

Common usecase examples for multiMS-toolbox

1. [Initialization – loading the multiMS-toolbox and setting the right directory](#)
2. [Using configuration parameters – either as function parameters or from the config file](#)
3. [multiMS-toolbox for matching and normalization of extracted peaks](#)
4. [multiMS-toolbox for normalization of full spectra measured in same m/z points](#)
5. [multiMS-toolbox for normalization of full spectra requiring interpolation](#)
6. [Writing matched peaks to disk](#)
7. [Writing normalized spectra and averaged spectra to disk](#)
8. [Some graphical parameters to note](#)
9. [Examples included in the distribution](#)

1. Initialization – loading the multiMS-toolbox and setting the right directory

To use the software, run the R-software (<http://www.r-project.org/>) and use its `setwd` command to go to directory where is the multiMS-toolbox file, e.g.

```
>setwd("D:/multiMS-toolbox")
```

Load the toolbox by the `source` command:

```
> source("multiMS-toolbox.R")
```

Then use the `setwd` command to go to the directory where are your data.

```
>setwd("D:/data-to-evaluate")
```

Note: You can copy the "multiMS-toolbox.R" file to your data directory and use it from there.

Note: You can assign the `.RData` extension to be automatically opened by the R for Windows GUI front-end. Then you could only double-click the `1blank.RData` (blank workspace) in the multiMS-toolbox directory with the directory already set there and continue with the `source` command.

2. Using configuration parameters – either as function parameters or from the config file

You call the main function `runPCA` from the toolbox, and you need to pass there some parameters.

You can either write all the used parameters to a file (e.g. see the default "config.R" file) and prevent any forgetting once you return back to your analysis later. To load the parameters from the config file to the `runPCA` function, use its `paramsFile` parameter:

```
>runPCA(paramsFile="myConfig.R")
```

Otherwise, you can pass any your parameter directly to the `runPCA` function using the default values of all the others:

```
>runPCA(csvfile="filesAll.csv", lowMz=900.0, highMz=2000.0)
```

You can use both the config file and the parameters passed directly, in that case, the directly passed parameters have higher priority over those from the config file.

Note: `lowMz` and `highMz` are the only required parameters, that have no default values and must be specified either in your config file or passed explicitly to the `runPCA` function.

3. multiMS-toolbox for matching and normalization of extracted peaks

If you want only to match peaks among the samples (not the full spectra), normalize them and run PCA you need:

1. For each your spectrum at least its corresponding peak file with at least the columns *mz* and *int* delimited by Tab. See example peaklists in the *protbind* subdirectory.

Note: For best performance use the [MS-alone tool](#) to extract the peak data from the spectra.

2. For each your spectrum, its spectrum file with intensities – with columns *mz* and *int* delimited by Tab. Do not use header in the spectrum file. See example spectra in the *protbind* subdirectory.
3. csv file to group files together, having at least the columns:
 - *fileName* – containing the names of data files to process.
 - *filesColorProperty* – the same string for two different files means, that its data points will be drawn with the same color.
 - *filesShapeProperty* – the same string for two different files means, that its data points will be drawn with the same shape.
 - *filesSpectrum* – containing the names of spectrum files for given data files.

See "filesAll.csv" example in the *protbind* subdirectory.

Decide whether you want to analyse peak areas or peak intensities controlled by the parameter *areaBased*: 0 – intensities, 1 – areas computed from full width at half maximum values, 2 – areas passed from peak files. If areas are computed from peak widths, the *fwhm* column must be also present in your peak files. If areas are passed from peak files, the *area* column must be present in your peak files.

Decide which normalization to use – controlled by the parameter *normalize*: The easiest way is to use no normalization (*normalize=0*), e.g.:

```
>runPCA(csvfile="filesAll.csv", lowMz=900.0, highMz=2000.0, normalize=0, areaBased=0, deisotoping=0)
```

Other normalization options:

normalize=

- 1 – normalized by the median of whole spectrum (not only the peaks) intensity ratios to the first sample spectrum (or the template spectrum passed in the *normalizedTemplateSpectrumFor1* parameter)
- 2 – normalized by the sum of all matched peak intensities or areas (do not require original spectra at all)
- 3 – normalized by the sum of the whole spectrum area (not only the peaks)

e.g.:

```
>runPCA(csvfile="filesAll.csv", lowMz=900.0, highMz=2000.0, normalize=2, areaBased=1, deisotoping=0)
```

For more detailed documentation see the [user guide](#).

Note: For other parameters to use, like *findRealValuesForMissingPeaks* for imputing the values of missing peaks, *normalizeLowMz* and *normalizeHighMz* parameters for limiting the normalization interval, *deisotoping* for aggregating intensities from the same isotope group, or parameters setting absolute or relative interval for matching peaks among the spectra, see the user guide, especially the section [Peak intensity and spectrum normalization parameters](#).

4. multiMS-toolbox for normalization of full spectra measured in same m/z points

If you want to run PCA on whole spectrum data (full spectra) and their intensities are recorded in the *m/z* points being equal among the samples, then you need:

1. For each your spectrum, its spectrum file with intensities, with columns *mz*, *int* delimited by Tab. Do not use header in the spectrum file. See example spectra in the *protbind* subdirectory.
2. csv file to group files together, having at least columns:
 - *filesSpectrum* – containing the names of spectrum files for given data files.
 - *filesColorProperty* – the same string for two different files means, that its data points will be drawn with the same color.
 - *filesShapeProperty* – the same string for two different files means, that its data points will be drawn with the same shape.

See "filesAll.csv" example in the *protbind* subdirectory, however it is targeted for matching peaks and thus it also contains *fileName* column, which is not required here.

Decide which normalization to use – controlled by the parameter *normalize*: The easiest way is to use no normalization (*normalize=0*), e.g.:

```
>runPCA(csvfile="filesAll.csv", lowMz=900.0, highMz=2000.0, normalize=0, useFullSpectra=1, fullSpectraMzTemplate=-1)
```

Other normalization options:

normalize=

- 1 – normalized by the median of spectrum intensity ratios to the first sample spectrum (or the template spectrum passed in the *normalizedTemplateSpectrumFor1* parameter)
- 2 or 3 – normalized by the sum of the whole spectrum area
- 4 – spectrum is divided by best matching exponential line
- 5 – each intensity is scaled among the samples, i.e. each m/z point shows standard deviation equal to 1

e.g.:

```
>runPCA(csvfile="filesAll.csv", lowMz=900.0, highMz=2000.0, normalize=2, useFullSpectra=1, fullSpectraMzTemplate=-1)
```

For more detailed documentation see the [user guide](#).

5. multiMS-toolbox for normalization of full spectra requiring interpolation

If you want to run PCA on whole spectrum data (full spectra), however each spectrum is recorded in different m/z points, you need:

1. For each your spectrum, its spectrum file with intensities, with columns mz, int delimited by Tab. Do not use header in the spectrum file. See example spectra in the protbind subdirectory.
2. csv file to group files together, having at least columns:
 - filesSpectrum – containing the names of spectrum files for given data files.
 - filesColorProperty – the same string for two different files means, that its data points will be drawn with the same color.
 - filesShapeProperty – the same string for two different files means, that its data points will be drawn with the same shape.

See "filesAll.csv" example in the protbind subdirectory, however it is targeted for matching peaks and thus it also contains fileName column, which is not required here.

Decide which normalization to use – controlled by the parameter *normalize*: The easiest way is to use no normalization (*normalize=0*), other values of this parameter are explained in the previous section.

Decide if you use one spectrum with template m/z points and the other spectra will be reinterpolated to these m/z points - then the template spectrum filename is passed in the *fullSpectraMzTemplate* parameter, e.g.:

```
>runPCA(csvfile="filesAll.csv", lowMz=900.0, highMz=2000.0, normalize=0, useFullSpectra=1, fullSpectraMzTemplate="FC-3-0H-b.spectrum.txt")
```

If you want the FIRST sample of the csv grouping file to be the m/z template, you can use:

```
>runPCA(csvfile="filesAll.csv", lowMz=900.0, highMz=2000.0, normalize=0, useFullSpectra=1, fullSpectraMzTemplate=1)
```

However, other positive numerical values are not allowed.

If you want to reinterpolate the intensity values regularly on the <lowMz, highMz> interval, where each 1 m/z unit is covered by 50 points, you can use:

```
>runPCA(csvfile="filesAll.csv", lowMz=900.0, highMz=2000.0, normalize=0, useFullSpectra=1, fullSpectraMzTemplate=NULL, fullSpectraDivide1MzBy=50)
```

WARNING: Be very careful when handling *fullSpectraDivide1MzBy* value. Too high value could result in out of memory (memory limits) error.

For more detailed documentation see the [user guide](#).

6. Writing matched peaks to disk

For matching the peaks, the matched values are written to a file, among several other text and graphical outputs from other analyses. See created output files in the current (data) directory.

7. Writing normalized spectra and averaged spectra to disk

If you also want each normalized spectrum file to be written to the disk for processing in some other tools, set the parameter *fast* to 0. This will also output the averaged normalized spectra, drawing the averages over the filesColorProperty groups. E. g. for full spectra processing already aligned in same m/z points and normalization by spectrum area:

```
>runPCA(csvfile="filesAll.csv", lowMz=900.0, highMz=2000.0, normalize=2, useFullSpectra=1,
fullSpectraMzTemplate=-1, fast=0)
```

8. Some graphical parameters to note

For description of other interesting parameters, like *itemsLabelAtMost* for specification of in how large graphs should be each sample also labeled, see the user guide, especially the section [Experiment output parameters](#).

9. Examples included in the distribution

In the protbind directory there are spectra of several proteinaceous binders. The commands available to show the examples are only a shorthand for using the *runPCA* function.

To run the demo examples for proteinaceous binders aging effect, load the multiMS-toolbox file and then move to the directory, where the example files are stored:

```
> setwd("examples")
```

```
> setwd("protbind")
```

And then run either of these commands:

```
> demoLowProteins1()
```

or according to selected normalization method (see Implemented functions for details)

```
> demoNormalizedLowProteins1()
```

```
> demoNormalizedLowProteins2()
```

```
> demoNormalizedLowProteins3()
```

The commands above are only a shorthand for

```
> runPCA(lowMz=900.0, highMz=2000.0, label="FC3", csvfile="filesAll.csv", areaBased=1, deisotoping=1,
normalize=0, findRealValuesForMissingPeaks=1, legendColorPropertyLabel="Age",
legendShapePropertyLabel="Concentration", fast=1);
```

```
> runPCA(lowMz=900.0, highMz=2000.0, label="FC3", csvfile="filesAll.csv", areaBased=1, deisotoping=1,
normalize=1, findRealValuesForMissingPeaks=1, legendColorPropertyLabel="Age",
legendShapePropertyLabel="Concentration", fast=1);
```

```
> runPCA(lowMz=900.0, highMz=2000.0, label="FC3", csvfile="filesAll.csv", areaBased=1, deisotoping=1,
normalize=2, findRealValuesForMissingPeaks=1, legendColorPropertyLabel="Age",
legendShapePropertyLabel="Concentration", fast=1);
```

```
> runPCA(lowMz=900.0, highMz=2000.0, label="FC3", csvfile="filesAll.csv", areaBased=1, deisotoping=1,
normalize=3, findRealValuesForMissingPeaks=1, legendColorPropertyLabel="Age",
legendShapePropertyLabel="Concentration", fast=1);
```

For the full spectrum analysis examples, you can also run

```
> demoFullSpectraNormalizedLowProteins1()
```

```
> demoFullSpectraNormalizedLowProteins2()
```

The commands above are only a shorthand for

```
> runPCA(lowMz=900.0, highMz=2000.0, label="FC3", csvfile="filesAll.csv", useFullSpectra=1,
fullSpectraMzTemplate=NULL, fullSpectraDivide1MzBy=50, normalize=1, legendColorPropertyLabel="Age",
```

```
legendShapePropertyLabel="Concentration", fast=1);
```

```
> runPCA(lowMz=900.0, highMz=2000.0, label="FC3", csvfile="filesAll.csv", useFullSpectra=1,  
fullSpectraMzTemplate=NULL, fullSpectraDivide1MzBy=50, normalize=2, legendColorPropertyLabel="Age",  
legendShapePropertyLabel="Concentration", fast=1);
```

These could be shortened a lot, because many mentioned values use their default values - see the user guide, especially the section [Peak intensity and spectrum normalization parameters](#).

All the outputs are printed and drawn to the R-GUI and stored to csv, pdf and txt files to the current directory.

To run the demo examples for bacteria mass spectrum, load the multiMS-toolbox file and then move to the directory, where the example files are stored:

```
> setwd("examples")
```

```
> setwd("bacteria")
```

And then run the command

```
> demoHighProteins1()
```

or, when normalization is used, run

```
> demoNormalizedHighProteins1()
```

The commands above are only a shorthand for

```
> runPCA(lowMz=2000.0, highMz=15000.0, normalize=0, label="Cronobacter bacterial culture",  
csvfile="filesAllbakterie.csv", areaBased=0, deisotoping=0, maxDistance1=7.0, maxDistance2=7.0,  
findRealValuesForMissingPeaks=1, legendColorPropertyLabel="Strain/Method", legendShapePropertyLabel="Aliquot",  
fast=1);
```

```
> runPCA(lowMz=2000.0, highMz=15000.0, normalize=1, label="Cronobacter bacterial culture",  
csvfile="filesAllbakterie.csv", areaBased=0, deisotoping=0, maxDistance1=7.0, maxDistance2=7.0,  
findRealValuesForMissingPeaks=1, legendColorPropertyLabel="Strain/Method", legendShapePropertyLabel="Aliquot",  
fast=1);
```

For the full spectrum analysis, you can also run

```
> demoFullSpectraNormalizedHighProteins1()
```

```
> demoFullSpectraNormalizedHighProteins2()
```

The commands above are only a shorthand for

```
> runPCA(label="Cronobacter bacterial culture", csvfile="filesAllbakterie.csv", normalize=1, lowMz=2000.0,  
highMz=15000.0, useFullSpectra=1, fullSpectraMzTemplate=NULL, fullSpectraDivide1MzBy=50,  
legendColorPropertyLabel="Strain/Method", legendShapePropertyLabel="Aliquot", fast=1);
```

```
> runPCA(label="Cronobacter bacterial culture", csvfile="filesAllbakterie.csv", normalize=2, lowMz=2000.0,  
highMz=15000.0, useFullSpectra=1, fullSpectraMzTemplate=NULL, fullSpectraDivide1MzBy=50,  
legendColorPropertyLabel="Strain/Method", legendShapePropertyLabel="Aliquot", fast=1);
```

All the outputs are printed and drawn to the R-GUI and stored to csv, pdf and txt files to the current directory.

For more detailed documentation, see the [user guide](#).

Last modified: 06.09.2021