Package 'batman'

April 28, 2016

Version 1.2.1.06

Author Jie Hao,
William Astle,
Maria De Iorio,
Timothy Ebbels

Maintainer Jie Hao <j.hao@ic.ac.uk>

Title Bayesian AuTomated Metabolite Analyser for NMR spectra

Description BATMAN deconvolves resonance peaks from NMR spectra and obtain concentration estimates for the corresponding metabolites automatically.

Depends R (>= 2.15.0), doSNOW, foreach, iterators, snow, utils, plotrix

License GPL-2

LazyLoad yes

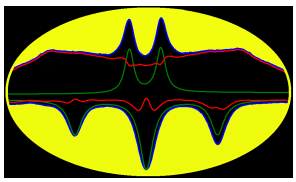
R topics documented:

`batman-package` *Bayesian AuTomated Metabolite Analyser for NMR spectra (BATMAN)*

Description

BATMAN deconvolves resonance peaks from NMR spectra of complex mixtures and obtains concentration estimates for the corresponding metabolites automatically. This is achieved through a database of spectral profiles for known metabolites and a Bayesian Markov Chain Monte Carlo algorithm. Users have the options to specify the multiplet ppm position, position shift range, peak width range and so on. Parallel processing is available if processing several spectra. The installation and testing instructions can be found at:

https://r-forge.r-project.org/scm/viewvc.php/documentation



Details

Package:	batman
Type:	Package
Version:	1.2.1.03
Date:	2014-10-24
License:	GPL-2
LazyLoad:	yes

Author(s)

Jie Hao <j.hao@ic.ac.uk>, William Astle <william.astle@mail.mcgill.ca>, Maria De Iorio <m.deiorio@ucl.ac.uk>, Timothy Ebbels <t.ebbels@ic.ac.uk>

References

Hao, J., et al., BATMAN–an R package for the automated quantification of metabolites from nuclear magnetic resonance spectra using a Bayesian model. Bioinformatics, 2012. 28(15): p. 2088-90.

http://bioinformatics.oxfordjournals.org/content/28/15/2088

Astle, W., et al., A Bayesian Model of NMR Spectra for the Deconvolution and Quantification of Metabolites in Complex Biological Mixtures. Journal of the American Statistical Association, 2012. 107(500): p. 1259-1271.

http://www.tandfonline.com/doi/abs/10.1080/01621459.2012.695661#.UgEf-hbZa4k

---

`batman` *Perform BATMAN and Plot Analysis Result*

Description

The main function, it performs metabolite and wavelet fitting to input NMR spectra, plots fitting results, posterior distributions for relateive concentrations and peak positions, and saves output. If the input `createDir = TRUE`, a folder name "runBATMAN" will be created in specified directory, within which, two folders "BatmanInput" and "BatmanOutput" are created. "BatmanInput" contains the input data files copied from installed package folder "extdata". The user only needs to modify files in this folder to change the settings for running `batman`. The `batman` output files are saved in "BatmanOutput" subfolders.

Usage

```
batman(BrukerDataDir, BrukerDataZipDir, txtFile, rData, createDir = TRUE,
       runBATMANDir = getwd(), overwriteDir = FALSE,
       figBatmanFit = TRUE, listMeta = FALSE,
       figRelCon = FALSE, figMetaFit = FALSE,
       showPlot = TRUE)
```

Arguments

`BrukerDataDir`	The directory of the folder containing 1D Bruker spectral data files. If not specified, spectral data will be read in from one of the following inputs prioritized in the order: `BrukerDataZipDir`, `txtFile`, `rData` and NMRdata.txt in "BatmanInput" folder.
`BrukerDataZipDir`	The directory of the folder containing zipped 1D Bruker spectral data files. If not specified, spectral data will be read in from one of the following inputs prioritized in the order: `BrukerDataDir`, `txtFile`, `rData` and NMRdata.txt in "BatmanInput" folder.
`txtFile`	The .txt file containing spectral data in the format of first column ppm, and the second column the real part of spectrum. If not specified, spectral data will be read in from one of the following inputs prioritized in the order: `BrukerDataDir`, `BrukerDataZipDir`, `rData` and NMRdata.txt in "BatmanInput" folder.

rData	The R data file containing spectral data in the format of first column ppm, and the second column the real part of spectrum. If not specified, spectral data will be read in from one of the following inputs prioritized in the order: BrukerDataDir, BrukerDataZipDir, txtFile and NMRdata.txt in "Batman-Input" folder.
createDir	If set TRUE, a new BATMAN work directory will be created specified by runBATMANDir. If set FALSE, batman input will be obtained from the "extdata" folder in batman package installation directory, and the batman output files will also be put within this folder. The default is TRUE.
runBATMANDir	User specified BATMAN work directory, the default is current work directory. It will only work when createDir is set TRUE.
overwriteDir	If folder "runBATMAN" exists, set TRUE to overwrite folder. The default is FALSE.
figBatmanFit	Plot metabolites and wavelets fit if set TRUE. The default is TRUE.
listMeta	Individual metabolite fit will also be shown in the plot if set TRUE. The default is FALSE.
figRelCon	Plot posterior samples of the relative concentration for fitted metabolites with 95% credible interval if set TRUE. The default is FALSE.
figMetaFit	If set TRUE, plot the posterior mean of the metabolites fit with 95% credible interval. The default is FALSE.
showPlot	If set FALSE, no plot will be shown on display, the pdf file(s) for the plot(s) will be created in output folder. The default is TRUE.

Value

It returns a data list with the following objects:

specTitle	A matrix $(2 \times n)$ containing the spectrum number in its first row and the corresponding title of the spectrum in its second row.
sFit	A matrix $t \times 5n$ of BATMAN fit results (down sampled). For 1 spectrum, it is a matrix with 5 columns: $$[ppm, originalspectrum, metabolitesfit, waveletsfit, overallfit].$$ The "overall fit" is the posterior mean of the BATMAN fit results after MCMC burn in iterations. Certain numbers of burn in iterations are used at the beginning of an MCMC run for finding a good starting point. n is the number of spectra, and t is the number of data points in each spectrum.
sFitHR	A matrix $t \times 3n$ of BATMAN fit results in the original resolution (without down sample). For 1 spectrum, it is a matrix with 3 columns: $$[ppm, originalspectrum, metabolitesfit].$$ n is the number of spectra, and t is the number of data points (without down sample) in each spectrum.
beta	A matrix $(m \times n)$ containing the posterior means of relative concentrations for m fitted metabolites and n spectra after burn in.

`betaSam` A matrix ($m \times (s * n)$) containing (for the first spectrum) s posterior samples of the relative concentrations in its rows. m is the number of fitted metabolites. n is the number of spectra analyzed. The subsequent columns contain the same format of data for the rest $n - 1$ spectra.

`betaCI` A matrix ($m \times 2n$) containing the 95% credible interval of the relative concentrations for m fitted metabolites. Every pair of columns is for one spectrum.

`metaTemp` A matrix ($t \times (m * n)$) containing the posterior means of m fitted metabolite templates in its columns (down sampled) after burn in. n is the number of spectra analyzed and t is the number of data points in each spectrum.

`metaTempHR` A matrix ($t \times (m * n)$) containing the posterior means of m fitted metabolite templates in its columns (without down sample) after burn in. n is the number of spectra analyzed and t is the number of data points (without down sample) in each spectrum.

`metaFitSam` A matrix ($t \times (s * n)$) containing s posterior samples of total metabolites fit during MCMC iterations in its columns. n is the number of spectra analyzed and t is the number of data points in each spectrum. The remaining $n-1$ spectra metabolites fit results are saved in the same sequence in subsequent columns.

`metaIndFitSam` A matrix ($t \times (m * s * n)$) containing s posterior samples of m individual metabolites fit during MCMC iterations in its columns. n is the number of spectra analyzed and t is the number of data points in each spectrum. The remaining $n - 1$ spectra results are saved in the same sequence in subsequent columns.

`thetaSam` A matrix ($t \times (s * n)$) containing s samples of wavelet fit during MCMC iterations in its columns. n is the number of spectra analyzed. The remaining $n-1$ spectra wavelet fit results are saved in the same sequence in subsequent columns.

`delta` A matrix ($M \times n$) containing posterior means of M multiplets ppm shift of fitted metabolites in its rows. M is the sum of all multiplets in the fitted metabolts. Each column of the matrix corresponds to one spectrum. If only 1 spectrum is analyzed, delta is a column vector.

`deltaSam` A matrix ($s \times (M * n)$) containing the posterior samples of multiplets ppm shift. Every M columns correspond the shift posterior samples of M multiplets for one spectrum. M is the sum of all multiplets in the fitted metabolts and n is the number of spectra analyzed.

`outputDir` The directory of output folder with all the output result files.

See Also

`readBatmanOutput`, `batmanrerun`

Examples

```
library(batman)
## Run BATMAN
if(interactive()) bm<-batman()
## This will create the folder "runBATMAN" in current working directory,
## within the folder "runBATMAN", a subfolder "BatmanInput" contains all the
## input files batman uses. Users can modify "metabolitesList.csv",
## "batmanOptions.txt" and so on to change the settings of batman.
```

```
## Please check "BatmanInput" for details on how to adjust input parameters.
#########################################################################
## The following is an example of what will be displayed in R
## and what value the user could input:
#########################################################################
## batman...
## Number of burn-in iterations:  4000
## Number of post-burn-in iterations:  100
##
## The template file used is
##  1: The default template of multiplets in multi_data.csv file.
##
## Loading multi_data.csv...
## Percentage completed...
##   |                                                                      |   0%
## Size of each spectrum is 393.
## Size of metabolite list is 22.
## Constructing chain data structure...
## time used is 0 seconds.
## Running MCMC...
##   |=================================================               |  80%
## time used for burnin is 76 seconds.
##   |======================================================================| 100%
## time used is 95 seconds.
## saving posteriors...
##
## time elapsed
##   95.61
##  second.
## Reading in saved data in folder
## .../user_specified_dir/runBATMAN/BatmanOutput/07_Dec_17_19_18
## Completed.
#########################################################################
## Alternatively if more than 1 spectrum are included without using fixed effect
## (in batmanOptions.txt file, set
## "Same concentration for all spectra (fixed effect) (1/0): 0"),
## user will be asked to input the following parameter:
#########################################################################
## How many parallel processes (multicores) do you want to run
## the multi-spectra analysis?
## (Enter 1 for running them sequentially.)
##
##  Parallel processing of multi spectra currently cannot display
##  progress bar (or any words), if you input is > 1, please be patient
##  for the results :)
##
## 1: 2   ## user input
## time elapsed
##   78.79
##  second.
## Reading in saved data in folder
## .../user_specified_dir/runBATMAN/BatmanOutput/07_Dec_17_35_53
## Completed.
```

```
########################################################################
```

Batman-Input *BATMAN Input Files are Explained Here*

Description

batman gets input parameters and metabolite templates information from the input files explained here. The input files are in either folder ".../runBATMAN/BatmanInput" or folder "extdata" depending on batman arguments. The user can modify the parameter values in the following input files (do not change the name of these files): batmanOptions.txt, metabolitesList.csv, multi_data.csv, multi_data_user.csv, NMRdata.txt.

Arguments

batmanOptions.txt

Option file to be used by batman. A copy of this file in the output directory is used for batmanrerun. The parameters in batmanOptions.txt file are explained here with example input values. The parameters have to be listed in the particular order given here, and do not leave empty lines in between except beginning with the comment character "%". Please note that for version 1.0.9 and later, one more input line,

"Use specified chemical shift for spectra (chemShiftperSpectra.csv) file (1/0): 0",

is added at the end of this file. For earlier version users updated to this version, running batman will add the above input line at the end of the file if missing.

ppmRange - ppm ranges for analysis: (1.2, 1.6) (2.1, 2.8)

- Put each set of ppm range in a pair of parentheses in the same line, separate start and end ppm values with a comma, separate each set of ppm range with space. Note that, very small number of spectra variables may cause error in wavelet analysis, do not give very narrow ppm ranges and also check the "Down sampling:" factor below, which used together, may also left very small number of spectra variables.

specNo - Ranges of spectra number to be included (e.g. 1,3-4 etc.): 1-3, 5

- Integer, if no. > 1 and fixed effect (same concentration for all spectra) is 0, user will be asked to choose whether to parallelize fittings between spectra when running batman or rerunbatman.

negThresh - Truncation threshold for negative intensities: -0.5

- Spectrum intensity smaller than the lower limit will be replaced by the lower limit.

scaleFac - Intensity scale factor: 20000

- The whole spectrum will be divided by the normalisation factor.

downSamp - Down sampling factor: 3

- Integer, number of spectra variable will be reduced by the factor of the input parameter, 3, in this case. For the example shown, the spectra variables with the index $1:3:end$ will be used for analysis.

`hiresFlag - Save metabolite fit at resolution of original spectrum? (Yes - 1 / No - 0): 1`

- Whether to save the metabolites fitting result in the original resolution without down sampling. Input 1 for yes, and 0 for no.

`randSeed - Random number seed: 25`

- Random number generation seed, integer.

`nItBurnin - Number of burn-in iterations: 4000`

- Integer, this is the number of burn-in iterations. The number of iterations after burn in will be asked when running `batman`. If changing the range of spectrum causing fitting results inconsistent, this indicates that the burn in stage hasn't found the best chemical shift. User may need to increase burn in iterations or reduce prior truncation on ppm shift for each multiple (adjust parameter "rdelta" below or use "csFlag" function).

`nItPostBurnin - Number of post-burn-in iterations: 1000`

- Integer, this is the number of post-burn-in iterations. The posterior samples will be saved in the frequency specified by the next parameter.

`multFile - Choose template of multiplets file from options below: 2`

- Integer, choose a template file from the following options:
  1, The default template of multiplets in multi_data.csv file,
  2, The user input template of multiplets in multi_data_user.csv file,
  3, Both the default and user input template of multiplets files.

`thinning - Save MCMC state in every ? iterations: 5`

- Integer, save posterior samples for every 5 iterations.

`cfeFlag - Same concentration for all spectra (fixed effect)? (Yes - 1 / No - 0): 0`

- Whether all the input spectra have the same metabolite concentrations (e.g. technical replicates). Input 1 for yes, and 0 for no.

`nItRerun - Number of iterations for batmanrerun: 5000`

- Integer, this is the number of iterations for `batmanrerun`. The rerun will use fixed multiplets positions obtained from running `batman`. There is no burnin for batman rerun.

`startTemp - Start temperature: 1000`

- Sets the start temperature parameter of the likelihood of tempering. Higher temperature may need more burnin iterations to cool down.

`specFreq - Spectrometer frequency (MHz): 600`

- Spectrometer used to collect the spectrum.

`a - Gamma-distributed with shape a: 0.00001`

`b - Gamma-distributed with scale b: 0.000000001`

- Hyper parameters for the global precision priors ($\lambda \sim Gamma(a, b/2)$) on wavelet coefficients.

`muMean - Mean of prior on global peak width (mu) in ln(Hz): 0`

`muVar - Variance of prior on global peak width (mu) in ln(Hz): 0.1`

`muVar_prop - Variance of proposal distribution for mu in ln(Hz): 0.002`

`nuMVar - Variance of prior on peak width offset (nu_m) in ln(Hz): 0.0025`

`nuMVarProp - Variance of proposal distribution for nu_m in ln(Hz): 0.0001`

- For peak width, γ, in Hz of metabolite m, the model for γ is $\ln(\gamma) = \mu + \nu_m$ where μ is the spectrum wide average log-peakwidth and ν_m is a random effect on metabolite deviation from μ. The mean of each prior on ν_m is 0. Set the variance of the prior on ν_m to 0 to turn off the random effect on peak width to keep peaks at the same width. The user can keep the proposal variance parameters unchanged for most of the case.

`tauMean - mean of the prior on tau: -0.01`

- Hyper priors (τ) on negative wavelet coefficient (truncated normal). A more negative value means the wavelet fit will have more negative component.

`tauPrec - inverse of variance of prior on tau: 2`

- This parameter is inversely proportional to the variance of the prior on τ.

`rdelta - Truncation of the prior on peak shift (ppm): 0.030`

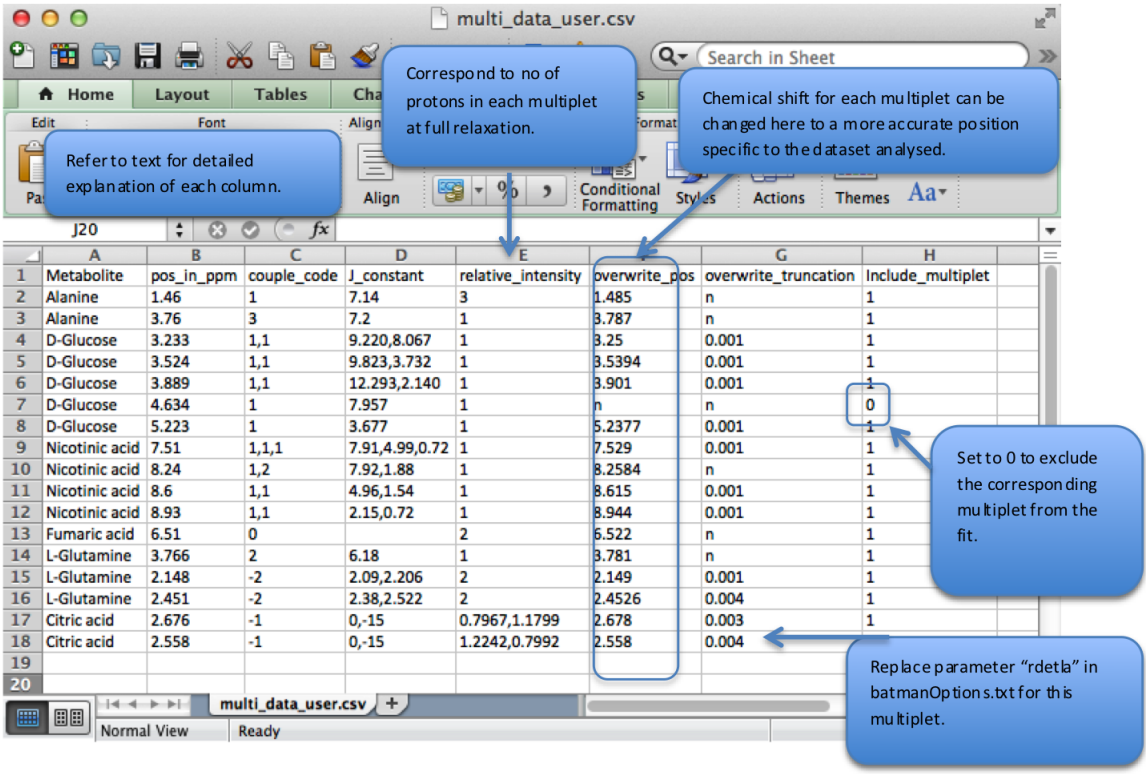
- Prior of the truncation on ppm shift for all multiplets, individual prior for each multiplet can be changed in the "multi_data.csv" file. Increase this parameter to allow multiplets to shift more. Please note, increasing this value may need more burn in iterations to find the best chemical shift for multiplets.

`csFlag - Specify chemical shift for each multiplet in each spectrum?`
`(chemShiftperSpectra.csv file) (Yes - 1 / No - 0): 0`

- Input "1" to use file "chemShiftperSpectra.csv" to specify chemical shift per multiplet and per spectrum. Input "0" will not use that file. User can use the MATLAB tool "SplineFitBATMAN" provided to get more accurate chemical shift per spectra for each multiplet. This tool will save chemical shift information into "chemShiftperSpectra.csv".

`metabolitesList.csv`
: List of metabolite names to be fitted. Put "%" in front of the metabolite name to comment out any metabolite for batman analysis.

`multi_data.csv`
: Multiplet template parameters file, obtained from the online Human Metabolome Database (HMDB) version 2.5. The user can modify the parameters in the template file and specify ppm positions, and normal distribution truncation of ppm shift parameters (a positive value applied as +/- on the distribution).



	A	B	C	D	E	F	G	H
1	Metabolite	pos_in_ppm	couple_code	J_constant	relative_intensity	overwrite_pos	overwrite_truncation	Include_multiplet
2	Alanine	1.46	1	7.14	3	1.485	n	1
3	Alanine	3.76	3	7.2	1	3.787	n	1
4	D-Glucose	3.233	1,1	9.220,8.067	1	3.25	0.001	1
5	D-Glucose	3.524	1,1	9.823,3.732	1	3.5394	0.001	1
6	D-Glucose	3.889	1,1	12.293,2.140	1	3.901	0.001	1
7	D-Glucose	4.634	1	7.957	1	n	n	0
8	D-Glucose	5.223	1	3.677	1	5.2377	0.001	1
9	Nicotinic acid	7.51	1,1,1	7.91,4.99,0.72	1	7.529	0.001	1
10	Nicotinic acid	8.24	1,2	7.92,1.88	1	8.2584	n	1
11	Nicotinic acid	8.6	1,1	4.96,1.54	1	8.615	0.001	1
12	Nicotinic acid	8.93	1,1	2.15,0.72	1	8.944	0.001	1
13	Fumaric acid	6.51	0		2	6.522	n	1
14	L-Glutamine	3.766	2	6.18	1	3.781	n	1
15	L-Glutamine	2.148	-2	2.09,2.206	2	2.149	0.001	1
16	L-Glutamine	2.451	-2	2.38,2.522	2	2.4526	0.004	1
17	Citric acid	2.676	-1	0,-15	0.7967,1.1799	2.678	0.003	1
18	Citric acid	2.558	-1	0,-15	1.2242,0.7992	2.558	0.004	1
19								
20								

The columns are:

`Metabolite`: The name of metabolite the multiplets belongs to.

`pos_in_ppm`: The ppm position of the center of the multiplets. Refer to the next two parameters for more explanation. If the next parameter "couple_code" was set to "-1" (empirical multiplet), this corresponds to the ppm position of the "0" Hz offset of the "J_constant". If the "couple_code" was set to "-2" (raster multiplet), this corresponds to the ppm position of the center of the raster multiplet. More details of using empirical multiplet and raster multiplet can be found in the following 3 fields.

`couple_code`: Coupling code. 0 = singlet, 1 = doublet, 2 = triplet, 3 = quartet, 4 = quintet, 5 = sextet, 6 = septet, 1,1 = doublet of doublets, 1,2 = doublet of triplets, 2,1 = triplet of doublets, 2,2 = triplet of triplets, 1,3 = doublet of quartets, 3,1 = quartet of doublets, 1,1,1 = doublet of doublet of doublets, 2,3 = triplet of quartets, 3,2 = quartet of triplets, 3,3 = quartet of quartets. If "-1" is inputted here, a user specified empirical multiplet can be created. An example can be found in file "multi_data_user.csv". If "-2" is inputted here, a raster multiplet with range specified in ppm in the field "J_constant" is used. Examples can be found in file "multi_data_user.csv".

`J_constant`: J constant.

If the empirical multiplet is used ("couple_code" is "-1"), J_constant contains the offsets in Hz for peaks (each peak corresponds to a offset in Hz, offsets are separated by comma) of a mutiplet positioned at "pos_in_ppm", J_constant/f (f is the magnet frequency in Hz) is the offset of peak in ppm. Note that the spectra are shown in reverse ppm axis, so a positive offset means peak at higher ppm value, and a negative offset is peak at lower ppm value.

If the raster multiplet is used ("couple_code" is "-2"), the field here requires a two values input (in ppm) separated by comma, which specifies the range of the raster multiplet in the pure spectrum. Note in this case, the field "Metabolite" will also be the .txt file name containing the pure spectrum (refer to `createPureSpectraTemplate`).

`relative_intensity`: (previously called no_of_protons) In the ideal case, at full relaxation, it should correspond to the number of protons in each multiplet. If the empirical multiplet ("couple_code" is "-1"), the same number of values (corresponding to each offset in "J_constant"") needed here as peak intensities. In this case, the sum of "relative_intensity" is the number of protons in this multiplet. If the raster multiplet is used ("couple_code" is "-2"), a single value is needed here corresponds to the number of protons in the included raster multiplet at full relaxation.

`overwrite_pos`: The default is "n" for not overwrite position, and in that case the value in "pos_in_ppm" is used for each multiplet. If user want to use a different value from "pos_in_ppm", it should be put in this column.

`overwrite_truncation`: The default is "n", and the default truncation value is obtained from the user input truncation on ppm shift (rdelta) in `batmanOptions.txt`. If the user wants to use different truncations for specific multiplets, it should be put in this column. This value will be used to calculate the ppm shift variance value (truncation/5) for the corresponding multiplets.

`Include_multiplet`: The default is "1" and all multiplets belong to the listed metabolites will be used. Set to "0" to exclude certain multiplet(s) from listed metabolite(s).

`multi_data_user.csv`

Metabolite template parameters file for user to add new metabolites in the same format as `multi_data.csv`.

`NMRdata.txt` The file has ppm value as its first column, and real part of the NMR spectrum in each of the subsequent columns. This file will be used when none of the input data argument is given.

`Batman-Output` *BATMAN Outputs are Explained Here*

Description

`batman` and `batmanrerun` return the results as a data list with the objects described in their individual function. They also put results in .txt format in a folder named after the start execution time ($date\_month\_hours\_mins\_seconds$) within either folder ".../runBATMAN/BatmanOutput" or folder "extdata" depending on `batman` input `createDir` settings.

Value

batman and batmanrerun save their results in the following files in the output folder:

beta_i_rr_j.txt
: A column vector ($m \times 1$) containing the estimated posterior mean of relative concentrations for m fitted metabolites of spectrum i. For batman results, j is 0, and for batmanrerun results, j is 1.

beta_sam_i_rr_j.txt
: A matrix ($m \times s$) with each row containing the s posterior samples of the relative concentrations for one fitted metabolite of spectrum i. m is the total number of fitted metabolites. For batman results, j is 0, and for batmanrerun results, j is 1.

delta_draw_mean_i.txt
: A column vector ($M \times 1$) containing the posterior mean of M multiplets ppm shift from the pre-set ppm position value in multi_data.csv or multi_data_user.csv of spectrum i.

delta_sam_i.txt
: A matrix ($s \times M$) containing the posterior samples of M multiplets ppm shift. Every column correspond the shift posterior samples of one multiplet for spectrum i. M is the sum of all multiplets in the fitted metabolites.

L_i.txt
: A matrix ($t \times M$) with each column as the template of one fitted metabolite for spectrum i before fitting. t is the number of data points in each spectrum.

lambda_sam_i_rr_j.txt
: A column vector ($s \times 1$) containing s posterior samples of λ (a scalar global precision parameter) for spectrum i. For batman results, j is 0, and for batmanrerun results, j is 1.

metabolitesListUsed.txt
: A column vector ($m \times 1$) containing the m metabolite names which have multiplets in/near the ppm region specified in batmanOptions.txt and used in the fitting.

metaFit_sam_i_rr_j.txt
: A matrix ($t \times s$) containing s posterior samples of total metabolites fit during MCMC iterations in its columns for spectrum i. t is the number of data points in each spectrum. For batman results, j is 0, and for batmanrerun results, j is 1.

metaIndFit_sam_i_rr_j.txt
: A matrix ($t \times (m * s)$) containing s posterior samples of m individual metabolites fit in its columns for spectrum i. t is the number of data points in each spectrum. Every m columns are the m individual metabolite fit samples for one posterior sample. For batman results, j is 0, and for batmanrerun results, j is 1.

metaTemp_i_rr_j.txt
: A matrix $t \times m$) containing the posterior means of m fitted metabolite templates in its columns (down sampled) after burn in for spectra i. t is the number of data points in each spectrum. For batman results, j is 0, and for batmanrerun results, j is 1.

`metaTempHR_`i`_rr_`j`.txt`
: A matrix ($t \times m$) containing the posterior means of m fitted metabolite templates in its columns (without down sample) after burn in for spectra i. t is the number of data points (without down sample) in each spectrum. For `batman` results, j is 0, and for `batmanrerun` results, j is 1.

`MultipletsPpmShifts.txt`
: A table ($M \times n$) containing the posterior means of multiplets ppm shift for M multiplets as its rows. M is the sum of all multiplets in the fitted metabolites and n is the number of spectra analyzed.

`NMRdata_mod_`i`.txt`
: A matrix ($t \times 2$) containing the input spectrum i in its original resolution. The first column is ppm value, and the second column is the ith spectrum intensity.

`RelCon.txt`
: A table ($m \times n$) of the posterior means of relative concentrations for m fitted metabolites and n spectra.

`RelConCreInt.txt`
: A table ($m \times 2n$) containing the 95% credible intervals (2.5% and 97.5%) for the relative concentrations of m fitted metabolites for n spectra.

`specFit_`i`_rr_`j`.txt`
: A matrix ($t \times 5$) of BATMAN fit results with five columns as:

$$[ppm, Originalspectrum, Metabolitesfit, Waveletfit, Overallsum]$$

of spectrum i. For `batman` results, j is 0, and for `batmanrerun` results, j is 1.

`specFitHR_`i`_rr_`j`.txt`
: A column vector ($t \times 1$) of metabolite fit result in the original resolution for spectrum i. t is the number of data points (without down sample) in each spectrum. For `batman` results, j is 0, and for `batmanrerun` results, j is 1.

`theta_sam_`i`_rr_`j`.txt`
: A matrix ($t \times s$) containing s samples of wavelet fit during MCMC iterations in its columns for spectrum i. For `batman` results, j is 0, and for `batmanrerun` results, j is 1.

`batmanOptions.txt`
: The same file copied from batman input. This file will be used by `batmanrerun`.

`metabolitesList.txt`
: The same file copied from batman input.

`NMRdata.txt`
: The same file copied from batman input.

If any plotting is performed, pdf files of the figure will be saved. For details, please refer to each plotting functions.

batmanrerun *Perform BATMAN with Fixed (Previously Estimated) Multiplet Positions*

Description

This performs metabolite and wavelet fitting to input NMR spectra with fixed multiplet position obtained from running batman, and also plots fitting results. The user should modify parameters in the copy file "batmanOptions.txt" in batman output folder to change the rerun settings.

Usage

```
batmanrerun(BM, figBatmanFit = TRUE, listMeta = FALSE,
            figRelCon = FALSE, figMetaFit = FALSE, showPlot = TRUE)
```

Arguments

BM	batman output data frame.
figBatmanFit	Plot metabolites and wavelets fit if set TRUE. The default is TRUE.
listMeta	Individual metabolite fit will also be shown in the plot if set TRUE. The default is FALSE.
figRelCon	Plot posterior samples of the relative concentration for listed metabolites with 95% credible interval if set TRUE. The default is FALSE.
figMetaFit	If set TRUE, plot the posterior mean of the metabolites fit with 95% credible interval. The default is FALSE.
showPlot	If set FALSE, no plot will be shown on display, the pdf file(s) for the plot(s) will be created in output folder. The default is TRUE.

Value

When batmanrerun is called with multiplet ppm shifts fixed from the batman results, the following objects are added to the batman result:

sFitRerun	A matrix $t \times 5n$ of BATMAN rerun fit results (down sampled). For 1 spectrum, it is a matrix with 5 columns: $$[ppm, originalspectrum, metabolitesfit, waveletsfit, overallfit].$$ n is the number of spectra, and t is the number of data points in each spectrum.
sFitRerunHR	A matrix $t \times 3n$ of BATMAN rerun fit results in the original resolution (without down sample). For 1 spectrum, it is a matrix with 3 columns: $$[ppm, originalspectrum, metabolitesfit].$$ n is the number of spectra, and t is the number of data points (without down sample) in each spectrum.

betaRerun — For batman rerun, a matrix ($m \times n$) containing the posterior means of relative concentrations for m fitted metabolites and n spectra.

betaSamRerun — For batman rerun, a matrix ($m \times (s * n)$) containing (for the first spectrum) s posterior samples of the relative concentrations in its rows. m is the number of fitted metabolites. n is the number of spectra analyzed. The subsequent columns contain the same data format for the rest $n - 1$ spectra.

betaCIRerun — For batman rerun, a matrix ($m \times 2n$) containing the 95% credible interval of the relative concentrations for m fitted metabolites. Every pair of columns is for one spectrum.

metaTempRerun — For batman rerun, a matrix ($t \times (m * n)$) containing the posterior means of m fitted metabolite templates in its columns (down sampled). n is the number of spectra analyzed and t is the number of data points in each spectrum.

metaTempRerunHR — For batman rerun, a matrix ($t \times (m * n)$) containing the posterior means of m fitted metabolite templates in its columns (without down sample). n is the number of spectra analyzed and t is the number of data points (without down sample) in each spectrum.

metaFitSamRerun — For batman rerun, a matrix ($t \times (s * n)$) containing s posterior samples of total metabolites fit in its columns. n is the number of spectra analyzed and t is the number of data points in each spectrum. The remaining $n-1$ spectra metabolites fit results are saved in the same sequence in subsequent columns.

metaIndFitSamRerun — For batman rerun, a matrix ($t \times (m * s * n)$) containing s posterior samples of m individual metabolites fit in its columns. n is the number of spectra analyzed and t is the number of data points in each spectrum. The remaining $n-1$ spectra results are saved in the same sequence in subsequent columns.

thetaSamRerun — For batman rerun, a matrix ($t \times (s * n)$) containing s samples of wavelet fit in its columns. n is the number of spectra analyzed. The remaining $n - 1$ spectra wavelet fit results are saved in the same sequence in subsequent columns.

outputDir — The directory of output folder with all the output result files.

See Also

batman, readBatmanOutput

Examples

```
library(batman)
## Run batman
if(interactive())
{
bm<-batman()
## then call batmanrerun
bm<-batmanrerun(bm)
}
#########################################################################
## The following is an example of what will be displayed in R
```

```
## and what value the user could input:
########################################################################
## Rerunning batman for  500  iterations.
## percentage completed...
##   |                                                                  |   0%
## Size of each spectrum is 382.
## Size of metabolite list is 22.
## Constructing chain data structure...
## time used is 1 seconds.
## Running MCMC...
##   |==================================================================| 100%
## time used is 65 seconds.
## saving posteriors...
##
## For rerun, time elapsed
##  65.96 seconds.
## Reading in saved data in folder
## .../user_specified_dir/runBATMAN/BatmanOutput/07_Dec_17_35_53
## Completed.
########################################################################
## Alternatively if more than 1 spectrum are included without using fixed
## effect, user will be asked to input whether to parallelize the analysis
## between spectra.
########################################################################
## How many parallel processes (multicores) do you want to run the
## multi-spectra analysis?
## (Enter 1 for running them sequentially.)
##
## Parallel processing of multi spectra currently cannot display
## the progress bar (or any words), if you input is > 1, please be patient
## for the results :)
##
## 1: 2  ## user input
##
## For rerun, time elapsed
##   64.4
## Reading in saved data in folder
## .../user_specified_dir/runBATMAN/BatmanOutput/07_Dec_17_35_53
## Completed.
########################################################################
```

---

checkBatmanOptions *Check previouse versions of batmanOptions.txt file and unify the parameter names to the current one.*

Description

Check batmanOptions.txt file and may add a new input line at the end of the file for old versions.

Usage

```
checkBatmanOptions(dir)
```

Arguments

dir	The directory of batmanOptions.txt file.

Examples

```
library(batman)
## createfolder "runBATMAN" in current working directory
batmanDir = newDir(runBATMANDir = getwd(), overwriteFile = TRUE)
checkBatmanOptions(dir = paste(batmanDir[2], "/batmanOptions.txt", sep = ""))
```

---

createChemShiftPerSpec

Creating the file chemShiftPerSpec.csv which contains chemical shift parameters for all multiplets and spectra.

Description

This function creates a file called chemShiftPerSpec.csv, so user can specify chemical shift parameter for each spectrum and multiplet. The first column is multiplet names in the same order as the template inputs in multi_data.csv and/or multi_data_user.csv (depending on user choice of using one or both of them) file(s). The second column is the default chemical shift value (pos_in_ppm) for the corresponding multiplet. From the third column forward is the chemical shift value for each spectrum in the same order as they read in by BATMAN, if 'n' is present in the field, the default chemical shift value (or overwrite_pos value if given) will be used.

Usage

```
createChemShiftPerSpec(templateOption, dirIP)
```

Arguments

templateOption	Choose template file(s). `templateOption = 1` for multi_data.csv, `templateOption = 2` for multi_data_user.csv, and `templateOption = 3` for both files.
dirIP	The input directory of BATMAN. This is the path ending with '/BatmanInput' if runBATMAN directory is created.

See Also

batman

Examples

```
library(batman)
## createfolder "runBATMAN" in current working directory
batmanDir = newDir(runBATMANDir = getwd(), overwriteFile = TRUE)
## create chemShiftPerSpec.csv
createChemShiftPerSpec(templateOption = 1, dirIP = batmanDir[2])
```

`createPureSpectraTemplate` *Creating a folder called 'PureSpectraTemplate' in the specified input directory. The folder contains pure metabolite spectrum template in .txt file with metabolite name as the file name.*

Description

This function will read in pure metabolites spectra in Bruker format and save them in .txt format in folder "PureSpectraTemplate". The .txt file name is the same as the input to "metaNames". The "PureSpectraTemplate" folder will be used if a raster multiplet is used ("couple_code" value in multi_data.csv and/or multi_data_user.csv is set to "-2").

Usage

```
createPureSpectraTemplate(dirPureSpec, metaNames, dirIP)
```

Arguments

`dirPureSpec`	A vector containing the directories of Bruker pure metabolite spectra files.
`metaNames`	The vector of metabolites names in the same order as the spectra directories in `dirPureSpec`.
`dirIP`	The input directory of BATMAN. This is the path ending with '/runBATMAN/BatmanInput' if runBATMAN directory is created.

Examples

```
library(batman)
## createfolder "runBATMAN" in current working directory
batmanDir = newDir(runBATMANDir = getwd(), overwriteFile = TRUE)
## create pure spectra text file, replace "/user/bruker/spectra/file?"
## with the directories of Bruker spectra files.
## createPureSpectraTemplate(dirPureSpec = c(/user/bruker/spectra/file1,
## /user/bruker/spectra/file2), metaNames = c("testPure1"), dirIP = batmanDir[2])
```

plotBatmanFit *Plot Batman Metabolite Fit of NMR Spectra (With Down Sampling)*

Description

This function plots the BATMAN fit results, and saves the figure to pdf file in specified directory. For multiple spectra analysis, the file name is in the format of "specFit_$i$to$j$_$metaName$.pdf", where i and j are the range numbers of spectra in the figure and the metabolite name will be shown in place of $metaName$ if supplied. Maximum of 2 spectra will be shown in each figure. The figure file will not be overwritten if it already exists by default. A prefix can be added to the file name for new saves.

Usage

```
plotBatmanFit(BM, xfrom, xto, yfrom, yto, listMeta = FALSE,
              metaName, saveFig = TRUE, saveFigDir = BM$outputDir,
              prefixFig, rerun = FALSE, placeLegend,
              plotColour, overwriteFig = FALSE,
              showPlot = TRUE)
```

Arguments

BM	batman output data frame.
xfrom	The start ppm value to plot. Default is set to the start ppm value of the whole processed range.
xto	The end ppm value to plot. Default is set to the end ppm value of the whole processed range.
yfrom	The start value of vertical axis to plot. Default is set to 0.
yto	The end value of vertical axis to plot. Default is set to the maximum value of the spectrum point in display.
listMeta	Individual metabolite fit will also be shown in the plot if set TRUE. The default is FALSE.
metaName	One or more specified metabolite fits will be shown in the plot. If no name was given and listMeta = TRUE, all the individual metabolite fit will be shown.
saveFig	Save figure(s) to pdf file(s) if set TRUE. The default is TRUE.
saveFigDir	Save figure(s) in this directory. The default is output directory of BM.
prefixFig	Add prefix to each saved figure name. The default is no prefix.
rerun	Set to FALSE to plot batman result, and TRUE to plot batmanrerun result.
placeLegend	Where to place the legend in figure. The default is "topright".
plotColour	User can specify colours for each metabolite if listMeta = TRUE. If not, a set of randomly generated colours will be used.
overwriteFig	Overwrite saved figure file in pdf format if overwriteFig = TRUE. The default is FALSE.
showPlot	If set FALSE, no plot will be shown on display, the pdf file(s) for the plot(s) will be created in output folder. The default is TRUE.

See Also

batman, batmanrerun

Examples

```
library(batman)
## Run BATMAN
if(interactive())
{
bm<-batman()
## then plot results
plotBatmanFit(bm)
}
```

---

plotBatmanFitHR *Plot BATMAN Metabolite Fit of NMR Spectra in Original Resolution (Without Down Sampling)*

Description

This function plots a high resolution BATMAN fit results (without down sampling), and save figure to pdf file in user specified directory. For multiple spectra analysis, the file name is in the format of "specFitHR_i_$metaName$.pdf", where i is the spectrum number in the figure and the metabolite name will be shown in place of $metaName$ if supplied. The figure file will not be overwritten if it already exists. A prefix can be given to the file name for new saves.

Usage

```
plotBatmanFitHR(BM, xfrom, xto, yfrom, yto, metaName, saveFig = TRUE,
                saveFigDir = BM$outputDir, prefixFig, rerun = FALSE,
                overwriteFig = FALSE, showPlot = TRUE)
```

Arguments

BM	batman output data frame.
xfrom	The start ppm value to plot. Default is set to the start ppm value of the whole processed range.
xto	The end ppm value to plot. Default is set to the end ppm value of the whole processed range.
yfrom	The start value of vertical axis to plot. Default is set to 0.
yto	The end value of vertical axis to plot. Default is set to the maximum value of the spectrum point in display.
metaName	Individual metabolite fit will also be shown in the plot if a metabolite name is given. Only one metabolite name can be given, if missing from input all metabolites will be plotted.
saveFig	Save figure(s) to pdf file(s) if set TRUE. The default is TRUE.

saveFigDir	Save figure(s) in this directory. The default is the output directory of BM.
prefixFig	Add prefix to each saved figure name. The default is no prefix.
rerun	Set to FALSE to plot batman result, and TRUE to plot batmanrerun result.
overwriteFig	Overwrite saved figure file in pdf format if overwriteFig = TRUE. The default is FALSE.
showPlot	If set FALSE, no plot will be shown on display, the pdf file(s) for the plot(s) will be created in output folder. The default is TRUE.

See Also

batman, batmanrerun

Examples

```
library(batman)
## Run BATMAN fit
if(interactive())
{
bm<-batman()
## Plot batman Fit in its original resolution if the option parameter
## is set to 1 for "Save metabolites fit same as the original spectrum
## resolution (1/0)" in "batmanOptions.txt", .
plotBatmanFitHR(bm)
}
```

---

plotBatmanFitStack *Stack plot Batman Metabolite Fit of NMR Spectra (With Down Sampling)*

Description

This function plots the BATMAN fit results in stack, and saves the figure to pdf file in specified directory. User can choose to plot all or some of the spectra analyzed, show the metabolite fit and give a range for x and y limits. The figure file will not be overwritten if it already exists by default. A prefix can be added to the file name for new saves.

Usage

```
plotBatmanFitStack(BM, offset = 1, mirroredWav = TRUE, specNo,  xfrom,
                   xto, yfrom, yto, listMeta = FALSE, metaName,
                   saveFig = TRUE, saveFigDir = BM$outputDir,
                   prefixFig, rerun = FALSE, placeLegend = "topright",
                   plotColour, overwriteFig = FALSE,
                   metaLwd = 2, metaLty = 5, orientation = "L",
                   showPlot = TRUE)
```

Arguments

BM	batman output data frame.
offset	Offset value for the stack plot.
mirroredWav	Plot mirrored wavelet fit if set TRUE. The default is TRUE
specNo	Vector of spectra ID in the input order to specify which spectra to be plotted in stack plot.
xfrom	The start ppm value to plot. Default is set to the start ppm value of the whole processed range.
xto	The end ppm value to plot. Default is set to the end ppm value of the whole processed range.
yfrom	The start value of vertical axis to plot. Default is set to 0.
yto	The end value of vertical axis to plot. Default is set to the maximum value of the spectrum point in display.
listMeta	Individual metabolite fit will also be shown in the plot if set TRUE. The default is FALSE.
metaName	One or more specified metabolite fits will be shown in the plot. If no name was given and listMeta = TRUE, all the individual metabolite fit will be shown.
saveFig	Save figure to pdf file if set TRUE. The default is TRUE.
saveFigDir	Save figure in this directory. The default is output directory of BM.
prefixFig	Add prefix to each saved figure name. The default is no prefix.
rerun	Set to FALSE to plot batman result, and TRUE to plot batmanrerun result.
placeLegend	Where to place the legend in figure. The default is "topright".
plotColour	User can specify colours for each metabolite if listMeta = TRUE. If not, a set of randomly generated colours will be used.
overwriteFig	Overwrite saved figure file in pdf format if overwriteFig = TRUE. The default is FALSE.
metaLwd	The line widths for metabolite fit.
metaLty	The line types for metabolite fit.
orientation	The orientation of plot, either portrait or landscape, the defult is "L".
showPlot	If set FALSE, no plot will be shown on display, the pdf file(s) for the plot(s) will be created in output folder. The default is TRUE.

Examples

```
library(batman)
## Run BATMAN
if(interactive())
{
bm<-batman()
## then plot results
plotBatmanFitStack(bm)
}
```

plotChemShiftDist *Plot histogram of chemical shifts for the multiplets across a series of spectra.*

Description

This function plots the histogram of the mean posterior estimated chemical shifts for the multiplets of certain or all metabolites across a series of spectra. User can choose to plot all or some of the metabolite. The figure file will not be overwritten if it already exists by default. A prefix can be added to the file name for new saves.

Usage

```
plotChemShiftDist(BM, metaName, breaks = 20, xlim,
                  saveFig = TRUE, saveFigDir = BM$outputDir,
                  prefixFig, overwriteFig = FALSE,
                  showPlot = TRUE)
```

Arguments

BM	batman output data frame.
metaName	One or more specified metabolites will be shown. If no name was given, all the individual metabolites will be shown.
breaks	A single number to set the number of bins for the histogram.
xlim	The range of x values.
saveFig	Save figure to pdf file if set TRUE. The default is TRUE.
saveFigDir	Save figure in this directory. The default is output directory of BM.
prefixFig	Add prefix to each saved figure name. The default is no prefix.
overwriteFig	Overwrite saved figure file in pdf format if overwriteFig = TRUE. The default is FALSE.
showPlot	If set FALSE, no plot will be shown on display, the pdf file(s) for the plot(s) will be created in output folder. The default is TRUE.

Examples

```
library(batman)
## Run BATMAN
if(interactive())
{
bm<-batman()
## then plot results
plotChemShiftDist(bm)
}
```

plotDiagnosticScatter *Diagnostic scatter plot of batman metabolites fit vs NMR spectra bins or minimum wavelet fit.*

Description

When fitting a large number of spectra, this plot facilitates discovery of spectra or metabolites which are poorly fit.

Usage

```
plotDiagnosticScatter(BM, binWidth = 0.018, cexID = 0.5, saveFig = TRUE,
                      saveFigDir = BM$outputDir, prefixFig,
                      rerun = FALSE, placeLegend = "topright",
                      overwriteFig = FALSE, showPlot = TRUE)
```

Arguments

BM	batman output data frame.
binWidth	The full width of the bins to integrate. The centre of a bin is the estimated mean posterior chemical shift for each multiplet in each spectrum.
cexID	Character size for the spectra ID number.
saveFig	Save figure to pdf file if set TRUE. The default is TRUE.
saveFigDir	Save figure in this directory. The default is output directory of BM.
prefixFig	Add prefix to each saved figure name. The default is no prefix.
rerun	Set to FALSE to plot batman result, and TRUE to plot batmanrerun result.
placeLegend	Where to place the legend in figure. The default is "topright".
overwriteFig	Overwrite saved figure file in pdf format if overwriteFig = TRUE. The default is FALSE.
showPlot	If set FALSE, no plot will be shown on display, the pdf file(s) for the plot(s) will be created in output folder. The default is TRUE.

Examples

```
library(batman)
## Run BATMAN
if(interactive())
{
bm<-batman()
## then plot results
plotDiagnosticScatter(bm)
}
```

plotMetaFit *Plot Posterior Means of Metabolites Fit with 95% Credible Interval*

Description

This function plots posterior means of the metabolite fit with 95% credible interval , and saves the figure to pdf file in specified directory. For multiple metabolites, the file name is in the format of "spec_$i$to$j$_mFitSam.pdf", where i and j are range numbers of spectra in the figure. A maximum of 2 spectra will be shown in each figure. Figure file will not be overwritten if it already exists. Prefix can be added to the file name for new saves.

Usage

```
plotMetaFit(BM, from, to, metaName, saveFig = TRUE,
            saveFigDir = BM$outputDir, prefixFig,
            rerun = FALSE, overwriteFig = FALSE,
            showPlot = TRUE)
```

Arguments

BM	batman output data frame.
from	The start ppm value to plot. Default is set to the start ppm value of the whole processed range.
to	The end ppm value to plot. Default is set to the end ppm value of the whole processed range.
metaName	Only multiplets belonging to the named Metabolite will be shown. Only one metabolite name can be given. If missing, all metabolites will be plotted.
saveFig	Save figure to pdf file if set TRUE. The default is TRUE.
saveFigDir	Save figure in this directory. The default is current working directory.
prefixFig	Add prefix to each saved figure name. The default is no prefix.
rerun	Set to FALSE to plot batman result, and TRUE to plot batmanrerun result.
overwriteFig	Overwrite saved figure file in pdf format if overwriteFig = TRUE. The default is FALSE.
showPlot	If set FALSE, no plot will be shown on display, the pdf file(s) for the plot(s) will be created in output folder. The default is TRUE.

See Also

batman, batmanrerun

Examples

```
library(batman)
## Run BATMAN fit, then plot metabolite fit
if(interactive())
{
bm<-batman()
## Plot metabolites Fit.
plotMetaFit(bm)
}
```

---

plotRelCon *Boxplot or Histogram of Posterior distributions of Relative Concentrations for Listed Metabolites with 95% Credible Interval*

Description

This function plots the posterior distributions of relative concentrations, and saves the figure to pdf file. The file name is in the format of "spec_i_RelCon_$j1$to$j2$.pdf", where i are the spectrum numbers and $j1$ and $j2$ are the order numbers of fitted metabolites in the order of their input in file metaboliteList.csv. The figure file will not be overwritten if it already exists. A prefix can be added to file name for new saves.

Usage

```
plotRelCon(BM, metaName, plotHist = FALSE, breaks,
           saveFig = TRUE, saveFigDir = BM$outputDir,
           prefixFig, rerun = FALSE, overwriteFig = FALSE,
           showPlot = TRUE)
```

Arguments

BM	batman output data frame.
metaName	Only multiplets belonging to the named Metabolite will be shown. Only one metabolite name can be given. If missing, all metabolites will be plotted.
plotHist	If plotHist = TRUE, the ppm shift posteriors will be displayed as histogram. The default is FALSE.
breaks	A single number to set the number of bins for the histogram. If missing from the input, it is set to the data length divided by 3.
saveFig	Save figure(s) to pdf file(s) if set TRUE. The default is TRUE.
saveFigDir	Save figure(s) in this directory. The default is output directory of BM.
prefixFig	Add prefix to each saved figure name. The default is no prefix.
rerun	Set to FALSE to plot batman result, and TRUE to plot batmanrerun result.
overwriteFig	Overwrite saved figure file in pdf format if overwriteFig = TRUE. The default is FALSE.
showPlot	If set FALSE, no plot will be shown on display, the pdf file(s) for the plot(s) will be created in output folder. The default is TRUE.

See Also

batman, batmanrerun

Examples

```
library(batman)
## Run BATMAN and then plot relative concentration
if(interactive())
{
bm<-batman()
## Plot relative concentrations
plotRelCon(bm)
}
```

---

plotShift *Boxplot or Histogram of ppm Shift Posterior distributions for Multiplets of Named Metabolite*

Description

This function provides boxplots or histograms of the ppm shift posterior distributions of multiplets, and saves the figure to pdf file in specified directory. The file name is in the format of "spec_i_$metaName$_ppmShift.pdf", where i is the spectrum number and "metaName" is the input metabolite name if given. The figure file will not be overwritten if it already exists. A prefix can be given to the file name for new saves.

Usage

```
plotShift(BM, metaName, plotHist = FALSE, breaks, perMult = FALSE,
          saveFig = TRUE, saveFigDir = BM$outputDir, prefixFig,
          overwriteFig = FALSE, showPlot = TRUE)
```

Arguments

BM	batman output data frame.
metaName	Only multiplets belonging to the named Metabolite will be shown. Only one metabolite name can be given. If missing, all metabolites will be plotted.
plotHist	If plotHist = TRUE, the ppm shift posteriors will be displayed as histogram. The default is FALSE.
breaks	A single number to set the number of bins for the histogram. If missing from the input, it is set to the data length divided by 3.
perMult	If set TRUE plot the shifts per multiplet, otherwise, plot the shifts per spectrum.
saveFig	Save figure to pdf file if set TRUE. The default is TRUE.
saveFigDir	Save pdf file in this directory. The default is the output directory of BM.
prefixFig	Add prefix to each saved figure name. The default is no prefix.

overwriteFig	Overwrite saved figure file in pdf format if overwriteFig = TRUE. The default is FALSE.
showPlot	If set FALSE, no plot will be shown on display, the pdf file(s) for the plot(s) will be created in output folder. The default is TRUE.

See Also

batman, batmanrerun

Examples

```
library(batman)
## Run BATMAN
if(interactive())
{
bm<-batman()
## Plot ppm shift for each multiplet.
plotShift(bm)
}
```

---

readBatmanOutput *Reads in BATMAN Output Data Files*

Description

Reads in output data files from batman in specified folder.

Usage

```
readBatmanOutput(dirOP, dirIP,readMetaIndFitSam = TRUE,
                            readMetaTempHR = TRUE, readMetaTemp = TRUE)
```

Arguments

dirOP	The folder with batman output files.
dirIP	The folder with batman input files.
readMetaIndFitSam	If set TRUE, read in the posterior samples of individual metabolites fit.
readMetaTempHR	If set TRUE, read in the posterior means of fitted metabolite templates (without down sample).
readMetaTemp	If set TRUE, read in the posterior means of fitted metabolite templates (down sampled).

Value

It returns a data list with the objects described in batman.

See Also

batman, batmanrerun

Examples

```
library(batman)
## Run BATMAN
if(interactive())
{
bm<-batman()
## Read in output files in saved directory.
bmread<-readBatmanOutput(bm$outputDir,bm$inputDir)
}
```

---

readBruker *Read Raw Binary Bruker NMR Spectra*

Description

Read in multiple raw binary Bruker NMR spectra (1D) from a specified folder, and return a matrix with columns:

$$[ppm, spectrum1, spectrum2, ...].$$

Interpolation may be performed if spectra have different ppm scales.

Usage

```
readBruker(BrukerDataDir)
```

Arguments

BrukerDataDir	The directory of the folder containing 1D Bruker spectral data files. Recursively finds all the "1r" files in datapath and read in.

Value

It returns a matrix with columns:

$$[ppm, spectrum1, spectrum2, ...].$$

Examples

```
library(batman)
## Read in all Burker NMR spectra files, replace "/your/data/path/here" with the
## directory of the data files you want to read.
## brukerdata<-readBruker("/your/data/path/here")
```

readBrukerZipped *Read Raw Binary Bruker NMR Spectra in Zipped format*

Description

Read in multiple raw binary Bruker NMR spectra (1D), with spectrum data in a zipped format, from a specified folder, and return a matrix with columns:

$$[ppm, spectrum1, spectrum2, ...].$$

Interpolation may be performed if spectra have different ppm scales.

Usage

```
readBrukerZipped(BrukerDataZipDir)
```

Arguments

`BrukerDataZipDir`
: The directory of the folder containing zipped 1D Bruker spectral data files. Recursively finds all the "*.zip" files in `datapath`, unzipped them in the same folder, call "*.zip" files to read in spectra, and delete the unipped folders. If no "*.zip" file was found, it works the same as "*.zip" files.

Value

It returns a matrix same as `readBruker`, with columns:

$$[ppm, spectrum1, spectrum2, ...].$$

Examples

```
library(batman)
## Read in all Burker NMR spectra files, replace "/your/data/path/here" with the
## directory of the data files you want to read.
## brukerdata<-readBrukerZipped("/your/data/path/here")
```

saveBruker2Txt *Read Raw Binary Bruker NMR Spectra and save them to ASCII file.*

Description

Save the multiple raw binary Bruker NMR spectra (1D) from a specified folder into ASCII file as a matrix with columns:

$$[ppm, spectrum1, spectrum2, ...].$$

Interpolation may be performed if spectra have different ppm scales.

Usage

```
saveBruker2Txt(BrukerDataDir)
```

Arguments

BrukerDataDir	The directory of the folder containing 1D Bruker spectral data files. Recursively finds all the "1r" files in datapath and read in.
saveFileName	The saved file name with extension.

Value

It returns a matrix with columns:

$$[ppm, spectrum1, spectrum2, ...].$$

Examples

```
library(batman)
## Read in all Burker NMR spectra files, replace "/your/data/path/here" with the
## directory of the data files you want to read.
## brukerdata<-readBruker("/your/data/path/here")
```

Index