name of MSstats input report from OpenMS, which includes feature(peptide ion)-level data.
annotation	name of 'annotation.txt' data which includes Condition, BioReplicate, Run. Run should be the same as filename.
useUniquePeptide	TRUE (default) removes peptides that are assigned for more than one proteins. We assume to use unique peptide for each protein.
removeFewMeasurements	TRUE (default) will remove the features that have 1 or 2 measurements across runs.
removeProtein_with1Feature	TRUE will remove the proteins which have only 1 feature, which is the combination of peptide, precursor charge, fragment and charge. FALSE is default.
summaryforMultipleRows	max or sum - when multiple PSMs identify the same feature within a single MS run (duplicate PSMs), use the highest (max) or sum of the duplicate intensities. Default is max for label-free converters and sum for TMT converters. Note that this parameter does NOT control collapsing across fractions of the same biological mixture.
use_log_file	logical. If TRUE, information about data processing will be saved to a file.
append	logical. If TRUE, information about data processing will be added to an existing log file.
verbose	logical. If TRUE, information about data processing will be printed to the console.
log_file_path	character. Path to a file to which information about data processing will be saved. If not provided, such a file will be created automatically. If append = TRUE, has to be a valid path to a file.
...	additional parameters to <code>data.table::fread</code> .

Value

data.frame in the MSstats required format.

Author(s)

Meena Choi, Olga Vitek.

Examples

```
openms_raw = data.table::fread(system.file("tinytest/raw_data/OpenMS/openms_input.csv",
                                           package = "MSstatsConvert"))
openms_imported = OpenMStoMSstatsFormat(openms_raw, use_log_file = FALSE)
head(openms_imported)
```

OpenMStoMSstatsTMTFormat

Generate MSstatsTMT required input format for OpenMS output

Description

Generate MSstatsTMT required input format for OpenMS output

Usage

```
OpenMStoMSstatsTMTFormat(
  input,
  useUniquePeptide = TRUE,
  rmPSM_withfewMea_withinRun = TRUE,
  rmProtein_with1Feature = FALSE,
  summaryforMultiplePSMs = sum,
  use_log_file = TRUE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL,
  ...
)
```

Arguments

input	MSstatsTMT report from OpenMS
useUniquePeptide	TRUE (default) removes peptides that are assigned for more than one proteins. We assume to use unique peptide for each protein.
rmPSM_withfewMea_withinRun	TRUE (default) will remove the features that have 1 or 2 measurements within each Run.
rmProtein_with1Feature	TRUE will remove the proteins which have only 1 peptide and charge. Default is FALSE.
summaryforMultiplePSMs	sum(default) or max - when there are multiple measurements for certain feature in certain run, select the feature with the largest summation or maximal value.
use_log_file	logical. If TRUE, information about data processing will be saved to a file.
append	logical. If TRUE, information about data processing will be added to an existing log file.
verbose	logical. If TRUE, information about data processing will be printed to the console.
log_file_path	character. Path to a file to which information about data processing will be saved. If not provided, such a file will be created automatically. If append = TRUE, has to be a valid path to a file.
...	additional parameters to <code>data.table::fread</code> .

Value

data.frame of class MSstatsTMT.

Examples

```
raw.om = data.table::fread(system.file("tinytest/raw_data/OpenMSTMT/openmstmt_input.csv",
                                       package = "MSstatsConvert"))
input.om <- OpenMSToMSstatsTMTFormat(raw.om)
head(input.om)
```

OpenSWATHtoMSstatsFormat

Import OpenSWATH files

Description

Import OpenSWATH files

Usage

```
OpenSWATHtoMSstatsFormat(
  input,
  annotation,
  filter_with_mscore = TRUE,
  mscore_cutoff = 0.01,
  useUniquePeptide = TRUE,
  removeFewMeasurements = TRUE,
  removeProtein_with1Feature = FALSE,
  summaryforMultipleRows = max,
  use_log_file = TRUE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL,
  ...
)
```

Arguments

input	name of MSstats input report from OpenSWATH, which includes feature-level data.
annotation	name of 'annotation.txt' data which includes Condition, BioReplicate, Run. Run should be the same as filename.
filter_with_mscore	TRUE(default) will filter out the features that have greater than mscore_cutoff in m_score column. Those features will be removed.
mscore_cutoff	Cutoff for m_score. Default is 0.01.

<code>useUniquePeptide</code>	TRUE (default) removes peptides that are assigned for more than one proteins. We assume to use unique peptide for each protein.
<code>removeFewMeasurements</code>	TRUE (default) will remove the features that have 1 or 2 measurements across runs.
<code>removeProtein_with1Feature</code>	TRUE will remove the proteins which have only 1 feature, which is the combination of peptide, precursor charge, fragment and charge. FALSE is default.
<code>summaryforMultipleRows</code>	max or sum - when multiple PSMs identify the same feature within a single MS run (duplicate PSMs), use the highest (max) or sum of the duplicate intensities. Default is max for label-free converters and sum for TMT converters. Note that this parameter does NOT control collapsing across fractions of the same biological mixture.
<code>use_log_file</code>	logical. If TRUE, information about data processing will be saved to a file.
<code>append</code>	logical. If TRUE, information about data processing will be added to an existing log file.
<code>verbose</code>	logical. If TRUE, information about data processing will be printed to the console.
<code>log_file_path</code>	character. Path to a file to which information about data processing will be saved. If not provided, such a file will be created automatically. If <code>append = TRUE</code> , has to be a valid path to a file.
<code>...</code>	additional parameters to <code>data.table::fread</code> .

Value

data.frame in the MSstats required format.

Author(s)

Meena Choi, Olga Vitek.

Examples

```
os_raw = system.file("tinytest/raw_data/OpenSWATH/openswath_input.csv",
                     package = "MSstatsConvert")
annot = system.file("tinytest/raw_data/OpenSWATH/annot_os.csv",
                   package = "MSstatsConvert")
os_raw = data.table::fread(os_raw)
annot = data.table::fread(annot)

os_imported = OpenSWATHtoMSstatsFormat(os_raw, annot, use_log_file = FALSE)
head(os_imported)
```

PDtoMSstatsFormat	<i>Import Proteome Discoverer files</i>
-------------------	---

Description

Import Proteome Discoverer files

Usage

```
PDtoMSstatsFormat(
  input,
  annotation,
  useNumProteinsColumn = FALSE,
  useUniquePeptide = TRUE,
  summaryforMultipleRows = max,
  removeFewMeasurements = TRUE,
  removeOxidationMpeptides = FALSE,
  removeProtein_with1Peptide = FALSE,
  which.quantification = "Precursor.Area",
  which.proteinid = "Protein.Group.Accessions",
  which.sequence = "Sequence",
  use_log_file = TRUE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL,
  ...
)
```

Arguments

input	PD report or a path to it.
annotation	name of 'annotation.txt' or 'annotation.csv' data which includes Condition, BioReplicate, Run information. 'Run' will be matched with 'Spectrum.File'.
useNumProteinsColumn	TRUE removes peptides which have more than 1 in # Proteins column of PD output.
useUniquePeptide	TRUE (default) removes peptides that are assigned for more than one proteins. We assume to use unique peptide for each protein.
summaryforMultipleRows	max or sum - when multiple PSMs identify the same feature within a single MS run (duplicate PSMs), use the highest (max) or sum of the duplicate intensities. Default is max for label-free converters and sum for TMT converters. Note that this parameter does NOT control collapsing across fractions of the same biological mixture.

```

removeFewMeasurements
    TRUE (default) will remove the features that have 1 or 2 measurements across
    runs.
removeOxidationMpeptides
    TRUE will remove the peptides including 'oxidation (M)' in modification. FALSE
    is default.
removeProtein_with1Peptide
    TRUE will remove the proteins which have only 1 peptide and charge. FALSE
    is default.
which.quantification
    Use 'Precursor.Area'(default) column for quantified intensities. 'Intensity' or
    'Area' can be used instead.
which.proteinid
    Use 'Protein.Accessions'(default) column for protein name. 'Master.Protein.Accessions'
    can be used instead.
which.sequence
    Use 'Sequence'(default) column for peptide sequence. 'Annotated.Sequence'
    can be used instead.
use_log_file
    logical. If TRUE, information about data processing will be saved to a file.
append
    logical. If TRUE, information about data processing will be added to an existing
    log file.
verbose
    logical. If TRUE, information about data processing will be printed to the con-
    sole.
log_file_path
    character. Path to a file to which information about data processing will be
    saved. If not provided, such a file will be created automatically. If append =
    TRUE, has to be a valid path to a file.
...
    additional parameters to data.table::fread.

```

Value

data.frame in the MSstats required format.

Author(s)

Meena Choi, Olga Vitek

Examples

```

pd_raw = system.file("tinytest/raw_data/PD/pd_input.csv",
    package = "MSstatsConvert")
annot = system.file("tinytest/raw_data/PD/annot_pd.csv",
    package = "MSstatsConvert")
pd_raw = data.table::fread(pd_raw)
annot = data.table::fread(annot)

pd_imported = PDtoMSstatsFormat(pd_raw, annot, use_log_file = FALSE)
head(pd_imported)

```

PDtoMSstatsTMTFormat *Convert Proteome Discoverer output to MSstatsTMT format.*

Description

Convert Proteome Discoverer output to MSstatsTMT format.

Usage

```
PDtoMSstatsTMTFormat(
  input,
  annotation,
  which.proteinid = "Protein.Accessions",
  useNumProteinsColumn = TRUE,
  useUniquePeptide = TRUE,
  rmPSM_withfewMea_withinRun = TRUE,
  rmProtein_with1Feature = FALSE,
  summaryforMultipleRows = sum,
  use_log_file = TRUE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL,
  ...
)
```

Arguments

input	PD report or a path to it.
annotation	annotation with Run, Fraction, TechRepMixture, Mixture, Channel, BioReplicate, Condition columns or a path to file. Refer to the example 'annotation' for the meaning of each column.
which.proteinid	Use 'Protein.Accessions' (default) column for protein name. 'Master.Protein.Accessions' can be used instead to get the protein name with single protein.
useNumProteinsColumn	logical, TRUE (default) removes shared peptides by information of # Proteins column in PSM sheet.
useUniquePeptide	TRUE (default) removes peptides that are assigned for more than one proteins. We assume to use unique peptide for each protein.
rmPSM_withfewMea_withinRun	TRUE (default) will remove the features that have 1 or 2 measurements within each Run.
rmProtein_with1Feature	TRUE will remove the proteins which have only 1 peptide and charge. Default is FALSE.

```
summaryforMultipleRows
    max or sum - when multiple PSMs identify the same feature within a single MS
    run (duplicate PSMs), use the highest (max) or sum of the duplicate intensities.
    Default is max for label-free converters and sum for TMT converters. Note
    that this parameter does NOT control collapsing across fractions of the same
    biological mixture.
use_log_file    logical. If TRUE, information about data processing will be saved to a file.
append         logical. If TRUE, information about data processing will be added to an existing
log file.
verbose        logical. If TRUE, information about data processing will be printed to the con-
sole.
log_file_path  character. Path to a file to which information about data processing will be
saved. If not provided, such a file will be created automatically. If append =
TRUE, has to be a valid path to a file.
...           additional parameters to data.table::fread.
```

Value

data.frame of class MSstatsTMT

Examples

```
raw.pd = data.table::fread(system.file("tinytest/raw_data/PDTMT/pdmtmt_input.csv",
                                     package = "MSstatsConvert"))
annotation.pd = data.table::fread(system.file("tinytest/raw_data/PDTMT/pd_annotation.csv",
                                     package = "MSstatsConvert"))
head(raw.pd)
head(annotation.pd)
input.pd <- PDtoMSstatsTMTFormat(raw.pd, annotation.pd)
head(input.pd)
```

PhilosophertoMSstatsTMTFormat

Convert Philosopher (Fragpipe) output to MSstatsTMT format.

Description

Convert Philosopher (Fragpipe) output to MSstatsTMT format.

Usage

```
PhilosophertoMSstatsTMTFormat(
  input,
  annotation,
  protein_id_col = "Protein",
  peptide_id_col = "Peptide.Sequence",
```

```

Purity_cutoff = 0.6,
PeptideProphet_prob_cutoff = 0.7,
useUniquePeptide = TRUE,
rmPSM_withfewMea_withinRun = TRUE,
rmPeptide_OxidationM = TRUE,
rmProtein_with1Feature = FALSE,
summaryforMultipleRows = sum,
use_log_file = TRUE,
append = FALSE,
verbose = TRUE,
log_file_path = NULL,
...
)

```

Arguments

input	data.frame of msstats.csv file produced by Philosopher
annotation	annotation with Run, Fraction, TechRepMixture, Mixture, Channel, BioReplicate, Condition columns or a path to file. Refer to the example 'annotation' for the meaning of each column. Channel column should be consistent with the channel columns (Ignore the prefix "Channel ") in msstats.csv file. Run column should be consistent with the Spectrum.File columns in msstats.csv file.
protein_id_col	Use 'Protein'(default) column for protein name. 'Master.Protein.Accessions' can be used instead to get the protein ID with single protein.
peptide_id_col	Use 'Peptide.Sequence'(default) column for peptide sequence. 'Modified.Peptide.Sequence' can be used instead to get the modified peptide sequence.
Purity_cutoff	Cutoff for purity. Default is 0.6
PeptideProphet_prob_cutoff	Cutoff for the peptide identification probability. Default is 0.7. The probability is confidence score determined by PeptideProphet and higher values indicate greater confidence.
useUniquePeptide	TRUE (default) removes peptides that are assigned for more than one proteins. We assume to use unique peptide for each protein.
rmPSM_withfewMea_withinRun	TRUE (default) will remove the features that have 1 or 2 measurements within each Run.
rmPeptide_OxidationM	TRUE (default) will remove the peptides including oxidation (M) sequence.
rmProtein_with1Feature	TRUE will remove the proteins which have only 1 peptide and charge. Default is FALSE.
summaryforMultipleRows	max or sum - when multiple PSMs identify the same feature within a single MS run (duplicate PSMs), use the highest (max) or sum of the duplicate intensities. Default is max for label-free converters and sum for TMT converters. Note that this parameter does NOT control collapsing across fractions of the same biological mixture.

<code>use_log_file</code>	logical. If TRUE, information about data processing will be saved to a file.
<code>append</code>	logical. If TRUE, information about data processing will be added to an existing log file.
<code>verbose</code>	logical. If TRUE, information about data processing will be printed to the console.
<code>log_file_path</code>	character. Path to a file to which information about data processing will be saved. If not provided, such a file will be created automatically. If <code>append = TRUE</code> , has to be a valid path to a file.
<code>...</code>	additional parameters to <code>data.table::fread</code> .

Value

data.frame of class MSstatsTMT

Examples

```
input_file_path = system.file("tinytest/raw_data/Philosopher/msstats.csv",
                             package = "MSstatsConvert")
annotation_file_path = system.file("tinytest/raw_data/Philosopher/MSstatsTMT_annotation.csv",
                                   package = "MSstatsConvert")
input = data.table::fread(input_file_path)
annotation = data.table::fread(annotation_file_path)
msstats_format = PhilosopherToMSstatsTMTFormat(input, annotation)
head(msstats_format)
```

ProgenisistoMSstatsFormat

Import Progenis files

Description

Import Progenis files

Usage

```
ProgenisistoMSstatsFormat(
  input,
  annotation,
  useUniquePeptide = TRUE,
  summaryforMultipleRows = max,
  removeFewMeasurements = TRUE,
  removeOxidationMpeptides = FALSE,
  removeProtein_with1Peptide = FALSE,
  use_log_file = TRUE,
  append = FALSE,
  verbose = TRUE,
```

```

    log_file_path = NULL,
    ...
)

```

Arguments

input	name of Progenesis output, which is wide-format. 'Accession', 'Sequence', 'Modification', 'Charge' and one column for each run are required.
annotation	name of 'annotation.txt' or 'annotation.csv' data which includes Condition, BioReplicate, Run information. It will be matched with the column name of input for MS runs.
useUniquePeptide	TRUE (default) removes peptides that are assigned for more than one proteins. We assume to use unique peptide for each protein.
summaryforMultipleRows	max or sum - when multiple PSMs identify the same feature within a single MS run (duplicate PSMs), use the highest (max) or sum of the duplicate intensities. Default is max for label-free converters and sum for TMT converters. Note that this parameter does NOT control collapsing across fractions of the same biological mixture.
removeFewMeasurements	TRUE (default) will remove the features that have 1 or 2 measurements across runs.
removeOxidationMpeptides	TRUE will remove the peptides including 'oxidation (M)' in modification. FALSE is default.
removeProtein_with1Peptide	TRUE will remove the proteins which have only 1 peptide and charge. FALSE is default.
use_log_file	logical. If TRUE, information about data processing will be saved to a file.
append	logical. If TRUE, information about data processing will be added to an existing log file.
verbose	logical. If TRUE, information about data processing will be printed to the console.
log_file_path	character. Path to a file to which information about data processing will be saved. If not provided, such a file will be created automatically. If append = TRUE, has to be a valid path to a file.
...	additional parameters to <code>data.table::fread</code> .

Value

data.frame in the MSstats required format.

Author(s)

Meena Choi, Olga Vitek, Ulrich Omasits

Examples

```

progenesis_raw = system.file("tinytest/raw_data/Progenesis/progenesis_input.csv",
                             package = "MSstatsConvert")
annot = system.file("tinytest/raw_data/Progenesis/progenesis_annot.csv",
                    package = "MSstatsConvert")
progenesis_raw = data.table::fread(progenesis_raw)
annot = data.table::fread(annot)

progenesis_imported = ProgenesisToMSstatsFormat(progenesis_raw, annot,
                                                use_log_file = FALSE)

head(progenesis_imported)

```

ProteinProspectortoMSstatsTMTFormat

Generate MSstatsTMT required input format from Protein Prospector output

Description

Generate MSstatsTMT required input format from Protein Prospector output

Usage

```

ProteinProspectortoMSstatsTMTFormat(
  input,
  annotation,
  useUniquePeptide = TRUE,
  removeFewMeasurements = TRUE,
  removeProtein_with1Feature = FALSE,
  summaryforMultipleRows = sum,
  use_log_file = TRUE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL
)

```

Arguments

input	Input txt peptide report file from Protein Prospector with "Keep Replicates", "Mods in Peptide", and "Protein Mods" options selected.
annotation	data frame which contains column Run, Fraction, TechRepMixture, Mixture, Channel, BioReplicate, Condition.
useUniquePeptide	TRUE (default) removes peptides that are assigned for more than one proteins. We assume to use unique peptide for each protein.

removeFewMeasurements	TRUE (default) will remove the features that have 1 or 2 measurements across runs.
removeProtein_with1Feature	TRUE will remove the proteins which have only 1 feature, which is the combination of peptide, precursor charge, fragment and charge. FALSE is default.
summaryforMultipleRows	max or sum - when multiple PSMs identify the same feature within a single MS run (duplicate PSMs), use the highest (max) or sum of the duplicate intensities. Default is max for label-free converters and sum for TMT converters. Note that this parameter does NOT control collapsing across fractions of the same biological mixture.
use_log_file	logical. If TRUE, information about data processing will be saved to a file.
append	logical. If TRUE, information about data processing will be added to an existing log file.
verbose	logical. If TRUE, information about data processing will be printed to the console.
log_file_path	character. Path to a file to which information about data processing will be saved. If not provided, such a file will be created automatically. If append = TRUE, has to be a valid path to a file.

Value

data.frame of class "MSstatsTMT"

Examples

```
input = system.file("tinytest/raw_data/ProteinProspector/Prospector_TotalTMT.txt",
  package = "MSstatsConvert")
input = data.table::fread(input)
annot = system.file("tinytest/raw_data/ProteinProspector/Annotation.csv",
  package = "MSstatsConvert")
annot = data.table::fread(annot)
output <- ProteinProspectortoMSstatsTMTFormat(input, annot)
head(output)
```

SkylinetoMSstatsFormat

Import Skyline files

Description

Import Skyline files

Usage

```

SkylinetoMSstatsFormat(
  input,
  annotation = NULL,
  removeiRT = TRUE,
  filter_with_Qvalue = TRUE,
  qvalue_cutoff = 0.01,
  useUniquePeptide = TRUE,
  removeFewMeasurements = TRUE,
  removeOxidationMpeptides = FALSE,
  removeProtein_with1Feature = FALSE,
  use_log_file = TRUE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL,
  ...
)

```

Arguments

input	name of MSstats input report from Skyline, which includes feature-level data.
annotation	name of 'annotation.txt' data which includes Condition, BioReplicate, Run. If annotation is already complete in Skyline, use annotation=NULL (default). It will use the annotation information from input.
removeiRT	TRUE (default) will remove the proteins or peptides which are labeled 'iRT' in 'StandardType' column. FALSE will keep them.
filter_with_Qvalue	TRUE(default) will filter out the intensities that have greater than qvalue_cutoff in DetectionQValue column. Those intensities will be replaced with zero and will be considered as censored missing values for imputation purpose.
qvalue_cutoff	Cutoff for DetectionQValue. default is 0.01.
useUniquePeptide	TRUE (default) removes peptides that are assigned for more than one proteins. We assume to use unique peptide for each protein.
removeFewMeasurements	TRUE (default) will remove the features that have 1 or 2 measurements across runs.
removeOxidationMpeptides	TRUE will remove the peptides including 'oxidation (M)' in modification. FALSE is default.
removeProtein_with1Feature	TRUE will remove the proteins which have only 1 feature, which is the combination of peptide, precursor charge, fragment and charge. FALSE is default.
use_log_file	logical. If TRUE, information about data processing will be saved to a file.
append	logical. If TRUE, information about data processing will be added to an existing log file.

verbose	logical. If TRUE, information about data processing will be printed to the console.
log_file_path	character. Path to a file to which information about data processing will be saved. If not provided, such a file will be created automatically. If append = TRUE, has to be a valid path to a file.
...	additional parameters to <code>data.table::fread</code> .

Value

data.frame in the MSstats required format.

Author(s)

Meena Choi, Olga Vitek

Examples

```
skyline_raw = system.file("tinytest/raw_data/Skyline/skyline_input.csv",
                           package = "MSstatsConvert")
skyline_raw = data.table::fread(skyline_raw)
skyline_imported = SkylineToMSstatsFormat(skyline_raw)
head(skyline_imported)
```

SpectroMineToMSstatsTMTFormat

Import data from SpectroMine

Description

Import data from SpectroMine

Usage

```
SpectroMineToMSstatsTMTFormat(
  input,
  annotation,
  filter_with_Qvalue = TRUE,
  qvalue_cutoff = 0.01,
  useUniquePeptide = TRUE,
  rmPSM_withfewMea_withinRun = TRUE,
  rmProtein_with1Feature = FALSE,
  summaryforMultipleRows = sum,
  use_log_file = TRUE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL,
  ...
)
```

Arguments

<code>input</code>	data name of SpectroMine PSM output. Read PSM sheet.
<code>annotation</code>	data frame which contains column Run, Fraction, TechRepMixture, Mixture, Channel, BioReplicate, Condition. Refer to the example 'annotation.mine' for the meaning of each column.
<code>filter_with_Qvalue</code>	TRUE(default) will filter out the intensities that have greater than <code>qvalue_cutoff</code> in EG.Qvalue column. Those intensities will be replaced with NA and will be considered as censored missing values for imputation purpose.
<code>qvalue_cutoff</code>	Cutoff for EG.Qvalue. default is 0.01.
<code>useUniquePeptide</code>	TRUE (default) removes peptides that are assigned for more than one proteins. We assume to use unique peptide for each protein.
<code>rmPSM_withfewMea_withinRun</code>	TRUE (default) will remove the features that have 1 or 2 measurements within each Run.
<code>rmProtein_with1Feature</code>	TRUE will remove the proteins which have only 1 peptide and charge. Default is FALSE.
<code>summaryforMultipleRows</code>	max or sum - when multiple PSMs identify the same feature within a single MS run (duplicate PSMs), use the highest (max) or sum of the duplicate intensities. Default is max for label-free converters and sum for TMT converters. Note that this parameter does NOT control collapsing across fractions of the same biological mixture.
<code>use_log_file</code>	logical. If TRUE, information about data processing will be saved to a file.
<code>append</code>	logical. If TRUE, information about data processing will be added to an existing log file.
<code>verbose</code>	logical. If TRUE, information about data processing will be printed to the console.
<code>log_file_path</code>	character. Path to a file to which information about data processing will be saved. If not provided, such a file will be created automatically. If <code>append = TRUE</code> , has to be a valid path to a file.
<code>...</code>	additional parameters to <code>data.table::fread</code> .

Value

data.frame of class MSstatsTMT

Examples

```
raw.mine = data.table::fread(system.file("tinytest/raw_data/SpectroMine/spectromine_input.csv",
                                         package = "MSstatsConvert"))
annotation.mine = data.table::fread(system.file("tinytest/raw_data/SpectroMine/spectromine_annotation.csv",
                                                package = "MSstatsConvert"))
head(raw.mine)
```

```
head(annotation.mine)
input.mine <- SpectroMinetoMSstatsTMTFormat(raw.mine, annotation.mine)
head(input.mine)
```

SpectronauttoMSstatsFormat

Import Spectronaut files

Description

Import Spectronaut files

Usage

```
SpectronauttoMSstatsFormat(
  input,
  annotation = NULL,
  intensity = "PeakArea",
  peptideSequenceColumn = "EG.ModifiedSequence",
  heavyLabels = NULL,
  excludedFromQuantificationFilter = TRUE,
  filter_with_Qvalue = FALSE,
  qvalue_cutoff = 0.01,
  useUniquePeptide = TRUE,
  removeFewMeasurements = TRUE,
  removeProtein_with1Feature = FALSE,
  summaryforMultipleRows = max,
  calculateAnomalyScores = FALSE,
  anomalyModelFeatures = c(),
  anomalyModelFeatureTemporal = c(),
  removeMissingFeatures = 0.5,
  anomalyModelFeatureCount = 100,
  runOrder = NULL,
  n_trees = 100,
  max_depth = "auto",
  numberOfCores = 1,
  use_log_file = TRUE,
  append = FALSE,
  verbose = TRUE,
  log_file_path = NULL,
  ...
)
```

Arguments

input	name of Spectronaut output, which is long-format. ProteinName, PeptideSequence, PrecursorCharge, FragmentIon, ProductCharge, IsotopeLabelType, Condition, BioReplicate, Run, Intensity, F.ExcludedFromQuantification are required. Rows with F.ExcludedFromQuantification=True will be removed.
annotation	name of 'annotation.txt' data which includes Condition, BioReplicate, Run. If annotation is already complete in Spectronaut, use annotation=NULL (default). It will use the annotation information from input.
intensity	Intensity column to use. Accepts legacy enum values 'PeakArea' (default, uses F.PeakArea), 'NormalizedPeakArea' (uses F.NormalizedPeakArea). Can also be any raw Spectronaut column name passed as a string (e.g. "FG.MS1Quantity"); the column name is standardized internally. For protein turnover workflows the recommended default is "FG.MS1Quantity".
peptideSequenceColumn	Name of the Spectronaut column that contains the peptide sequence. Defaults to "EG.ModifiedSequence". The value is standardized internally (dots and spaces removed) before column lookup.
heavyLabels	Character vector naming each labeled residue and its heavy label as they appear in the peptide sequence column: the single-letter amino acid code, then the label in square brackets. For example "K[Lys6]", or c("K[Lys6]", "R[Arg10]") for a double-label experiment. Any label name Spectronaut reports is accepted. The residue letter is required, because it is what tells MSstats which peptides could have carried a label. Peptides carrying the label are marked heavy (IsotopeLabelType = "H"), those that could carry it but do not are light ("L"), and those with no labelable residue are NA. A peptide with two or more labelable residues (counted across all labels supplied) may be only partially labeled, which the two-state turnover model cannot represent, so it is dropped and the number removed is reported. Defaults to NULL: turnover mode off, every peptide marked light.
excludedFromQuantificationFilter	Remove rows with F.ExcludedFromQuantification=TRUE Default is TRUE.
filter_with_Qvalue	FALSE(default) will not perform any filtering. TRUE will filter out the intensities that have greater than qvalue_cutoff in EG.Qvalue column. Those intensities will be replaced with zero and will be considered as censored missing values for imputation purpose.
qvalue_cutoff	Cutoff for EG.Qvalue. default is 0.01.
useUniquePeptide	TRUE (default) removes peptides that are assigned for more than one proteins. We assume to use unique peptide for each protein.
removeFewMeasurements	TRUE (default) will remove the features that have 1 or 2 measurements across runs.
removeProtein_with1Feature	TRUE will remove the proteins which have only 1 feature, which is the combination of peptide, precursor charge, fragment and charge. FALSE is default.

summaryforMultipleRows	max or sum - when multiple PSMs identify the same feature within a single MS run (duplicate PSMs), use the highest (max) or sum of the duplicate intensities. Default is max for label-free converters and sum for TMT converters. Note that this parameter does NOT control collapsing across fractions of the same biological mixture.
calculateAnomalyScores	Default is FALSE. If TRUE, will run anomaly detection model and calculate anomaly scores for each feature. Used downstream to weigh measurements in differential analysis.
anomalyModelFeatures	character vector of quality metric column names to be used as features in the anomaly detection model. List must not be empty if calculateAnomalyScores=TRUE.
anomalyModelFeatureTemporal	character vector of temporal direction corresponding to columns passed to anomalyModelFeatures. Values must be one of: mean_decrease, mean_increase, dispersion_increase, or NULL (to perform no temporal feature engineering). Default is empty vector. If calculateAnomalyScores=TRUE, vector must have as many values as anomalyModelFeatures (even if all NULL).
removeMissingFeatures	Remove features with missing values in more than this fraction of runs. Default is 0.5. Only used if calculateAnomalyScores=TRUE.
anomalyModelFeatureCount	Feature selection for anomaly model. Anomaly detection works on the precursor-level and can be much slower if all features used. We will by default filter to the top-100 highest intensity features. This can be adjusted as necessary. To turn feature-selection off, set this value to a high number (e.g. 10000). Only used if calculateAnomalyScores=TRUE.
runOrder	Temporal order of MS runs. Should be a two column data.table with columns Run and Order, where Run matches the run name output by Spectronaut and Order is an integer. Used to engineer the temporal features defined in anomalyModelFeatureTemporal.
n_trees	Number of trees to use in isolation forest when calculateAnomalyScores=TRUE. Default is 100.
max_depth	Max tree depth to use in isolation forest when calculateAnomalyScores=TRUE. Default is "auto" which calculates depth as $\log_2(N)$ where N is the number of runs. Otherwise must be an integer.
numberOfCores	Number of cores for parallel processing anomaly detection model. When > 1, a logfile named 'MSstats_anomaly_model_progress.log' is created to track progress. Only works for Linux & Mac OS. Default is 1.
use_log_file	logical. If TRUE, information about data processing will be saved to a file.
append	logical. If TRUE, information about data processing will be added to an existing log file.
verbose	logical. If TRUE, information about data processing will be printed to the console.

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

... additional parameters to `data.table::fread`.

Value

data.frame in the MSstats required format.

Author(s)

Meena Choi, Olga Vitek

Examples

```
spectronaut_raw = system.file("tinytest/raw_data/Spectronaut/spectronaut_input.csv",  
                             package = "MSstatsConvert")  
spectronaut_raw = data.table::fread(spectronaut_raw)  
spectronaut_imported = SpectronauttoMSstatsFormat(spectronaut_raw, use_log_file = FALSE)  
head(spectronaut_imported)
```

Index

.cleanRawPD, [3](#)

as.data.frame.MSstatsValidated, [4](#)
as.data.table.MSstatsValidated, [4](#)

CheckDataHealth, [5](#)

DIANNtoMSstatsFormat, [5](#)
DIAUmpiretoMSstatsFormat, [9](#)

FragPipetoMSstatsFormat, [11](#)

getInputFile, [12](#)
getInputFile,MSstatsInputFiles-method
(getInputFile), [12](#)
getInputFile,MSstatsPhilosopherFiles-method
(getInputFile), [12](#)

MaxQtoMSstatsFormat, [13](#)
MaxQtoMSstatsTMTFormat, [15](#)
MetamorpheusToMSstatsFormat, [17](#)
MSstatsAnomalyScores, [19](#)
MSstatsBalancedDesign, [20](#)
MSstatsClean, [21](#)
MSstatsClean,MSstatsDIANNFiles-method
(MSstatsClean), [21](#)
MSstatsClean,MSstatsDIAUmpireFiles-method
(MSstatsClean), [21](#)
MSstatsClean,MSstatsMaxQuantFiles-method
(MSstatsClean), [21](#)
MSstatsClean,MSstatsMetamorpheusFiles-method
(MSstatsClean), [21](#)
MSstatsClean,MSstatsMZMineFiles-method
(MSstatsClean), [21](#)
MSstatsClean,MSstatsOpenMSFiles-method
(MSstatsClean), [21](#)
MSstatsClean,MSstatsOpenSWATHFiles-method
(MSstatsClean), [21](#)
MSstatsClean,MSstatsPhilosopherFiles-method
(MSstatsClean), [21](#)
MSstatsClean,MSstatsProgenesisFiles-method
(MSstatsClean), [21](#)
MSstatsClean,MSstatsProteinProspectorFiles-method
(MSstatsClean), [21](#)
MSstatsClean,MSstatsProteomeDiscovererFiles-method
(MSstatsClean), [21](#)
MSstatsClean,MSstatsSkylineFiles-method
(MSstatsClean), [21](#)
MSstatsClean,MSstatsSpectroMineFiles-method
(MSstatsClean), [21](#)
MSstatsClean,MSstatsSpectronautFiles-method
(MSstatsClean), [21](#)
MSstatsImport, [26](#)
MSstatsLogsSettings, [27](#)
MSstatsMakeAnnotation, [28](#)
MSstatsPreprocess, [29](#)
MSstatsSaveSessionInfo, [31](#)
MZMinetoMSstatsFormat, [32](#)

OpenMStoMSstatsFormat, [34](#)
OpenMStoMSstatsTMTFormat, [36](#)
OpenSWATHtoMSstatsFormat, [37](#)

PDtoMSstatsFormat, [39](#)
PDtoMSstatsTMTFormat, [41](#)
PhilosophertoMSstatsTMTFormat, [42](#)
ProgenesistoMSstatsFormat, [44](#)
ProteinProspectortoMSstatsTMTFormat,
[46](#)

SkylinetoMSstatsFormat, [47](#)
SpectroMinetoMSstatsTMTFormat, [49](#)
SpectronauttoMSstatsFormat, [51](#)