4

Pre-Processing Mass Spectrometry Data

Kevin R. Coombes, Keith A. Baggerly, and Jeffrey S. Morris

Department of Biostatistics and Applied Mathematics, University of Texas M.D.
Anderson Cancer Center, Houston, TX 77030, USA
krc@odin.mdacc.tmc.edu
kabagg@wotan.mdacc.tmc.edu
jeffmo@wotan.mdacc.tmc.edu

4.1 Introduction

Mass spectrometry is being applied to discover disease-related proteomic patterns in complex mixtures of proteins derived from tissue samples or from easily obtained biological fluids such as serum, urine, or nipple aspirate fluid (Paweletz et al., 2001; Wellmann et al., 2002; Petricoin et al., 2002; Adam et al., 2002, 2003; Zhukov et al., 2003; Schaub et al., 2004). Potentially, we can use these proteomic patterns for early diagnosis, to predict prognosis, to monitor disease progression or response to treatment, or even to identify which patients are most likely to benefit from particular treatments.

The mass spectrometry instruments most commonly used to address these clinical and biological problems use a matrix-assisted laser desorption and ionization (MALDI) ion source and a time-of-flight (TOF) detection system. Briefly, to run an experiment on a MALDI-TOF instrument, the biological sample is first mixed with an energy absorbing matrix (EAM) such as sinapinic acid or α-cyano-4-hydroxycinnamic acid. This mixture is crystallized onto a metal plate. (The commonly used method of surface enhanced laser desorption and ionization (SELDI) is a variant of MALDI that incorporates additional chemistry on the surface of the metal plate to bind specific classes of proteins (Merchant and Weinberger, 2000; Tang et al., 2004).) The plate is inserted into a vacuum chamber, and the matrix crystals are struck with pulses from a nitrogen laser. The matrix molecules absorb energy from the laser, transfer it to the proteins causing them to desorb and ionize, and produce a plume of ions in the gas phase. This process takes place in the presence of an electric field, which accelerates the ions into a flight tube where they drift until they strike a detector that records the time of flight (Figure 4.1).

In theory, the spectral data produced by a single laser shot in a mass spectrometer consists of a vector of counts. Each count represents the number of ions hitting the detector during a small, fixed interval of time. We refer to this interval of time as the *time resolution* of the instrument; the time reso-

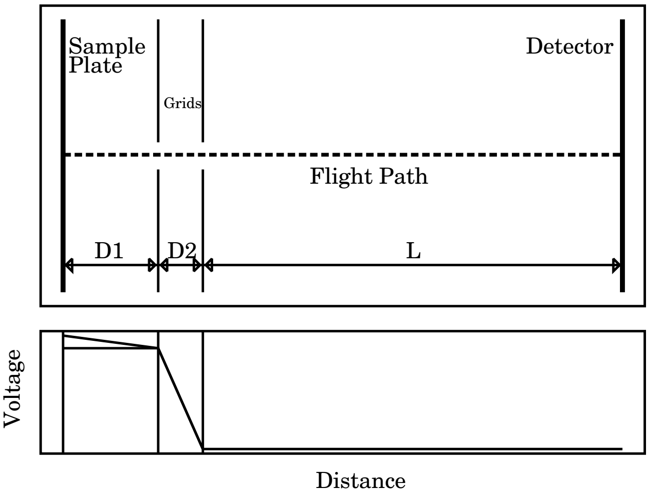

Fig. 4.1. (Top) Simplified schematic of a MALDI-TOF instrument with time-lag focusing. Samples are inserted on a metal plate into a vacuum chamber where they are ionized by a laser. Electric fields between the sample plate and two charged grids accelerate the ions into a drift tube, where they continue until they strike a detector. **(Bottom)** Voltage potentials along the instrument. The sample plate and grid start at the same potential, but the potential is raised after a brief delay.

lution is typically on the order of 1–4 nanoseconds. A complete spectrum is acquired within tens of milliseconds, so a typical spectrum is a vector containing between $10,000$ and $100,000$ entries. In practice, most mass spectrometers produce spectra by averaging the counts over many (often a few hundred) individual laser shots. Thus, the raw data produced by running a sample through a mass spectrometer can best be thought of as a time series (see Chapter 11) vector containing tens of thousands of real numbers. Unless an entry in the vector is known to represent an actual count of the number of ions, it is usually just called an *intensity* and is assumed to be measured in continuous arbitrary units. Peaks in a plot of the intensity as a function of time represent the proteins or peptides that are present in the sample (Figure 4.2, top).

It is important to realize that the natural scale on which to view a mass spectrum is the time axis along which the data was originally collected. Applications of mass spectrometry are, however, based on the mass of the particles. Ions of different mass are separated in the flight tube. In general, lighter ions fly faster and thus reach the detector before heavier ions. More precisely, the velocity achieved by an ion is proportional to its *mass-to-charge ratio* (m/z). A quadratic transformation is used to compute m/z from the observed flight time. The coefficients of this quadratic transformation must be determined experimentally. Researchers prepare a sample containing a small number (typically between 3 and 7) of molecules of known masses and use it to generate a spectrum. They then determine the times at which the peaks corresponding to the known masses occur in that spectrum, and use least squares and this set of $(time, mass)$ pairs to determine the coefficients of the quadratic transformation. The process of mapping the observed time of flight to the m/z values is called *calibration.*

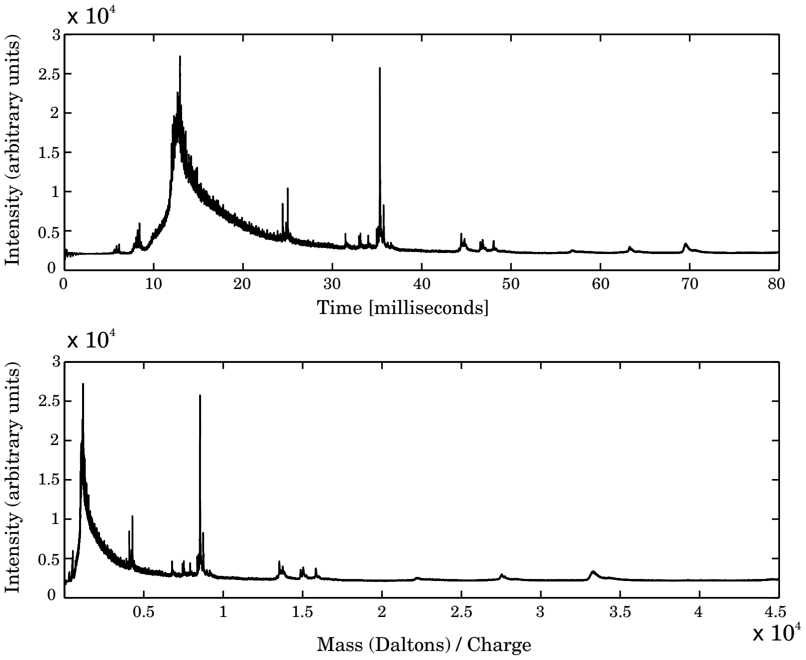

Fig. 4.2. A sample spectrum displayed on two scales. **(Top)** Intensity data as a function of the actual time-of-flight. **(Bottom)** Intensity as a function of the calibrated mass-to-charge ratio. Mass is measured in Daltons; charge is measured in multiples of the charge of one electron.

A typical data set arising in a clinical application of mass spectrometry contains tens or hundreds of spectra; each spectrum contains many thousands of intensity measurements representing an unknown number of protein peaks. Any attempt to make sense of this volume of data requires extensive low-level processing in order to identify the locations of peaks and to quantify their sizes accurately. Inadequate or incorrect pre-processing methods, however, can result in data sets that exhibit substantial biases and make it difficult to reach meaningful biological conclusions (Baggerly et al., 2003; Sorace and Zhan, 2003; Baggerly et al., 2004b,a). The low-level processing of mass spectra involves a number of complicated steps that interact in complex ways. Typical processing steps are as follows.

- *Calibration* maps the observed time of flight to the inferred mass-to-charge ratio.
- *Filtering* or *denoising* removes random noise, typically electronic or chemical in origin.
- *Baseline subtraction* removes systematic artifacts, usually attributed to clusters of ionized matrix molecules hitting the detector during early portions of the experiment, or to detector overload.
- *Normalization* corrects for systematic differences in the total amount of protein desorbed and ionized from the sample plate.
- *Peak detection* is the process of identifying locations on the time or m/z scale that correspond to specific proteins or peptides striking the detector.
- *Peak quantification* is the primary goal of low-level processing; it typically involves an assessment of the signal-to-noise (S/N) ratio and may involve heights or areas.
- *Peak matching* across samples is required because neither calibration nor peak detection is perfect. Thus, the analyst must decide which peaks in different samples correspond to the same biological molecule.

In the realm of mass spectrometry, there is a clear distinction between peak *detection* and peak *identification*. The peaks seen by a mass spectrometer are anonymous. The only thing we know about them is their mass, which is never enough to completely characterize the protein or peptide that made the peak. The term *peak identification* refers to the process of determining the exact species of protein molecule that caused a peak to be detected. This process typically involves additional experimentation (often by shunting molecules of a target mass into another instrument where they are physically fragmented along amino acid boundaries and sent through a second mass spectrometer to determine the sizes of the fragments) and database searches to compare the results with the fragmentation patterns of known proteins.

The potential importance of the clinical applications of mass spectrometry has drawn the attention of increasing numbers of analysts. As a result, the development of better methods for processing and analyzing the data has become an active area of research (Rai et al., 2002; Baggerly et al., 2003; Coombes et al., 2003; Hawkins et al., 2003; Lee et al., 2003; Liggett et al., 2003; Wagner et al., 2003; Yasui et al., 2003a,b; Zhu et al., 2003; Coombes et al., 2005b; Morris et al., 2005). One should note that not all methods use all of the processing steps listed above, nor do they necessarily perform them in the same order.

4.2 Basic Concepts

Statistically, the low-level processing of mass spectra reduces to decomposing the observed signal into three components: True signal, baseline, and noise.

One might try to decompose a spectrum using a model represented schematically by the equation

$$f(t) = B(t) + N \cdot S(t) + \epsilon(t) \tag{4.1}$$

where $f(t)$ is the observed signal, $B(t)$ is the baseline, $S(t)$ is the true signal, N is a normalization factor, and $\epsilon(t)$ is the noise. At present, this model is of limited utility, since we do not have an effective characterization of the individual components. The true signal can, in principle, be modeled as a sum of independent, possibly overlapping, peaks, each corresponding to a single protein. Approximate shapes of the peaks might be estimated empirically by simulating the physical process by which a time-of-flight (TOF) mass spectrometer collects data (Coombes et al., 2005a; Morris et al., 2005). White noise is a plausible model for the final term in the model, based on the notion that it arises primarily from electronic noise in the detector. One might also argue that at least some components of the noise have additional structure that is time dependent or even periodic (Baggerly et al., 2003). A fundamental limitation of the model in Equation 4.1, however, is that we do not have a good theoretical model for the baseline, aside from the vague intuition that it consists of a very low frequency component of the observed signal. This intuition is difficult to use without making it more precise, because the shape of the true peaks changes within a spectrum, becoming significantly lower and broader at later times and higher masses.

Our current procedure for processing sets of mass spectra is founded on two principles. First, the raw data is the ultimate arbiter; processing should be kept to a minimum in order to avoid introducing additional variance or additional bias into the measurements that will be used in later statistical analyses. Second, we should borrow strength across samples whenever possible.

1. Align the spectra on the time scale by choosing a linear change of variables for each spectrum in order to maximize the correlation between spectra.
2. Compute the mean of the aligned raw spectra.
3. Denoise the mean spectrum using the *undecimated discrete wavelet transform* (UDWT).
4. Locate intervals containing peaks by finding local maxima and minima in the denoised mean spectrum.
5. Quantify peaks in individual raw spectra by recording the difference between the maximum height and minimum height in each interval that should contain a peak.
6. Calibrate all spectra using the mean of the full set of available calibration experiments.

4.3 Advantages and Disadvantages

The chief advantage of performing peak finding by locating intervals in the mean spectrum that contain peaks is that it avoids the extremely messy and error-prone problem of matching peaks across spectra. The corresponding disadvantage is that this will only work if the spectra have been aligned properly before computing the mean (Figure 4.3). A small amount of misalignment is safe; it merely broadens the peaks in the mean spectrum. Severe misalignment, however, can make the data unusable.

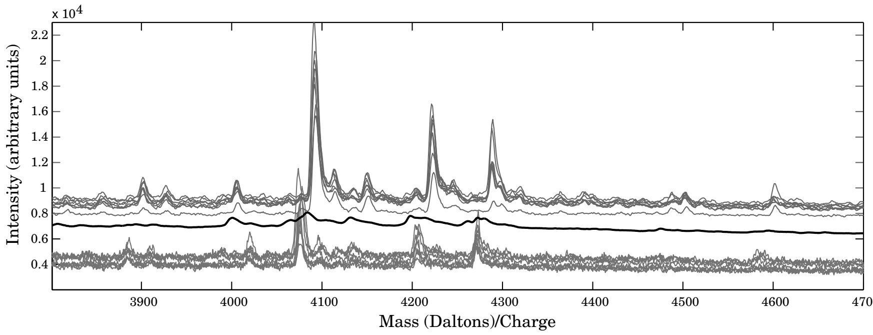

Fig. 4.3. Mean spectrum on improperly aligned data. The same sample was procssed in multiple laboratories for several weeks. The two sets of gray curves are spectra from different laboratory-weeks. The heavy black curve is the mean spectrum over all laboratories and weeks. The sharp peaks that are present in the individal spectra have been diluted in the mean spectrum by a failure to align the spectra properly.

There are two advantages that follow from performing alignment on the time scale rather than first calibrating and then aligning on the mass scale. First, it is simpler, since it only requires a linear change of variables instead of a quadratic. Second, it is more reproducible, since it does not incorporate any additional errors that might be introduced in the calibration step. This factor is particularly important in many of the applications of mass spectrometry to protein profiling of complex mixtures. In many studies, the instrument is only calibrated in a fairly narrow range, but data is collected over a much wider range. For example, Ciphergen has a low mass standard mixture that contains five proteins with masses between 1084.2 and 7033.6 Daltons; their high mass standard mixture contains proteins with masses between $12,360.2$ and *something* Daltons. Both calibrant mixtures have been used while acquiring spectra from 1000 to $50,000$ Daltons or higher. When the calibration is extrapolated in this way, the errors can be substantial. Our final calibration step, which averages the results of multiple calibration experiments, should perform more accurately, even when extrapolated, than using a single calibration experiment.

The peak quantification step in our procedure implicitly performs local baseline correction without fitting an explicit curve. The local minimum in the interval containing the peak is taken to be the local definition of baseline. Without a coherent model that explicitly describes the shape baseline takes, preferably one motivated by the physical processes that affect the detector in a mass spectrometer (Malyarenko et al., 2005), fitting baseline can be problematic. Using the local minimum as an estimate of baseline has several advantages. First, it is simple to compute. Second, it does not require fitting either a parametric or nonparametric model that may simply not be appropriate in some circumstances. For example, the spectrum in Figure 4.2 has a baseline that might be modeled by an exponential decay starting at a high point near 12 ms. The baseline before 12 ms, however, clearly has a different shape. We have also seen spectra with two large bumps instead of one, which makes it difficult to specify a model that will work in full generality.

Another advantage of quantifying the peak height as the difference between local maximum and local minimum on a nonempty interval is that it avoids assigning a quantification of zero. Nonexistent peaks in a sample will be assigned a value that is proportional to the noise in the spectrum. By biasing the estimates slightly high in this manner, it is easier to work with transformations of the peak height in later statistical analyses of the data. When using alternative methods that assign a value of 0, analysts who want to use a log-transformation typically make an arbitrary choice to truncate the data before transformation. In essence, our method accepts additional bias in order to reduce some of the variance and avoid depending on arbitrary thresholds.

A critical disadvantage, however, is that the height of overlapping peaks can be biased significantly low (Figure 4.4). If a peak overlaps with other peaks on both sides, then the local minima will not come all the way down to the true baseline. In many cases, such overlapping peaks often represent related molecules that will be highly correlated in expression. There are a number of phenomena that give rise to such related molecules. For example, some proteins can carry along one or more matrix molecules (or *adducts*). The acids used in the matrix typically have a mass between 100 and 200 Daltons. A collection of regularly spaced peaks with mass difference in this range often represents the same protein or peptide carrying different numbers of matrix adducts. Proteins can also pick up sodium ions (with a mass of 22 Daltons) or lose a water molecule (with a mass of 18 Daltons). So, peaks whose mass difference is 18 or 22 Daltons also often represent the same protein or peptide. At a finer scale, isotopes of carbon (^{12}C vs. ^{13}C), nitrogen (^{14}N vs. ^{15}N), oxygen (^{16}O vs. ^{18}O) or other common elements can be incorporated into proteins in different numbers, leading to chemically identical proteins that differ in mass by 1 or 2 Daltons. Most mass spectrometers can be focused, at least at low mass levels, to be able to resolve differences smaller than a single Dalton, which occur when ionized proteins acquire multiple charges.

The mass spectrometry community appears to be converging on the use of wavelets for denoising. Because the intrinsic shape of a peak changes with

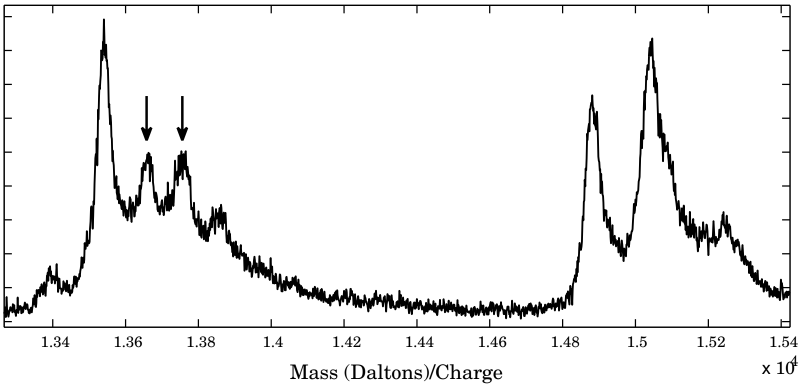

Fig. 4.4. Closeup of a raw spectrum. The two peaks indicated by arrows overlap with the peaks on either side, so the local minima closest to these peaks do not go all the way down to baseline.

the mass (becoming broader and lower at higher mass), the adaptive, multiscale nature of wavelets makes them a natural choice for denoising mass spectra, since these properties allow them to efficiently capture peaks of different widths. The wavelet approach for denoising involves three steps. The first is to compute the wavelet coefficients from the data, which involves choosing a basic wavelet basis function, then applying a series of linear filters derived from this function in a pyramid-based algorithm, called the *discrete wavelet transform* (Mallet, 1989). Applying this transform to a set of spectra results in a vector of wavelet coefficients summarizing signals at different frequencies and locations within the spectra. Second, set small wavelet coefficients to zero (*thresholding*), and third, compute the inverse wavelet transform to recover the denoised spectrum. The larger coefficients not set to zero can either be shrunken towards zero (*soft thresholding*) or left as they are (*hard thresholding*). In our experience, hard thresholding seems to perform better in denoising applications, since it results in less bias in the reconstructed denoised signal. Researchers still have a number of choices to make when using wavelets, however. They must select a basic wavelet basis function on which to base the transform (we usually use a Daubechies wavelet of degree 8, (Daubechies, 1992), the kind of transform (we use the UDWT (Lang et al., 1995, 1996; Gyaourova et al., 2002)), and the thresholding procedure (we use hard thresholding, with the threshold determined manually). The UDWT is superior to the more common decimated discrete wavelet transform (DDWT) when it comes to denoising. Its primary advantage is that, by construction, the UDWT is shift-invariant. The DDWT, by contrast, can produce different results if the start of the signal is shifted by a few time points. As a consequence, denoising with the DDWT can introduce significant artifacts into the signal near either end of the spectrum.

4.4 Caveats and Pitfalls

We have already mentioned some of the major difficulties that can arise using this procedure. First, the spectra must be properly aligned on the time scale. If this step is not peformed correctly, then the peaks can be completely "out of phase" in some regions of the spectra, causing them to disappear from the mean spectrum. One also has a choice of trying to compute all pairwise alignments or just selecting a "standard" spectrum and aligning all other spectra with the standard. Using all pairwise alignments can lead to computationally challenging optimization problems. By contrast, the alignments can potentially vary if one standard spectrum is replaced with another. Our own practice is to use the "most typical" spectrum as a standard to which all others are aligned. In order to select the most typical spectrum, we first compute the mean spectrum without any alignment, and compute the Pearson correlation between this unaligned mean and each spectrum. The most typical spectrum is defined to be the one that maximizes the correlation with the mean.

One concern is that protein peaks that are present in only a few spectra will not be detectable in the mean. In an extensive simulation study, we compared peak finding using the mean spectrum to peak finding in individual spectra followed by matching peaks across spectra (Morris et al., 2005). Large peaks, even if rare, can still be found in the mean. Peaks that are small and rare are harder to find, but our simulations indicate, as a reasonable rule of thumb, that any peak that is present in at least $\sqrt{N}$ spectra, where N is the number of spectra in the study, is as likely to be detected in the mean as it is in individual spectra. If you believe that it is important to find small peaks that are present in fewer than $\sqrt{N}$ spectra, than you will have to supplement the mean spectrum approach with the study of individual spectra. In the situation where there are natural biological groups of spectra (for example, cancer patients vs. healthy controls), one may be able to restrict peak finding to the group mean spectra and the overall mean. In this approach, the peaks in the overall mean would be used to match most of the peaks found in the group means, and rare peaks that are present in only one group could still be located.

Our preliminary studies using the UDWT suggest that the degree of the Daubechies wavelet does not affect the results very much, so it is probably safe to use the one of degree 8 (Coombes et al., 2005b). Using hard thresholding also appears to do a better job than soft thresholding of preserving the actual shape of peaks. The only problematic part of wavelet denoising is selecting the threshold at which to truncate the wavelet coefficients. We use a variant of a SiZer plot (Chaudhuri and Marron, 1999) to select a threshold interactively. Our SiZer routine computes the denoised spectra over a user-specified range of thresholds, including one extreme value that provides a "super-smooth" curve. The differences between the super-smooth curve and the various denoised spectra are displayed in a heatmap, with time along the horizontal axis and

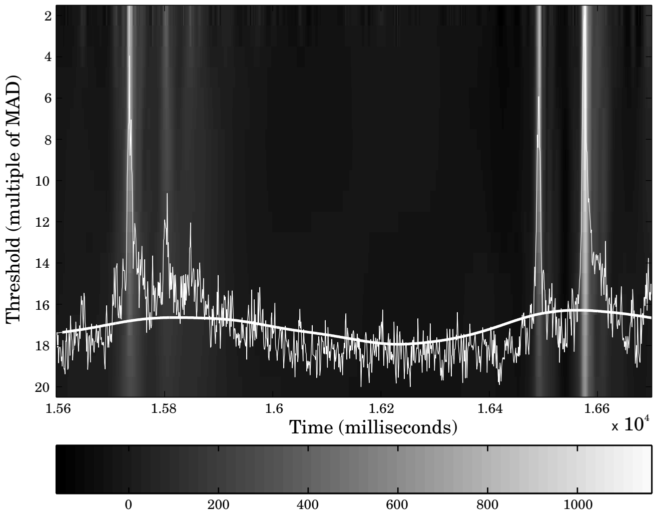

Fig. 4.5. SiZer plot of the effect of different wavelet thresholds (vertical axis) on the deviations of denoised spectra from a highly smoothed version (white curve).

thresholds along the vertical axis. The raw spectrum and the super-smooth curve are overlaid on top of the heatmap. In the example in Figure 4.5, most of the noise has been removed by the time the threshold reaches 4 or 5. The rightmost of the set of three peaks centered around 15, 800 clock ticks appears to fade by the time the threshold reaches about 10 or 12. For this spectrum, a threshold between 6 and 10 looks appropriate. By focusing the SiZer plot on different regions of the spectrum, the analyst can refine this estimate and select a threshold that retains most of the visible peaks without following all the zigs and zags in the noise. It would, of course, be extremely useful if the selection of the threshold could be automated, preferably by defining a reasonable objective function of the threshold that could be optimized.

We have also described the biases that can occur in the heights of peaks that overlap their neighbors. One can, of course, insert any preferred baseline correction method between Steps 4 and 5 of the procedure described above. One would then have a choice of quantification methods available, including the maximum peak height or the area under the curve. Regardless of which method is used, however, a critical issue affecting downstream analysis of the resulting peak quantification matrix is the high level of correlation between peaks. Many successful analyses of mRNA expression microarray data have been conducted that either explicitly or implicitly assume that genes are independent. We suspect that the success of these methods has depended, at least in part, on the fact that the correlation matrix for gene expression is relatively sparse. The correlation matrix for protein peaks, by contrast, appears to be much denser. In addition to matrix adducts, sodium adducts, and isotope distributions that give rise locally to correlated peaks, there can also be distant correlation arising from the same protein present in the mixture in different charge states. (Keep in mind that we can only infer the mass-to-charge ratio from the time-of-flight, and cannot isolate the mass.) In some cases, there can be significant negative correlation between peaks that is both biologically and statistically significant. For example, phosphorylating a protein adds an 80-Dalton phosphate group to the unmodified protein, producing two peaks separated by 80 Daltons. Biologically, phosphorylation typically activates a protein, changing its behavior within the cell. It is certainly conceivable that one important difference between cancer cells and their healthy counterparts may lie not in the amount of a particular protein that is present but on the extent to which that protein is activated. If this is the case, then it could give rise to a pair of negatively correlated peaks separated in mass by 80 Daltons. In general, analysts dealing with peak quantification data from mass spectrometry experiments should be prepared to incorporate the correlation structure into their models.

The method described here does not perform normalization as a routine part of pre-processing. Analysts can still perform normalization later using the quantified peak heights. Such normalization can borrow techniques from the world of mRNA microarrays. For example, global normalization by dividing by the median peak height is likely to be robust and reasonably effective. One can also use linear mixed models in the spirit of Kerr et al. (2000) or Wolfinger et al. (2001) to incorporate peak-based normalization into the analysis of differential expression. Other alternatives for normalization are described in the next section.

4.5 Alternatives

Most alternative methods normalize by dividing by the *total ion current* (TIC), which is just the sum of the intensities under all or a substantial portion of the curve. Methods for computing TIC vary widely; it can be computed on raw data, baseline corrected data, or smoothed data. It can also be computed on the time scale or on the m/z scale. One must be careful on the m/z scale because some computations fail to account for the fact that the observations are no longer equally spaced. The total area under a curve estimated at a few thousand time points can be quite large; consequently, the normalized values are often multiplied by a large (arbitrary) constant to put the intensity units on a scale that doesn't require quite so many decimal points to display.

A basic suite of methods for processing SELDI data is implemented in the ProteinChip software from Ciphergen (Fung and Enderwick, 2002); these methods are comparable to those that have traditionally been used in the mass spectrometry community. Their default analysis is close to the order in our initial descripiton of processing steps. They process one spectrum at a time, beginning with calibration to map the time-of-flight data to m/z values. They then perform baseline correction by fitting a varying-width segmented convex hull to the spectrum. Optionally, one can first smooth the spectrum by computing a moving average in a fixed width window before fitting the convex hull. Our own experience with Ciphergen's baseline correction suggests that it has a tendency to slice through the bottoms of peaks in areas of rapidly changing baseline (such as the region from 10 to 20 ms in Figure 4.2). They next denoise the spectrum either using a moving average or a Savitzky-Golay filter. The window size for the moving average can be constant on either the time scale or the m/z scale, or can vary over segments of the spectrum to account for the differences in the expected width of peaks. Their peak detection algorithm attempts to identify regions that rise above local valleys by a user-specified multiple of the noise. Peaks can be filtered based on the signal-to-noise ratio (S/N), whether the width of the peak at half-height is a specified multiple of the expected peak width, or by requiring the peak to have some minimum area. Normalization is performed by dividing by TIC or by the height or area of a specified control peak. Because the Ciphergen algorithm finds peaks in individual spectra, they must make a second pass to decide which peaks "match", or represent the same protein, in different spectra. They typically match peaks if their relative mass differs by a fixed percentage; this algorithm is based on the idea that the intrument has a nominal mass accuracy typically on the order of $0.1\% - 0.3\%$ across the entire range. In practice, such accuracies are probably achievable in the calibrated region, but the errors can be much larger when the calibrations are extrapolated to a wider range.

Yasui and colleagues (2003b) have described a method that does not attempt to quantify peaks; instead, they compute a binary indicator for the presence or absence of a peak. They define a point on the graph of the spectrum to be a peak if it satisfies two properties. First, it must be a local maximum in a fixed width window. (They use a window that extends 20 clock ticks on either side.) Second, it must have an intensity value higher than the average intensity in a broad neighborhood, where this average is computed using the super-smoother method in a window containing 5% of the data points. Because their downstream analysis only depends on presence or absence of peaks, they do not need to concern themselves with baseline correction, and denoising is implicitly accounted for by the super-smoother. They must still find an appropriate way to match peaks across spectra.

Our own pre-processing methods have evolved over time. Initially, we used a series of steps closely related to the Ciphergen routines (Baggerly et al., 2003). This method worked on calibrated spectra one at a time. We started by performing baseline subtraction using a "semimonotonic" local baseline. We began by computing the local minimum in a fixed sized window (200 time steps). We next imposed a monotonicity requirement. (Note that this method would only make sense for the spectrum in Figure 4.2 by discarding the portion to the left of about 12 ms.) Since the combination of monotonicity with local minima would tend to be biased low as we moved to the right (and thus had a greater opportunity to see extremely low values of the noise), we added a "fuzz" parameter and computed the baseline as the smaller of the "monotone minimum + fuzz" and the "local minimum". We then normalized to TIC. The spectrum was then divided into windows whose width increased smoothly (along a quartic polynomial) across the spectrum. We quantified peaks as the maximum value in the baseline corrected spectrum in each window.

Our second method also worked on calibrated spectra one at a time (Coombes et al., 2003). This method performed peak-finding on the raw spectra, without baseline correction or denoising. Using first differences, a large list of candidate peaks was generated from all local maxima in the raw spectrum. The median absolute value of the first differences was used as an estimate of noise, and any local maximum that did not rise above the nearest local minimum by more than the noise was eliminated. Next, local maxima that were separated by fewer than $T = 3$ time steps of $M = 0.05\%$ relative mass units were combined into a single maximum. Then any peak where the slope from the maximum down to the nearby local minima was less than half the noise was eliminated. After this preliminary peak list was generated, the intervals containing the peaks were removed from the spectrum and replaced by linear interpolations. The baseline was estimated from the peak-free spectra by taking the local minimum in a fixed width window. The process of peak-finding and removal for baseline estimation was iterated to produce a stable baseline-corrected spectrum with an associated peak list. Peaks were matched across spectra if they differed in time by T time steps or in relative mass by M units.

Our third method initially worked one spectrum at a time on calibrated spectra, but introduced the UDWT for wavelet denoising (Coombes et al., 2005b). Denoising was performed as the first step of processing, using hard thresholding as described above. Baseline correction used a monotone local minimum; normalization was perfomed by dividing by TIC. Peak finding was performed on the denoised, baseline-corrected, normalized spectrum. After wavelet denoising, every local maximum is a candidate peak. Since the wavelet transform also gives local estimates of the noise, the only filtering performed on the peaks was to remove candidate peaks with S/N below a threshold. Peaks were quantified by the height of the local maximum in the processed spectrum. Peaks were matched across spectra if they differed in location by at most $T = 7$ time steps or in relative mass by at most $M = 0.3\%$.

The next step in the evolution of our pre-processing routines was to introduce the idea of using the mean spectrum for preprocessing (Morris et al., 2005). In this approach, we first aligned the spectra and computed the mean. We then denoised the spectrum using the UDWT, baseline corrected with a monotone minimum, and found peaks in the mean spectrum by keeping all local maxima with $S/N > 5$. In order to quantify these peaks in the individual spectra, the spectra were also wavelet denoised, baseline-corrected using the monotone minimum, and normalized to TIC. The size of a peak in an individual spectrum was taken to be the maximum value of the processed spectrum in the interval defining the peak.

All of these methods experience some difficulty with overlapping peaks, since the quantification for one peak will also contain possibly contaminating information from overlapping peaks. One approach for dealing with this problem is to model the spectra as a sum of peaks, with the peaks represented by some parametric form, and perform *deconvolution.* Ideally, this modeling and deconvolution should appropriately partition each intensity among all overlapping peaks. One example of this approach is given by (Clyde et al., 2006), in which the authors represent the peaks using a sum of Lévy processes. While potentially improving the quantifications, deconvolution also has the potential to introduce errors and extra variability to the process. There is a need for careful studies comparing methods involving deconvolution with those that do not.

Almost all methods in existing literature for analyzing mass spectrometry data involve first performing peak detection and quantification, then analyzing the peaks. An alternative approach is to model the mass spectra as functions, for example using functional mixed models (Morris et al., 2006). This approach has the potential to identify differentially expressed regions of the spectra that might be missed by peak detection algorithms, and also can automatically adjust for systematic effects due to nuisance factors, e.g. block effects, affecting both the intensities (y-axis) and locations (x-axis) of the peaks. Further study is necessary to compare the functional and peak-based approaches to determine the advantages and disadvantages of each.

4.6 Case Study: Experimental and Simulated Data Sets for Comparing Pre-Processing Methods

As you can tell from the previous section, a wide variety of methods have been proposed for pre-processing mass spectra. Not surprisingly, it can be difficult to determine which methods are better than others. The evolution of our own thought on the matter (described in painful detail above) has been guided by two kinds of data sets: Actual experimental data consisting of replicate spectra from the same sample, and a large set of simulated data.

Our collaborators have been willing to produce data sets containing numerous replicate spectra, obtained by processing aliquots of the same sample on different days and different chips. Specifically, samples of nipple aspirate fluid (NAF) were collected from women with unilateral breast cancer and from healthy women using methods that we have described elsewhere (Kuerer et al., 2004; Pawlik et al., 2005). Small amounts of the samples from all women in the study were pooled to produce a single quality control (QC) sample. The QC sample was divided into aliquots and stored at -80°C. In an initial experiment, the QC sample was processed on two spots of each of three different eight-spot ProteinChip arrays (Ciphergen Biosystems, Inc., Fremont, CA).

This procedure was repeated for four successive days, producing a total of 24 spectra from the same sample. In all subsequent experiments with biological samples of interest, two spots of each eight-spot ProteinChip array were used for the QC sample. Since 36 additional arrays were used, this produced 72 more replicate spectra from the same QC sample, collected over several months. This data set allows us to compare pre-processing methods by examining the extent to which they produce reproducible results on replicate spectra (Coombes et al., 2005b). Details on how these samples were used for QC have been described elsewhere (Coombes et al., 2003).

We analyzed the initial set of 24 QC samples using several different algorithms (Coombes et al., 2005b). Because all the samples were the same, our main concern was whether the processing methods could reproducibly find the same peaks. First, we applied our wavelet-denoising algorithm with a threshold of 10 to individual spectra, using the "montone minimum" to correct baseline. This method detected, on average, about 211 local maxima per spectrum in the region above 950 Daltons/charge. Of these local maxima, about 158 per spectrum had $S/N > 2$ and about 96 had $S/N > 10$.

Next, we analyzed the same spectra using the algorithm in the Ciphergen ProteinChip software. With the default parameter settings, the Ciphergen algorithm found only 9 peaks per spectrum. When we increased the "peak sensitivity" setting to maximum, making no other changes, then the Ciphergen algorithm found only 41 peaks per spectrum. Thus, the wavelet denoising method consistently found more peaks than the Ciphergen algorithm.

One possible explanation of the difference between the algorithms is that the Ciphergen algorithm is more conservative than the wavelet-based algorithm, and thus only finds the tallest, most reliable peaks. If this were the case, then we would expect the Ciphergen algorithm to be more reproducible across spectra. In order to test this possibility, we matched peaks across spectra if they differed in time by fewer than 7 time steps or in relative mass by less than 0.3%. With these matching criteria, the wavelet-based method found a total of 174 distinct peaks and the Ciphergen algorithm (at maximum sensitivity) found a total of 149 distinct peaks. We plotted a histogram counting the number of times, in 24 samples, that the same peak was identified as present (Figure 4.6). We found that with the wavelet-based algorithm, 47 peaks were present in all 24 spectra, 83 peaks were found in at least 20 spectra, and 130 peaks were found in at least 10 spectra. With the Ciphergen algorithm, by contrast, only 6 peaks were present in all 24 spectra, and 47 of the 149 distinct peaks were present in only 1 spectrum. On this data set, the wavelet-based methods not only identified more total peaks, but it identified them more reproducibly.

We also analyzed the same spectra using the method described by Yasui and colleagues (2003b). We applied their method with a grid of parameter values, letting the window parameter range take on the value 10, 20, . . ., 100 and the smoothing parameter take on the values 0.01, 0.02, 0.05, 0.07, 0.10, 0.15, and 0.20. For each combination of parameters, we computed the mean and

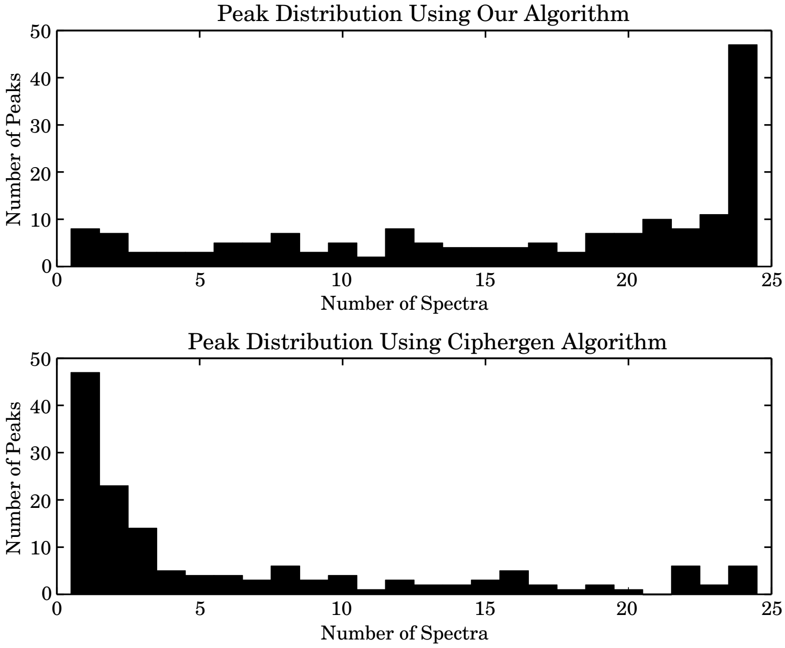

Fig. 4.6. Histograms showing the number of peaks found in replicate spectra. **(Top)** Our wavelet-based algorithm found 174 distinct peaks, and 47 of those peaks were found in all 24 spectra. **(Bottom)** The Ciphergen algorithm found 149 distinct peaks, but 47 of the peaks were identified in only one spectrum and only 6 peaks were identified in all 24 replicate spectra.

standard deviation of the number of peaks found in the 24 replicate spectra. The standard deviation was about the same (mean 64.26, range $60.36 - 70.43$) for all choices of the parameters. The mean number of peaks appeared relatively insensitive to the smoothing parameter, but decreased significantly as a function of the width parameter. Figure 4.7 shows a single spectrum in three different mass ranges. The overlaid curve is a super-smooth using 5% of the data points; circles indicate peaks found by Yasui's method using a window width of 80. With these parameters, their method detected an average of 267 "peaks" per spectrum. In the higher mass range (above 20,000 Da), these peaks do not appear to differ significantly from the surrounding noise. At lower mass ranges (between 2,000 and 3,000 Da), however, the window width prevented several clearly visible peaks from being detected. In the middle mass range, we also saw clear peaks (e.g, around $14,500$ and $14,800$ Daltons) that went undetected because they fell below the level of the super-smooth curve. If we decreased the window width or the super-smooth parameter in order to detect the obvious peaks in the low and middle mass ranges, we obtained vastly larger numbers of spurious peaks in the high mass region.

The reproducibility across spectra of the peaks found by Yasui's method was comparable to those found by the Ciphergen algorithm (data not shown).

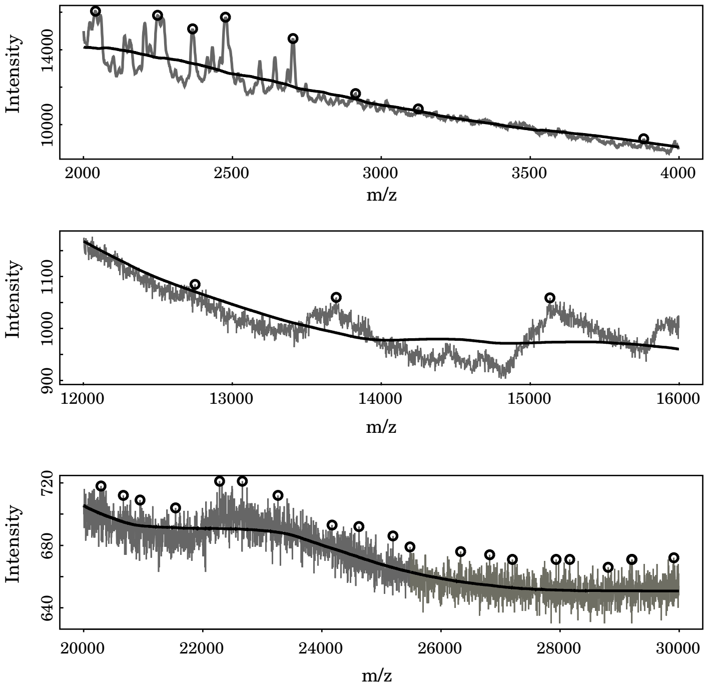

Fig. 4.7. Results of the peak-finding method proposed by Yasui and colleagues. The gray curve is the raw spectrum; the black curve is a super-smooth using 5% of the data. Circles mark local maxima that exceed the super-smooth level, which should correspond to peaks.

Reproducibility, by itself, is not enough to determine which method works better. One can potentially get more reproducible results by being very conservative about which features in a spectrum are called peaks. The largest peaks may be found very reproducibly, but the cost of a highly conservative approach is that a large number of smaller peaks may become "false negatives" — true peaks that cannot be used in later analyses because they were never found to begin with. Another potential problem is that the measure of reproducibility depends on matching peaks across spectra, using an algorithm that itself is not error-free. The matching step is required because even after calibration and alignment, peaks will not be perfectly aligned across replicates. Our matching algorithm joins peaks into "bins" if the difference in mass is less than 0.3%. Slight errors in alignment can combine with an occasional spurious peak to lump distinct peaks into a common bin (Figure 4.8).

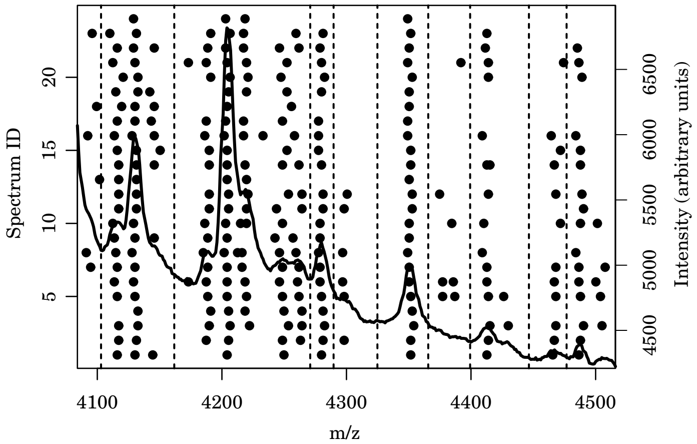

Fig. 4.8. Difficulties in peak matching. Circles indicate the presence or absence of peaks in the 24 replicate NAF spectra. Vertical lines mark the bins that separate distinct "matched" peaks. The overlaid curve is the mean spectrum.

Without knowing the true biochemical composition of the samples used in the experiments, it is hard to develop additional criteria by which to evaluate processing methods. To deal with this problem, we developed a simulation engine in S-Plus (Insightful Corp., Seattle, WA) that allowed us to simulate mass spectra from instruments with different properties (Coombes et al., 2005a). The simulation engine was based on a mathematical model of a physical mass spectrometry instrument. We initially used the model to explore some of the low-level characteristics of mass spectrometry data, including the limits on mass resolution and mass calibration, the role of isotope distributions, and the implications for methods of normalization and quantification. We then used the simulation engine to compare peak finding based on individual spectra to peak finding using the mean spectrum (Morris et al., 2005). We referred to the algorithm that matched peaks that were found by the wavelet-based algorithm on separate or single spectra as SUDWT. The algorithm that used the same denoising and baseline correction procedures but found peaks in the mean spectrum was called MUDWT.

For the simulation, we began with a *virtual population*, which is a distribution that describes the peaks that might be found in a *virtual sample* drawn from this population. An individual peak was characterized by four parameters: Its mass X, its mean M intensity on the log scale, its standard deviation S on the log scale, and its prevalence P, which is the probability that it is present in any given sample. We modeled the prevalence with a beta distribution and modeled the triple $(\log(X), M, S)$ with a multivariate normal distribution; the hyperparameters describing these distributions were estimated from real data. We simulated virtual populations containing 150

Table 4.1. Overall results from the simulation study. The top element in each box is the mean quantity over the 100 virtual experiments, and the bottom interval is the range. The comparison proportion p measures the proportion of the virtual experiments for which the MUDWT had higher sensitivity than the SUDWT plus one-half the proportion for which the methods tied.

Settings	Method	Sensitivity	FDR
n=100 σ=66	SUDWT	0.75 (0.60, 0.85)	0.09 (0.02, 0.26)
	MUDWT	0.83 (0.75, 0.92)	0.06 (0.00, 0.41)
	Comparison	0.97	0.80
n=100 σ=22	SUDWT	0.58 (0.43, 0.69)	0.25 (0.11, 0.41)
	MUDWT	0.74 (0.61, 0.84)	0.23 (0.10, 0.52)
	Comparison	1.00	0.63
n=100 σ=200	SUDWT	0.70 (0.61, 0.80)	0.08 (0.00, 0.17)
	MUDWT	0.78 (0.69, 0.87)	0.05 (0.00, 0.45)
	Comparison	0.97	0.86
n=33 σ=66	SUDWT	0.73 (0.63, 0.84)	0.09 (0.01, 0.20)
	MUDWT	0.80 (0.74, 0.86)	0.06 (0.00, 0.36)
	Comparison	0.99	0.85
n=200 σ=66	SUDWT	0.75 (0.58, 0.87)	0.12 (0.02, 0.46)
	MUDWT	0.85 (0.75, 0.91)	0.11 (0.00, 0.31)
	Comparison	1.00	0.69

peaks. In order to simulate a *virtual experiment*, we drew N samples from the population, processed them through our virtual mass spectrometer, and added Gaussian white noise with mean zero and standard deviation σ. For each combination of N and σ, we stimulated 100 different experiments. In each experiment, we applied both SUDWT and MUDWT to detect peaks. Performance of the algorithms was measured by the sensitivity (the proportion of true peaks matching at least one found peak) and the false discovery rate (FDR; the proportion of found peaks that matched no true peak). We found that, at comparable FDR levels, MUDWT had higher sensitivity overall than SUDWT (Table 4.1). SUDWT did have a slight advantage when detecting peaks at low abundance and low prevalence; see Morris et al. (2005) for details.

4.7 Lessons Learned

From our case study, we see that different pre-processing methods can lead to very different numbers of detected peaks. Thus, it is of crucial importance to identify approaches for comparing different methods and identifying which are most effective. We discussed two here. First, an experimental data set containing many replicate spectra from the same sample allows us to compare methods based on how reproducibly they detect peaks. Second, simulated spectra are useful for determining conditions under which different methods more accurately find and quantify peaks. We discussed a MALDI-TOF simulation engine that can be used to generate virtual spectra for which the true proteins and quantifications are known, and thus can be used to validate different methods. We focused on validating the peak detection step here, but it could be used equally well for comparing different denoising, baseline correction, and quantification methods, and could also be used to evaluate methods for identifying differentially expressed peaks and/or building classification models based on subsets of peaks.

4.8 List of Tools and Resources

Increased activity in the development of analytical tools to process mass spectra have produced a number of software packages.

1. A software package (Cromwell) implementing our methods in MATLAB (The MathWorks, Natick, MA) is available on our Web site at `http://bioinformatics.mdanderson.org/software.html`. The replicates in the NAF data set and the simulated data sets are also available by following the link to “Public Data Sets”.
2. *Bioconductor* (`http://www.bioconductor.org/`), which began as a project to develop analysis tools in the statistical programming language R, has recently added a package called `PROcess` for the low-level processing of mass spectra.
3. The *Cancer Bioinformatics Grid* (caBig) is an effort by the United States National Cancer Institute to develop reusable software tools, standards, ontologies, and shared data. Progress of the caBig proteomics working group can be followed at the Web site `https: //cabig.nci.nih.gov/workspaces/ICR/Meetings/SIGs/Proteomics/index_html`.
4. Under the auspices of caBig, Duke University has been developing a suite of R programs to process mass spectra, called RProteomics (`http://gforge.nci.nih.gov/projects/rproteomics`).
5. The wavelet-based methods described in Coombes et al. (2005b); Morris et al. (2005) and the methods described in Yasui et al. (2003a,b) have been implemented as a commercial add-on, Proteome 1.0, to S-PLUS (Insightful, Seattle, WA).

6. Incogen (Williamsburg, VA), in cooperation with proteomics researchers at William and Mary College and the Eastern Virginia Medical School, has included support for the processing and analysis of mass spectra in its Visual Integrated Bioinformatics Environment (VIBE) software.

Naturally, manufacturers of mass spectrometers supply software with their instruments that does some form of basic pre-processing. When shifting away from the manufacturer's software to an alternative package, one has to worry about file formats. Ciphergen, for example, saves spectra in a proprietary binary format but also allows you to export them as comma-separated-values with two columns (m/z and intensity) or in a simple XML format. The XML file format is usually preferable, since it retains information about the protocol and the condition of the instrument when the spectrum was acquired. Two different efforts are underway to develop standard XML formats for mass spectrometry data. The de facto standard appears to be mzXML (described in detail at `http://tools.proteomecenter.org/mzXMLschema.php`), which is supported by conversion tools that accept the native format from several different MALDI-TOF instrurments and was adopted by caBig. An alternative XML format, mzData (`http://psidev.sourceforge.net/ms`) is being developed by the Proteomics Standards Institute.

4.9 Conclusions

Numerous methods have now been suggested for pre-processing mass spectra, and both free and commercial software packages implementing these methods have become available. Because the methods can produce very different results, researchers interested in performing downstream analysis on the peak lists must make sure that the processing applied at the early stages is appropriate for their data. Ideas for quantifying which processing methods produce better results have started to be proposed, and data sets (both experimental and simulated) are available to start evaluating the performance of different methods. For most applications, it appears that peak detection using the mean spectrum is superior to methods that work with individual spectra and then match or bin peaks across spectra. Nevertheless, the development of better pre-processing methods remains an active area of research.