CHAPTER 4

Signal Processing Methods for Mass Spectrometry

Peter Monchamp, Lucio Andrade-Cetto, Jane Y. Zhang, and Robert Henson

4.1 Introduction

With the advent of important advances in instrumentation, researchers nowadays can perform large-scale experiments on biological data. They aim to understand biological processes and functions by measuring data at the molecular and cellular level. The large number of required experiments compared with the limited number of measurable events gives signals that are frequently immersed in noise and have poor quality. For example, high-throughput DNA sequencing appeared in the late 1990s at the peak of the Human Genome Project [1, 2] and pushed the rate of data acquisition to its limits. Inferring the DNA sequence from four time traces (base-calling) was significantly improved by preprocessing the signal. Gel electropherograms and later capillary electrophoresis were enhanced with deconvolution methods, background subtracting, signal decorrelation, normalization, and other methods well known at the time by the signal processing community [3]. Microarray technologies, which measure gene expression at the cell level by testing mRNA, also required algorithms borrowed from signal processing for normalization and smoothing [4]. In this chapter we review the signal processing techniques that are used with mass-spectrometry signals. Other new technologies now being developed, such as liquid chromatography mass spectrometry (LC-MS) and tissue microarrays, will also require preprocessing to improve the data.

The use of mass spectrometry (MS) to diagnosis disease by identifying the proteins in biological samples has been gaining interest in recent years [5]. As a first step, biological fluids, such as serum, are analyzed for protein patterns without identifying the underlying proteins [6]. Differences in protein patterns between diseased and healthy patients can occur because of differences in the expressed proteins. Further analysis identifies the proteins responsible for the disease as biomarkers [7]. In this case, biomarkers can be one or more proteins that, when detected and measured, indicate the presence of a specific disease. Clinicians can use these biomarkers for diagnosis and prognosis, while pharmaceutical researchers can investigate biomarkers as possible drug targets or to understand biochemical pathways.
4.1.1 Data Acquisition Methods

MS is an analytical technique for identifying molecules using information about their mass or the mass of their fragments. Any molecule that can be ionized into the gas phase can have its mass determined by a mass spectrometer. An ion source vaporizes molecules into the gas phase and converts them into ions. The gas phase ions are accelerated through an electric field and separated by their mass ($m$) and charge ($z$). Finally, the separated ions are detected and measured by an electron multiplier. The MS data is plotted as a spectrum with $m/z$ values on the x-axis and ion intensity on the y-axis.

There are four common techniques for ionizing biological molecules. Electron Ionization (EI) is the most common ionization technique. It works well for small molecules that are easily vaporized into the gas phase. With thermally sensitive molecules, EI causes extensive fragmentation where you may not observe the parent ion. For large biological molecules with low volatility and thermal instability you need to use other methods of ionization. Soft ionization techniques such as Fast Atom Bombardment (FAB), Electrospray Ionization (ESI), and Matrix-Assisted Laser Desorption Ionization (MALDI) overcome the limitations of EI. Currently, the most common methods for ionizing large biological molecules are Electrospray Ionization Mass Spectrometry (ESI-MS), Matrix-Assisted Laser Desorption Ionization Mass Spectrometry (MALDI-MS) and Surface Enhanced Laser Desorption Ionization Mass Spectrometry (SELDI-MS). These methods can detect high molecular mass, low volatile, and thermally liable compounds such as proteins in biological samples. They all use soft ionization techniques to volatize the proteins into the gas phase without fragmenting the molecules and to detect them with high sensitivity.

4.1.2 History of Ionization Techniques

The developers of two of the common ionization techniques in mass spectrometry received Nobel Prizes. In 2002, John Fenn (electrospray ionization) and Koichi Tanaka (soft laser desorption ionization) shared half of the Nobel Prize in Chemistry for their development of techniques to analyze biological macromolecules using mass spectrometry. In both cases, the discovered breakthroughs were related to extending the size of biological molecules that could be analyzed to over 10,000 Daltons.

John Fenn developed electrospray ionization (ESI), where proteins in a liquid solvent are sprayed through a nozzle with a strong voltage applied to produce charged droplets. Solvent is then removed from the charged droplets, leaving charged protein ions. With ESI, the ionized proteins are produced with a series of multiple charges. The breakthrough for analysis of large molecules described by John Fenn in 1989 [8] was to add a countercflow of gas to desolvate the droplets and use signal averaging over the multiple ions for a single protein to create a signal that was stronger and more accurate than any of the individual ion signals.

Koichi Tanaka developed soft laser desorption ionization (SLDI), a precursor to MALDI, where proteins are mixed with a matrix material and applied to a metal plate. A laser ionizes and vaporizes the matrix and protein molecules from the plate. The breakthrough described by Koichi Tanaka in 1988 [9] was to use a ma-
trix material of ultra-fine cobalt particles and glycerol with a low-energy nitrogen laser having a wavelength of 337 nm to ionize the proteins. Energy from the laser is selectively absorbed by the matrix, while the proteins tend not to absorb light with a wavelength of 337 nm. Using a combination of laser wavelength and matrix material, large proteins are vaporized and ionized without fragmentation.

4.1.3 Sample Preparation

Electrospray (ESI) does not need any prior sample preparation. Sample molecules in a liquid are separated using liquid chromatography (LC) techniques with the liquid from the end of a chromatography column introduced directly into an ES ionizer.

Samples for MALDI are prepared by mixing a matrix solution with a sample solution and spotting the mixture on a MALDI plate. The plate is allowed to dry while solvents in the mixture evaporate, leaving a crystallized matrix.

SELDI is a similar technique to MALDI. It is a proprietary analysis method from Ciphergen Inc. for selectively separating proteins from a mixture. With SELDI, a biological sample is applied to a surface with an affinity for proteins with different chemical properties. Proteins with an affinity for the surface bond to it, while proteins without an affinity are washed off the surface. A matrix solution is next applied over the sample and allowed to dry and crystallize.

4.1.4 Ionization

With ESI, a stream of liquid is pumped from an LC column though a needle with a very high voltage. The charged liquid is broken into droplets with a nebulizing gas, and then solvent removes molecules from the sample with a stream of drying gas. By a method that is not clearly understood, charge on the solvent molecules is transferred to the sample molecules with the addition of one or more protons. The remaining sample ions in the gas phase are attracted to the entrance of the MS detector.

After sample preparation, MALDI and SELDI use the same instrument technique. The crystallized mixture is inserted into an ion source with a high vacuum. It is irradiated with a laser. The matrix molecules absorb most of the energy and protect the sample from being fragmented. Matrix molecules desorb from the surface of the plate and vaporize along with the sample molecules. Energy is transferred from the matrix molecules to the sample molecules to help them ionize. Protein molecules are usually ionized by adding a proton (H+) to the molecular ion (M) to create a singly charged ion [M+H]+, but there may also be some doubly charged proteins [M+2H]2+.

4.1.5 Separation of Ions by Mass and Charge

A common method for separating ions with MALDI samples uses a time-of-flight (TOF) tube. Positively charged sample ions formed in the source are repelled by a positively charge anode and accelerated into a mass analyzer by an electric field into a flight tube. The molecules traveling down the flight tube reach the ion detector at different times because of differences in mass and charge. The higher the mass of an
ion, the lower its velocity and the longer it takes to travel down the flight tube to the detector. Ions with twice the charge move twice as fast as ions with the same mass but half the charge.

The time for an ion to reach a detector from the source is given by (4.1), in which \((t - t_0) =\) time of flight for an ion from the source to the detector, \(M =\) mass of the ion, \(e =\) charge of the ion, \(E =\) electric field to accelerate ions into the flight tube, \(d =\) length of accelerating region with electric field, \(L =\) length of nonaccelerating region without an electric field, and \(V_0 =\) potential of the electric field.

\[
t - t_0 = \left(\frac{2md}{ZeE}\right)^{1/2} + L\left(\frac{m}{2ZeV_0}\right)^{1/2}
\]

After rearranging (4.1) for \(m/z\), the quadratic relationship between the mass-to-charge ratio and TOF is apparent in (4.2). The constants \(a\) and \(b\) depend on the instrument, potential applied at the source, electric field, and length of the flight tube.

\[
m/z = a(t - to)^2 + b
\]

Some MALDI-TOF instruments have an ion mirror that deflects ions with an electric field back down the flight tube. Doubling the flight path of ions increases the resolution between ion peaks [10].

### 4.1.6 Detection of Ions and Recorded Data

An electron multiplier detects and measures the ions reaching the end of a TOF tube. After an MS instrument is calibrated with compounds of known mass, the constants in the quadratic equation relating time to mass/charge are determined, and the mass/charge of detected ions calculated. The result is a series of data points with mass/charge and relative ion intensity values. A mass spectrum is a plot of mass/charge on the x-axis and relative ion intensity on the y-axis.

For large biomolecules, a MALDI-MS instrument can measure the molecular mass with an accuracy sufficient to identify individual peptides.

### 4.1.7 Data Preprocessing

Experimental MS data begins with data acquisition, uses preprocessing to correct some of the acquisition problems, and ends with analysis to identify protein molecules. Before analyzing spectra data, you need to preprocess it to remove or minimize problems with the data [11]. Problems with data acquisition can be divided into two areas:

- Flawed experimental design and technique. This area includes samples prepared with different procedures, sample data sets not acquired randomly to minimize systemic errors, and comparing spectra acquired with different instruments. Problems with the experimental process need to be corrected before you can preprocess data and complete the final analysis [12, 13];
- Instrument miscalibration, noise, and variation. The processing methods described in this chapter can minimize problems in this area, but cannot correct
for poorly acquired data from problems in the previous area. Processing techniques cannot overcome problems with inadequate data acquisition technique.

In contrast to the processing methods in this chapter, other classical MS analysis strategies keep information only for the mass of peaks calculated by an instrument. The detected ion intensity of a peak is characterized by determining the centroid of the peak, and then representing it with a single intensity value equal to the peak height and assuming the \( m/z \) value at the centroid corresponds to the actual mass. The advantage to this approach is that it saves a huge amount of memory. The disadvantage is that important information might be lost due to a defective peak extraction or failed segmentation. Lost information could happen if peaks appear overlapped in the raw spectra and the shape of the peaks is distorted due to a low signal-to-noise ratio. Processing the raw data allows you to improve the results from further analysis of the data.

4.1.8 Example Data
This chapter shows a typical workflow for dealing with protein MS data. The example data are from the FDA-NCI Clinical Proteomics Program Databank and was used to identify proteomic patterns for diagnosis of ovarian cancer in serum samples [14]. The data was acquired using Surface-Enhanced Laser Desorption Ionization Time-of-Flight Mass Spectrometry (SELDI-TOF MS) [15].

4.2 Signal Resampling
Signal resampling is the process of calculating a new signal with intensity values at controlled mass/charge (\( m/z \)) points where the reassembled signal follows, as much as possible, the original signal. By controlled we mean that the mass/charge points can be less than the original ones (down-sampling), approximately equal (synchronizing), or more than (up-sampling). In mass spectrometry, up-sampling is usually not used.

With high-resolution MS data, the large number of values in a signal can be impractical to work with using computationally intensive algorithms, and they may reach the limits of computer memory. If the sampling rate is higher than the resolution of the instrument, you could have redundant values immersed in noise, or your analysis may not need the data provided with a higher resolution. In both cases, you could remove the extra values.

Another problem is that the number of \( m/z \) values and the distance between \( m/z \) values may vary between samples analyzed with one instrument or, more likely, with different instruments, making comparison between spectra difficult.

Resampling has several advantages. By resampling you can:

- Reduce the values in a signal to a more manageable number while preserving the information content of the spectra. If the datasets are too large to keep in the available memory, then you need to down-sample to be able to work with
all of the data. You may also want to do this for algorithm design purposes and work with a smaller dataset;
• Take spectra with different m/z vectors and match the scales, creating a consistent m/z vector range. If the samples were taken from different machines, then the values may be slightly different, so you need to resample to get everything on the same scale. Also, comparative algorithms between spectra may need to use the same reference values;
• Fill in missing m/z values. Another issue is that samples may be missing for certain m/z values so you can use resampling to fill in dropped values. This helps when you need to visualize the data. Dropped samples can only be recovered if the original m/z values follow a linear or a quadratic function.

A disadvantage of resampling occurs if you reduce the number of values for visualization and analysis purposes to a size that masks or removes important features of the data.

You want a function that allows you to select a new m/z vector by specifying an m/z range and the number of values. It inputs a raw mass spectrum and outputs a spectrum having the specified number of samples with an m/z spacing that increases linearly within the specified range. The m/z vector can be a linear or a quadratic function.

Also apply an antialias filter to prevent high-frequency noise from folding into lower frequencies. The antialias filter could use a linear-phase FIR filter with a least-squares error minimization. The cut-off frequency is set by the largest downsampling ratio when comparing the same regions in the m/z input and output vectors [16].

4.2.1 Algorithm Explanation and Discussion
Resampling calculates new m/z points with their respective values that best fit to the original raw spectra. The new m/z values should be regularly spaced following a known function $f(x)$. For digital signal processing, this is similar to sample rate conversion where $f(x) = K$. In genomic signal processing, $f(x)$ could be a soft function, so you can have more samples in the areas with a high content of information. For example, TOF signals have a quadratic relationship between mass and charge (4.1), where you would want to have more samples in the low m/z values of the spectra.

When looking at different spectra, resample all spectra to the same $f(x)$. This allows you to further compare spectra without having to segment the signals further. Working with low-resolution spectra from different experiments might require you to resample to improve the reproducibility of experiments.

When down-sampling a signal, high-frequency components appear in the down-sampled signal as low-frequency components known in the signal processing community as aliasing. To prevent aliasing, you should figure out the Nyquist frequency ($f_N = f_{\text{Sampling}}/2$) and prefilter the original signal before down-sampling. In the case of high-resolution MS signals, the high-frequency content of the signal is mostly noise. Since the sampling rate may be variable for a single spectrum, the Nyquist frequency is also variable. For practical cases, select the Nyquist frequency with a value equal to the minimum distance between two contiguous samples of the targeted m/z vector.
4.2.2 Example Demonstrating Down Sampling

In this section, a high-resolution example taken from the FDA-NCI ovarian dataset is used to demonstrate how to resample MS data. Functions from the Bioinformatics Toolbox [17] show the process of converting high-resolution spectra to low-resolution spectra by downsampling. Load the high-resolution spectra and plot the data.

```matlab
load high_resolution_sample;
plot(MZ, Y, '.);
```

The first variable MZ is a vector of \( m/z \) values, while the second variable Y is a vector of ion intensity values corresponding to each \( m/z \) value. See Figure 4.1 for a plot of the raw MS data.

Determine the number of data values in the original spectrum.

```matlab
original_size = numel(MZ)
original_size =
355760
```

Down-sample the spectra between 2,000 and 11,000 and reduce the number of data values.

```matlab
[MZD,YD] = msresample(MZ,Y,10000,'Range',[2000 11000]);
```

Plot the resampled spectrum and notice the reduced number of data points. See Figure 4.2 for an example of a spectrum with additional data points removed.

Zoom into a section of the spectrum.

```matlab
axis([3875 3895 0 90])
```

Figure 4.1 Original mass spectrum before resampling.
See Figure 4.3 comparing a spectrum before and after resampling with the antialiasing filter turned on.

Resample the original spectrum but this time turn off the antialias filter. The down-sampled spectrum shows some noise due to aliasing effects. See Figure 4.4.

\[
[MZD, YD] = \text{msresample}(MZ, Y, 10000, 'Range', [2000 11000], 'Cutoff', 1.0, 'ShowPlot', true);
\]

\[
\text{axis}([3875 3895 0 90])
\]
4.3 Correcting the Background

Mass spectrometry data usually shows a varying baseline. Chemical noise in the matrix or ion overloading can cause this variation. Subtracting the baseline makes spectra easier to compare. Use baseline correction:

- With samples that have an obvious offset, drift, or broad low-frequency peaks;
- After you down-sample or with spectra that have consistent m/z ranges;
- Before you correct the calibration, because the noise will affect the results of that step. MALDI and TOF samples are particularly susceptible to noise, although other techniques and more sensitive instruments give cleaner spectra.

One strategy for removing a low-frequency baseline within the high-frequency noise and signal peaks follows three steps: (1) estimate the most likely baseline in a small window, (2) regress the varying baseline to the window points using a spline interpolation and smoothing, and (3) subtract the estimated and regressed baseline from the spectrum. Also, consider band broading of mass ion peaks by assuming a Gaussian distribution of peaks and plotting the standard deviation across the m/z values, and then use a monotonic smoothing algorithm to subtract the baseline [18].

4.3.1 Algorithm Explanation and Discussion

Estimating the most likely background in every window is the most crucial step. Unfortunately, you cannot observe the true baseline using the minimum values because of the high-frequency signal noise. There are two good approaches to overcome this problem:

- Use a quantile value of the observed sample within the window (see Figure 4.5). This approach is fast, but it has the disadvantage of assuming there are a rela-
tively equal proportion of points that belong to a peak and points that belong to the baseline in the current window. For example, by setting the quantile to 10%, you are assuming that in every window 20% of the points are baseline and 80% are peaks. You can safely underestimate the quantile with the result that your estimated baseline is slightly conservative. On the other hand, if you over estimate the quantile, the proportion of baseline points includes peak values. See Figure 4.5.

- Use a probabilistic model. The second approach improves the result at the cost of computational time. You can assume that the points in every window come from a doubly stochastic model, that the source of each point can be “noise” or “peak,” and that each class has its own distribution. In practice, assuming a uniform Gaussian distribution is relatively safe. Estimating the baseline implies learning the distributions and the class labels for every point, which is an unsupervised clustering problem solved by an Expectation-Maximization estimation [18]. At the end, the mean of the “noise” class turns out to be the best baseline estimate for the window. See Figure 4.6.

How do you select the window size? It should be sufficiently small so that the varying trend of the baseline is not significant, and you can assume that it is constant in your estimation. This makes the estimation approach faster and be more robust. It should be sufficiently large so that you can observe a representative sample of the baseline in the window. In the case of MS signals, the abundance of peaks and resolution of the raw trace varies through a spectrum, so you should allow different windows sizes, depending on the region of the spectrum.

Why use spline interpolation and smoothing to regress the baseline? Some authors have tried to approximate the baseline of a signal to a known function. For example, using a known function is a good strategy with genomic signal preprocessing and DNA sequences, where a combination of exponential and linear curves is sufficient to model the background of DNA chromatograms. This strategy satisfactorily recovers the baseline introduced by gel electrophoresis. When you can de-
rive a good model that correlates to the observed data, the function-based regression is more robust. With MS signals, you would have difficulty finding a good model that always correlates to the background, but you could smooth the acquired data points and then interpolate for every \( m/z \) value. The smoothing is sufficient to minimize the effect of potential outliers.

4.3.2 Example Demonstrating Baseline Subtraction

In this and the following sections, four low-resolution spectra taken from two different low-resolution ovarian cancer FDA-NCI ovarian datasets are used to demonstrate MS preprocessing tasks. These spectra were generated using the WCX2 protein-binding chip, two with manual sample handling and two with a robotic sample dispenser and processor. Functions from the Bioinformatics Toolbox show the process for correcting a baseline. Load a set of low-resolution spectra and plot the data for the second spectra.

```matlab
load low_resolution_sample;
plot(MZ,Y(:,2));

MZ is the mass/charge vector, while \( Y \) is a matrix, with the ion intensities for each sample in separate columns. See Figure 4.7 for a plot of the raw MS data.

Adjust the baseline for a set of spectra by selecting a window of 500 points and assuming 20\% of the points in a window are baseline, and plot the second spectrum with the estimated baseline subtracted.

```matlab
YB = msbackadj(MZ,Y,'WindowSize',500,'Quantile',0.20);
plot(MZ, YB(:,2));
```  

See Figure 4.8 for an example of a spectrum with the baseline subtracted from the raw spectrum.
4.4 Aligning Mass/Charge Values

Errors in calibration or limitations of a mass spectrometer can lead to variations between the observed $m/z$ vector and the true TOF of the ions. Therefore, systematic shifts appear in repeated experiments and two identical proteins acquired in different spectra can have different $m/z$ values. A single instrument or using different instruments can cause these systematic errors. Although the high-throughput detector used in MS can generate numerous spectra per patient, undesirable variation may get introduced in the MS data due to the nonlinearity in the detector response, ionization suppression, minor changes in the mobile phase composition, and inter-
action between analytes. Additionally, the resolution of the peaks usually changes for different experiments and also varies towards the end of the spectrum.

Adjust the \( m/z \) values when:

- A known profile of peaks is expected in a spectrum. You may have known compounds in a biological sample that you know should align;
- Your samples are “spiked” with known compounds (internal standards) to aid calibration;
- External calibration standards analyzed with samples show variation.

Resample and correct the baseline for your raw spectra before trying to align \( m/z \) values. One advantage of working with the raw data, and not with peak information, is that the alignment algorithm is less prone to fail due to a defective peak extraction algorithm. When analyzing MALDI-TOF data, you may have information about \( m/z \) values for known calibration standards or contaminants. A preprocessing function should use a set of \( m/z \) values where you expect reference peaks to appear, and it should allow you to define a set of relative weights which the aligning algorithm can use to emphasize peaks with a small area.

One method aligns a raw mass spectrum by scaling and shifting the \( m/z \) scale so that the cross-correlation between the raw mass spectrum and a synthetic spectrum is maximized. Build a synthetic spectrum with Gaussian pulses centered at the masses specified by the reference peaks. Once a new \( m/z \) scale is determined, calculate a new spectrum by piecewise cubic interpolating and shifting the new spectrum from the original \( m/z \) vector. This method preserves the shape of the peaks.

### 4.4.1 Algorithm Explanation and Discussion

A smooth function warps the signals by resampling the spectra. The smooth function can be any higher-order polynomial. Since most of the observed shifts in the MS data are due to the difficulty of achieving a consistent calibration of the TOF to mass between experiments (4.1), the function msalign in the Bioinformatics Toolbox uses a second-order warp function. Other authors [19] have proposed using cubic splines for datasets in which the dominant shift anomalies are not due to the former quadratic relation.

The alignment algorithm builds a synthetic signal with two or more peaks represented by a Gaussian kernel. The \( m/z \) values of the synthetic signal (the location of the Gaussian peaks) are shifted and scaled until the cross-correlation between the raw mass spectrum and the synthetic signal reaches its maximum value. In this case, shifting and scaling represent the two degrees of freedom needed in the smooth warping function. For higher-order warp functions, you would need to identify more parameters. The user is responsible for selecting the approximate location of the reference peaks expected to appear in the spectra.

When multiple spectra are aligned, the previous algorithm is repeated for each one. The estimation of the warping function for every spectrum can be distributed over a cluster of computers since these computations are data independent, therefore achieving linear speedup of the computations. The algorithm then selects the ultimate locations of the reference peaks based on the computed warping functions such that the sum of the squared shifts for the reference peaks is minimized. A sub-
stantial difference between this alignment approach and other published approaches [20] is that this approach infers the warping function from the raw data and not from a list of peaks.

Setting the width of the Gaussian pulses has a twofold purpose. On one side, pulses should be narrow enough so that close peaks in the spectra are not included with the reference peaks. On the other side, pulses should be wide enough so that the algorithm captures a peak that is off the expected site. Tuning the spread of the Gaussian pulses controls a tradeoff between robustness (wider pulses) and precision (narrower pulses). However, pulse width is unrelated to the shape of the observed peaks in the spectrum. The algorithm allows you to give spectrum-dependent widths and weights to every reference peak. You may want to set different widths for Gaussian pulses since the typical spectrum resolution changes along the \( m/z \) range. Peak weights are used to emphasize peaks whose intensity is small but that provide a consistent \( m/z \) value and appear with good resolution in most of the spectra.

The algorithm searches over a two-dimensional grid of possible shifts and scales for the \( m/z \) vector using a multiresolution exhaustive grid search. This approach does not guarantee you will find a global maxima. However, since misalignments of peaks generally are systematic and small, the algorithm adjusts the \( m/z \) values while preserving its robustness for noisy datasets. You can improve this technique by using a better optimization method instead of an exhaustive grid search. For example, you could apply genetic algorithms, which considerably speed up the estimation of the warping functions.

### 4.4.2 Example Demonstrating Aligning Mass/Charge Values

Plot four low-resolution spectra with the baseline corrected, and then zoom into a few ion peaks to show the misalignment of \( m/z \) values between spectra.

```matlab
plot(MZ,Y);
```

See Figure 4.9 for a plot of four misaligned mass spectra.

Enter the location and weight of the reference peaks.

```matlab
P = [3991.4 4598 7964 9160];
W = [60 100 60 100];
```

Use a heat map to observe the alignment of peaks in the original spectrum. See Figure 4.10.

```matlab
msheatmap(MZ,YB,'Markers',P,'Limit',[3000 10000]),
title('Before Alignment')
```

Align the set of baseline-subtracted spectra to the reference peaks given.

```matlab
YA = msalign(MZ,YB,P,'Weights',W);
```

After applying the alignment algorithm, you can observe improvements in peak alignment between spectra based on peak height. See Figure 4.11.

```matlab
msheatmap(MZ,YA,'markers',P,'limit',[3000 10000])
```
4.4 Aligning Mass/Charge Values

Figure 4.9 Four low-resolution mass spectra showing misalignment.

Figure 4.10 Heat map showing misalignment.

Figure 4.11 Heat map showing alignment corrected.
4.5 Normalizing Relative Intensity

Repeated experiments commonly have systematic differences in the total amount of desorbed and ionized proteins. Sample sizes may be different, sample preparation may be different with different technicians, there could be ion saturation, or the sensitivity of the instrument may change. The result is a variation in the amplitude of ion intensities.

To compensate for systematic differences, you could normalize the relative intensities of the spectra to the average area under the spectra curves or the height of a selected peak. This type of normalization has been used in experiments looking for differences in expressed proteins, but there is an assumption about the samples. The assumption is that the amount of proteins whose expression changes is much less than the amount of total proteins in a sample. This may not always be the case.

A second, more robust normalization method uses the area or height of an internal standard. An internal standard is a compound with a known mass and with the same amount of compound added to each sample. Differences in the area of an internal standard are proportional to the differences in area for the proteins in a sample.

Normalize your samples

- After subtracting the baseline and correcting miscalibration by adjusting the $m/z$ values;
- After subtracting the low $m/z$ values with ion intensity values having considerable noise;
- When the samples are “spiked” with known compounds (internal standards).

You can normalize a group of mass spectra by setting the area under each curve to the group median or to the percentage of height of a selected peak, or you can normalize samples with a constant amount of “spiked” internal standard with the area of the standard peak [21–24].

4.5.1 Example Demonstrating Intensity Normalization

Plot the low-resolution spectra after correcting for baseline variation and miscalibration. See Figure 4.12.

```matlab
plot(MZ, YA)
```

One of many methods to normalize the intensity values of spectra is to rescale the maximum intensity of every signal to a certain value. For example, you could select the highest peak in a sample and normalize all spectra to 100% of that peak. It is also possible to ignore problematic regions. For example, in biological samples you might want to ignore the low-mass region ($m/z < 1000$ Daltons). Choose a cutoff value that eliminates the large amount of noise at lower $m/z$ values but does not remove any proteins of interest.

```matlab
YN1 = msnorm(MZ,YA,'Quantile',1,'Limits',[1000 inf],'MAX',100); plot(MZ,YN1);
```
4.5 Normalizing Relative Intensity

See Figure 4.12 for an example of four spectra with baseline and calibration corrected.

Figure 4.12 Four spectra with baseline and calibration corrected.

See Figure 4.13 for an example of four spectra normalized to the highest ion peak from one of the spectra.

The msnorm can also normalize using the area under the curves (AUC) and then rescaling the spectra having relative intensities below 100.

\[
YN2 = \text{msnorm}(MZ, YA, 'LIMITS', [1000 \text{ inf}], 'MAX', 100);
\]

\[
\text{plot}(MZ, YN2)
\]

See Figure 4.14 for an example of four spectra normalized to the mean area from the four spectra.

You can also use the peak height or area of an internal standard to normalize the spectra for comparison. For example, if the peak at 9164 is an internal standard, you could normalize a set of spectra based only on the mean area of this peak.

View the variation in area (ion intensity) of the peak at 9164 m/z.

\[
\text{plot}(MZ, YA); \text{ axis([8500 10000 -5 105])};
\]

Figure 4.13 Four spectra normalized to the highest ion peak.

Figure 4.14 Four spectra normalized to the mean area from the four spectra.
See Figure 4.15 showing an ion peak from four spectra with the same amount of compound but different areas.

Normalize the area of the peak at 9164 to 40% of its height. By setting the quantile to 0.8, msnorm uses the highest 80% of values in the selected window to normalize the peak. This eliminates normalization errors from smaller peaks riding on the edge of the selected peak.

\[
YN3 = \text{msnorm}(MZ, YA, \text{`limits'}, [9000 9300], \text{`quantile'},[0.8 1], \text{`MAX'}, 40);
\]

\[
\text{plot}(MZ, YN3); \text{axis([7000 10000 -5 105]}));
\]

See Figure 4.16 for an example of an ion peak in four spectra normalized to have the same area.

See Figure 4.15 showing an ion peak from four spectra with the same amount of compound but different areas.

Figure 4.14  Four spectra normalized to the mean area under the curves.

Figure 4.15  Internal standard with unequal areas.
4.6 Smoothing Noise

Mass spectra usually contain a mixture of noise and signal. Some applications require you to filter the noise or smooth the spectra in order to improve the validity and precision of the observed $m/z$ values of the peaks in the spectra. For the same reason, smoothing also improves peak detection algorithms. Noise reduction can be achieved either by filtering or by smoothing. This section reviews the smoothing techniques and explains why these are the preferred techniques to enhance the signal over conventional filtering.

Filtering is a noise reduction technique that selectively reduces the power content of specific signal frequencies. There are two families of filters, IIR and FIR, that you can apply efficiently to the signal. You need to design a filter in terms of its frequency characteristics. For this you can analyze the frequency spectrum of the signal and try to identify the frequencies of the undesired content and chose the best cutoff frequency for your filter. In the case of mass spectra, you need a low-pass filter since the low-frequency noise of the signal and baseline has already been corrected. The length of the filter depends on the degree of frequency selectiveness you want.

Smoothing (also known as polynomial filtering) is an alternative for noise reduction that involves the treatment of the signal samples in order to make them fit a particular model. Smoothing consists of adjusting sample by sample the signal based on a regional polynomial fit. With smoothing, you do not have to design a filter that is robust to outliers, can easily adapt to varying sampling rate, and preserve the sharpness of peaks while eliminating high-frequency components. However, smoothing is more computationally intensive than linear filtering.

There are two types of polynomial smoothing methods for mass spectra that remove the false ion peaks that do not indicate compounds in the sample. These methods preserve the sharpness (high-frequency components) of the ion peaks by smoothing the curve using nonparametric and polynomial filtering methods [25, 26].

Figure 4.16  Spectrum normalized to an internal standard
Lowess filters smooth a mass spectrum by using a locally weighted linear regression method. The smoothing process is considered local because each smoothed value is determined by neighboring data points within a span. The process is weighted because a regression weight function is defined for the data points contained within the span. The weight sequence is given by the tricube function shown below [27, 28].

\[
w_i = \left(1 - \left| \frac{x - x_i}{h} \right|^3 \right)^3
\]

4.3

The \(m/z\) vector might not be uniformly spaced. Therefore, the sliding window (span) for smoothing is centered using the closest samples in terms of the \(m/z\) value and not in terms of the \(m/z\) vector indices.

For example, if the span is 10 samples, the method consists of performing a locally weighted regression smoothing algorithm by applying a full least-squares fit with the 10 closest samples to the point to be fixed. This step is repeated for every point in the signal. One of its strengths lays in its ability to effectively adapt to data with nonuniformly spaced values.

A linear fit (Lowess) or a quadratic fit (Loess) is usually employed, but a zero order may also be used, which is equivalent to a weighted local mean estimator. Samples are weighted in the fitting process, which allows emphasis of those samples that are closest to the point being fixed. Different weighting approaches have been proposed such as using a tricubic function, a Gaussian pulse, or a triangle shape.

This polynomial fitting approach allows an estimate of how much you need to correct at every point. By doing some statistics on this data, it is easy to detect potential outliers which you can simply remove from the signal. This allows reapplying the algorithm until no more outliers are detected, and recalling that the previous procedure in the algorithm does not require evenly spaced samples [29].

![Smoothed spectrum using a least-squares polynomial filter.](image)

**Figure 4.17** Smoothed spectrum using a least-squares polynomial filter.
• Savitzky and Golay filters smooth a mass spectrum using a least-squares digital polynomial filter. The Savitzky and Golay method of smoothing is a generalization of the Lowess method. You derive the filter coefficients by performing an unweighted linear least squares fit using a polynomial of a given degree. It allows you to use higher order polynomials for the fitting. As a result, the algorithm preserves signal features such as the resolution between ion peaks and the height of the peaks. The original algorithm by Savitzky and Golay assumes a uniformly spaced mass/charge vector while the function mssgolay also allows one that is not uniformly spaced [30].

One of the most important parameters in polynomial filtering is the size of the window, (or spanning). It is indirectly associated with the cut-off frequency. However, there is not a practical relation between these two so you can usually adjust the window based on experimental experience. For example, in a low resolution mass spectrum signal, it is common to have the span set to 15-20 samples.

**Example demonstrating noise smoothing**

Smooth the normalized spectra with a polynomial filter of second order. Most of the mass spectrometry preprocessing functions in the Bioinformatics Toolbox have an input parameter Showplot that creates a customized plot to help you follow and assess the quality of the preprocessing action.

\[
YS = \text{mssgolay}(MZ, YN2, \text{`SPAN'}, 35, \text{`ShowPlot'}, 3);
\]

Zooming into a reduced region reveals the detail of the smoothing algorithm. See Figure 4.18.

\[
\text{axis([8000 9000 -1 8])}
\]

![Figure 4.18 Detail showing noisy and smoothed spectrum.](image-url)
4.7 Identifying Ion Peaks

After you have adjusted the baseline, corrected for calibration, normalized the intensities, and smoothed the spectra, you can identify peaks.

A simple approach to finding putative peaks is to look at the first derivative of the smoothed signal.

```matlab
slopeSign = diff(YS(:,1)) > 0;
slopeSignChange = diff(slopeSign) < 0;
h = find(slopeSignChange) + 1;
```

Remove peaks in the low-mass region below 1500 Daltons and small ion intensity peaks with a height below 5.

```matlab
h(MZ(h) < 1500) = [];
h[YS(h,1) < 5] = [];
```

Plot the spectrum with identified peaks.

```matlab
plot(MZ,YS(:,1),'-',MZ(h),YS(h,1),'ro');
```

See Figure 4.19 showing the ion peaks detected in a spectrum.

More elaborate peak detection methods use discrete wavelet transforms (DWT) for isolating the noise, and then finding the putative peaks. When using DWT special care needs to be taken to account for signal shifts and varying signal resolution [31].

![Figure 4.19](image.png)
References


