

# Package: MGMS2 (via r-universe)

May 14, 2026

**Title** 'MGMS2' for Polymicrobial Samples

**Version** 1.0.2

**Author** So Young Ryu [aut] (<<https://orcid.org/0000-0003-2347-7015>>),  
George Wendt [cre] (<<https://orcid.org/0000-0003-3608-9601>>)

**Maintainer** George Wendt <gwendt@unr.edu>

**Description** A glycolipid mass spectrometry technology has the potential to accurately identify individual bacterial species from polymicrobial samples. To develop bacterial identification algorithms (e.g. machine learning) using this glycolipid technology, it is necessary to generate a large number of various in-silico polymicrobial mass spectra that are similar to real mass spectra. 'MGMS2' (Membrane Glycolipid Mass Spectrum Simulator) generates such in-silico mass spectra, considering errors in m/z (mass-to-charge ratio) and variances of intensity values, occasions of missing signature ions, and noise peaks. It estimates summary statistics of monomicrobial mass spectra for each strain or species and simulates polymicrobial glycolipid mass spectra using the summary statistics of monomicrobial mass spectra. References: Ryu, S.Y., Wendt, G.A., Chandler, C.E., Ernst, R.K. and Goodlett, D.R. (2019) <doi:10.1021/acs.analchem.9b03340> ``Model-based Spectral Library Approach for Bacterial Identification via Membrane Glycolipids." Gibb, S. and Strimmer, K. (2012) <doi:10.1093/bioinformatics/bts447> ``MALDIquant: a versatile R package for the analysis of mass spectrometry data."

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**Depends** MALDIquant, MALDIquantForeign

**RoxygenNote** 7.0.2

**Suggests** testthat

**NeedsCompilation** no

**Config/pak/sysreqs** libxml2-dev

**Repository** <https://cranhaven.r-universe.dev>

**Date/Publication** 2026-05-14 12:02:00 UTC

**RemoteUrl** <https://github.com/cranhaven/cranhaven.r-universe.dev>

**RemoteRef** package/MGMS2

**RemoteSha** 5fdd93ace3421c5bae47b2a28e56db2e694b27eb

**RemoteSubdir** MGMS2

## Contents

characterize_peak . . . . .	2
create_insilico_mixture_template . . . . .	3
filtermass . . . . .	4
gather_summary . . . . .	5
gather_summary_file . . . . .	6
preprocessMS . . . . .	7
process_monospectra . . . . .	7
simulate_ind_spec_single . . . . .	8
simulate_many_poly_spectra . . . . .	9
simulate_poly_spectra . . . . .	10
summarize_monospectra . . . . .	12
summary_mono . . . . .	13

<b>Index</b>	<b>14</b>
--------------	-----------

---

characterize_peak	<i>characterize_peak</i>
-------------------	--------------------------

---

## Description

This function characterizes peaks by species/strain in a simulated spectrum after taking the highest peak or merging peaks in each bin.

## Usage

```
characterize_peak(spec, option = 1, bin.size = 1, min.mz = 1000, max.mz = 2200)
```

## Arguments

spec	A data frame that contains m/z values of peaks, normalized intensities of peaks, species names, and strain names. Either an output of <a href="#">simulate_poly_spectra</a> or one elements of a list output from <a href="#">simulate_many_poly_spectra</a> .
option	An option on how to merge peaks. There are two options: 1) no merge, thus take the highest intensity peak in each bin after binning a spectrum by bin.size, or 2) take a sum of intensity within each bin after binning a spectrum by bin.size.
bin.size	An integer. A bin size. (1 by default)
min.mz	A real number. Minimum mass-to-charge ratio. (1000 by default)
max.mz	A real number. Maximum mass-to-charge ratio. (2200 by default)

**Value**

A data frame that contains m/z values of peaks (mz), intensities of peaks (int), species names (species), and strain names (strain). Species and strain columns may contain more than one species/strain if an option 2 is chosen.

**Examples**

```
spectra.processed.A <- process_monospectra(
  file=system.file("extdata", "listA.txt", package="MGMS2"),
  mass.range=c(1000,2200))
spectra.processed.B <- process_monospectra(
  file=system.file("extdata", "listB.txt", package="MGMS2"),
  mass.range=c(1000,2200))
spectra.processed.C <- process_monospectra(
  file=system.file("extdata", "listC.txt", package="MGMS2"),
  mass.range=c(1000,2200))
spectra.mono.summary.A <- summarize_monospectra(
  processed.obj=spectra.processed.A,
  species='A', directory=tempdir())
spectra.mono.summary.B <- summarize_monospectra(
  processed.obj=spectra.processed.B,
  species='B', directory=tempdir())
spectra.mono.summary.C <- summarize_monospectra(
  processed.obj=spectra.processed.C,
  species='C', directory=tempdir())
mono.info=gather_summary(c(spectra.mono.summary.A, spectra.mono.summary.B, spectra.mono.summary.C))
mixture.ratio <- list()
mixture.ratio['A']=1
mixture.ratio['B']=0.5
mixture.ratio['C']=0
sim.template <- create_insilico_mixture_template(mono.info)
insilico.spectrum <- simulate_poly_spectra(sim.template, mixture.ratio)
merged.spectrum <- characterize_peak(insilico.spectrum, option=2)
```

---

```
create_insilico_mixture_template
      create_insilico_mixture_template
```

---

**Description**

This function generates an initial template for simulated mass spectra.

**Usage**

```
create_insilico_mixture_template(mono.info, mz.tol = 0.5)
```

**Arguments**

mono.info	An output of <a href="#">gather_summary</a> .
mz.tol	A m/z tolerance in Da. (Default: 0.5)

**Value**

A data frame which contains simulated m/z, log intensity, and normalized intensity values of peaks.

**Examples**

```
spectra.processed.A <- process_monospectra(
  file=system.file("extdata", "listA.txt", package="MGMS2"),
  mass.range=c(1000,2200))
spectra.processed.B <- process_monospectra(
  file=system.file("extdata", "listB.txt", package="MGMS2"),
  mass.range=c(1000,2200))
spectra.processed.C <- process_monospectra(
  file=system.file("extdata", "listC.txt", package="MGMS2"),
  mass.range=c(1000,2200))
spectra.mono.summary.A <- summarize_monospectra(
  processed.obj=spectra.processed.A,
  species='A', directory=tempdir())
spectra.mono.summary.B <- summarize_monospectra(
  processed.obj=spectra.processed.B,
  species='B', directory=tempdir())
spectra.mono.summary.C <- summarize_monospectra(
  processed.obj=spectra.processed.C,
  species='C', directory=tempdir())
mono.info=gather_summary(c(spectra.mono.summary.A, spectra.mono.summary.B, spectra.mono.summary.C))
template <- create_insilico_mixture_template(mono.info)
```

---

 filtermass

*filtermass*


---

**Description**

Internal function. This function removes peaks with their mass values (m/z values) outside a given mass range. This function is used in [process\\_monospectra](#).

**Usage**

```
filtermass(spectra, mass.range)
```

**Arguments**

**spectra**            Mass Spectra (A MALDIquant MassSpectrum (S4) object). An output of [importMzXml](#).  
**mass.range**        Mass (m/z) range (a vector). For exmaple, c(1000,2200).

**Value**

A list of filtered mass spectra (MALDIquant MassSpectrum (S4) objects) which contains mass, intensity, and metaData.

---

gather_summary	<i>gather_summary</i>
----------------	-----------------------

---

## Description

This function combines outputs from [summarize\\_monospectra](#).

## Usage

```
gather_summary(x)
```

## Arguments

x                    A list of multiple monomicrobial mass spectra information from [summarize\\_monospectra](#).

## Value

A list of combined summaries (data frames) of mass spectra from [summarize\\_monospectra](#) and the corresponding species (a vector).

## Examples

```
spectra.processed.A <- process_monospectra(  
  file=system.file("extdata", "listA.txt", package="MGMS2"),  
  mass.range=c(1000,2200))  
spectra.processed.B <- process_monospectra(  
  file=system.file("extdata", "listB.txt", package="MGMS2"),  
  mass.range=c(1000,2200))  
spectra.processed.C <- process_monospectra(  
  file=system.file("extdata", "listC.txt", package="MGMS2"),  
  mass.range=c(1000,2200))  
spectra.mono.summary.A <- summarize_monospectra(  
  processed.obj=spectra.processed.A,  
  species='A', directory=tempdir())  
spectra.mono.summary.B <- summarize_monospectra(  
  processed.obj=spectra.processed.B,  
  species='B', directory=tempdir())  
spectra.mono.summary.C <- summarize_monospectra(  
  processed.obj=spectra.processed.C,  
  species='C', directory=tempdir())  
mono.info=gather_summary(c(spectra.mono.summary.A, spectra.mono.summary.B, spectra.mono.summary.C))
```

---

gather\_summary\_file     *gather\_summary\_file*

---

### Description

This function combines output files from [summarize\\_monospectra](#).

### Usage

```
gather_summary_file(directory)
```

### Arguments

directory     A directory that contains summary files from [summarize\\_monospectra](#).

### Value

A list of combined summaries of mass spectra (data frames) from [summarize\\_monospectra](#) and the corresponding species (a vector).

### Examples

```
spectra.processed.A <- process_monospectra(  
  file=system.file("extdata", "listA.txt", package="MGMS2"),  
  mass.range=c(1000,2200))  
spectra.processed.B <- process_monospectra(  
  file=system.file("extdata", "listB.txt", package="MGMS2"),  
  mass.range=c(1000,2200))  
spectra.processed.C <- process_monospectra(  
  file=system.file("extdata", "listC.txt", package="MGMS2"),  
  mass.range=c(1000,2200))  
spectra.mono.summary.A <- summarize_monospectra(  
  processed.obj=spectra.processed.A,  
  species='A', directory=tempdir())  
spectra.mono.summary.B <- summarize_monospectra(  
  processed.obj=spectra.processed.B,  
  species='B', directory=tempdir())  
spectra.mono.summary.C <- summarize_monospectra(  
  processed.obj=spectra.processed.C,  
  species='C', directory=tempdir())  
summary <- gather_summary_file(directory=tempdir())
```

---

preprocessMS	<i>preprocessMS</i>
--------------	---------------------

---

### Description

Internal function. This function preprocesses spectra by transforming/smoothing intensity, removing baseline, and calibrating intensities.

### Usage

```
preprocessMS(spectra, halfWindowSize = 20, SNIP.iteration = 60)
```

### Arguments

spectra	Spectra. A MALDIquant object. An output of either <a href="#">importMzXml</a> or <a href="#">filtermass</a> .
halfWindowSize	halfWindowSize The highest peaks in the given window (+/-halfWindowSize) will be recognized as peaks. (Default: 20). See <a href="#">detectPeaks</a> for details.
SNIP.iteration	SNIP.iteration An iteration used to remove the baseline of an spectrum. (Default: 60). See <a href="#">removeBaseline</a> for details.

### Value

The processed mass spectra. A list of MALDIquant MassSpectrum objects (S4 objects).

---

process_monospectra	<i>process_monospectra</i>
---------------------	----------------------------

---

### Description

This function processes multiple mzXML files which are listed in the file that an user specifies.

### Usage

```
process_monospectra(  
  file,  
  mass.range = c(1000, 2200),  
  halfWindowSize = 20,  
  SNIP.iteration = 60  
)
```

**Arguments**

file	A file name. This file is a tab-delimited file which contains the following columns: file names, strain.no, and strain. See below for details.
mass.range	The m/z range that users want to consider for the analysis. (Default: c(1000,2200)).
halfWindowSize	A half window size used for the smoothing the intensity values. (Default: 20). See <a href="#">smoothIntensity</a> for details.
SNIP.iteration	An iteration used to remove the baseline of an spectrum. (Default: 60). See <a href="#">removeBaseline</a> for details.

**Value**

A list of processed monobacterial mass spectra (S4 objects, MALDIquant MassSpectrum objects), and their strain numbers (a vector), unique strains (a vector), and strain names (a vector).

**Examples**

```
spectra.processed.A <- process_monospectra(
  file=system.file("extdata", "listA.txt", package="MGMS2"),
  mass.range=c(1000,2200))
```

---

```
simulate_ind_spec_single
      simulate_ind_spec_single
```

---

**Description**

Internal function. The function simulates m/z and intensity values using given summary statistics.

**Usage**

```
simulate_ind_spec_single(interest, mz.tol, species, strain)
```

**Arguments**

interest	Summary statistics of spectra.
mz.tol	The tolerance of m/z. This is used to generate m/z values of peaks.
species	Species.
strain	Strain name.

**Value**

A data frame that contains m/z, (normalized) intensity values, missing rates of peaks, species name, and strain name.

---

```
simulate_many_poly_spectra  
    simulate_many_poly_spectra
```

---

## Description

The function creates simulated mass spectra in pdf file and returns simulated mass spectra (m/z and intensity values of peaks).

## Usage

```
simulate_many_poly_spectra(  
  mono.info,  
  nsim = 10000,  
  file = NULL,  
  mixture.ratio,  
  mixture.missing.prob.peak = 0.05,  
  noise.peak.ratio = 0.05,  
  snr.basepeak = 500,  
  noise.cv = 0.25,  
  mz.range = c(1000, 2200),  
  mz.tol = 0.5  
)
```

## Arguments

mono.info	A list output of <a href="#">gather_summary</a> or <a href="#">gather_summary_file</a> .
nsim	The number of simulated spectra. (Default: 10000)
file	An output file name. (By default, file=NULL. No pdf file will be generated.)
mixture.ratio	A list of bacterial mixture ratios for given bacterial species in sim.template.
mixture.missing.prob.peak	A real value. The missing probability caused by mixing multiple bacteria species. (Default: 0.05)
noise.peak.ratio	A ratio between the numbers of noise and signal peaks. (Default: 0.05)
snr.basepeak	A (base peak) signal to noise ratio. (Default: 5000)
noise.cv	A coefficient of variation of noise peaks. (Default: 0.25)
mz.range	A range of m/z values. (Default: c(1000,2200))
mz.tol	m/z tolerance. (Default: 0.5)

## Value

A list of data frames. A list of simulated mass spectra (data frames) that contains m/z values of peaks, normalized intensities of peaks, species names, and strain names. This function also creates pdf files which contain simulated spectra.

**Examples**

```

spectra.processed.A <- process_monospectra(
  file=system.file("extdata", "listA.txt", package="MGMS2"),
  mass.range=c(1000,2200))
spectra.processed.B <- process_monospectra(
  file=system.file("extdata", "listB.txt", package="MGMS2"),
  mass.range=c(1000,2200))
spectra.processed.C <- process_monospectra(
  file=system.file("extdata", "listC.txt", package="MGMS2"),
  mass.range=c(1000,2200))
spectra.mono.summary.A <- summarize_monospectra(
  processed.obj=spectra.processed.A,
  species='A', directory=tempdir())
spectra.mono.summary.B <- summarize_monospectra(
  processed.obj=spectra.processed.B,
  species='B', directory=tempdir())
spectra.mono.summary.C <- summarize_monospectra(
  processed.obj=spectra.processed.C,
  species='C', directory=tempdir())
mono.info=gather_summary(c(spectra.mono.summary.A, spectra.mono.summary.B, spectra.mono.summary.C))
mixture.ratio <- list()
mixture.ratio['A']=1
mixture.ratio['B']=0.5
mixture.ratio['C']=0
insilico.spectra <- simulate_many_poly_spectra(mono.info, mixture.ratio=mixture.ratio, nsim=10)

```

---

simulate\_poly\_spectra *simulate\_poly\_spectra*

---

**Description**

This function takes simulated m/z and intensities of peaks from [create\\_insilico\\_mixture\\_template](#) and modifies them based on given parameters.

**Usage**

```

simulate_poly_spectra(
  sim.template,
  mixture.ratio,
  spectrum.name = "Spectrum",
  mixture.missing.prob.peak = 0.05,
  noise.peak.ratio = 0.05,
  snr.basepeak = 500,
  noise.cv = 0.25,
  mz.range = c(1000, 2200)
)

```

**Arguments**

sim.template	A data frame which contains m/z, log intensity, normalized intensity values and missing rates of peaks. There are also species and strain information. An object of <code>create_insilico_mixture_template</code> .
mixture.ratio	A list of bacterial mixture ratios for given bacterial species in sim.template.
spectrum.name	A character. An user can define the spectrum name. (Default: 'Spectrum').
mixture.missing.prob.peak	A real value. The missing probability caused by mixing multiple bacteria species. (Default: 0.05)
noise.peak.ratio	A ratio between the numbers of noise and signal peaks. (Default: 0.05)
snr.basepeak	A (base peak) signal to noise ratio. (Default: 500)
noise.cv	A coefficient of variation of noise peaks. (Default: 0.25)
mz.range	A range of m/z values. (Default: c(1000,2200))

**Value**

A data frame that contains m/z values of peaks, normalized intensities of peaks, species names, and strain names. A modified version of sim.template.

**Examples**

```
spectra.processed.A <- process_monospectra(
  file=system.file("extdata", "listA.txt", package="MGMS2"),
  mass.range=c(1000,2200))
spectra.processed.B <- process_monospectra(
  file=system.file("extdata", "listB.txt", package="MGMS2"),
  mass.range=c(1000,2200))
spectra.processed.C <- process_monospectra(
  file=system.file("extdata", "listC.txt", package="MGMS2"),
  mass.range=c(1000,2200))
spectra.mono.summary.A <- summarize_monospectra(
  processed.obj=spectra.processed.A,
  species='A', directory=tempdir())
spectra.mono.summary.B <- summarize_monospectra(
  processed.obj=spectra.processed.B,
  species='B', directory=tempdir())
spectra.mono.summary.C <- summarize_monospectra(
  processed.obj=spectra.processed.C,
  species='C', directory=tempdir())
mono.info=gather_summary(c(spectra.mono.summary.A, spectra.mono.summary.B, spectra.mono.summary.C))
mixture.ratio <- list()
mixture.ratio['A']=1
mixture.ratio['B']=0.5
mixture.ratio['C']=0
sim.template <- create_insilico_mixture_template(mono.info)
insilico.spectrum <- simulate_poly_spectra(sim.template, mixture.ratio)
```

---

summarize\_monospectra *summarize\_monospectra*

---

## Description

This function summarizes monomicrobial spectra and writes summary in the specified directory.

## Usage

```
summarize_monospectra(  
  processed.obj,  
  species,  
  directory = NULL,  
  minFrequency = 0.5,  
  align.tolerance = 5e-04,  
  snr = 3,  
  halfWindowSize = 20,  
  top.N = 50  
)
```

## Arguments

processed.obj	A list from <a href="#">process_monospectra</a> which contains peaks information for each strain.
species	Species name.
directory	Directory. (By default, no summary file will be generated.)
minFrequency	Percentage value. A minimum occurrence proportion required for building a reference peaks. All peaks with their occurrence proportion less than minFrequency will be moved. (Default: 0.50). See <a href="#">filterPeaks</a> and <a href="#">referencePeaks</a> for details.
align.tolerance	Mass tolerance. Must be multiplied by $10^{-6}$ for ppm. (Default: 0.0005).
snr	Signal-to-noise ratio. (Default: 3).
halfWindowSize	The highest peaks in the given window ( $\pm$ halfWindowSize) will be recognized as peaks. (Default: 20). See <a href="#">detectPeaks</a> for details.
top.N	The top N peaks will be chosen for the analysis. An integer value. (Default: 50).

## Value

A data frame that contains the peaks informations: m/z, mean log intensity, standard deviation of log intensity, missing rate of peaks. In addition, it also contains species and strain information.

**Examples**

```
spectra.processed.A <- process_monospectra(
  file=system.file("extdata", "listA.txt", package="MGMS2"),
  mass.range=c(1000,2200))
spectra_mono.summary.A <- summarize_monospectra(
  processed.obj=spectra.processed.A, species='A',
  directory=tempdir())
```

summary\_mono

*summary\_mono***Description**

Internal function. This function calculates summary statistics for peaks afterling aligning spectra of interest.

**Usage**

```
summary_mono(
  spectra.interest,
  minFrequency = 0.5,
  align.tolerance = 5e-04,
  snr = 3,
  halfWindowSize = 20,
  top.N = 50
)
```

**Arguments**

spectra.interest	A list which contains peaks information for a strain of interest.
minFrequency	Percentage value. A minimum occurrence proportion required for building a reference peaks. All peaks with their occurrence proportion less than minFrequency will be moved. (Default: 0.50). See <a href="#">filterPeaks</a> and <a href="#">referencePeaks</a> for details.
align.tolerance	Mass tolerance. Must be multiplied by $10^{-6}$ for ppm. (Default: 0.0005).
snr	Signal-to-noise ratio. (Default: 3).
halfWindowSize	The highest peaks in the given window ( $\pm$ halfWindowSize) will be recognized as peaks. (Default: 20). See <a href="#">detectPeaks</a> for details.
top.N	The top N peaks will be chosen for the analysis. An integer value. (Default: 50).

**Value**

Summary information (Data frame) of spectra of interest.

# Index

characterize\_peak, [2](#)  
create\_insilico\_mixture\_template, [3](#), [10](#),  
[11](#)  
detectPeaks, [7](#), [12](#), [13](#)  
filtermass, [4](#), [7](#)  
filterPeaks, [12](#), [13](#)  
gather\_summary, [3](#), [5](#), [9](#)  
gather\_summary\_file, [6](#), [9](#)  
importMzXml, [4](#), [7](#)  
preprocessMS, [7](#)  
process\_monospectra, [4](#), [7](#), [12](#)  
referencePeaks, [12](#), [13](#)  
removeBaseline, [7](#), [8](#)  
simulate\_ind\_spec\_single, [8](#)  
simulate\_many\_poly\_spectra, [2](#), [9](#)  
simulate\_poly\_spectra, [2](#), [10](#)  
smoothIntensity, [8](#)  
summarize\_monospectra, [5](#), [6](#), [12](#)  
summary\_mono, [13](#)