ELSEVIER

Biophysical Chemistry 108 (2004) 121–126

Biophysical Chemistry

www.elsevier.com/locate/bpc

Reverse Smoothing: a model-free data smoothing algorithm

Dennis E. Roark*

Department of Computer Science and Mathematics, University of Sioux Falls, 1101 W. 22nd Street, Sioux Falls, SD 57105, USA

Abstract

Biophysical chemistry experiments, such as sedimentation-equilibrium analyses, require computational techniques to reduce the effects of random errors of the measurement process. The existing approaches have primarily relied on assumption of polynomial models and least-squares approximation. Such models by constraining the data to remove random fluctuations may distort the data and cause loss of information. The better the removal of random errors the greater is the likely introduction of systematic errors through the constraining fit itself. An alternative technique, reverse smoothing, is suggested that makes use of a more model-free approach of exponential smoothing of the first derivative. Exponential smoothing approaches have been generally unsatisfactory because they introduce significant data lag. The approaches given here compensates for the lag defect and appears promising for the smoothing of many experimental data sequences, including the macromolecular concentration data generated by sedimentation-equilibria experiments. Test results on simulated sedimentation-equilibrium data indicate that a 4-fold reduction in error may be typical over standard analyses techniques.

Keywords: Noise reduction; Data smoothing; Approximation theory; Exponential smoothing; Least squares; Sedimentation equilibrium

1. Introduction

Analysis of physical–chemical experimental data usually involves a battle of defending the meaningful data from its burial by a myriad of error distortions and perturbations arising from factors such as imperfect experimental apparatus (e.g. flawed optics), imperfect data retrieval, and various non-idealities and approximations of the experimental model. The classic work on analyzing macromolecular solutions by the sedimentation-equilibria techniques of David Yphantis exemplify the challenges of suppressing the effects of these random and systematic errors, while minimizing the loss or distortion of the underlying meaningful data [1]. As a graduate student with David Yphantis, I was privileged to work with him in developing techniques to unmask the valid information contained within sets of error disturbed data [2,3]. Many of these techniques involve graphical presentations of molecular weight-averages across the centrifuge cell. Without adequate means of recovering these averages from the random errors of the typical experiment, these analytical techniques cannot be used to study mixtures of macromolecular species. The BIOSPIN computer programming project was our attempt to remove the effects of experimental noise and to provide several types of molecular weight-averages as a function of radius

*Tel.: +1-605-331-6757.
E-mail address: dennis.roark@usiouxfalls.edu (D.E. Roark).

0301-4622/04/$ - see front matter © 2003 Elsevier B.V. All rights reserved.
doi:10.1016/j.bpc.2003.10.014

at sedimentation equilibrium. It achieved sufficient success that the program continues in use today in some laboratories. This problem of information extraction from noise-filled data by employing 'data smoothing' remained a primary concern of my research career and was extended for sedimentation-equilibrium analyses [4,5]. Even as I have moved away from the field of physical biochemistry, I remain deeply grateful for the formative influences of David Yphantis and have continued the interest he sparked in computational means of reducing the effects of noise in a series of data measurements. For the scientific habits of thought he gave to his students, this proposal of an alternative smoothing approach is dedicated to David Yphantis.

The error reduction problem under consideration is that for a sequence of related data values, such as the interference fringe position (or optical density) as a function of radial position in a sedimentation-equilibrium experiment or a time series of measurements of some observable or indeed any set of data points that could be graphed in a 2D space, that is expected to follow some pattern that may not be known beforehand, and in which the error can be assumed to be primarily in one of the dimensions. The errors suppressed by a regression or smoothing technique are assumed to be random rather than systematic in nature. Removal of systematic errors often requires improvements in instrumentation or other aspects of experimental design, or the assumption of some model that the data should follow. Systematic deviations from the model can then be ascribed to systematic sources of error.

2. Standard polynomial approximation techniques

The regression or smoothing problem is so critical to the interpretation of experimental data that it has been treated in a number of ways. The majority of these error reduction techniques involve the assumption that the data, at least locally, can be fit by some model. These models are often linear or higher order polynomials and the fit criterion is often the familiar one of least-squares. In previous analyses of sedimentation-equilibria experiments, we and others have employed the technique of 'sliding' or local least-squares fit using polynomials of order two or three [2,4,5]. Through the use of local fits (more generally, piecewise polynomial approximation) with the data point of interest near the center of the region to be fit, the distortions imposed by the model can be minimized, though not eliminated. The set of points in the fits can then 'slide along' the entire data set. The more points used in the fit, the better the removal of random errors, but the greater the introduction of a systematic error by the fit itself. Such fits are always a compromise between noise reduction and error introduction, unless the true underlying data closely approximate the type of function (cubic polynomial in most cases) used for the regression.

Because of the compromises of polynomial regression, where a balance is attempted between noise suppression and data distortion by the fit itself, a number of more sophisticated techniques have been proposed [6]. These include using more complex modeling functions (e.g. Lagrange, Hermite or Chebyshev polynomials) or cubic Spline fits. A Spline fit determines a set of cubic functions connecting data sub-regions. The values of these functions and their first two derivatives are matched at the endpoints where one polynomial must intersect with adjacent polynomials. This technique is normally used to guarantee that a smooth curve can exactly pass through a series of points, not a desirable outcome in a data smoothing algorithm where the points themselves are expected to be influenced by random fluctuations. But with modification, the Spline fits can be employed for noise reduction of data. (For instance, the Spline fit can be required to pass through the midpoint of adjacent data points and the fit iterated.) The mathematics of information theory, stochastic processes and the disturbance of a signal in time by white-noise type perturbations have suggested filtering techniques to separate the information from the noise [7]. However, the random perturbations of biophysical experimentation (such as sedimentation-equilibrium optical measurements) seems unlikely to be finely enough grained to be treated as white noise or to benefit from these sophisticated filtering techniques.

3. Problems associated with moving average techniques

An alternative approach to noise removal from a sequence of data values, one that does not attempt a fit by piecewise polynomial (or a more complex functional) approximation, appears at first consideration crude in its simplicity and almost guarantees the introduction of substantial data distortion. This is the technique of 'trailing moving averages' and is frequently employed to analyze trends in time-series data where the concern is primarily to determine the current trend direction and whether that direction is likely to continue in the near future (times slightly greater than those in the data sequence). *N*-point (perhaps 10 or 20 point) trailing moving averages suppress random fluctuations. The more a particular point lies above the moving average, the greater the likelihood that the trend is to increasing values, and if below to decreasing values. However, the lags imposed by moving averages, where the past counts more than the present, makes them virtually useless to test the smoothed data against a physical model. A moving average that utilizes a fixed number of data points that are equally weighted can exhibit abrupt discontinuities in the slope of the smoothed data. If *N* previous points are averaged to determine the smoothed value at some point, and if this subset of *N* points is moved along the data, a particularly erroneous point as it enters and then *N* averages later exits the subset can cause sudden jumps in the slope.

An improvement over this 'simple moving average' approach is the use of 'exponential moving averages.' This type of moving average does not suffer from the abrupt slope discontinuities as an erroneous point exits a subset used for the average, but it continues to suffer from distorting lags similar to those of simple moving average smoothing. The exponential moving average uses a recursion formula to smooth data point *i* of the data set $y[i]$:

$$y[i] = w \cdot y[i] + (1-w)y[i-1]$$

where w is some weighting factor $(0 < w < 1)$. The lower the value of w, the greater the smoothing and the greater the lag distortion produced by the smoothing. The recursion begins at data point 2 and proceeds through the remainder of the data set. The technique is termed exponential smoothing because the influence of a point $j < i$ decreases in an approximately exponential fashion, $\exp\{j-i\}$. This smooth decline in the influence of previous points is in contrast to the simple moving average approach that employs equal weights for the *N* points previous to data point *i* and no weight for points more than *N* distant from point *i*. Although reasonably model free, the exponential moving averages assumes that there is a strong correlation between behavior of point *i* and points just previous to it and that this correlation decreases smoothly the greater the distance to point *i*. Were it not for the serious lag distortion of this type of data smoothing, it might have much to offer the experimentalist attempting to suppress the random noise of a data set.

4. A multi-pass approach: reverse smoothing

The algorithm proposed here, 'reverse smoothing' for lack of a better term, significantly reduces or eliminates the lag effect while maintaining the noise reduction benefit of exponential smoothing. It is 'model free' in the sense that there is no polynomial or other function to constrain or distort the data. It appears appropriate for situations where the data sequence is expected after noise removal to exhibit a 'smooth behavior' in the mathematical sense. The data and its first derivative are assumed to be describable by continuous functions, even though these functions may not be known. As such, the technique seems particularly suited for the random error reduction of the sort of data expected from sedimentation-equilibria experiments. The algorithm performs a very noisy sliding three-point linear least-square fit to obtain an estimate of the slope of the data at each point *i*. The three-point fits to find the slope at the middle point must have sufficient data density so that the linear fit produces minimal distortion of any underlying data patterns. In other words, there must be sufficient data points to locally characterize the slope. This is an important restriction. In regions where the slope changes rapidly, higher data den-

sity will be needed. With the technique presented here, a minimum of eight points in the vicinity of some local slope feature is usually adequate. Within the sub-region of these, at least, eight points, the slope should appear smooth and slowly varying. For high-speed sedimentation-equilibrium experiments, a minimum of 60 points across the cell is appropriate. Some smoothing occurs in these small local fits, but it is not expected that the set of slopes, the numerical derivatives at each point i, will be at all smooth. For the simple case of equally spaced data, the slopes at the midpoint is half the distance between the y-values of the outer points. For the general case, the three-point least squares is somewhat more complicated. The slopes at the first and last points are extrapolated from nearby points. Three passes of exponential smoothing are performed on the set of slopes. Exponential smoothing is not performed on the original data, but on the estimated first derivatives of that data.

The first pass is a standard forward pass from point 2 to the end of the data set. The weighting factor is $1/2$. The second pass is a reverse pass, from the next to last point to the first point using a weighting factor of $1/3$. If there are n data points and the $y[i]$ values are now the slopes of the original data set, the second pass uses the recursion relation:

$$y[i] = w \cdot y[i] + (1-w)y[i+1]$$

and begins at point $n-1$, proceeding in reverse order through point 1. Following the reverse pass that uses a more severe weighting factor than the previous forward pass, a third exponential smoothing pass in the forward direction with weighting factor $1/2$ is performed from point 2 through point n. The smoothed derivatives are then integrated by a semi-trapezoidal numerical integration to find the smoothed data values to within an additive constant. The average value of the original data set and the average value of the smooth set are used to determine the appropriate integration constant and adjust each of the data points.

The algorithm replaces the original $y[i]$ values with their smoothed counterparts. For sedimentation-equilibrium data, the $y[i]$ are the natural log of the concentration data as determined by interference fringe displacement or optical absorption. The abbreviated C++ outline of the algorithm for the general case (data not necessarily equally spaced in x) is

```
// Reverse Smoothing Algorithm
void RevSmooth(double* x, double* y, int n)
{
        double* S = new double [n+1];
        double b = 0;
        for (i=1; i<=n; i++)    {       // store the original array and find the average
                S[i] = y[i];
                b += y[i];
        }
        b /= n;
        for (i=2; i<n; i++) {           // estimate the derivatives
                y[i] = 3*(x[i-1]*S[i-1] + x[i]*S[i] + x[i+1]*S[i+1])
                        - (x[i-1] + x[i] + x[i+1]) * (S[i-1] + S[i] + S[i+1]);
                double D = 3*(x[i-1]*x[i-1] + x[i]*x[i] + x[i+1]*x[i+1])
                  - (x[i-1] + x[i] + x[i+1]) * (x[i-1] + x[i] + x[i+1]);
                y[i] /= D;
        }                               // estimate end point derivatives
        y[1] = y[2] + (y[3]-y[5])*(x[2]-x[1]) / (x[5]-x[3]);
        y[n] = y[n-1] + (y[n-2]-y[n-4])*(x[n]-x[n-1]) / (x[n-2]-x[n-4]);

        for (i=2; i<=n; i++)            // forward pass, exponential smoothing
                y[i] = (y[i] + y[i-1]) / 2;
        for (i=n-1; i>=1; i--)          // reverse pass exponential smoothing
                y[i] = (y[i] + 2*y[i+1]) / 3;
        for (i=2; i<=n; i++)            // forward pass
                y[i] = (y[i] + y[i-1]) / 2;

        double c = S[1] = y[1];// integrate the derivatives
        for (i=2; i<=n; i++)    {
                c += (2*y[i]+y[i-1]) * (x[i]-x[i-1]) / 3;
                S[i] = c;
        }
        double d = 0;                   // find integration constant
        for (i=1; i<=n; i++) {
                y[i] = S[i];
                d += y[i];
        }
        d /= n;
        for (i=1; i<=n; i++)
                y[i] += (b - d);
        delete [ ] S;                   // cleanup
}
```

The weighting factors of $1/2$ for both forward exponential smoothing steps of the derivatives, the weighting factor of the intervening reverse exponential smoothing of $1/3$, and the combination of left and right derivatives in the integration step have been empirically chosen to closely eliminate any perceived lag for a universe of data set examples. Particularly helpful in this tuning process was the use of random noise perturbed symmetric parabolic data of sufficiently high density. In the conventional moving average or exponential moving average approaches, the smoothed parabolas are displaced to lower x values and become asymmetric. The Reverse Smoothing algorithm

Table 1
Reverse Smoothing test with simulated, random noise perturbed data

Position (cm)	Concentration (arbitrary units)	Sigma W (cm^{-2})	Sigma W standard deviation (cm^{-2}) for random perturbations in concentration (standard deviation for unsmoothed data in parentheses)		
			±0.01	±0.02	±0.04
6.6	0.856	1.72	0.02 (0.19)	0.07 (0.4)	0.17 (0.61)
6.7	3.865	3.43	0.01 (0.11)	0.02 (0.21)	0.05 (0.36)
6.8	100	6.02	0.01 (0.02)	0.01 (0.03)	0.01 (0.03)

Comparison of smoothed and unsmoothed results. See text for details.

presented here maintains the symmetry of both sides of the parabola and keeps the vertex location properly positioned in x. The leftward displacements or lags of the forward exponential steps are countered by the rightward displacement of the reverse step. The smoothing procedure acting on the first derivatives can more effectively remove random fluctuations with minimum distortion to the underlying data than would occur if the smoothing were performed on the undifferentiated data. In other words, to achieve the same degree of smoothing, more severe weighting factors would have to be used with the undifferentiated data, leading to greater loss of information.

5. Testing with simulated sedimentation-equilibria data

Table 1 presents the results of a simulated high-speed sedimentation-equilibrium experiment that is perturbed with varying amounts of random noise. The simulated experiment assumes that the cell meniscus is at 6.5 cm and cell base at 6.8 cm. Two molecular weight species are present with sigmas of 1.6 and 6.4 to give a substantial spread of weight-average molecular weights across the cell. The concentrations for the species are adjusted so that the weight-average sigmas are 1.61 and 6.02 at the meniscus and cell base, respectively. The concentration is normalized in arbitrary units for a base concentration of 100. Although some investigators use more data points that was common in the era of BIOSPIN, the simulation uses 80 points, typical of the classic high-speed experiment. (Additional points allow this or other smoothing techniques to better suppress random errors. Automated interference fringe measurements may permit several hundred points.) As is often the practice without high data density, the measurement interval is decreased as the base of the cell is approached. For the first 1/4th of the cell, the point density in the simulation is 55 μm, for the middle half of the cell, 35 μm, and for the 1/4th of the cell nearest the base, the measurement density is 25 μm.

The 80 points of simulated concentration data were generated and perturbed with random errors of three standard deviations prior to calculation of the ln(concentration) data. Finally, the ln(concentration) vs. radius^2/2 simulated data were analyzed using the Reverse Smoothing algorithm. Table 1 presents the results at three radii, 6.6, 6.7 and 6.8 cm, where the weight-average sigmas were 1.72, 3.43 and 6.02, respectively. The standard deviation from the correct value is presented for three different amounts of random perturbation. (At least five distinct sets of normalized random numbers were used for each reported deviation. Values of weight-average sigma were calculated by sliding local least-squares fits according to the classic manner [1–4].) Standard deviation of the weight-average sigmas is reported in parentheses with Reverse Smoothing disabled for comparison. As expected, the extent of error introduced by random perturbations is most significant at the lowest radius analyzed, approximately 1/3rd the distance from meniscus to base, and decreases at higher radii. At the two lower radii, Reverse Smoothing of the noise perturbed data reduces the standard deviations of the weight-average sigma between 4- and 10-fold.

Although unconventional but arguably appropriate for a modern presentation of a data analysis algorithm such as this, additional examples of its use are available to the reader through the author's web site from which the reader may download a zip file: www.usiouxfalls.edu/~droark/reverse.html.

One file contained in the zip is an Excel spreadsheet with a VBA version of Reverse Smooth for equally spaced data. The reader may create their own data examples, generate the smoothed values and observe the resulting chart. The chart on the web page is an example of this comparison of raw and smoothed data for a somewhat noisy set similar to that which might be generated in a sedimentation-equilibrium experiment. The other file in the zip permits demonstration for the general case and uses the C++ version described here. Up to 200 (x, y) values may be entered and the original and smoothed set displayed graphically. (The x values need not be equally spaced but should be sequential. The programs and algorithm may be freely used with attribution and are placed in the public domain.)

Future plans are to revise the original BIOSPIN program I authored while with David Yphantis for analyzing high-speed sedimentation-equilibrium experiments; to incorporate into a Windows version of the program the Reverse Smoothing algorithm described here; and to analyze recent experiments (Roark et al., in preparation).

References

[1] D.A. Yphantis, Biochemistry 3 (1964) 297.
[2] D.E. Roark, D.A. Yphantis, Ann. NY Acad. Sci. 164 (Art. 1) (1969) 156.
[3] A.T. Ansevin, D.E. Roark, D.A. Yphantis, Anal. Biochem. 34 (1970) 237.
[4] R.M. Carlisle, J.I.H. Patterson, D.E. Roark, Anal. Biochem. 61 (1974) 248.
[5] D.E. Roark, Biophys. Chem. 5 (1976) 185.
[6] R. Burden, J. Faires, Numerical Analysis, fifth ed., PWS-Kent Publishing, Boston, 1993, Chapters 3 and 8.
[7] L. Arnold, Stochastic Differential Equations, Wiley, New York, 1974, Chapters 3 and 12.