Package 'MetProc'

May 19, 2016

Type Package

Title Separate Metabolites into Likely Measurement Artifacts and True Metabolites

Version 1.0.1

Date 2016-05-18

Author Mark Chaffin

Maintainer Mark Chaffin <mac8278@mail.harvard.edu>

Description Split an untargeted metabolomics data set into a set of likely true metabolites and a set of likely measurement artifacts. This process involves comparing missing rates of pooled plasma samples and biological samples. The functions assume a fixed injection order of samples where biological samples are randomized and processed between intermittent pooled plasma samples. By comparing patterns of missing data across injection order, metabolites that appear in blocks and are likely artifacts can be separated from metabolites that seem to have random dispersion of missing data. The two main metrics used are: 1. the number of consecutive blocks of samples with present data and 2. the correlation of missing rates between biological samples and flanking pooled plasma samples.

Depends R (>= 3.1.0)

Imports gplots, fastcluster

License GPL (>= 2)

LazyLoad true

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2016-05-19 05:46:39

R topics documented:

`MetProc-package` *Separate Untargeted Metabolites into Likely Artifacts and Likely True Metabolites*

Description

Package to separate metabolites from an untargeted metabolomics experiment into likely artifacts versus likely true metabolites. The general strategy is to compare missing rates of pooled plasma samples and missing rates of biological samples across an injection order. With a randomized injection order for biological samples, generally metabolites that are present for only certain sections of the entire run (exhibiting a block structure) are likely artifacts whereas metabolites with random patterns of missingness are likely true metabolites. The package uses 3 main metrics to separate metabolites and provides tools to plot patterns of missing data across injection order to visualize differences in likely artifacts compared to true metabolites. Details of the separation process and applied metrics can be found in the details section of `met_proc`.

Details

Package:	MetProc
Type:	Package
Version:	1.0
Date:	2016-05-18
License:	GPL (>= 2)

If data is formatted appropriately (see `sampledata` for an example), generally only need to use the `read.met` function followed by the `met_proc` function to output a separate dataframe for likely true metabolites and likely measurement artifacts.

Author(s)

Mark Chaffin

Maintainer: Mark Chaffin <mac8278@mail.harvard.edu>

Examples

```
library(MetProc)
#Read in metabolomics dataset
metdata <- read.met(system.file("extdata/sampledata.csv", package="MetProc"),
headrow = 3, metidcol = 1, fvalue = 8, sep = ",", ppkey = "PPP", ippkey = "BPP")

#Separate likely artifacts from true signal using default settings
results <- met_proc(metdata,plot=FALSE)

#Separate likely artifacts from true signal using custom cutoffs and criteria
#Uses 5 groups of metabolites based on the pooled plasma missing rate, applies
#custom metric thersholds, sets the minimum pooled plasma missing rate to 0.05,
#sets the maximum pooled plasma missing rate to 0.95, sets the missing rate
#to consider a block of samples present at 0.6
results <- met_proc(metdata, numsplit = 5, cor_rates = c(0.4,.7,.75,.7,.4),
runlengths = c(80, 10, 12, 10, 80), mincut = 0.05, maxcut = 0.95, scut = 0.6,
ppkey = 'PPP', sidkey = 'X', plot = FALSE)

#Uses default criteria for running met_proc, but plots the results
#and saves them in a PDF in the current directory. Adding plots
#may substantially increase running time if many samples are
#included
results <- met_proc(metdata, plot = TRUE, missratecut = 0.001,
histcolors = c('red','yellow','green','blue','purple'))

#Write the retained metabolites to current directory
write.met(results,'sample_retained.csv',
system.file("extdata/sampledata.csv", package="MetProc"),
headrow=3,metidcol=1,fvalue=8,sep=",",type='keep')

#Write the removed metabolites to current directory
write.met(results,'sample_removed.csv',
system.file("extdata/sampledata.csv", package="MetProc"),
headrow=3,metidcol=1,fvalue=8,sep=",",type='remove')
```

`corr_metric` *Calculate Correlation of Missing Rates between Pooled Plasma and Biological Samples*

Description

Calculates the correlation of missing rates between the two flanking pooled plasma samples and intervening biological samples for each block in the injection order. A block is defined as a set of biological samples and their flanking pooled plasma samples. See `sampledata` for an example of the data format and block structure. Requires 2 arguments as input: 1. The metabolomics dataset formatted from the `read.met` function and 2. A list of 2 elements output from the `get_group` function containing column indices of pooled plasma samples and biological samples, respectively. If either pooled plasma or biological samples are entirely absent or entirely present, the function will return NA for the metric of that metabolite as the standard deviation of a vector will be 0.

Usage

```
corr_metric(df, grps)
```

Arguments

df	The metabolomics dataset, ideally read from the read.met function. Each column represents a sample and each row represents a metabolite. Columns should be labeled with some unique prefix denoting whether the column is from a biological sample or pooled plasma sample. For example, all pooled plasma samples may have columns identified by the prefix "PPP" and all biological samples may have columns identified by the prefix "X". Missing data must be coded as NA. Columns must be ordered by injection order.
grps	A list of 2 elements from the get_group function. Element "pp" should contain indices of pooled plasma columns and "sid" should contain indicies of biological sample columns.

Value

Returns a vector of equal length to the number of rows in df (representing metabolites) with the correlation of missing rates between flanking pooled plasma and intervening biological samples across all blocks.

See Also

See MetProc-package for examples of running the full process.

Examples

```
library(MetProc)
#Read metabolomics data
metdata <- read.met(system.file("extdata/sampledata.csv", package="MetProc"),
headrow=3, metidcol=1, fvalue=8, sep=",", ppkey="PPP", ippkey="BPP")

#Get indices of samples and pooled plasma
grps <- get_group(metdata,'PPP','X')

#get correlation metrics of metabolites
corrs <- corr_metric(metdata,grps)
```

---

get_group *Retrieve Index of Biological Samples and Pooled Plasma Samples*

Description

Takes a metabolomics data matrix and retrieves the column indices of biological samples and pooled plasma samples. Columns must be ordered by injection order and each column ID should have a unique prefix designating the particular type of sample it is. For example, "PPP"' to designate pooled plasma samples and "X" to designate biological samples. Generally if data is read into R using the read.met function, columns will be labeled appropriately.

Usage

```
get_group(df, ppkey = "PPP", sidkey = "X")
```

Arguments

df	The metabolomics dataset, ideally read from the read.met function. Each column represents a sample and each row represents a metabolite. Columns should be labeled with some unique prefix denoting whether the column is from a biological sample or pooled plasma sample. For example, all pooled plasma samples may have columns identified by the prefix "PPP" and all biological samples may have columns identified by the prefix "X". Missing data must be coded as NA. Columns must be ordered by injection order.
ppkey	The unique prefix of pooled plasma samples. Default is "PPP".
sidkey	The unique prefix of biological samples. Default is "X".

Value

A list of length 2 with the following keys:

pp	A vector with column indices of pooled plasma
sid	A vector with column indices of samples

See Also

See MetProc-package for examples of running the full process.

Examples

```
library(MetProc)
#Read metabolomics data
metdata <- read.met(system.file("extdata/sampledata.csv", package="MetProc"),
headrow=3, metidcol=1, fvalue=8, sep=",", ppkey="PPP", ippkey="BPP")

#Get groups based on samples and pooled plasma
grps <- get_group(metdata,'PPP','X')
```

---

get_missing *Compute Missing Rates of Biological Samples and Pooled Plasma Samples*

Description

Computes two missing rates per metabolite: 1. Missing rate of biological samples and 2. Missing rate of pooled plasma samples. Requires a metabolomics data matrix from read.met function as well as the indicies of pooled plasma and biological samples from get_group. Returns a list with the two missing rates across all metabolites

Usage

```
get_missing(df, ppind, sampind)
```

Arguments

df	The metabolomics dataset, ideally read from the read.met function. Each column represents a sample and each row represents a metabolite. Columns should be labeled with some unique prefix denoting whether the column is from a biological sample or pooled plasma sample. For example, all pooled plasma samples may have columns identified by the prefix "PPP" and all biological samples may have columns identified by the prefix "X". Missing data must be coded as NA. Columns must be ordered by injection order.
ppind	The indices of the pooled plasma samples.
sampind	The indices of the biological samples.

Value

A list with the missing rates of the pooled plasma samples and biological samples for all metabolites in dataframe. The keys are:

ppmiss	The pooled plasma missing rate for each metabolite
sampmiss	The biological sample missing rate for each metabolite

See Also

See MetProc-package for examples of running the full process.

Examples

```
library(MetProc)

#Read metabolomics data
metdata <- read.met(system.file("extdata/sampledata.csv", package="MetProc"),
headrow=3, metidcol=1, fvalue=8, sep=",", ppkey="PPP", ippkey="BPP")

#Get groups based on samples and pooled plasma
grps <- get_group(metdata,'PPP','X')

#Get the missing rates of each category for all metabolites
missrate <- get_missing(metdata,grps[['pp']],grps[['sid']])
```

heatmap_res *Plot Patterns of Missing Data Across Metabolites*

Description

Generates a heatmap to show patterns of missing data for metabolites. Useful to visualize the block structure of data to compare differences between removed metabolites and retained metabolites.

Usage

```
heatmap_res(df, ppkey = "PPP", sidkey = "X", missratecut = .01,title)
```

Arguments

df	The metabolomics dataset, ideally read from the read.met function. Each column represents a sample and each row represents a metabolite. Columns should be labeled with some unique prefix denoting whether the column is from a biological sample or pooled plasma sample. For example, all pooled plasma samples may have columns identified by the prefix "PPP" and all biological samples may have columns identified by the prefix "X". Missing data must be coded as NA. Columns must be ordered by injection order.
ppkey	Unique prefix of pooled plasma columns. Default is "PPP".
sidkey	Unique prefix of biological sample columns. Default is "X".
missratecut	The missing rate limit for displaying a metabolite. Only metabolites with overall missing rates equal to or greater than this cutoff will be plotted. Useful for avoiding plotting too many metabolites as the heatmap generation can be an expensive computation. If a metabolite has a very small missing rate, plotting is uninformative as all data is present. Default set to 0.01.
title	The title of the heatmap plotted

Value

Returns a heatmap illustrating the patterns of missing data for metabolites.

See Also

See MetProc-package for examples of running the full process.

Examples

```
library(MetProc)

#Read in metabolomics data
metdata <- read.met(system.file("extdata/sampledata.csv", package="MetProc"),
headrow=3, metidcol=1, fvalue=8, sep=",", ppkey="PPP", ippkey="BPP")

#Get the good versus bad metabolites
```

```
results <- met_proc(metdata)

#Plot Removed metabolites
#Similarly run for retained metabolites but
#replacing 'remove' with 'keep'
heatmap_res(results[['remove']],missratecut=.02,title='Removed Metabolites')
```

---

met_proc *Separates Metabolites into Likely True Metabolites and Likely Measurement Artifacts*

Description

Takes a metabolomics data matrix and processes metabolites into likely artifacts versus likely true metabolites. Biological samples should follow a randomized injection order with pooled plasma samples interspersed. Columns of data should be samples and rows are metabolites. Columns must be ordered by injection order. Metabolites are first grouped by missing rate of pooled plasma and then processed based on metrics of blocky structure to identify likely artifacts. Specifically, `corr_metric` and `run_metric` are used to quantify the degree to which structure is present in the patterns of missing data. Must pass all thresholds to be considered a true metabolite.

Usage

```
met_proc(df, numsplit = 5, cor_rates = c(0.6, 0.65, 0.65, 0.65, 0.6),
runlengths = c(NA, 15, 15, 15, NA), mincut = 0.02, maxcut = 0.95, scut = 0.5,
ppkey = "PPP", sidkey = "X", missratecut=0.01, histcolors=c('white'), plot=TRUE,
outfile='MetProc_output')
```

Arguments

`df`	The metabolomics dataset, ideally read from the `read.met` function. Each column represents a sample and each row represents a metabolite. Columns should be labeled with some unique prefix denoting whether the column is from a biological sample or pooled plasma sample. For example, all pooled plasma samples may have columns identified by the prefix "PPP" and all biological samples may have columns identified by the prefix "X". Missing data must be coded as NA. Columns must be ordered by injection order.
`numsplit`	The number of equal sized sections to divide metabolites into based on missing rate of pooled plasma columns. Divides the range of missing rates between `mincut` and `maxcut` into equal sections. Default is `5`.
`cor_rates`	A vector of length equal to `numsplit`. Each value represents the cutoff of the correlation metric in that section. Any metabolite with a value greater than or equal to the cutoff is considered an artifact and anything less than the cutoff is considered a true metabolite. If any value is set to NA, the correlation metric will not be considered for that group. One cutoff per group. Default is `c(.6, .65, .65, .65, .6)`.

`runlengths` A vector of length equal to `numsplit`. Each values represents the cutoff for the longest run metric in that section. Any metabolite with a run greater than or equal to the cutoff is considered an artifact and anything less than the cutoff is considred a true metabolite. If any value is set to NA, the longest run metric will not be considered for that group. One cutoff per group. Default is `c(NA, 15, 15, 15, NA)`.

`mincut` A cutoff to specify that any metabolite with pooled plasma missing rate less than or equal to this value should be retained. Default is `0.02`.

`maxcut` A cutoff to specify that any metabolite with pooled plasma missing rate greater than this value should be removed. Default is `0.95`.

`scut` The cutoff of missingness to consider a metabolite as having data present in a given biological sample block. Relevant only to `run_metric` computation. Default is `0.5`.

`ppkey` The unique prefix of pooled plasma columns. Default is `"PPP"`.

`sidkey` The unique prefix of biological samples columns. Default is `"X"`.

`missratecut` A parameter for heatmap plots when `plot=TRUE`. Only metabolites with missing rates (across pooled plasma and biological samples) equal to or greater than this cutoff will be plotted. Useful to avoid plotting too may metabolites in an effort to save time. If a metabolite has a very small missing rate, plotting is uninformative as all data is present. Default is `0.01`.

`plot` Indicate whether you would like to obtain plots of missingness patterns and distributions of calculated metrics. Plots will be output as a PDF. Default is `TRUE`.

`histcolors` A vector of length equal to `numsplit`. Each value represents the color to use for that group in the histograms of the longest run and correlation metrics for each subset of metabolites. If no color is provided, they will be colored white.

`outfile` Name and path of the file to store images if `plot=TRUE`. Do not include file extension in the name. Default is `"MetProc_output"` which will save a file called MetProc_output.pdf in the current working directory.

Details

The function uses a four step process:

1. Retain all metabolites with pooled plasma missing rate below `mincut` and remove all metabolites with pooled plasma missing rate above `maxcut`.
2. Split the remaining metabolites into `numsplit` groups that are defined by pooled plasma missing rates. The `numsplit` groups will divide the range of pooled plasma missing rates evenly.
3. For each group of metabolites based on pooled plasma missing rates from step 2, calculate the correlation metric with `corr_metric`. Any metabolite below the cutoff for that group, defined by `cor_rates`, will be retained and any metabolite above will be removed.
4. For each group of metabolites based on pooled plasma missing rates from step 2, calculate the longest run metric with `run_metric`. Any metabolite below the cutoff for that group, defined by `runlengths`, will be retained and any metabolite above will be removed.

Value

keep	A dataframe of the retained metabolites
remove	A dataframe of the removed metabolites

If plot = True, a PDF file will be saved containing the correspondence between pooled plasma missing rate and sample missing rate, the distribution of the correlation metric and longest run metric in each of the groups based on pooled plasma missing rates, and heatmaps displaying the patterns of present/missing data for both the removed and retained metabolites.

See Also

See run_metric for details on the longest run metric.
See corr_metric for details on the correlation metric.
See MetProc-package for examples of running the full process.

Examples

```
library(MetProc)

#Read in metabolomics data
metdata <- read.met(system.file("extdata/sampledata.csv", package="MetProc"),
headrow=3, metidcol=1, fvalue=8, sep=",", ppkey="PPP", ippkey="BPP")

#Separate likely artifacts from true signal using default settings
results <- met_proc(metdata,plot=FALSE)

#Separate likely artifacts from true signal using custom cutoffs and criteria
#Uses 5 groups of metabolites based on the pooled plasma missing rate, applies
#custom metric thersholds, sets the minimum pooled plasma missing rate to 0.05,
#sets the maximum pooled plasma missing rate to 0.95, sets the missing rate
#to consider a block of samples present at 0.6
results <- met_proc(metdata, numsplit = 5, cor_rates = c(0.4,.7,.75,.7,.4),
runlengths = c(80, 10, 12, 10, 80), mincut = 0.05, maxcut = 0.95, scut = 0.6,
ppkey = 'PPP', sidkey = 'X', plot = FALSE)

#Uses default criteria for running met_proc, but plots the results
#and saves them in a PDF in the current directory.
#Colors of the histograms set by histcolors.
#Adding plots may substantially increase running time if many
#samples are included
results <- met_proc(metdata, plot = TRUE, missratecut = 0.001,
histcolors = c('red','yellow','green','blue','purple'))
```

---

plot_metric *Plot Distribution of Metabolite Quality Metrics for Each Split of Data*

Description

For a given number of splits of data based on pooled plasma missing rate, calculate the longest run metric (run_metric) and the correlation metric (corr_metric) for metabolites in each group. Plot the distribution of these metrics for each group color coding those that exceed thresholds.

Usage

```
plot_metric(df,ppkey='PPP',sidkey='X',numsplit=5,mincut=.02,maxcut=0.95,
scut=0.5,cor_rates=c(.6,.65,.65,.65,.6),runlengths=c(NA,15,15,15,NA),
histcolors=c('white'))
```

Arguments

df	The metabolomics dataset, ideally read from the read.met function. Each column represents a sample and each row represents a metabolite. Columns should be labeled with some unique prefix denoting whether the column is from a biological sample or pooled plasma sample. For example, all pooled plasma samples may have columns identified by the prefix "PPP" and all biological samples may have columns identified by the prefix "X". Missing data must be coded as NA. Columns must be ordered by injection order.
ppkey	The unique prefix of pooled plasma samples. Default is "PPP".
sidkey	The unique prefix of biological samples. Default is "X".
numsplit	The number of equal sized sections to divide metabolites into based on missing rate of pooled plasma columns. Divides the range of missing rates between mincut and maxcut into equal sections. Default is 5.
mincut	A cutoff to specify that any metabolite with pooled plasma missing rate less than or equal to this value should be retained. Default is 0.02.
maxcut	A cutoff to specify that any metabolite with pooled plasma missing rate greater than this value should be removed. Default is 0.95.
scut	The cutoff of missingness to consider a metabolite as having data present in a given biological sample block. Relevant only to run_metric computation. Default is 0.5.
cor_rates	A vector of length equal to numsplit. Each value represents the cutoff of the correlation metric in that section. Any metabolite with a value greater than or equal to the cutoff is deemed an artifact and anything less than the cutoff is deemed a true metabolite. If any value is set to NA, the correlation metric will not be considered for that group. Default is c(.6, .65, .65, .65, .6).
runlengths	A vector of length equal to numsplit. Each values represents the cutoff for the longest run metric in that section. Any metabolite with a run greater than or equal to the cutoff is an artifact and anything less than the cutoff is a true metabolite. If any value is set to NA, the longest run metric will not be considered for that group. Default is c(NA, 15, 15, 15, NA).
histcolors	A vector of length equal to numsplit. Each value represents the color to use for that group. If no color is provided, they will be colored white.

Value

Returns histograms showing the correlation metric and longest run metric distributions for each group of the metabolites based on pooled plasma missing rate.

See Also

See MetProc-package for examples of running the full process.

Examples

```
library(MetProc)

#Read in metabolomics data
metdata <- read.met(system.file("extdata/sampledata.csv", package="MetProc"),
headrow=3, metidcol=1, fvalue=8, sep=",", ppkey="PPP", ippkey="BPP")

#Plot distributions of the two metrics for each group
plot_metric(metdata,ppkey='PPP',sidkey='X',numsplit=5,mincut=0.02,maxcut=0.95,
scut=0.5,cor_rates=c(.6,.65,.65,.65,.6),runlengths=c(NA,15,15,15,NA),
histcolors=c('red','yellow','green','blue','purple'))
```

---

plot_pp_sample_missing

Plot Pooled Plasma and Biological Sample Missing Rates

Description

Calculates the missing rate of the pooled plasma columns and biological sample columns for each metabolite. Plots a scatterplot showing the correspondence between the two.

Usage

```
plot_pp_sample_missing(df, ppkey = "PPP", sidkey = "X")
```

Arguments

df	The metabolomics dataset, ideally read from the read.met function. Each column represents a sample and each row represents a metabolite. Columns should be labeled with some unique prefix denoting whether the column is from a biological sample or pooled plasma sample. For example, all pooled plasma samples may have columns identified by the prefix "PPP" and all biological samples may have columns identified by the prefix "X". Missing data must be coded as NA. Columns must be ordered by injection order.
ppkey	The unique prefix of pooled plasma samples. Default is "PPP".
sidkey	The unique prefix of biological samples. Default is "X".

Value

Returns a scatterplot comparing the pooled plasma missing rate to the sample missing rate

See Also

See MetProc-package for examples of running the full process.

Examples

```
library(MetProc)

#Read in metabolomics data
metdata <- read.met(system.file("extdata/sampledata.csv", package="MetProc"),
headrow=3, metidcol=1, fvalue=8, sep=",", ppkey="PPP", ippkey="BPP")

#Plot the pooled plasma missing rate against the sample missing rate
plot_pp_sample_missing(metdata,ppkey='PPP',sidkey='X')
```

---

read.met *Read in a Metabolomics Dataset of Standard Structure*

Description

Read a metabolomics file. The file must be structured in a specific way. The columns of the file designate samples and the rows designate metabolites. The first n rows may contain any information. However, starting at row n+1 there must be a header line with column labels. The remaining rows are designated as one per metabolite. One column should contain the ID of each metabolite. Other columns can be included, but starting at some column, and continuously after this point, each sample or pooled plasma sample should be given its own column sorted by injection order. All pooled plasma columns should have a unique prefix differentiating them from biological samples. Up to 2 types of pooled plasma samples can be included in the file – each with a unique prefix. This may be useful when both a pooled plasma control generated from biological samples and a commercially available pooled plasma standard are used. All biological samples may have a designating prefix or simply lack a prefix designating pooled plasma samples. If no prefix designates the biological samples, a prefix of "X" will be used for biological samples in subsequent analysis. Missing data must be coded as NA.

Usage

```
read.met(data, headrow = 3, metidcol=1, fvalue=8, sep=",", ppkey='PPP',
ippkey = 'BPP', sidkey="none")
```

Arguments

data	The metabolomics dataset file. The columns of the file designate samples and the rows designate metabolites. The first n rows can contain any information. However, starting at row n+1 there must be a header line with column labels. The remaining rows are designated as one per metabolite. One column should

contain the ID of each metabolite. Other columns can be included, but starting at some column, and continuously after this point, each biological sample or pooled plasma sample should be given it's own column sorted by injection order. All pooled plasma columns should have a unique prefix differentiating them from samples. Up to 2 types of pooled plasma samples can be included in the file – each with a unique prefix. All biological samples may have a designated prefix or simply lack the the prefix designating pooled plasma samples. If no prefix designates the biological samples, a prefix of "X" will be used for biological samples in subsequent analysis. Missing data must be coded as NA. See file `sampledata` for an example.

`headrow` The row number that contains the header line. Default is `3`.

`metidcol` The column number that contains the metabolite ID. Default is `1`.

`fvalue` The column number where data begins. Default is `8`.

`sep` File delimiter. Default is `","`.

`ppkey` The unique prefix of biological sample-based pooled plasma columns. Default is `"PPP"`.

`ippkey` The unique prefix of standard pooled plasma columns. Default is `"BPP"`.

`sidkey` The unique prefix of biological samples in the csv file. If 'none' provided as value, any column that does not contain the prefix of `ppkey` or `ippkey` will be considered a biological sample and given the prefix 'X' for subsequent use. Default is `"none"`.

Value

A matrix with the metabolomics data fully loaded. Should have the number of rows equal to the number of metabolites and columns equal to the number of samples + pooled plasma samples.

See Also

See `MetProc-package` for examples of running the full process.

Examples

```
library(MetProc)

#Read in metabolomics data
metdata <- read.met(system.file("extdata/sampledata.csv", package="MetProc"),
headrow=3, metidcol=1, fvalue=8, sep=",", ppkey="PPP", ippkey="BPP")
```

`run_metric` *Calculate Longest Run of Blocks where Data is Present*

Description

For each metabolite, data is split into blocks that consist of the preceding pooled plasma sample and following biological samples in an injection order. For each block, data is deemed present in biological samples if the missing rate is less than `scut`. An entire block is deemed to have data present if both the preceding pooled plasma and folllowing biolgical samples are both considered to have data present. The length of the longest consecutive run of blocks with data present is returned for each metabolite.

Usage

```
run_metric(df, grps, scut = 0.5)
```

Arguments

`df`	The metabolomics dataset, ideally read from the `read.met` function. Each column represents a sample and each row represents a metabolite. Columns should be labeled with some unique prefix denoting whether the column is from a biological sample or pooled plasma sample. For example, all pooled plasma samples may have columns identified by the prefix "PPP" and all biological samples may have columns identified by the prefix "X". Missing data must be coded as NA. Columns must be ordered by injection order.
`grps`	A group list from the `get_group` function
`scut`	The cutoff missing rate to determine if data is present in a group of biological samples. If the missing rate of the biological samples is greater than or equal to this missing rate threshold, data will be considered absent from the block of biological samples. Default is `0.5`.

Value

Returns a vector containing the longest consecutive run of blocks with data present for each metabolite

See Also

See `MetProc-package` for examples of running the full process.

Examples

```
library(MetProc)

#Read in metabolomics data
metdata <- read.met(system.file("extdata/sampledata.csv", package="MetProc"),
headrow=3, metidcol=1, fvalue=8, sep=",", ppkey="PPP", ippkey="BPP")

#Get indices of pooled plasma and samples
grps <- get_group(metdata,'PPP','X')

#Get the longest run metric for each metabolite
runs <- run_metric(metdata,grps,scut=.5)
```

`sampledata` *Simulated Metabolomics Data*

Description

This is a simulated dataset to show the format of the metabolomics data; patterns of missing data are generated roughly from a real metabolomics experiment. Rows represent metabolites and columns represent samples. The file contains 100 metabolites (rows) and 505 samples (480 biological sample columns and 25 pooled plasma columns) sorted by injection order. There are 20 biological samples between pooled plasma runs. Pooled plasma columns have prefix 'PPP' and biological samples are simple integers with no prefix.

Usage

```
sampledata
```

Format

The first row (Date) contains the date of processing. The second row (Inject) contains the injection number and is ordered from 1 to 505. The third row contains the column headers:

`Metab` is the metabolite ID.
`Meth` is the type of metabolite.
`HMDB` is the HMDB ID of the metabolite, if it exists.
`m/z` is the mass-to-charge ratio of the metabolite.
`rt` is the retention time.
`Com` contains any comments.
`ProcID` is the processing ID of the metabolite.

The remaining columns are either pooled plasma samples (prefix: 'PPP') or biological samples (prefix: No prefix). The basic structure of the csv file is as follows:

						Date	415	415	..	415	415	415	..
						Inject	1	2	..	21	22	23	..
Metab	Meth	HMDB	m/z	rt	Com	ProcID	PPP1	1	..	20	PPP2	21	..
M1	Lipid	H1	304	8.7		1	6.7	6.7	..	5.0	6.7	4.6	..
M2	Lipid	H2	309	7.6		2	1.0	1.1	..	1.1	1.0	1.2	..
..	..	..	..	..	..	..	..	..	..	..	..	..	..
M100	Lipid	H100	249	6.2		100	2.4	1.9	..	2.2	2.4	1.6	..

See Also

See `read.met` for example of reading this csv file for use.
See `MetProc-package` for examples of running the full process.

subset_met *Group Metabolites based on Pooled Plasma Missing Rate*

Description

Separates metabolites into groups based on pooled plasma missing rates so that different thresholds of metrics can be applied to each group.

Usage

```
subset_met(df, miss, numsplit = 5, mincut = 0.02, maxcut = 0.95)
```

Arguments

df	The metabolomics dataset, ideally read from the read.met function. Each column represents a sample and each row represents a metabolite. Columns should be labeled with some unique prefix denoting whether the column is from a biological sample or pooled plasma sample. For example, all pooled plasma samples may have columns identified by the prefix "PPP" and all biological samples may have columns identified by the prefix "X". Missing data must be coded as NA. Columns must be ordered by injection order.
miss	Vector of missing rates of equal length to number of rows in df representing the pooled plasma missing rate for each metabolite.
numsplit	The number of equal sized sections to divide metabolites into based on missing rate of pooled plasma columns. Divides the range of missing rates between mincut and maxcut into equal sections. Default is 5.
mincut	A cutoff to specify that any metabolite with pooled plasma missing rate less than or equal to this value should be retained. Default is 0.02.
maxcut	A cutoff to specify that any metabolite with pooled plasma missing rate greater than this values should be removed. Default is 0.95.

Value

A list consisting of a number of elements equal to numsplit. Each element contains a matrix of the given metabolite group based on the pooled plasma missing rate. The list keys are simple integers corresponding to the split number.

See Also

See MetProc-package for examples of running the full process.

Examples

```
library(MetProc)

#Read in metabolomics data
metdata <- read.met(system.file("extdata/sampledata.csv", package="MetProc"),
headrow=3, metidcol=1, fvalue=8, sep=",", ppkey="PPP", ippkey="BPP")

#Get indices of pooled plasma and samples
groups <- get_group(metdata,"PPP","X")

#Calculate a pooled plasma missing rate and sample missing rate
#for each metabolite in data
missrate <- get_missing(metdata,groups[['pp']],groups[['sid']])

#Group metabolites into 5 groups based on pooled plasma
#missing rate
subsets <- subset_met(metdata,missrate[['ppmiss']],5,.02,.95)
```

`write.met` *Write Metabolomics Dataset of Standard Structure*

Description

Write a metabolomics file based on the metabolites identified to be retained or removed using `met_proc`. Requires the filepath for the original metabolomics file in order to extract row and column information. Will take in this original file and the results of the `met_proc` function to write a file that contains only the retained or removed metabolites.

Usage

```
write.met(res, filename, origfile, headrow = 3, metidcol=1, fvalue=8,
sep=",", type="keep")
```

Arguments

`res`	The result output from `met_proc` function.
`filename`	The name and path for new metabolomics file.
`origfile`	The name and path for the original metabolomics file.
`headrow`	The row number that contains the header line in the original metabolomics file. Default is `3`.
`metidcol`	The column number that contains the metabolite ID in the original metabolomics file. Default is `1`.
`fvalue`	The column number where data begins in the original metabolomics file. Default is `8`.
`sep`	File delimiter for both the original metabolomics file and the new file. Default is `","`.
`type`	Either 'keep' or 'remove' to determine whether the retained metabolites or removed metabolites should be written to the file. Default is `"keep"`.

Value

Writes a file to `filename` that is of the same structure as the original metabolomics file but only containing either the retained or removed metabolites.

See Also

See `MetProc-package` for examples of running the full process.

Examples

```
library(MetProc)
#Read in metabolomics data
metdata <- read.met(system.file("extdata/sampledata.csv", package="MetProc"),
headrow=3, metidcol=1, fvalue=8, sep=",", ppkey="PPP", ippkey="BPP")

#Separate likely artifacts from true signal using default settings
results <- met_proc(metdata,plot=FALSE)

#Write the retained metabolites to current directory
write.met(results,'sample_retained.csv',
system.file("extdata/sampledata.csv", package="MetProc"),
headrow=3,metidcol=1,fvalue=8,sep=",",type='keep')
```

Index