mrbin - Magnetic Resonance Binning, Integration and Normalization

Matthias S. Klein

Version 1.6.0

1 Introduction

Nuclear Magnetic Resonance (NMR) is widely used for metabolomics research. This package uses spectral binning to convert 1D or 2D NMR data into a matrix of values suitable for further data analysis and performs basic processing steps in a reproducible way. Negative values, a common issue in NMR data, are replaced by positive values. All used parameters are stored in a readable text file and can be restored from that file to enable exact reproduction of the data at a later time.

2 Citation

If you are using mrbin in a publication, please cite the following manuscript:

Klein, M.S. (2021): Affine Transformation of Negative Values for NMR Metabolomics Using the mrbin R Package. J. Proteome Res. 20(2):1397-1404, DOI: 10.1021/acs.jproteome.0c00684

3 Getting Started

The main functions of this package are controlled via the mrbin() function. Most other functions in the package will not usually be ever called by the user, but serve internal purposes. Results returned include the final bin list and a set of used parameters.

3.1 Installation

To install mrbin, please install the latest version of R first. Then install mrbin.

To install the latest stable version of mrbin from CRAN:

```
install.packages("mrbin")
```

To install the latest development version from Github:

```
library(devtools)
install_github("kleinomicslab/mrbin")
```

To be able to run devtools, you may need to install additional software.

After installation, load the package as follows:

```
> library(mrbin)
```

3.2 Prerequisites and Considerations

To use this package, you will need your NMR data in the Bruker file format accessible on your computer. Please make sure your data is Fourier transformed, phase corrected, baseline corrected, and correctly referenced. The data has to be stored in folders according to standard Bruker folders, that means foldername/1/pdata/1 etc. Experiment numbers and processing numbers can be freely chosen.

This package has been tested for 1D NOESY and 2D 1H-13C HSQC spectra.

Before starting mrbin, take a look at your NMR data, for example in Bruker Topspin, and decide on the following parameters (you will be able to see the values of these parameters in regards to your data during running mrbin):

- Bin area: Area where signals are observed in your data set
- Bin width: Should match roughly the width of a singlet peak in your data set. Given in ppm.
- Bin height (only 2D): Should match roughly the height of a singlet peak in your data set. Given in ppm.
- Solvent area: Area to exclude to remove solvent artifacts
- Additional areas to be removed: Any other area containing artifacts, such as streaks surrounding strong peaks.

mrbin will also show you preview plots for these parameters during the run.

3.3 Running mrbin in Interactive Mode

You can start mrbin using the following code:

```
mrbinResults<-mrbin()
```

This will start a series of questions that will guide you through the parameters to be used.

mrbin() returns an (invisible) list containing three variables:

- bins: A matrix containing bin data for all samples, Depending on the option you chose, the data will be cleaned up and scaled.
- parameters: A list containing all parameters used to create the bin matrix.
- factors: A vector containing group names for all samples.

Up to three files may be written to the chosen directory:

- A .txt file containing all parameters and potential warning messages from the mrbin run. This file can be reloaded to R using recreatemrbin("filename"). This will enable reusing parameters used in a previous run and can help increase reproducibility.
- A .csv file containing the bin data for use in other software tools.
- A .pdf file containing quality control plots, including a PCA plot

3.4 Recreating Data and Parameters

In order to create reproducible results, mrbin will save the used parameters to a text file. Please keep this file. You may want to share this file in a data repository when publishing your findings.

While it is fine to view the parameter text file in a text editor, please do not change its contents, as this may break its formatting.

In order to recreate a previous data set, or to reload previously used parameters, use:

```
mrbinResults<-mrbin()
```

and select ""Reload from file" when asked "Set parameters or use existing parameters?". This will restore all parameters that were previously used. If the file was created using an older version of mrbin, this may cause inconsistencies. Missing parameters will be added using standard parameters. Ideally, download the older mrbin version at kleinomicslab.com and use the old version to recreated the data in an exact way.

Please be aware that bins will have to be recalculated, so the original NMR spectra will have to be present to do this.

3.5 Setting Parameters at the Command Line

Parameters can be submitted at the command line. When using silent=TRUE, this will set up all parameters and run all steps without asking for user input.

When setting silent=FALSE, the user will be guided through the user input questionnaire to make adjustments to the submitted parameters.

3.5.1 Example: 1D Data

The following example provides all parameters for analyzing a 1D data set.

```
> mrbinResults<-mrbin(silent=TRUE,
+      setDefault=FALSE,
+      parameters=list(verbose=TRUE,
+              dimension="1D",
+              binMethod="Rectangular bins",
+              binwidth1D=.01,
+              referenceScaling="Yes",
+              removeSolvent="Yes",
+              removeAreas="No",
+              sumBins="No",
+              noiseRemoval="Yes",
+              PQNScaling="Yes",
+              fixNegatives="Yes",
+              logTrafo="Yes",
+              saveFiles="Yes",
+              NMRfolders=c(system.file("extdata/1/10/pdata/10",package="mrbin"),
+                           system.file("extdata/2/10/pdata/10",package="mrbin"),
+                           system.file("extdata/3/10/pdata/10",package="mrbin"))
+      ))
```

This example uses data from the mrbin package. To use your own data, add the full folder names of the Bruker folders holding the files "1r" (or "2rr" for 2D data) as follows:

```
NMRfolders=c("C:/Bruker/TopSpin3.6.1/data/guest/nmr/sample_1/10/pdata/10",
             "C:/Bruker/TopSpin3.6.1/data/guest/nmr/sample_2/10/pdata/10",
             "C:/Bruker/TopSpin3.6.1/data/guest/nmr/sample_3/10/pdata/10")
```

3.5.2 Example: 2D Data

The following example provides all parameters for analyzing a 2D data set.

```
> mrbinResults<-mrbin(silent=TRUE,
+      setDefault=FALSE,
+      parameters=list(verbose=TRUE,
+                dimension="2D",
+                binwidth2D=0.1,
+                binheight=3,
+                PQNScaling="No",
+                fixNegatives="No",
+                logTrafo="No",
+                signal_to_noise2D=20,
+                NMRfolders=c(system.file("extdata/1/12/pdata/10",package="mrbin"),
+                        system.file("extdata/2/12/pdata/10",package="mrbin"),
+                        system.file("extdata/3/12/pdata/10",package="mrbin"))
+                ))
```

This example uses data from the mrbin package. To use your own data, add the full folder names of the Bruker folders holding the files "1r" (or "2rr" for 2D data) as follows:

```
NMRfolders=c("C:/Bruker/TopSpin3.6.1/data/guest/nmr/sample_1/12/pdata/10",
             "C:/Bruker/TopSpin3.6.1/data/guest/nmr/sample_2/12/pdata/10",
             "C:/Bruker/TopSpin3.6.1/data/guest/nmr/sample_3/12/pdata/10")
```

The binned NMR data can be found in mrbinResults$bins. This numeric matrix contains bins in columns and samples in rows, and can be directly used for further data analysis.

The binned data can also be loaded from the hard drive later (if an output file was specified):

```
binData<-read.csv("C:/Users/User/Documents/mrbin_bins.csv",
                  check.names = FALSE, row.names=1)
```

Column names include left and right borders or each bin separated by commas. For 2D data, column names include left, right, top, and bottom border of each bin.

3.5.3 Example: Lipid Data Analysis

Analyzing lipid NMR signals can be accomplished using custom bin lists rather than a regular grid of bins. This can be done as in the following code:

```
> results <- mrbin(silent=TRUE,parameters=list(binMethod="Custom bin list",
+  dimension="1D",specialBinList=matrix(c(
+                                5.45,5.2,0,160,
+                                2.9,2.74,0,160,
+                                2.14,1.93,0,160,
+                                1.41,1.2,0,160,
+                                0.94,0.8,0,160,
+                                2.44,2.2,0,160,
+                                4.325,4.26,0,160
+                                ),ncol=4,byrow=TRUE,dimnames=list(c(
+                                "-CH=CH- Methene",
+                                "=CH-CH2-CH= Diallylic",
+                                "-CH2-CH=CH- Allylic",
+                                "-CH2- Methylene",
+                                "-CH3 Methyl",
+                                "COO-CH2-CH2- Methylene_to_carboxyl",
+                                "Glycerol"
+                                ),NULL)),
+  referenceScaling="Yes",reference1D=c(0.03,-0.03),removeSolvent="No",
+  removeAreas="No",sumBins="No",trimZeros="Yes",noiseRemoval="No",
+  PQNScaling="No",fixNegatives="Yes",logTrafo="No",defineGroups="No",PCA="Yes",
+  createBins="Yes",useAsNames="Folder names",saveFiles="No",verbose=TRUE,
+  NMRfolders=c(system.file("extdata/1/10/pdata/10",package="mrbin"),
+               system.file("extdata/2/10/pdata/10",package="mrbin"),
+               system.file("extdata/3/10/pdata/10",package="mrbin"))
+  ))
```

When using custom bin lists, each spectral data point may be part of multiple bins. For rectangular bin lists, each data point can only be counted once. The lipid signal areas used in this example are based on Klein et al, 2011.

3.6 mrbin Workflow

The sequence of data processing is as follows:

- Gathering all parameters from user
- Creating a set with coordinates of each bin
- Removing solvent region
- Removing additional regions
- Cropping of HSQC spectra to the region along the diagonal
- Summing or merging regions containing peaks with unstable positions such as citric acid
- Reading Bruker NMR data
- Scaling to reference region
- Binning

- Removal of bins containing mostly noise
- Replacement of negative values
- PQN transformation
- Log transform
- Plotting a quality control plot, including a PCA plot
- Saving bins, parameters and the plot to the hard drive

3.7 Affine Transformation of Negative Values

The function atnv replaces (column-wise) negative values by a small positive number. The number is calculated as an affine transformation to the range of the lowest positive number to 0,01*the lowest positive number (of this column). Ranks stay unchanged. Positive numbers are not altered.

If sample-wise noise levels are available, the median noise level of samples with negative values is calculated and replaces the lowest positive number in case it is smaller. If no noise data is available, the 1% percentile of all positive values in the data set is used as an estimate.

It is recommended to use this function AFTER noise removal and other data clean-up methods, as it may alter (reduce) the noise level of the binned data. If no NMR data and noise levels are provided as arguments, the function will use NMR data and noise levels from the global variables mrbin.env$bins and mrbin.env$mrbinTMP.

To use own (user provided) data:

```
atnv(NMRdataMatrix,noiseLevelVector)
```

To use current mrbin data from the internal memory, use atnv()) without parameters. This requires data loaded using mrbin(). This is usually not necessary as it is included in the mrbin work flow.

3.8 Plotting

The mrbin package contains basic plotting commands for 1D and 2D NMR data.

Most convenient is using the mrplot function, which is menu-based:

```
mrplot()
```

In mrplot, multiple spectra shown overlaid. If both 1D and 2D spectra are added, both are shown in region-matched plots, e.g. for metbolite identification purposes.

To use the more basic commands, you need to first load one NMR spectrum:

```
readBruker(dimension="1D",folder=system.file("extdata/1/10/pdata/10",package="mrbin"))
plotNMR()
```

The system.file command loads example data from the mrbin package. To use your own spectra, please adjust as follows:

```
readBruker(dimension="1D",folder="C:/Bruker/TopSpin3.6.1/data/guest/nmr/sample_1/12/pdata/10")
plotNMR()
```

For 2D data, this code reads as follows:

```
readBruker(dimension="2D",folder=system.file("extdata/1/12/pdata/10",package="mrbin"))
plotNMR()
```

There are multiple command for editing the plot:

```
zoom(left=4.6, right=2, top=10, bottom=150) #Exact zoom
zoomIn() #Zoom in
zoomOut() #Zoom out
intPlus() #Increase intensity
intMin() #Decrease intensity
left() #Move spectrum to the left
right() #Move spectrum to the right
```

For 2D data, you can additionally use the following commands:

```
contMin() #Decrease minimum contour level (show more small peaks)
contPlus() #Increase minimum contour level (remove small peaks)
up() #Move spectrum up
down() #Move spectrum down
```

4 Known Issues

4.1 Firewall Warnings

If parallel computing is turned on and the package parallel is installed, mrbin will try to use the socket approach for computing. This requires establishing network connections to the local cluster, which might trigger the firewall. It is safe to unblock these connections.

4.2 Pop-Up Windows

mrbin is set up to ask for user input through pop-up windows. This requires graphics support, otherwise the user input will be asked through command line menus, which is less user friendly but still offers the full functionality.

4.3 Apple/Mac Computers And RStudio

In some cases, running mrbin from within RStudio on Apple computers will not generate pop-up windows. To enable pop-up windows, it might be helpful to install the newest version of xquartz from https://www.xquartz.org.

4.4 Spectra are Missing

If a Bruker spectrum is not shown during browsing, please make sure a file with filename title is present in the PROCNO folder of that spectrum. You can create a title file by opening the spectrum in Bruker Topspin, selecting the Title tab, entering a title and clicking the disk symbol for saving.

5 License

6 References

Klein MS, Dorn C, Saugspier M, Hellerbrand C, Oefner PJ & Gronwald W (2011): Discrimination of Steatosis and NASH in Mice Using Nuclear Magnetic Resonance Spectroscopy. *Metabolomics* 7:237-246