BACKGROUND: A major challenge in high-throughput clinical and toxicology laboratories is the reliable processing of chromatographic data. In particular, the identification, location, and quantification of analyte peaks needs to be accomplished with minimal human supervision. Data processing should have a large degree of self-optimization to reduce or eliminate the need for manual adjustment of processing parameters. Ultimately, the algorithms should be able to provide a simple quality metric to the batch reviewer concerning confidence about analyte peak parameters.

CONTENT: In this review we cover the basic conceptual and mathematical underpinnings of peak detection necessary to understand published algorithms suitable for a high-throughput environment. We do not discuss every approach appearing in the literature. Instead, we focus on the most common approaches, with sufficient detail that the reader will be able to understand alternative methods better suited to their own laboratory environment. In particular it will emphasize robust algorithms that perform well in the presence of substantial noise and nonlinear baselines.

SUMMARY: The advent of fast computers with 64-bit architecture and powerful, free statistical software has made practical the use of advanced numeric methods. Proper choice of modern data processing methodology also facilitates development of algorithms that can provide users with sufficient information to support QC strategies including review by exception.

© 2015 American Association for Clinical Chemistry

Accurate location and quantification of chromatographic peaks is sufficiently difficult that there are dozens of published approaches to accomplishing the task. Many of these require human input at one or more steps of the data processing algorithm. Others are incapable of reliably processing noisy traces containing peaks with amplitudes near the detection limit. These difficulties are compounded by the continued use, in many laboratories, of data processing strategies developed on older generation computers that were slow and had limited storage and poor numeric accuracy. The advent of modern computers and free, user-friendly software has enabled fast and reliable calculations that would not have been practical 10 years ago in a high-throughput laboratory.

This review will answer the following questions about processing chromatographic traces. Many of the answers rely on mathematical arguments which will be introduced at a level suitable for a typical experimentalist. To aid understanding, 3 Supplemental files are provided containing more detail about material that might be unfamiliar or mastered in the distant past (see the Data Supplement that accompanies the online version of this article at http://www.clinchem.org/content/vol62/issue1). Answers, highlighted in the text, will refer back to the questions.

1. How are derivatives used to locate peaks?
2. Why are digital filters used to process a chromatogram? How do they reduce noise?
3. Why do digital filters distort chromatographic peaks? Does distortion depend upon the type and, often, the length of the filter?
4. When a peak is described by 5 points, why can’t noise be reduced without distortion? Is it better to describe peaks using many points?
5. Do digital filters always cause baseline oscillations at the start and end of intense peaks?
6. In a high-throughput environment are there ways to automatically select the optimum filter?
7. How are boundaries identified for individual peaks within clusters?
8. Why do automatically generated baselines require long traces?
9. What restriction does curve fitting place on the number of points per peak? Can overlapping peaks be separated by least-squares?
10. How can peaks be automatically assigned a quality score useful for recognizing questionable peak areas?

Locating Peaks

DERIVATIVES VIA DIFFERENNCING
In stark contrast to human visual image processing, locating peaks in a noisy chromatogram is an incredible challenge for a numeric algorithm. Reliably performing
Digital filters process signals and obtain derivatives with reduced noise (3). This answers the first part of question 2. Digital filters process the chromatogram using the mathematical technique of convolution,

\[ s(t) = \sum_{t'} F(t') y(t - t') \]

where \( s \) is the smooth (or derivative) result, \( F \) is the filter, and \( y \) is the raw data. The left expression is the solution written in functional form, and the right expression is written in terms of vectors and indices. For this latter case let the data vector, \( y_i \), have indices, \( i = 1, \ldots, D \), and the filter vector, \( F \), have indices, \( j = 1, \ldots, L \), where \( L \) is odd and \( L \leq D \). The smoothed vector will have indices given by \( ii = M + 1, \ldots, D - M \), where \( M = (L - 1)/2 \). The summation has to be evaluated for each of the \( D - 2M \) values of \( ii \). The first and last \( M \) values are not determined.

The convolution operation will also reduce noise as long as \( \Sigma F_k^2 < 1 \). This can be seen by the propagation of variance in Eq. 2,

\[ s = \sum_{k=1}^{L} F_k y_{ii} \]

\[ \sigma_i^2 = \sum_{k=1}^{L} F_k^2 \sigma_{yi}^2 + \sum_{k=1}^{L} F_k^2 \sigma_{filter}^2 \]
where there are \( k \) terms within the summation and the simplification in the third expression assumes constant (homoscedastic) chromatographic noise. Thus, the “variance of the filter” is given by the sum of the squares of its elements. This answers the second part of question 2.

There are 2 general classes of digital filters, those that rely primarily on noise being normally distributed, such as running average (boxcar) and least-squares (Savitzky-Golay) filters, and those that primarily take advantage of differences between signal and noise frequencies. Fortunately chromatographic signals have frequencies occurring in a range that can be well estimated, whereas “white” (thermal and counting) noise is distributed with uniform probability over all frequencies. Thus it is possible to rationally design filters that maximize the reduction of noise while simultaneously minimizing signal distortion. To do this requires working in the frequency domain. The mathematical connection between a chromatogram (waveform) and its spectrum is called a Fourier transform. To simplify the discussion consider a chromatogram consisting of 1 gaussian-shaped peak centered at \( t = 0 \). With

\[
 y(t) = A \exp\left(-\frac{\pi t^2}{f_p^2}\right)
\]

where the subscript \( p \) denotes a chromatographic peak and \( f_p^2 = 1/t_p^2 \). When the gaussian shape is defined this way, the amplitude of the waveform is 0.0432 at \( t = \pm t_p \) and the area under the peak is \( A t_p \). Because of the reciprocal relationship between \( t_p \) and \( f_p \), a narrow chromatographic peak has a broad spectrum and vice versa.

When using digital filters it is convenient to assume that each data point is separated by 1 s. With this assumption the spectrum is also a gaussian peak centered at \( f = 0 \),

\[
 y(f) = \frac{A}{\pi} \exp\left(-\frac{\pi f^2}{f_p^2}\right)
\]

This answers the first part of question 3.

The design goal is to generate a filter spectrum that starts with unit amplitude at 0 Hz and smoothly decreases (rolls off) to zero amplitude as close as possible to the highest frequency of the signal. Roll-offs close to the signal distort the signal but reject a maximum amount of noise, whereas roll-off locations causing no distortion reject less noise. This is an unavoidable trade-off. Once designed, the Fourier transform of \( \text{filter}(t) \) yields \( \text{filter}(f) \), which is used to process the chromatogram via convolution. Example filter spectra are discussed in a later section.

CHROMATOGRAM SPECTRUM

To simplify the discussion consider a chromatogram consisting of 1 gaussian-shaped peak centered at \( t = 0 \). With
GAUSSIAN FILTERS AND THE DISTORTION/NOISE TRADE

Gaussian filters are second in popularity (after least-squares) for processing chromatograms. They are particularly useful when filter-induced baseline ripple cannot be tolerated. Such a situation occurs when a weak analyte peak is adjacent to a very strong interference. A nice example of a more general use is given by Danielsson, Bylund, and Markides. Also, a 2008 patent by Agilent mentions the use of a 2-dimensional gaussian filter. Gaussian filters are introduced at this point to simplify discussions of peak distortion and noise reduction.

They are based on the shape function used in Eq. 5 and given by:

\[
U_j^0 = \exp(-\pi t_j^2/\tau_f^2), \quad N^0 = \sum U_j^0
\]

\[
U_j^1 = -\frac{2\pi t_j}{\tau_f^2} \exp(-\pi t_j^2/\tau_f^2), \quad N^1 = \sum U_j t_j
\]

\[
U_j^2 = \frac{4\pi^2 t_j^2 - 2\pi t_j}{\tau_f^2} \exp(-\pi t_j^2/\tau_f^2), \quad N^2 = 0.5 \sum U_j t_j^2
\]

where \(U^0\), \(U^1\), and \(U^2\) are the unnormalized smoothing and derivative filters, and the associated \(N\) are the normalizers. That is, the filter is given by \(F^0 = U^0/N^0\), etc., where the superscript denotes the type of filter. Note that \(t\) runs from \(-M\) to \(+M\) (defined in the discussion of Eq. 2). The length of each filter is odd. The exact length is not specified for gaussian-based filters but must be sufficiently large that the end values are close to zero. Table 1 gives gaussian filters for \(\tau_f = 3.6\) and a length of 11. The spectrum of the smoothing filter is shown in Fig. 3.

When a gaussian-shaped peak with width \(\tau_p\) is convolved with a gaussian smoothing filter with width \(\tau_f\) the resulting smoothed peak is a broadened (distorted) gaussian with \(\tau_s = (\tau_p^2 + \tau_f^2)^{1/2}\).

To demonstrate the distortion-noise reduction trade consider the gaussian peak with \(\tau_p = 7\) shown in Fig. 1. Convolution with a filter having \(\tau_f = 3.6\) (see Table 1) will produce a smoothed peak having \(\tau_s = (7^2 + 3.6^2)^{1/2} = 7.89\) for a 12.7% width distortion and a filter variance of \(\Sigma (F_j^0)^2 = 0.196\). If instead a filter with \(\tau_f = 7\) is used, the smoothed peak now has \(\tau_s = 9.9\) for a 41% distortion.

\[\begin{array}{c|cccccccc}
\text{Type}\text{a} & \text{Filter } a^2 & \pm7 & \pm6 & \pm5 & \pm4 & \pm3 & \pm2 & \pm1 & 0 \\
\hline
\text{LSQ}^0 & 0.207 & -0.0839 & 0.0210 & 0.103 & 0.161 & 0.196 & 0.207 \\
\text{LSQ}^1,c & 0.0091 & -0.0455 & -0.0364 & -0.0273 & -0.0182 & -0.0091 & 0 \\
\text{LSQ}^2 & 0.0047 & 0.0350 & 0.0140 & -0.0023 & -0.0140 & -0.210 & -0.0233 \\
\text{G}^0 & 0.196 & 0.0006 & 0.0057 & 0.0314 & 0.105 & 0.218 & 0.278 \\
\text{G}^1,c & 0.048 & -0.0016 & -0.0112 & -0.0457 & -0.102 & -0.106 & 0 \\
\text{G}^2 & 0.036 & 0.0035 & 0.0191 & 0.0518 & 0.0487 & -0.0552 & -0.137 \\
\text{WS}^0 & 0.187 & -0.0031 & -0.0066 & -0.0095 & 0.0044 & 0.0528 & 0.132 & 0.209 & 0.241 \\
\text{WS}^1,c & 0.031 & 0.0019 & 0.0047 & -0.0018 & -0.0283 & -0.0648 & -0.0827 & -0.0258 & -0.0618 \\
\text{WS}^2 & 0.012 & -0.0022 & -0.0008 & 0.0151 & 0.0331 & 0.0306 & -0.0011 & -0.427 & -0.0618 \\
\end{array}\]

\[\text{a LSQ, Savitzky-Golay quadratic least-squares; G, gaussian, } \tau_f = 3.6; \text{ WS, windowed sinc, } f_{\text{edge}} = 0.12. \text{ Superscript 0 denotes smoothing; 1, first derivative; and 2, second derivative.}
\]

\[\text{b As defined here the indices are equal to } t_j \text{ in Eq. 6. When used in Eq. 2 indices } -5, \ldots, +5 \text{ become } 1, \ldots, 11, \text{ etc.}
\]

\[\text{c With first derivative filters the numeric signs shown are for negative indices. Reverse the sign for positive indices.}
\]
distortion. The filter variance drops to 0.101 and noise is reduced at the expense of distortion. The spectrum of the smoothed waveform is a gaussian function having \( f_s = 1/t_e \). In summary, use the time domain to determine noise reduction and the frequency domain to determine peak distortion.

Because convolution preserves peak area, moderate distortion due to oversmoothing may not be a significant problem. However, distortion can cause baseline separated peaks to coalesce, mandating multiparameter curve fitting to “deconvolve” individual peak areas. In a high-throughput laboratory it is expected that the distortion/noise reduction trade associated with filter “strength” will be sorted out during method development. In this regard note the recent paper by Felinger et al. (7), which reminds users that some vendor software has noise-filtering algorithms running during data acquisition. This may complicate rational filter design.

OTHER DIGITAL FILTERS

The most common digital filters used in chromatography are the so-called Savitzky-Golay, which are based on a running least-squares fit to the chromatographic data (8) and exist in published tables, making them readily available (1, 8, 9, 10). Hites and Biemann appear to be the first to apply them to chromatographic smoothing (11), with other authors not far behind (12). More recent applications feature use of the first (13), second (14, 15), and even third (16) derivatives. A 1997 patent by Hewlett-Packard mentions both moving average and least-squares filters (17), and a 2007 patent by Waters mentions using apodized least-squares filters for both smoothing and obtaining the second derivative of 2-dimensional LC-MS data (18).

Unfortunately, least-squares filters don’t easily capitalize on knowledge of the chromatograms frequencies. In addition, the filter variance is not easily varied and changes in jumps as filter length is increased. This makes it difficult to adjust the trade between distortion and noise reduction. An extensive discussion of how least-squares filters are generated and their properties are given in a review by Lytle (19). An example will be discussed later.

Another class of digital filters is that based on windowed sinc functions [see Chapter 7 in (1)]. The purpose of the window is to apodize the sinc function (make it “gracefully” approach zero) in an attempt to reduce ringing distortions. The version presented here is called the Hamming (or Happ-Genzel) with windowing achieved by an offset cosine,

\[
\sin(t) = \sin(2 \pi f_{edge})/2\pi f_{edge}
\]

\[
W^2(t) = \sin(t)[c + d \cos(2\pi t/W)]
\]

where \( c = 0.54, d = 0.46, f_{edge} \) is the filter spectrum edge frequency, \( t \) takes integer values from \(-W/2\) to \(+W/2\), and the length of the filter is given by \( L = W + 1 \). This is a high-performance filter used extensively in engineering applications. See Chapter 16 in Smith’s readable text (20). As the filter lengthens it induces oscillations on either side of the smoothed peak. For this class of filter the primary trade involves baseline oscillations against noise reduction.

Although a high-performance smoother, it is not clear that this class of filter has been used to determine derivatives. This is most likely due to the tedious calculus required to obtain a functional form. For example, from Eq. 7 it can be deduced that the first derivative is composed of 5 terms. The second derivative is worse, with a total of 13 terms. The final hurdle is determining the value of the second derivative filter when \( t = 0 \). To show the interested reader how this is done, the derivatives are derived in the online Supplemental Derivation 1 file.

COMPARISON OF THE FILTER-DEPENDENT DISTORTION/NOISE TRADE

To compare smoothing performance, quadratic least-squares, gaussian, and windowed sinc filters were so constructed such that each created a nominal 10% peak amplitude distortion when \( t_p = 7 \). The numeric filter values and their variances are given in Table 1. The least-squares, gaussian and windowed sinc filters all had comparable noise reduction (\( \sigma^2 = 0.207, 0.196, 0.187 \)), with the windowed sinc having a slight edge over the others. Fig. 3 shows the peak and filter spectra, where distortion is determined by multiplication of the signal and filter amplitudes at each frequency. This answers the second part of question 3.

As mentioned earlier, ripple distortion is a real concern when the peak of interest appears next to a very intense interference. This situation is shown in Fig. 4 for a noise-free peak with amplitude of a million counts smoothed with the filters shown in Table 1. The largest ripple is produced by the least-squares filter with the windowed sinc ripple about a factor of 5.6 less. The gaussian filter shows no negative excursions because none of the filter coefficients are negative. This answers question 5.

Examination of Fig. 2 shows that a peak described by 5 points (\( f_p = 2 \)) cannot have noise reduced by smoothing without a concomitant distortion. Additionally, if smoothing is not required but derivative filters are used to locate peaks they will still distort the peak start and end. Because of the fundamental trade between noise reduction and peak distortion, the ideal peak is described by 15–31 points. Within this range the user has flexibility in determining the distortion-noise reduction trade, adjusting it to suit the needs of a particular assay. Peaks having more than 31 points run into practical problems
such as inordinate processing times and small second derivatives. This answers the second part of question 4.

Automatic Filter Selection

A desirable feature of automatic processing of chromatographic data is the existence of an algorithm that by itself can select an appropriate digital filter “strength.” With the gaussian variety that has been used above, this means selecting an optimum value of $t_f$. Two approaches can be recommended: an iterative estimation based on the largest negative chromatogram second derivative; and an iterative method based on the Durbin–Watson parameter.

The second derivative approach assumes the gaussian peak shape given by $U^2$ in Eq. 6,

$$
\frac{d^2y}{dt^2} \bigg|_{\text{min}} = \frac{2\pi A}{t_p^2}
$$

where the most negative value occurs at the center of a peak ($t = 0$), and $|\text{min}|$ is the absolute value obtained by the second derivative filter. The algorithm requires an expected range for $t_f$, e.g., 2–31, and is initialized with the low value. Because gaussian filters were used, the next estimate of $t_f$ was empirically set to $t_{est}/4$ to control the amount of distortion. This estimate is then used to re-smooth the chromatogram and produce a new value for $t_{est}$. The procedure is repeated until (a) the estimate stabilizes; (b) the value pegs at either extreme; or, (c) estimates form a cycle where an average can be used. The chromatogram shown in Fig. 5 was processed starting with $t_f = 2$, resulting in subsequent values of 2.51, 2.75, and 2.79, stabilizing at 2.80. If the starting value was instead 4, the subsequent values were 3.02 and 2.85, again stabilizing at 2.80. Because the peaks in Fig. 5 have $t_p = 10$, the expected smoothed width is $(10^2 + 2.8^2)^{1/2} = 10.4$ for a 4% distortion. To allow more distortion divide $t_{est}$ by a smaller number, e.g., 3, which results in a stabilized value $t_f = 4.03$ and an 8% distortion. The allowable amount of distortion can be decided during method development.

Vivo-Truyols et al. (21) suggest using the Durbin-Watson parameter. This test is often performed with regression data to determine whether correlations exist in the residuals (22). For the present application the parameter is computed using,

$$
DW = \frac{\sum_{i=2}^{n} [(y_i - s_i) - (y_{i-1} - s_{i-1})]^2}{\sum_{i=1}^{n} (y_i - s_i)^2}
$$

where $y$ represents the original data and $s$ represents the smoothed trace. Although the actual statistical test involves upper and lower bounds, it is suggested that filter strength be increased from some minimum until the value of $DW$ is lower than 2. To this end the test was
applied to the chromatogram in Fig. 5. Starting with a value of \( t_f = 2 \) and increasing in steps of 0.1–4.0, a value of 2.9 gave \( DW = 2.06 \) and 3.0 gave \( DW = 1.97 \). For \( t_f = 3.0 \) the extent of distortion will be \( (10^2 + 3^2)^{1/2} \approx 10.4 \) or approximately 4%, which is visually indistinguishable from the value of \( t_f = 2.8 \) estimated using the second derivative approach. This answers question 6.

Both methods require iterative smoothing, which can be time-consuming. Windowing the trace to a region near the expected retention time can certainly decrease computation time. However, the window size needs to be significantly larger than the filter for all the mathematical components to work correctly. Also, for an analytical method that is under excellent control a fixed filter strength can be used after determining its value during method development.

**Filter Variance and Derivative Confidence Limits**

The last pieces of information required before using derivatives to locate peaks are estimates of their SDs. This translates into a need to know the SD of the chromatographic noise, \( \sigma_n \). To this end chromatography software often suggests that the user locate a flat stretch of baseline over which the noise can be estimated. If there is no such set of data, e.g., Fig. 5, a direct computation becomes problematic. Alternatively, if the chromatogram is first smoothed with a low distortion digital filter it is possible to obtain an excellent estimate of the noise on the raw trace. First compute the variance of the deviations between the raw and smoothed data, \( \sigma_{dev}^2 \). The variance of the raw chromatographic data is then given by,

\[
\sigma_{dev}^2 = \frac{1}{D - 2M} \sum_{i} (y_i - \bar{y})^2 \quad \sigma_i^2 = \sigma_{dev}^2 / (1 - 2F^0_i) + \sigma_{dev}^2
\]

where \( F^0 \) is the smoothing filter and the subscript, \( i \), denotes the center filter value (0.278 for \( G^0 \) in Table 1). The derivation of Eq. 10 is given in online Supplemental Derivation 2 file. If the “weakest” filter available still has a small amount of distortion, the variance of the deviations is better estimated by the median absolute deviation about the median (MAD),

\[
\text{MAD}(dev) = \text{median}(|dev - \text{median}(dev)|)
\]

\[
\sigma_{dev}^2 \equiv [1.48 \times \text{MAD}(dev)]^2
\]

where the numeric constant, 1.48, maps the MAD into an SD when errors are normally distributed. The MAD can tolerate a large percentage of outliers and still return an excellent estimate of the SD.

Once \( \sigma_i^2 \) is known it can be used to compute the variances of the smoothed trace and the 2 derivatives (see Eq. 3),

\[
\sigma_i^2 = \sigma_{pi}^2 \sigma_j^2 \quad \sigma_{d1}^2 = \sigma_{pi}^2 \sigma_j^2 \quad \sigma_{d2}^2 = \sigma_{pi}^2 \sigma_j^2
\]

where the variances for \( F^0 \), etc., are the values associated with the filters used to locate peaks (not necessarily the filter used to compute the deviations). Once the derivative SDs have been obtained, the limit (thresholds) can be set as simple multipliers, e.g., \( \times 2 \) for approximately 95% limits.

An alternative method for determining the raw trace noise has been suggested by Wentzell and Tarasuk (23). Their approach uses a high-pass filter based on an inverted Blackman windowed sinc function (similar in concept to Eq. 7). Their specific interest was the characterization of heteroscedastic noise with an example from LC-MS.

The raw chromatographic trace associated with Fig. 5 was smoothed with a gaussian filter having \( t_f = 2 \). The resulting SD of the trace computed via Eqs. 10 and 11 was 22.5 (known value 19.1). To identify peaks the trace was reprocessed with filters having \( t_f = 3 \), resulting in \( \sigma_1 = 10.9, \sigma_{d1} = 6.5, \) and \( \sigma_{d2} = 6.7 \). The derivative values were used to establish \( \pm 2 \sigma \) limits about zero, resulting in \( [R]_1 = 74–84, [A]_1 = 83–88, [F]_1 = 86–90; [R]_2 = 92–99, [A]_2 = 98–103, [F]_2 = 101–107; \) and \( [R]_3 = 109–114, [A]_3 = 113–118, [F]_3 = 116–126 \). The start and end of the 3 peaks in Fig. 5 are denoted by the open circles.

**Automatic Location of Peak Boundaries and Identification of Clusters**

If peak areas are determined by numeric integration, e.g., Simpson’s Rule (24), an excellent estimate of the peak start and end are necessary. The algorithm supporting this task is simplified when a first derivative digital filter has processed the chromatogram. For an isolated peak the algorithm draws a temporary baseline between the first point in [R] and the last point in [F]. If any interior point falls beneath this line, move the start or end to this new position. Then move the starting point to lower indices as long as the slope between the current and trial points is greater than some threshold (usually determined as a fraction of the largest slope in [R]) and no interior point falls beneath the new baseline. Repeat the process in an analogous manner with the ending point. In Fig. 5 this adjusts the first point in [R], from 74–71 and the last point in [F] from 126–128. This generally produces a baseline that gives excellent Simpson’s rule areas. If the peak of interest has an interference with significant tailing, then the derivative threshold for [F] can be set sufficiently high that the last point doesn’t extend the peak along the tail.
If a global baseline is not available, it is critical that all peaks belonging to a cluster are identified as such and that the local cluster baseline be estimated. A cluster exists when the last point of a preceding peak is adjacent to the first point of a trailing peak. When this condition is satisfied the preceding ending and trailing starting points are set to whichever point has the minimum amplitude. As an example, [F]1 and [R]2 above are both set to 91, and [F]2 and [R]3 are set to 108. The trial baseline is drawn between the first point of [R]1 and the last point of [F]3. As long as the amplitudes at intermediate peak boundaries are above the trial line it can be treated as the baseline. If any intermediate amplitude is below the trial line, it is broken into segments that satisfy the condition. For example, 2 peaks in a cluster might have a shared baseline while a third peak has its own baseline. If the peaks are not fit to a functional form, Simpson’s rule would have to use a vertical drop from the shared points to the baseline. This obviously does not give the correct area, with the error depending upon the extent of overlap. This answers question 7.

If the smoothed data associated with a peak will be fit to a shape function and area determined by the function integral, the starting and ending values are usually not critical. Similarly, if a global baseline has been determined, e.g., the dashed line in Fig. 5, there is little value in locating the local cluster baseline. Both of these conditions together save an enormous amount of algorithm development.

Alternative methods that use derivatives to identify peak boundaries have been described by Grushka and Atamna (25), and Stevenson, Gritti, and Guiochon (26).

### Locating the Global Baseline

An overwhelming number of baseline algorithms have been published. Several of the best have been compared by Liland, Almøy, and Mevik (27). Of these we find that the asymmetrically weighted least-squares approach described by Eilers (28) and by Zhang, Chen, and Liang (29) to work well. The performance of this approach has been reported by Stevenson, Conlan, and Barnett (30). Implementations are available in free software packages (31). Prior location of peaks is not required. Operationally a trial baseline is computed using a least-squares with uniform weights of 1. For subsequent iterations, points above the trial baseline are assumed to be associated with peaks and given small weights and those below it are given large weights. This tends to drive the baseline to the bottom of the chromatogram. To reduce the amount of wiggle in the baseline a second derivative penalty is added to the normal weighted least-squares term. A generalized form of the objective function being minimized is,

$$ O = \sum_i w_i (s_i - b_i)^2 + \lambda \sum_i (d^2 b_i / dt^2)^2 $$

where $s$ is the smoothed chromatogram, $b$ is the trial baseline, $w$ are the weights, and $\lambda$ controls the amount of the second derivative penalty. Eilers (28) suggested using a weighting scheme where $w = p < 0.5$ for a point above the trial baseline and $(1-p)$ for a point below the baseline.

Fig. 5 shows a baseline generated by the asymmetric least squares function from the R package “ptw.” We have found that for a given assay the value of $\lambda$ can be optimized during method development and left at that value when running batches of samples. Such baselines produce excellent Simpson’s rule areas and when subtracted from the chromatogram facilitate fitting peaks to shape functions. For windowed processing it is important that the baseline is fit to a substantial length of the trace just before and after the window. If this is not done the baseline tends to exhibit unwanted curvature near the window boundaries. This answers question 8.

### Fitting Peaks to Shape Functions

Most peaks found in clinical diagnostic chromatograms can be fit to a gaussian or, if skewed, an exponentially modified gaussian (EMG) function. If peak width changes with time a Parabolic-Lorentzian modified gaussian can be used (32). The first use of an EMG in chromatography appears to be that of Gladney, Dowden, and Swalen (33). A more recent discussion is by Grushka (34). ZABELL et al. discuss advantages of the EMG shape as a diagnostic for monitoring data quality (35). When used as a shape function, the gaussian is usually written,

$$ y(t) = \frac{A}{\sigma \sqrt{2\pi}} \exp \left(-\frac{0.5 (t - t_R)^2}{\sigma^2} \right) $$

where $A$ is the peak area, $t_R$ is the retention time, and $\sigma$ is the SD. Note that $\sigma = t_R/(2\pi)^{1/2}$. The EMG has a more complex form given by the convolution of a gaussian with an exponential decay,

$$ y(t) = \frac{A}{2\tau} \exp \left(\frac{\sigma^2}{2\tau R} - \frac{t - t_R}{\tau} \right) \left[ 1 - \text{erf} \left( \frac{\sigma}{\tau^{1/2}} \frac{t - t_R}{\sigma^{1/2}} \right) \right] $$

where $\tau$ is the skewness and erf is the error function [see Chapter 3 in (1)].

Both shape functions are fit to the smoothed data using a nonlinear least-squares (36). This particular fitting process is an iterative search of $\chi^2$ space fraught with practical difficulties. First, the chromatographic peak has to be described by more points than the number of function parameters—preferably $\times 3$. Thus the gaussian shape requires a minimum of 4 and preferably 9 points, whereas the EMG requires 5 and does better with 12. Second, every parameter
has to have an excellent initial estimate which is close to the final value. A suggested approach sets $t_0$ to the peak apex time, $\sigma$ to $1/4$ the peak base width, and $A$ to $(2\pi)^{1/2}\sigma$ times the apex peak height. For the EMG a reasonable estimate sets $\tau = \sigma$. And third, as noise increases $\chi^2$ space flattens, making it difficult for the iterative search to converge on the minimum, and the fit fails. If the EMG fit fails, the peak can be refit to a gaussian shape. If, in turn, the gaussian fit fails, Simpson’s rule can be used as a last resort. This answers the first part of question 9.

The cluster in Fig. 5 was baseline-corrected and processed by a nonlinear least-squares fit to a sum of 3 gaussian peaks each having the shape given by Eq. 14. All peaks were given the same SD to reduce the number of parameters from 9 to 7. Initial values were estimated by using the apex amplitudes, local peak maxima, and half the time between the start and apex of the first peak. The fit was excellent, with a residual SE of 5.96, amplitude errors <0.5%, location errors <0.1%, and a width error approximately 0.2%. Of course this is an ideal set of data, with results far better than those usually found in real high-throughput chromatograms. An excellent reference for the automatic deconvolution of overlapped chromatographic peaks is given by Vivó-Truyols et al. (37). This answers the second part of question 9.

One common error in curve fitting involves excessive oversmoothing modifying the functional form of the peak to the extent that it no longer has a gaussian or EMG shape. A second common error involves sample concentrations (particularly standards and calibrators) that are so high that amplitude is no longer proportional to concentration. With automatic processing, these need to be detected and flagged by the software (35). A simple method for catching both of these problems involves a peak quality score.

**Peak Quality Scores**

A simple quality score can be constructed from a slightly modified error of the fit using the smoothed data, $s$, and the fitted values, $\hat{s}$.

$$Q5 = 1 - \sqrt{\frac{\sum (s \hat{s})}{\text{norm}}^2}$$

where $\text{norm}$ is the maximum of the combined $s$ and $\hat{s}$ values. With this definition $Q5$ has values from 0 to 1. If a greater resolution is desired for the quality score, use the square of $Q5$, e.g., 0.9 and 0.8 are spread out to 0.81 and 0.64. For a baseline-resolved peak the quality score decreases with increased noise amplitude. When an interference overlaps the analyte peak the quality score depends upon the amplitude of the interference, whereas at high noise levels both contribute to the reduction in the score. With automatic processing a flag could be set for quality scores less than, say, 0.65 if no interference is anticipated. Conversely, if an experience indicates an interference has been seen in some samples, the flag could be set higher. If an interference is known to exist, a weighted least-squares could be used to reduce the influence of the interference. The 3 peaks in Fig. 5 had $SQ^2$ equal to 0.89, 0.95, and 0.94. For this high-SNR case the difference in quality scores is due to relative peak intensities. This answers question 10.

Two more-advanced methods have been described. Brodsky et al. use an approach based on discrepancies between replicate samples in the evaluation of metabolomics data (38). And, Lopatka, Vivó-Truyols, and Sjerp developed a peak probability model based on statistical overlap theory (39).

### References

17. Dryden PC, Quimby BD, inventors; Hewlett-Packard, as-
Automatic Processing of Chromatograms

Reviews

Clinical Chemistry 62:1 (2016) 153


