Package 'MSeasy'

May 3, 2012

Type Package

Title Preprocessing of Gas Chromatography-Mass Spectrometry (GC-MS) data

Version 5.3

Date 2012-04-20

Depends amap, clValid, cluster, fpc

Suggests xcms, tcltk

Author Elodie Courtois, Yann Guitton, Florence Nicole

Maintainer <florence.nicole@univ-st-etienne.fr><yann.guitton@univ-lille1.fr>

Description Package for the detection of molecules in complex mixtures of compounds. It creates an initial_DATA matrix from several GC-MS analyses by collecting and assembling the information from chromatograms and mass spectra (\{ }emph{MS.DataCreation}).It tests for the best unsupervised clustering method to group similar mass spectra into molecules (\{ }emph{MS.test.clust}).It runs the optimal unsupervised clustering method on the initial_DATA matrix, identifies the optimal number of clusters, produces different files for facilitating the quality control and identification of putative molecules, and returns fingerprinting or profiling matrices (\{ }emph{MS.clust}).It converts output files from \{ }emph{MS.clust} for NIST mass spectral library search and ARISTO webtool search

License GPL-2

URL http://sites.google.com/site/rpackagemseasy/

LazyLoad yes

R topics documented:

`MSeasy-package` *Unsupervised and untargeted processing of Gas Chromatography-Mass Spectrometry (GC-MS) data*

Description

Package for the detection of molecules in complex mixtures of compounds. It creates an initial_DATA matrix from several GC-MS analyses by collecting and assembling the information from chromatograms and mass spectra (*MS.DataCreation*). It tests for the best unsupervised clustering method to group similar mass spectra into molecules (*MS.test.clust*). It runs the optimal unsupervised clustering method on the initial_DATA matrix, identifies the optimal number of clusters, produces different files for facilitating the quality control and identification of putative molecules, and returns fingerprinting or profiling matrices (*MS.clust*). It converts output files from *MS.clust* for NIST mass spectral library search and ARISTO webtool search

Details

Package:	MSeasy
Type:	Package
Version:	5.3
Date:	2012-04-12
License:	GPL-2
LazyLoad:	yes

Author(s)

Elodie Courtois, Yann Guitton, Florence Nicole

`Agilent_MSDataCreation` *Demonstration folder for MS.DataCreation with option DataType=Agilent*

Description

This demonstration folder includes 2 GC-MS analyses of Lavandula obtained from Agilent. The two analyses represent a total of 54 chromatogram's peaks. The folder can be used with the function MS.DataCreation that collects and assembles the information from chromatograms and mass spectra of the two samples in a initial data matrix with peaks in row and mass spectrum in columns.

Usage

```
Agilent_MSDataCreation
```

Format

A folder with two different sub-folders, each corresponding to one GC-MS analysis. Each sub-folder contains an netCDF file (mass spectra) and a rteres file (chromatogram).

Examples

```
data(Agilent_MSDataCreation)
```

Agilent_quantF_MSclust

Demonstration dataset for MS.clust

Description

This demonstration dataset includes 2 GC-MS analyses of Lavandula, representing a total of 54 chromatogram's peaks. The file was created with MS.DataCreation (option quant=FALSE) from Agilent data. It can be used with the function MS.clust:

(i) to identify the optimal number of clusters.

(ii) to obtain the fingerprinting matrix (absence or presence of peaks for all samples)

Usage

```
data(Agilent_quantF_MSclust)
```

Format

A data frame with 54 chromatogram's peaks from 2 GC-MS analyses.

- `header line` the first row contains columns' names
- `first column` name of the sample/analysis
- `second column` retention time of the peak
- `following columns` mean relative mass spectrum of the peak (the intensity of one mass fragment (m/z) per column; Mean mass spectrum calculated by averaging 5 percent of the mass spectra surrounding the apex; The intensity of each mass fragment is transformed to a relative percentage of the highest mass fragment per spectrum)

Examples

```
data(Agilent_quantF_MSclust)
```

Agilent_quantT_MSclust *Demonstration dataset for MS.clust*

Description

This demonstration dataset includes 2 GC-MS analyses of Lavandula, representing a total of 54 chromatogram's peaks. The file was created with MS.DataCreation (option quant=TRUE) from Agilent data. It can be used with the function MS.clust:

(i) to identify the optimal number of clusters.

(ii) to obtain two profiling matrices, one with the corrected peak area and one with the percent of the total corrected area

Usage

```
data(Agilent_quantT_MSclust)
```

Format

A data frame with 54 chromatogram's peaks from 2 GC-MS analyses.

- `header line` the first row contains columns' names
- `first column` name of the sample/analysis
- `second column` retention time of the peak
- `third column` corrected peak area (corrArea)
- `fourth column` percent of the total corrected area (PercTotal)
- `following columns` mean relative mass spectrum of the peak (the intensity of one mass fragment (m/z) per column; Mean mass spectrum calculated by averaging 5 percent of the mass spectra surrounding the apex; The intensity of each mass fragment is transformed to a relative percentage of the highest mass fragment per spectrum)

Examples

```
data(Agilent_quantT_MSclust)
```

ASCII_MSclust *Demonstration dataset for MS.clust*

Description

This demonstration dataset includes 2 GC-MS analyses of Petrel, representing a total of 67 chromatogram's peaks. It can be used with the function MS.clust:

(i) to identify the optimal number of clusters.

(ii) to obtain the fingerprinting matrix (absence or presence of peaks for all samples)

Usage

```
data(ASCII_MSclust)
```

Format

A data frame with 67 chromatogram's peaks from 2 GC-MS analyses.

- `header line` the first row contains columns' names
- `first column` name of the sample/analysis
- `second column` retention time of the peak
- `following columns` mean relative mass spectrum of the peak (the intensity of one mass fragment (m/z) per column; Mean mass spectrum calculated by averaging 5 percent of the mass spectra surrounding the apex; The intensity of each mass fragment is transformed to a relative percentage of the highest mass fragment per spectrum)

Examples

```
data(ASCII_MSclust)
```

ASCII_MSDataCreation *Demonstration folder for MS.DataCreation with option DataType=ASCII*

Description

This demonstration folder includes 2 transformed GC-MS analyses of Petrel obtained from trans.ASCII. The two analyses represent a total of 67 chromatogram's peaks. The folder can be used with the function MS.DataCreation that collects and assembles the information from chromatograms and mass spectra of the two samples in a initial data matrix with peaks in row and mass spectrum in columns.

Usage

```
ASCII_MSDataCreation
```

Format

A folder with two different transformed ascii files, each corresponding to one GC-MS analysis.

Examples

```
data(ASCII_MSDataCreation)
```

ASCII_TransASCII *Demonstration folder for trans.ASCII*

Description

This demonstration folder includes 2 raw GC-MS analyses of Petrel in ASCII format. The data in ASCII format have to be transformed with the function trans.ASCII for further analyses with MS.DataCreation. The folder can be used with the function trans.ASCII to transform the raw ascii GC-MS data in the format suitable for MS.DataCreation.

Usage

```
ASCII_TransASCII
```

Format

A folder with two different raw ascii files corresponding to the two different GC-MS analyses.

Examples

```
data(ASCII_TransASCII)
```

Data_testclust *Demonstration dataset for MS.test.clust*

Description

To test for the best unsupervised clustering method, a dataset where molecules are already identified is created. Each molecule is represented by several samples mass spectrum. Here, the dataset contains 10 molecules obtained in different samples (84 Lavandula GC-MS analyses). In the function MS.test.clust, different clustering methods are tested for their abilities to find the correct structure of the dataset. Three different cluster validity indices are calculated to evaluate the results: the matching coefficient, the silhouette width and the Dunn index (see MS.test.clust for details)

Usage

```
data(Data_testclust)
```

Format

A data frame with 10 molecules from 84 GC-MS analyses.

- `header line` the first row must contains the columns' names
- `first column` name of the molecule
- `second column` sample name
- `third column` retention time
- `following columns` mean relative mass spectrum of the molecule (the intensity of one mass fragment (m/z) per column; Mean mass spectrum calculated by averaging 5 percent of the mass spectra surrounding the apex; The intensity of each mass fragment is transformed to a relative percentage of the highest mass fragment per spectrum)

Examples

```
data(Data_testclust)
```

`MS.clust` *Mass spectra clustering and creation of a fingerprinting or profiling matrix*

Description

MS.clust runs unsupervised clustering methods on mass spectra. It can identify the optimal number of clusters using a cluster validity index (silhouette width), produces different files for facilitating the quality control and identification of putative molecules within a complex dataset of numerous mass spectra, and returns a fingerprinting or profiling matrix for homogeneous clusters (see details below for the definition of homogeneous clusters).

Usage

```
MS.clust(data_tot, quant=FALSE, clV, ncmin, ncmax, Nbc, varRT = 0.1,
disMeth="euclidean", linkMeth="ward", clustMeth="hierarchical")
```

Arguments

`data_tot`	R object data frame as returned by *MS.DataCreation* (`initial_DATA.txt`), or a *user made file* (.txt, .csv...) with the first row containing columns' names; first column contains sample/analysis name; second column contains retention time of the peak (or retention index); optionally third and fourth columns may contain quantitative measures of peak size (height, width or area; For Agilent, columns 3 and 4 contain respectively corrected peak area and percent of the total corrected area), and following columns contains the mean relative mass spectrum (the intensity of one mass fragment (m/z) per column; each mass fragment is transformed to a relative percentage of the highest mass fragment per spectrum)
`clV`	TRUE indicates that the function clValid will be used to identify the optimal number of clusters. FALSE, when the number of clusters is already known, escapes the clValid step and goes directly to the clustering.
`ncmin`	If clV = TRUE, a numeric value giving the minimum number of clusters to be evaluated.
`ncmax`	If clV = TRUE, a numeric value giving the maximum number of clusters to be evaluated.
`Nbc`	If clV = FALSE, a numeric vector giving the number(s) of clusters to be evaluated. e.g., Nbc=c(20,25) would evaluate the number of clusters 20 and 25.
`varRT`	range of RT or RI to define homogeneous clusters, i.e. the accepted range of variation of RT/RI for a given molecule. Default value is set to 0.1. If the varRT is baseless (analyses from different GC columns for example), set the varRT to a high value.
`clustMeth`	A character vector giving the clustering methods. Available options are *hierarchical*(Default), *diana*, *kmeans* and *pam*
`disMeth`	The metric used to determine the distance matrix. Possible choices are *euclidean*(Default), *manhattan*, and *correlation*. For pam and diana, only euclidean and manhattan are available.

`linkMeth` For hierarchical clustering, the agglomeration method used. Available choices are *ward*(Default), *single*, *complete*, *centroid* and *average*. For all others `clustMeth`, `linkMeth=NULL`

`quant` `TRUE` only if option quant=TRUE was chosen in MS.DataCreation and/or if columns 3 or 4 of the input file contains one or two quantitative measures of the peak size. For Agilent Technologies, corrected peak area (CorrArea) is reported in column 3 and percent of the total corrected area (PercTot) is reported in column 4. CorrArea is used for absolute quantification when associated with the use of external and/or internal standards. PercTot is used for relative quantification (no external or internal standard needed). This option generates two distinct profiling matrices in outfiles, one with quantification1 (column 3) and one quantification2 (column 4). `FALSE` if these two columns are absent. Then, a fingerprinting matrix (absence or presence of each molecule) is generated

Format

- `header line` the first row must contains column headings
- `first column` name of the sample/analysis
- `second column` retention time (RT) of the peak (or retention index (RI))
- *optionally* `third column` quantification1 (For Agilent, corrected peak area)
- *optionally* `fourth column` quantification2 (For Agilent, percent of the total corrected area)
- `following columns` relative mass spectrum of the peak (mass spectrum at the apex or obtained by averaging 5 percent of the mass spectra surrounding the apex; Each mass fragment is transformed to a relative percentage of the highest mass fragment per spectrum); The intensity of one mass fragment (m/z) per column

Details

MS.clust runs several unsupervised clustering methods on a dataset composed of numerous mass spectra from different samples/analyses. When the total number of molecules in the dataset is unknown, MS.clust can first identify the optimal number of clusters with a cluster validity index (silhouette width) after running the clustering on a range of numbers of clusters (clValid procedure, `clV=TRUE`).

A graphic window displays the mean silhouette width as a function of the number of clusters. A red line indicates the optimal number of clusters with the highest silhouette width. The values of silhouette width for the different numbers of clusters are summarized in a table named *res_clValid.txt* and saved in a folder called *Output_resultdate_time*.

Following the graphic display, the user is asked *How many clustering separations ?*, i.e. how many times should the dataset be cut into clusters. The answer is an integer. If the graph indicates a unique and clear optimal at the apex of a curve, only one cut at the optimal number of clusters is expected. If the graph display an optimal value located on a plateau, the user might be interested to perform different cut, one at the optimal number of clusters together with one at the minimum and one at the maximum numbers of clusters delimiting the plateau.

Afterward, the user is asked *How many clusters?* The answer is an integer. If several values, each integer should be entered and followed by Enter key. When the number(s) of clusters to be analyzed is defined, the clustering is performed. The arguments of clustMeth includes hierarchical, diana, kmeans and pam. For disMeth and linkMeth, the arguments are similar to those of the clValid package. See arguments below and the documentation of this package for more details. Following the clustering, the function identifies homogeneous and inhomogeneous clusters. The homogeneous

clusters are defined by a variation in retention time lower than varRT (0.1 by default). Homogeneous clusters may correspond to a well-defined molecule, with clear mass spectra. Inhomogeneous clusters usually need manual investigations to be further classified as molecules.

Value

MS.clust produces different files in folder *Output_MSclust_resultdate_time* for facilitating the quality control of putative molecules within a dataset composed of numerous mass spectra:

`Output_cluster.txt`
: contains summary information on clusters. In column, the number of the cluster, quality of the cluster based on the variation of retention time (0 if inhomogeneous, 1 if homogeneous), number of distinct individuals within the cluster and total number of peaks in the cluster (to check for unique occurence of each given analysis in the cluster), mean retention time (RT), range of retention time (max(RT)-min(RT)), mean silhouette width. Follow the 8 highest mass fragments (m/z) and the complete mean relative mass spectrum.

`Output_peak.txt`
: contains detailed information for each peak. In column, the number of the cluster, the sample name, the retention time, the silhouette width, the neighbor cluster, *optionally* if `quant=TRUE` corrArea and PercTotal, the 8 highest mass fragments and the complete mean relative mass spectrum.

`Hist_cluster_ok_RT.pdf`
: a pdf file displaying the histogram of the distribution of retention times for each homogeneous cluster.

`Hist_cluster_ok_silhouette.pdf`
: a pdf file displaying the histrogram of the distribution of silhouette width for each homogeneous cluster.

`Hist_cluster_problem_RT.pdf`
: a pdf file displaying the histrogram of the distribution of retention times for each inhomogeneous cluster.

`Hist_cluster_problem_silhouette.pdf`
: a pdf file displaying the histrogram of the distribution of silhouette width for each inhomogeneous cluster.

Depending on the quant option

`Output_fingerprintingmatrix.txt`
: a fingerprinting matrix (0 for absence, 1 for presence) with samples' names in the first column, retention time in the second column and presence or absence for homogeneous clusters in the following columns.

or

`Output_profilingmatrix_quantification1.txt`
: a profiling matrix (0 for absence, quantification 1 if present) with samples' names in the first column, retention time in the second column and corrected area for homogeneous clusters in the following columns.

`Output_profilingmatrix_quantification2.txt`
: a profiling matrix (0 for absence, quantification 2 if present) with samples' names in the first column, retention time in the second column and percent of the total corrected area for homogeneous clusters in the following columns.

Author(s)

Elodie Courtois, Yann Guitton, Florence Nicole

See Also

cluster, kohonen, class, mclust, amap, ClValid, fpc, flexmix

Examples

```
data(Agilent_quantT_MSclust)
MS.clust(Agilent_quantT_MSclust, quant=TRUE, clV=TRUE, ncmin=10, ncmax=50,
 varRT = 0.1, disMeth="euclidean", linkMeth="ward", clustMeth="hierarchical")
1
21
## 21 clusters have been determined as the optimal number of cluters.
##with the option quant=TRUE, generate profiling matrices in output

data(Agilent_quantF_MSclust)
MS.clust(Agilent_quantF_MSclust, quant=FALSE, clV=FALSE, Nbc=21,
varRT = 0.1, disMeth="euclidean", linkMeth="ward", clustMeth="hierarchical")
##with clV=FALSE, if you already know the number of molecules in the dataset
##with the option quant=FALSE, generate a fingerprinting matrix in output

data(ASCII_MSclust)
MS.clust(ASCII_MSclust, quant=FALSE, clV=TRUE, ncmin=10, ncmax=50,
varRT = 0.1, disMeth="euclidean", linkMeth=NULL, clustMeth="kmeans")
3
26
28
30
## output files are generated for three different numbers of clusters.
## with 3 as the number of clustering separations
## 26 # First number of clusters
## 28 # Second number of clusters
## 30 # Third number of clusters
```

`MS.DataCreation` *an initial data from GC-MS analyses by collecting and assembling the information from chromatograms and mass spectra*

Description

This function constructs a global matrix called *initial_DATA.txt* by collecting and assembling the information from chromatograms and mass spectra from several GC-MS analyses. It performs basic peak detection if the input file is in ASCII format. For other input files, peak retention times (or retention indices) are retrieved from the chromatograms (peaklist.txt or rteres.txt files) and associated to their respective mass spectrum (AIA/ANDI NetCDF, mzXML, mzData and mzML files). Each row of the output matrix represents one peak in one analysis and reports the sample name in first column, the peak retention time (or retention index) in second column and the mass spectrum of the peak in the following columns. If the input file is in Agilent format, two quantification measures of peak size can be extracted directly from rteres.txt: corrected area is then inserted in column 3 and

percent of the total corrected area is placed in column 4 of *initial_DATA.txt*. If the input file is CDF, one or two quantification measures of peak size can be extracted from column 6 (quantification1) and 7 (quantification2) of peaklist.txt; values are then reported respectively in column 3 and 4 of *initial_DATA.txt*. Except for ASCII, xcms package is needed. Copy paste the following code to download xcms: source("http://bioconductor.org/biocLite.R");biocLite("xcms")

Usage

```
MS.DataCreation(DataType="CDF", path="", pathCDF="", mz, N_filt=3, apex= FALSE, quant = FALSE)
```

Arguments

DataType	Indicate the type of input files: *CDF* (default) when each sample folder contains a mass spectrum in AIA/ANDI NetCDF, mzXML, mzData or mzML format, and a peak list stored in a file named peaklist.txt. *Agilent* when sample folders are obtained with Agilent Technologies machines (extension .D) and contained a peak list stored in rteres.txt file (all .D folders should be grouped in one folder); mass spectra in AIA/ANDI format are grouped in a separate folder. *ASCII* for sample folders as returned by trans.ASCII.
path	If DataType="Agilent", name of the folder containing all the .D folders generated by Agilent Technologies. Each .D folder should contain a rteres.txt file (rteres.txt is the peak list generated by Agilent Technologies for each GC-MS analysis. Default parameters should be used in GC-MS Chemstation software. For each analysis, the name of the .D folder should be identical to the name of the AIA/ANDI file, which is usually the sample name. All .D folders should have different names). If DataType="ASCII", name of the folder *output_date_time* returned by trans.ASCII and containing converted files for each GC-MS analysis initially in ASCII format.
pathCDF	If DataType="Agilent", name of the folder containing the mass spectra of all the GC-MS analyses in AIA/ANDI NetCDF format. If DataType="CDF", name of the folder grouping all the GC-MS analysis folders. For each GC-MS analysis, the folder contains the mass spectrum in AIA/ANDI NetCDF, mzXML, mzData or mzML format, and a peak list stored in a file named peaklist.txt (see details below for the structure of the peak list file. All AIA/ANDI files should have different names).
mz	Range of mass fragments delimiting the mass spectrum, e.g. 30:250. If mz="all" or empty, the range is automatically detected and used to delimit the mass spectrum.
N_filt	Only if DataType="ASCII", N_filt must be informed for chromatogram smoothing before peak detection. For more details about smoothing, please refer to the documentation of the function *filter* with method=*convolution*. If N_filt is lower than 3, there will be no smoothing of the profile. A high N_filt will lower the noise in the chromatogram but can result in the loss of low concentrated peaks.
apex	TRUE indicates that the mass spectrum is considered at the apex of the peak and FALSE (default) indicates that a mean mass spectrum is obtained by averaging 5 percent of the mass spectra surrounding the apex (apex included) for AIA/ANDI NetCDF files, and by averaging the mass spectrum before, the mass spectrum after and the mass spectrum in the apex for ASCII files
quant	If DataType="Agilent" or DataType="CDF", the option quant indicates if quantification measures of peak size should be extracted from the peak list files and added to the initial_DATA matrix. TRUE, if DataType="Agilent", indicates

that the two quantification columns CorrArea (corrected peak area) and PercTot (percent of the total corrected area) are extracted from rteres.txt and added in columns 3 and 4 of the output matrix. Corrected area is used for absolute quantification when associated with the use of external and/or internal standards. Percent of the total corrected area is used for relative quantification (no external or internal standard needed). If `DataType="CDF"`, indicates that one or two columns with quantification measures of peak size (height, width or area) are in columns 6 and 7 of peaklist.txt. The information is extracted and added in column 3 and 4 of the output matrix. This option will allow to generate one or two profiling matrices with quantification for each putative molecule after MS.clust. `FALSE` indicates that quantitative measures are absent or should not be added to the output matrix. Then, a fingerprinting matrix (absence or presence of each putative molecule) will be obtained after MS.clust.

Details

After a GC-MS analysis, different types of files are produced from the chromatograph and the mass spectrometer . Each instrument vendor provide specific proprietary data formats that should be converted to common raw data format such as ANDI NetCDF or mzXML. Most commonly used file formats for mass spectral data, i.e. NetCDF, mzXML and ASCII, are acceptable in MS.DataCreation. Specific proprietary format from Agilent Technologies can also be used directly. Below the detailed structure of the three types of input formats:

(i) DataType=*CDF*. Each GC-MS analysis has its own folder, which contains a mass spectrum in AIA/ANDI NetCDF, mzXML, mzData or mzML format, and a peak list stored in a file named peaklist.txt. Peaklist.txt should have column headings similar to *peak/RT/firstscan/maxscan/lastscan/quantification1/quantific*
The first column contain the peak number, the retention time in minute or second is in the second column, the first scan of the peak is in the third column, the scan at the apex (maxscan) is in column 4, the last scan of the peak is in column 5, and optionally a quantitative measure of peak size (quantifaction1) is in column 6, and another quantitative measures of peak size (quantification2) is in column 7 (only *maxscan* used if `apex=TRUE` in `MS.clust`). The sample name reported in the output matrix is extracted from the name of the AIA/ANDI files. Thus, all AIA/ANDI files should have different names. All analysis folders should be grouped in one folder. The function first checks for the presence of AIA/ANDI and peaklist.txt files, controls if the range of mz is consistent and checks the structure of the peaklist.txt files. In a second time, the function collects the peak's retention time in peaklist.txt and looks for corresponding mass spectra in CDF files. Depending on the Apex option, the mean mass spectrum per each peak is calculated or the mass spectrum at the apex is extracted. The intensity, in counts, of each mass fragment is transformed to a relative percentage of the highest mass fragment per spectrum. If quant = TRUE, one or two quantification columns, quantification1 and quantification2, are extracted for each peak from peaklist.txt and placed respectively in columns 3 and 4 of the output initial_DATA matrix.

(ii) DataType=*Agilent*. For Agilent Technologies providers (using the default parameters): each GC-MS analysis returns a folder .D that contains a file rteres.txt with summary information of the chromatogram (analogous to a peak list). All the analysis folders should have different names and should be grouped in one folder. The mass spectra should be exported in ANDI NetCDF format. These files are automatically generated at once for several selected GC-MS analyses with the Chemstation data analysis software (Menu/File/Export to AIA/ANDI). By default, all CDF files are exported in one folder that may correspond to `pathCDF`. The sample name reported in the output matrix is extracted from the name of the .D folder. Thus, all .D folders should have different names. AIA/ANDI files should have identical name with the corresponding .D folder. The function first checks if all sample folders (.D) within the folder *path* have a file rteres.txt and if in `pathCDF` there are all the CDF files needed. If one file is missing, the analysis stops and indicates the name of the problematic sample. The analysis should be restarted after correction or removal. In a second time,

the function collects the peak's retention time in rteres.txt and looks for corresponding mass spectra in CDF files. Depending on the Apex option, the mean mass spectrum per each peak is calculated or the mass spectrum at the apex is extracted. The intensity, in counts, of each mass fragment is transformed to a relative percentage of the highest mass fragment per spectrum. If quant = TRUE, the two quantification columns CorrArea (corrected peak area) and PercTot (percent of the total corrected area) are extracted for each peak from rteres.txt and placed respectively in columns 3 and 4 of the output initial_DATA matrix.

(iii) DataType=*ASCII*.If your GC-MS raw data have been converted into the international ASCII format, all files (one per GC-MS analysis) should be grouped in one folder and first pass through the trans.ASCII function. The trans.ASCII function generates a folder output_date_time with translated files compatible with MS.DataCreation. This output_date_time file may correspond to *path*. First, a smoothing of chromatogram depending on the option N_filt is performed (see the documentation of the function *filter*, method=*convolution*). Afterwards, peak are detected by the succession of 3 points with increasing intensity directly followed by three points of decreasing intensity (all points should have an intensity higher than 10 kilocounts). The first and last peaks of the chromatogram are removed if incomplete. In a third time, depending on the Apex option, the function calculates the mean mass spectrum per each peak or extracts the mass spectrum at the apex and the intensity (in counts) of each mass fragment is transformed to a relative percentage of the highest mass fragment per spectrum.

The output file called initial_DATA.txt is saved in folder *Output_MSDataCreation_resultdate_time*. It contains the relative mass spectrum of each peak of all samples. The first column contains sample name (the name of the folder containing the GC-MS analysis), the second column is the peak retention time (or retention index) and the following columns correspond to the relative mass spectrum of the peak (within the range of the mass spectrum). If quant = TRUE, the first column contains sample name (the name of the folder containing the GC-MS analysis), the second column is the peak retention time (or retention index), the third column contains quantification 1 (corrected area for Agilent), the fourth column contains quantification 2 (percent of the total corrected area for Agilent) and the following columns correspond to the relative mass spectrum of the peak (within the range of the mass spectrum).

Value

MS.DataCreation returns a data matrix called initial_DATA.txt, saved in folder *Output_MSDataCreation_resultdate_time*
It contains one row per peak and per individual with sample name, retention time (or retention index) and relative mass spectrum. If quant =TRUE, two supplementary columns quantification1 and quantification2 are added after the column retention time. During the analysis, a temporary file called save_list_temp.rda is automatically generated in folder *Output_MSDataCreation_resultdate_time*. It allows recovering temporary informations if the function stopped before ending.

Author(s)

Elodie Courtois, Yann Guitton, Florence Nicole

Examples

```
## Not run:
##not run
## DataType="Agilent"
## require xcms package
## For Agilent Technologies GC-MS files
## two folders are required:one folder with all .D analysis folders, each containing a rteres.txt file
## the second folder contains all CDF or mzXML files.
## CDF files have to be downloaded from MSeasy web site
```

```
##  http://sites.google.com/site/rpackagemseasy/downloads/Agilent_example.zip

url1<-"http://sites.google.com/site/rpackagemseasy/downloads/Agilent_example.zip"
download.file(url=url1, destfile="AgilentCDF.zip")
unzip(zipfile="AgilentCDF.zip", exdir=".")
unlink("AgilentCDF.zip")  ##delete the zip files
## Two folders are created in your current working directory : Agilent_CDF and Agilent_rteres

#with pathCDF
library(xcms)
MS.DataCreation(path=file.path(getwd(),"Agilent_rteres"), pathCDF=file.path(getwd(),"Agilent_CDF"), DataTyp

# without pathCDF
library(xcms)
MS.DataCreation(path=file.path(getwd(),"Agilent_rteres"), DataType="Agilent", mz=30:250,apex=FALSE, quant=F

## Browse for the path to the Agilent_CDF folder
## downloaded and unzipped from MSeasy website
unlink(c("Agilent_rteres", "Agilent_CDF"), recursive=TRUE)   #remove

##DataType="CDF"
##require xcms package
## Each GC-MS files has one folder containing
## one CDF files and one peak list file named peaklist.txt
## All analysis folders are grouped in one folder
## CDF files and peaklist.txt have to be downloaded from MSeasy web site
##  http://sites.google.com/site/rpackagemseasy/downloads/CDF_peaklist_example.zip


url1<-"http://sites.google.com/site/rpackagemseasy/downloads/CDF_peaklist_example.zip"
download.file(url=url1, destfile="ExampleCDF.zip")
unzip(zipfile="ExampleCDF.zip", exdir=".")
##One folder is created in your current working directory CDF_peaklist
unlink("ExampleCDF.zip")  ##delete the zip files

#with pathCDF
library(xcms)
MS.DataCreation(pathCDF=file.path(getwd(),"CDF_peaklist"), DataType="CDF", mz="all",apex=FALSE, quant=FALSE

# without pathCDF
library(xcms)
MS.DataCreation(DataType="CDF", mz="all",apex=FALSE, quant=FALSE)


## Ask for the CDF_peaklist folder
## downloaded and unzipped from MSeasy website
unlink("CDF_peaklist", recursive=TRUE)

## End(Not run)

##For ASCII GC-MS files
pathASCII<-system.file("doc/ASCII_MSDataCreation",
package="MSeasy")
MS.DataCreation(path=pathASCII,mz=30:250,DataType="ASCII",apex=TRUE, N_filt=3)
```

`MS.test.clust` *Test for the best clustering method*

Description

This function tests the efficiency of several unsupervised clustering methods to group similar mass spectra from mass spectrometry (MS) data. Using a dataset where molecules are already well-identified and represented by several samples/individuals mass spectra, the clustering algorithms are tested for their ability to find the correct structure of the dataset (correctly assign the different mass spectra to the pre-defined molecules).

Usage

```
MS.test.clust(data_tot, nclust)
```

Arguments

`data_tot`	data matrix with the name of the molecule in the first column, the name of the sample in the second column, the retention time (or retention index) in the third column and the relative mass spectrum displayed in the following columns.
`nclust`	number of molecules in the dataset

Details

This function tests the efficiency of several unsupervised clustering methods to group similar mass spectra from mass spectrometry data. Using a dataset where molecules are already well-identified and represented by several samples/individuals mass-spectra, the clustering algorithms are tested for their ability to correctly assign the different mass spectra to the pre-defined molecules.

Since the total number of true molecules is usually unknown in complex biological substance, the use of unsupervised clustering algorithms is required. These include partitional and hierarchical algorithms. Partitional algorithms, such as K-means or Partitioning Around Medoids (PAM), determine all clusters at once and do not consider any hierarchic-al/neighboring relations among clusters. For these algorithms, the number of clusters should be specified beforehand. Unlike partitional methods, hierarchical algorithms are iterative methods for clustering datasets (hierarchical clustering analysis HCA), based on the neighboring relations among clusters. Two types of algorithms exist: agglomerative or divisive. Agglomerative algorithms begin with each element as a separate cluster and merge them into successively larger clusters. Divisive algorithms begin with the whole set and proceed to divide it into successively smaller clusters. An important step in hierarchical methods is the choice of a distance metric and a link method. Several options are implemented for the link method (average, single, complete, centroid and ward). The p-order Minkowski distance and correlations are the most commonly used measures of dissimilarity and similarity (Jobson, 1991). Minkowski distance is typically used with p being 2 (Euclidean distance) or 1 (Manhattan distance). A study found that clustering large dimension data was more efficient using p indices of Minkowski distances smaller than 1 (Aggarwal, et al., 2000; Hinneburg, et al., 2000). For that reason, we also implemented two values of p indices below 1 (p= 1/2; p=1/3 The clustering algorithms tested are Partition Around Medoid (PAM), hierarchical divisive clustering (Diana), hierarchical agglomerative clustering (hclust), with various combinations of distance metrics and link methods.

The results of clustering algorithms are evaluated with three quality indices that assess which clustering scheme best fits the data. The matching coefficient computes for correct assignment of each

mass spectrum to the expected molecules. When one cluster groups the mass spectra corresponding to the same molecule, then 1 is attributed and when one cluster contains mass spectra of different molecules, then 0 is attributed. The sum is then divided by the total expected number of molecules/clusters. The value of the matching coefficient varies from 0 to 1 and 1 indicates perfect clustering. Matching coefficient= Number of clusters grouping mass spectra of the same molecule divided by the total number of clusters.

The second cluster validity index is called silhouette width and was described by Rosseeuw (1987). This index is based on two criteria: cluster compactness and isolation.

Silhouette width s(i) is defined as: s(i)=(b-a)/max(a,b)

where a is the average distance of a point from the other points of the same cluster (variation intracluster / compactness) and b represents the minimum of the average distances of the point from the points of the other clusters (cluster separation)

Another quality index, the Dunn index D, is defined as:

D=[mink,l-numbers of clustersdist(Ck, Cl)]/[maxm-cluster numberdiam(Cm)]

k,l,m - numbers of clusters which come from the same partitioning, dist(Ck,Cl) - inter cluster distance between clusters Ck and Cl, diam(Cm) - intra cluster diameter computed for cluster Cm.

Value

This function will return three matrices with the distance metric in column and the clustering algorithms in row.

`Dunn.test`	display the Dunn index
`silhouette.test`	display the Silhouette Width
`matching.coef`	display the matching coefficient

This function produces a pdf file *Graph_MStestClust.pdf* displaying graphics with matching coefficient and silhouette width in the folder *output_MStestclust_resultDate_time* to help identifying the best clustering method.

Author(s)

Elodie Courtois, Yann Guitton, Florence Nicole

Examples

```
## Not run:
data(Data_testclust)
MS.test.clust(Data_testclust,10)

## End(Not run)
```

`MSeasyToARISTO` *Convert output files from the function MS.clust into compatible format for ARISTO websearch* http://www.ionspectra.org/aristo/batchmode/

Description

MSeasyToARISTO convert the output files `output_peak` or `output_cluster` generated by MS.clust to the ARISTO webtool. ARISTO is a webtool that provides ontology of submitted compounds http://www.ionspectra.org/aristo/batchmode/. It is possible to consider only a subset of selected clusters.

Usage

```
MSeasyToARISTO(filename="", outfilename="", cluster="")
```

Arguments

`filename`	input text file as returned by *MS.clust* (`Output_peak.txt` or `Output_cluster.txt`. If left empty a popup window opens to browse your computer)
`outfilename`	Name of the converted file for ARISTO (if left empty default is ForARISTO
`cluster`	If `cluster` = numeric(), to select one or a subset of clusters for submission to ARISTO.

Value

A file compatible with the ARISTO webtool is created in a new folder *output_MStoARISTO_result_time*

Author(s)

Yann Guitton

See Also

tcltk, http://www.ionspectra.org/aristo/batchmode/

Examples

```
#Not run
## Not run:
pathexample<-system.file("doc/Output_examples",
package="MSeasy")

MSeasyToARISTO(file.path(pathexample,"Output_peak21.txt"), cluster=1)
MSeasyToARISTO(file.path(pathexample,"Output_cluster21.txt"), cluster=1)

## End(Not run)
```

MSeasyToMSP *Convert output files from MS.clust into MSP format for NIST mass spectral library search*

Description

MSeasyToMSP export mass spectra from the output files generated by MS.clust into a MSP file compatible with NIST mass spectral library search tool. It is possible to consider only mass spectra from a selected subset of clusters. tcltk package is needed.

Usage

```
MSeasyToMSP(filename="", outfilename="", cluster="", autosearch=FALSE)
```

Arguments

filename	text file as returned by *MS.clust* (Output_peak.txt or Output_cluster.txt. If left empty, a popup window opens to specify the location of the input file)
outfilename	Name for the MSP output file (if left empty default is ForNIST
cluster	If cluster = numeric(), to select one or a subset of clusters for identification.
autosearch	ONLY on WINDOWS plateforms when NIST mssearch is installed! If autosearch = TRUE, then the MSP file created is automatically sent to your NIST MS search tool. See also the function SearchNIST of MSeasy

Value

A file compatible with the NIST mass spectral library search tool (*.MSP file) is created in a new folder *output_MStoMSP_result_time*

Author(s)

Yann Guitton

See Also

tcltk, SearchNIST

Examples

```
#Not run
#Not run
## Not run:
pathexample<-system.file("doc/Output_examples",
package="MSeasy")
MSeasyToMSP(file.path(pathexample,"Output_peak21.txt"), cluster=5)

MSeasyToMSP(file.path(pathexample,"Output_cluster21.txt"))

## End(Not run)
```

SearchNIST *Identification of putative molecules using the NIST mass spectral library search tool*

Description

SearchNIST performs identification of putative molecules using the NIST mass spectral library search tool. The input file is a MSP file. It can be obtained directly from MSeasy output files by using the conversion function MSeasyToMSP. Tcltk package is needed. Warning: this function only works on Windows plateforms !

Usage

```
SearchNIST(mspfile=NULL)
```

Arguments

mspfile	file returned by MSeasyToMSP or any MSP file compatible with NIST. Default is NULL (or if left empty), launched a popup window to specify the location of the MSP file

Value

A text file called *ResultsFromNIST.txt*with the results of the NIST mass spectral library search tool is created in the folder *output_SearchNIST_result_time*.

Author(s)

Yann Guitton

See Also

tcltk, MSeasyToMSP

Examples

```
#Not run
## Not run:
pathexample<-system.file("doc/Output_examples",
package="MSeasy")
MSeasyToMSP(file.path(pathexample,"Output_peak21.txt"), cluster=5)

SearchNIST(mspfile=NULL) # select the MSP file created by the code below
#or
SearchNIST(mspfile=file.path(pathexample,"ForNIST.msp"))

## End(Not run)
```

trans.ASCII *Transform GC-MS data in ASCII format to suitable data for MS.DataCreation*

Description

This function transform each ASCII file (i.e. each GC-MS analysis in ASCII format) into a new file compatible with MS.DataCreation.

Usage

```
trans.ASCII(path, mz)
```

Arguments

path	Name of the folder containing all the GC-MS analyses in ASCII format. If left empty, a popup window opens to browse your computer *tcltk package required*
mz	Range of mass fragments delimiting the mass spectrum (each mass fragment is characterized by its mass-to-charge ratio m/z)

Details

When your raw GC-MS data cannot be exported to netCDF (or mzXML) but only to the international ASCII format (.txt). The data in ASCII format have to be transformed with the function trans.ASCII for further analyses with MS.DataCreation (option *DataType="ASCII"*).

Value

trans.ASCII creates a folder named *output_transASCII_Date_Hour* which contains the same number of files than path.

Author(s)

Elodie Courtois, Yann Guitton, Florence Nicole

Examples

```
## Not run:
##not run
##For ASCII GC-MS files
path<-system.file("doc/ASCII_TransASCII",package="MSeasy")
trans.ASCII(path=path,mz=30:250)

## End(Not run)
```

Index