

metabomatching

Rico Rueedi

October 3, 2016

Please address any questions, comments, or bugs to

`metabomatching@unil.ch`

For bugs, please mention the version of metabomatching that produced the error. If possible, include the relevant `ps.<description>` subdirectories.

Contents

1	Quick Start	2
1.1	Matlab/octave	2
1.2	Docker	2
2	Complete Instructions	3
2.1	Pseudospectra and Scores	4
2.2	Settings	4
2.2.1	variant	5
2.2.2	mode	5
2.2.3	dsh	5
2.2.4	scoring	5
2.2.5	decorrLambda	6
2.2.6	p_pm	6
2.2.7	reference	6
2.2.8	nShow	6
2.2.9	pSignificant	7
2.3	Reference Spectral Databases	7
2.3.1	HMDB	7
2.3.2	BMRB	8
2.3.3	Chemical Abstracts Service Registry Number (CASRN)	9
2.3.4	Providing a Custom Spectral Reference Database	9

3	Metabomatching Figures	10
3.1	Additional Graphical Settings	12
3.1.1	ls	12
3.1.2	title	12
3.1.3	psa_<tag>	12

1 Quick Start

1.1 Matlab/octave

The metabomatching Matlab code, which is compatible with octave, can be obtained from GitHub,

<https://github.com/rrueedi/metabomatching.git>

or the web page of the Computational Biology Group, University of Lausanne:

<http://www2.unil.ch/cbg/index.php?title=Metabomatching>

The download includes: code files to run metabomatching and produce SVG images of the results, spectral reference databases built from HMDB and BMRB, a CASRN to metabolite name conversion list, and a test pseudospectrum.

Minimal Run

In Matlab or octave, set as working directory the directory into which metabomatching was downloaded, and run the file

`metabomatching.m`

This will apply metabomatching with default settings to the test pseudospectrum listed in the file

`test.pseudospectrum.tsv`

located in the subdirectory

`ps.test`

The metabomatching scores for each metabolite are stored in the 4th column of the tab-separated file

`test.scores.tsv`

and presented graphically in the file

`test.svg`

To run metabomatching on data other than the provided test data, all that is required is at least one pseudospectrum file named <tag>.pseudospectrum.tsv saved to a subdirectory named ps.<description>, and conforming to the format described in section 2.1.

1.2 Docker

2 Complete Instructions

Metabomatching operates on the contents of the directory in which it is installed. Specifically, for each subdirectory of the installation directory named `ps.<description>`, metabomatching will be run over all pseudospectrum files named `<tag>.pseudospectrum.tsv` within. Metabomatching settings are fixed for each subdirectory, and all pseudospectra within a subdirectory must be defined on the same chemical shift binning. Details on the content and format of pseudospectra files are given in section 2.1.

Example. Suppose we would like to apply metabomatching to the pseudospectra of two SNPs, `rs1` and `rs2`, obtained in one association study (`study1`), and to the pseudospectra of two SNPs, `rs1` and `rs3`, obtained in another study (`study2`). Using the SNP names as tags for the pseudospectra and the study names as description for the subdirectories, the folder structure in the installation should then be

```
<installation directory>/
|-- data/
|   |-- ...
|-- documentation/
|   |-- ...
|-- fos/
|   |---
|-- func/
|   |-- ...
|-- ps.study1/
|   |-- rs1.pseudospectrum.tsv
|   |-- rs2.pseudospectrum.tsv
|-- ps.study2/
|   |-- rs1.pseudospectrum.tsv
|   |-- rs3.pseudospectrum.tsv
|-- metabomatching.m
|-- metabomatching_docker.m
```

Metabomatching Results. For each pseudospectrum file

`<tag>.pseudospectrum.tsv`

in each subdirectory, metabomatching outputs one scores file

`<tag>.scores.tsv`

and one SVG file

`<tag>.svg`

showing the main metabomatching results. Details on the content and format of scores files are given in section 2.1. Details for the SVG files are given in section 3. For each subdirectory, metabomatching also writes files related to the settings and reference data used. These are described in section 2.2.

Spectral Reference Data. Metabomatching compares pseudospectra to NMR spectra listed in a reference database. The data subdirectory in the installation directory contains nine databases. Six contain reference spectra from HMDB, three from BMRB. Details as to their content and format are given in section 2.3.

2.1 Pseudospectra and Scores

For an untargeted metabolome composed of N features, the pseudospectrum of a certain variable is the collection of N association p -values, N effect sizes, and N standard errors resulting from the associations of the variable with the N metabolome features.

Pseudospectrum file format. A pseudospectrum is provided to metabomatching as a tab-separated file named

`<tag>.pseudospectrum.tsv`

The file contains four columns. The first row must be a header line containing the four labels `shift`, `beta` (effect size), `se` (standard error), and `p`. The subsequent N rows contain the association data for the N metabolome features, indexed by their chemical shift. Columns and data rows may appear in any order. The tag may be composed of letters, numbers, hyphens, and underscores.

Example. Pseudospectrum file with the header row and three data rows.

shift	beta	se	p
1.11	0.03041	0.08991	0.2647
1.12	-0.5301	0.8771	6.1451E-11
1.13	-0.7171	0.1871	0.003201

Scores file format. For every pseudospectrum, metabomatching computes M scores, with M the number of metabolite reference spectra provided, and writes the corresponding scores file, named

`<tag>.scores.tsv`

The file contains three columns, listing the metabolite CASRN, a spectrum identification number (see 2.3), and the metabomatching score.

Example. Scores file for three metabolites.

70-47-3	10003021	3.8131
56-84-8	10003111	0.0002
56-88-2	10003311	0.0002

2.2 Settings

The settings used by metabomatching in each subdirectory can be specified with the optional file

`parameters.in.tsv`

There are seven settings that control the metabomatching algorithm, and two common graphical settings described below. Additional graphical settings are described in section 3. For any non-specified setting, metabomatching uses a default value. On completion, metabomatching writes all settings it used to the file

`parameters.out.tsv`

independently of whether the settings were provided or used by default.

2.2.1 variant

Select among the 1-compound, 2-compound, \pm -, and \pm -2-compound-variants of metabomatching. Note that 2-compound and \pm -2-compound metabomatching are computationally demanding, and may exceed the capabilities of standard personal computers.

key variant

values 1c 2c pm pm2c

default 1c

2.2.2 mode

Controls the spectrum description type used by metabomatching, and determines, in conjunction with setting reference, the spectral reference file metabomatching tries to load.

key mode

values peak multiplet

default peak

2.2.3 dsh

Controls the neighborhood parameter (δ if in peak mode, γ if in multiplet mode).

key dsh

values any number in $(0, 1]$

default 0.025 if mode=peak, 0.01 if mode=multiplet

2.2.4 scoring

key scoring

values chisq z

default chisq

2.2.5 decorrLambda

Sets the shrinkage parameter λ used when applying decorrelation. $\lambda = 1$ implies no decorrelation. For any other value in $[0, 1)$ the feature-feature correlation matrix must be available in the subdirectory, as a comma-separated file named

correlation.csv

The file has no header or row descriptors, and should contain N columns and N rows, with N the number of non header rows in the pseudospectra.

key decorrLambda

values any number in $[0, 1]$

default 1

2.2.6 p_pm

Sets the p -value threshold below which pseudospectrum features are considered to carry signal (rather than noise) for the purpose of \pm -metabomatching (`variant = pm` or `pm2c`). When running metabomatching for one effect direction, the effect size for features of association p -value below `p_pm` and of opposite effect direction is set to 0.

key pPM

values any number in $(0, 0.05]$

default 1E-4

2.2.7 reference

Selects which of the four standard reference sets metabomatching uses. Metabomatching copies the corresponding reference file to each subdirectory. Metabomatching uses the specified reference set in combination with any `slo(m|p)` files already present in the subdirectory. If `reference` is set to `existing`, metabomatching uses only reference file(s) already present in the subdirectory. This applies to both `slo(m|p)` files and `casname` files.

key reference

values hmdb umdb smdb bmrdb umrb smrb existing

default hmdb

2.2.8 nShow

Sets the number of candidate metabolites shown in the results SVG file. Due to the graphical layout, only multiples of 4 are allowed as values for `nShow`.

key nShow

values any multiple of 4 less than the number of metabolites listed in the reference database

default 8

2.2.9 pSignificant

Sets the p -value threshold below which pseudospectrum features are considered significant. This has no effect on metabomatching, and serves only to label features in the SVG file. Significant features are clustered (distance below 2δ if mode = peak or $2\gamma + 0.3$ if mode = multiplet). High values for pSignificant may tag many features significant and break the layout of the SVG file.

key pSignificant

values any number in (0,0.05)

default 5E-8

2.3 Reference Spectral Databases

The spectral reference databases included with the metabomatching software were obtained from HMDB (Human Metabolome Database [1]) and BMRB (Biomagnetic Resonance Database, [2]). Resulting from different experiments, the spectra for individual metabolites are usually different in the two databases, sometimes significantly so.

2.3.1 HMDB

In HMDB, the spectrum for a metabolite is characterized twice. Once as a list of multiplets, and once as a list of peaks. For most spectra, the two descriptions are different enough to result in differences in metabomatching rankings. The differences can be significant. HMDB contains a proton NMR spectrum for 835 metabolites. Every metabolite in the database is identified by a metabolite ID of the form HMDB00001.

Peak Descriptions File. The full set of peak description metabolite spectra is listed in the reference file

hmdb.slop

This file is a four column tab-separated file, with each row containing the CASRN (as a string), the peak position in ppm (real number), the peak height (real number), and the spectrum ID (integer), for one peak of the peak description spectrum of one metabolite. The spectrum ID is defined as 1<xxxxxx>1, with <xxxxxx> the HMDB ID stripped of the characters HMDB but keeping leading zeroes. Peak heights are used in the SVG image, but not by metabomatching.

Example. Spectra of dimethylglycine (CASRN 1118-68-9, HMDB00092) and glycine (CASRN 56-40-6, HMDB000123)

1118-68-9	2.9100	1.0000	1000921
1118-68-9	3.7100	0.2770	1000921
56-40-6	3.5400	1.0000	1001231

Multiplet Descriptions File. The full set of multiplet description metabolite spectra is listed in the reference file

hmdb.slom

This file is a four column tab-separated file, with each row containing the CASRN (as a string), the left edge of the multiplet range (real number), the right edge of the multiplet range (real number), the proton count (integer), and the spectrum ID (integer), for one multiplet range of the multiplet description spectrum of one metabolite. The spectrum ID is defined as 1<xxxxxx>1, with <xxxxxx> the HMDB ID stripped of the characters HMDB but keeping leading zeroes. Proton counts are used in the SVG image, but not by metabomatching.

Example. Spectra of dimethylglycine (CASRN 1118-68-9, HMDB00092) and glycine (CASRN 56-40-6, HMDB000123)

```
1118-68-9 2.90 2.93 6 1000921
1118-68-9 3.70 3.72 2 1000921
56-40-6 3.46 3.60 2 1001231
```

Subsets. Two additional reference files are available containing the spectra of metabolites known to be measured in urine samples. This subset was defined by Bouatra et al. [], and named UMDB (Urine Metabolome Database). It lists the spectra for 180 metabolites. The files are named

umdb.slop
umdb.slom

for the peak and multiplet description spectra, respectively.

2.3.2 BMRB

In BMRB, the spectrum for a metabolite is characterized as a list of peaks. Each metabolite can have several spectra, obtained from different experiments. For most spectra, the descriptions are different enough to affect, slightly, metabomatching rankings. BMRB contains 975 proton NMR spectra for 740 metabolites. Every metabolite in the database is identified by a metabolite ID of the form BMRB00001.

Peak Descriptions File. The full set of peak description metabolite spectra is listed in the reference file

bmr.b.slop

This file is a four column tab-separated file, with each row containing the CASRN (as a string), the peak position in ppm (real number), the peak height (real number), and the spectrum ID (integer) for one peak of the peak description spectrum of one metabolite. The spectrum ID is defined as 2<xxxxxx><y>, with <xxxxxx> the BMRB ID stripped of the characters BMRB but keeping leading zeroes, and <y> indexes different spectra for the same BMRB metabolite. Peak height is used in the SVG image, but not by metabomatching.

Example. Spectra of dimethylglycine (CASRN 1118-68-9, BMRB02411) and glycine (CASRN 56-40-6). Glycine appears in BMRB with two different BMRB IDs (BMRB00089 and BMRB00977), and has four listed spectra (3 for BMRB00089, and 1 for BMRB00977).

1118-68-9	2.9160	1.0000	2002411
1118-68-9	3.7140	0.3362	2002411
56-40-6	3.5424	1.0000	2000891
56-40-6	3.5420	1.0000	2000892
56-40-6	3.5440	1.0000	2000893
56-40-6	3.5441	1.0000	2009771

2.3.3 Chemical Abstracts Service Registry Number (CASRN)

Metabomatching uses CASRN (Chemical Abstracts Service Registry Number) to identify metabolites. Metabolite names are used only in the SVG image. The file

`all.casname`

provides the CASRN to metabolite name table, as a two column tab-separated file.

Example. Entries for dimethylglycine and glycine

1118-68-9	dimethylglycine
56-40-6	glycine

2.3.4 Providing a Custom Spectral Reference Database

Additional spectra can be provided to metabomatching as a custom spectral reference database. The database itself needs a name and a number (3 and above, as 1 and 2 are already assigned to HMDB and BMRB respectively). Each spectrum in the database needs a CASRN identifying the metabolite, and a 6 digit ID number indexing the metabolite in the database as well as a 1 digit number indexing different spectra for the same metabolite.

Example. To provide two new spectra for glycine and two new spectra for the new metabolite *customine* in a custom database called MYDB, we would create the database file

`mydb.slop`

of content

56-40-6	3.5411	1.0000	4000011
56-40-6	3.5422	1.0000	4000012
12345-67-8		1.2345	1.0000 4000021
12345-67-8		1.2367	1.0000 4000022

Because customine is not known in `all.casname`, we would also create the CASRN-name table

`mydb.casname`

of content

Both `mydb.slop` and `mydb.casname` need to be placed in the `ps.<description>` subdirectories for which metabomatching should use MYDB, and can be used exclusively, or in combination with a standard database (see reference setting 2.2.7).

3 Metabomatching Figures

For every `<tag>.pseudospectrum.tsv` file in every `ps.<description>` subdirectory, metabomatching creates an SVG image that shows the pseudospectrum and the spectra of the nShow (2.2.8) top-ranked candidate metabolites. The image is designed with a width of 180mm. The SVG image for the test pseudospectrum included in the metabomatching download is shown in 1.

Descriptions. The pseudospectrum panel title is customized by the optional two-column semicolon-separated file

```
description.csv
```

with the first column containing the pseudospectrum tag, the second the description. The pseudospectrum panel title will then read

```
Pseudospectrum of <description>
```

in bold face. In the case where pseudospectra result from an mGWAS, the description file may be formatted as a three-column file with the first column containing the pseudospectrum tag, the second the SNP ID, and the third the gene name. The title will then read

```
Pseudospectrum of <SNPID> in <GENE>
```

in bold face and with the gene name in italics.

Example. For the test pseudospectrum, the description file reads

```
test;rs37369;AGXT2
```

Control Metabolites. Control metabolites are provided by the optional two column three column semicolon-separated file

```
cascontrol.csv
```

where the first column contains the tag of the pseudospectrum and the second the CASRN of the desired control metabolite. More than one control metabolite may be provided for any pseudospectrum.

Example. For the test pseudospectrum, the `cascontrol.csv` file reads

```
test;144-90-1
```

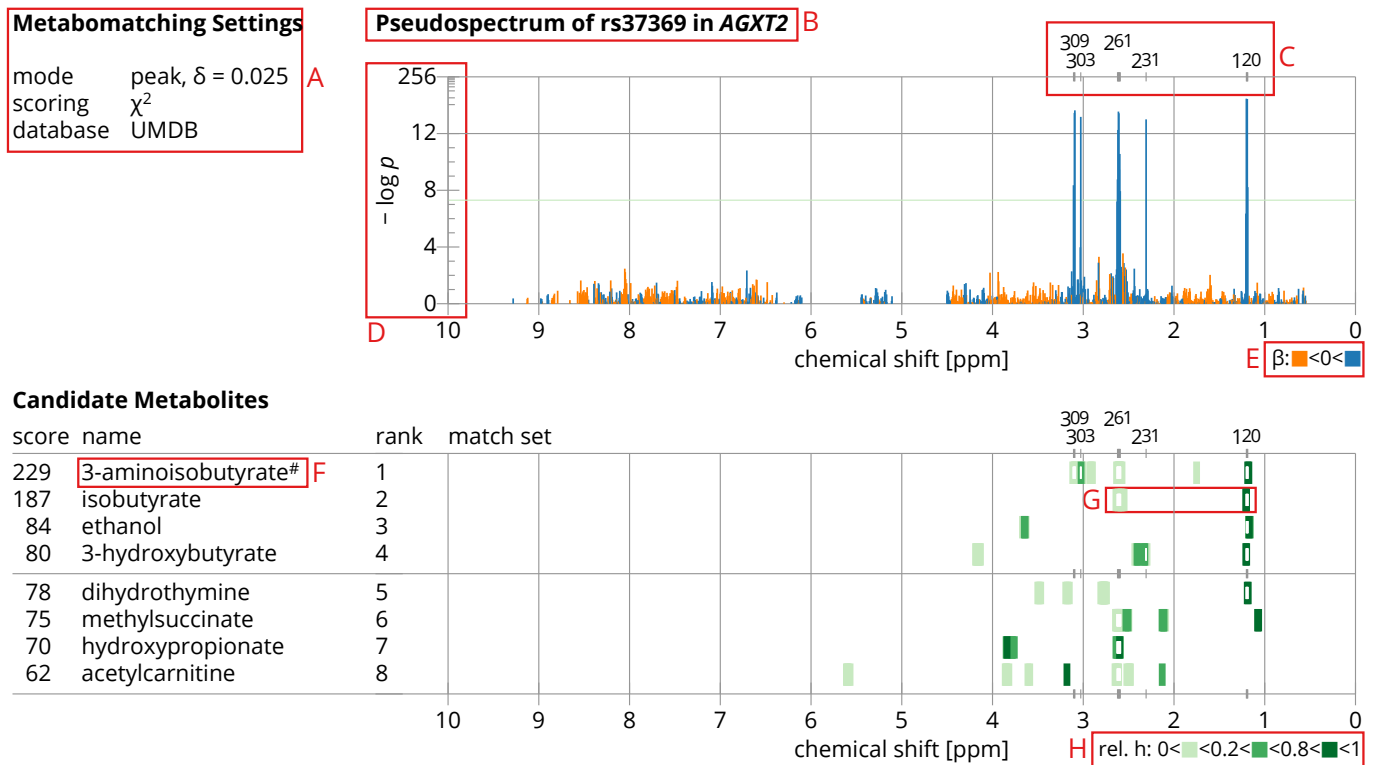


Figure 1: Metabomatching results for the pseudospectrum of SNP rs37369 in AGXT2. The figure contains three panels: the settings panel (*Metabomatching Settings*, top left), the pseudospectrum panel (*Pseudospectrum of...*, top right), and the spectrum panel (*Candidate Metabolites*, bottom).

- A. The settings panel lists the main settings used in the metabomatching run.
- B. The title of the pseudospectrum panel can be provided by the user (see below).
- C. Features in the pseudospectrum with p -value below the significance threshold defined by pSignificant (2.2.9) are highlighted. The position of each significant feature is marked by a tick on the upper edge of the pseudospectrum box. In addition, significant peaks are clustered (according to δ or $\gamma + 0.015$), and the position of the peak with the strongest association in the cluster is labeled. Marks are repeated in the spectrum panel.
- D. The scale used for the pseudospectrum y -axis is linear for values of $-\log p$ between 0 and 12, logarithmic for values above 12, and capped at 256.
- E. Features are colored orange if their association effect size β is negative, blue if it is positive.
- F. A list of metabolites that are shown regardless of their ranks, can be provided by the user (see below). Such metabolites are marked by a "#" next to their name.
- G, H. The match sets are shown on the same chemical shift axis as the pseudospectrum, with one match set per row. For each match set element, the relative height of the peak that produced the element is color-coded. Elements are plotted in order of increasing relative height. In multiplet mode, relative proton counts are used instead of relative heights. The positions of significantly associated features are marked by white ticks in the match set rows to highlight match set elements that include these features.

3.1 Additional Graphical Settings

3.1.1 `ls`

Set metabomatching to produce 220mm wide figures suitable for landscape paper orientation (see figure 2). In this orientation and space permitting, metabolite names are shown, but CASRN aren't. In 2-compound metabomatching, long metabolite names are replaced by metabolite CASRN. This option is applied, for all pseudospectra, by adding

```
ls      1
```

in the `parameters.in.tsv` file.

3.1.2 `title`

Adding

```
title   0
```

to the `parameters.in.tsv` file sets metabomatching to not print the figure title (A) when in portrait orientation. The option has no effect in landscape mode.

3.1.3 `psa_<tag>`

For cases where the labels for significant features (C) overlap the pseudospectrum title (B), the pseudospectrum title can be aligned to the right edge of the pseudospectrum panel. This is done for each pseudospectrum individually, by adding

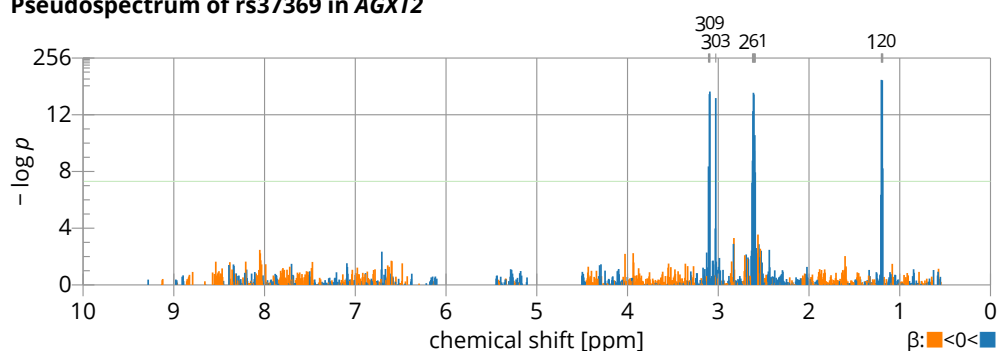
```
psa_<tag> right
```

in the `parameters.in.tsv` file.

Metabomatching Settings

variant 2-compound
mode peak, $\delta = 0.025$
scoring χ^2
database HMDB

Pseudospectrum of rs37369 in AGXT2



Candidate Metabolites

score	name (■&■)	rank	match set
239	isobutyrate & D-cysteine	1	
238	cysteine & isobutyrate	2	
232	malonate & isobutyrate	3	
231	citrate & 144-90-1	4	
230	144-90-1 & 4408-78-0	5	
229	isobutyrate & 144-90-1	6	
229	methylamine & 144-90-1	7	
229	maleate & 144-90-1	8	

144-90-1: 3-aminoisobutyrate

4408-78-0: phosphonoacetate

Figure 2: 2-compound metabomatching results. When running 2-compound metabomatching, pairs of candidate metabolites are scored and ranked. As a result the *Candidate Metabolites* panel lists the names and spectra of two metabolites, with the spectra colored green for one metabolite, and purple for the other. Where metabolite names would over-run the *name* column, CASRN are listed instead, and a CASRN to name table is printed at the bottom of the figure.