ESTIMATING FUNCTIONS WITH WAVELETS: USING A DAUBECHIES WAVELET IN NONPARAMETRIC REGRESSION

by

M. E. Bock and G. J. Pliego
Purdue University Purdue University

Technical Report #95-46

Department of Statistics
Purdue University

December 1995

Estimating Functions with Wavelets: Using a Daubechies Wavelet in Nonparametric Regression

M. E. Bock and G. J. Pliego
Purdue University, U.S.A.

1 Introduction

A nonparametric regression problem in statistics provides a natural application for wavelet analysis, being wavelets families of functions useful in function approximation. We sketch a description of a nonparametric "smoother" based on the Daubechies wavelet family D_4 [Daubechies, 1992]. The D_4 family is used since it is simple to describe and illustrative of other orthonormal wavelet families. The fast algorithms for D_4 are well suited to large data sets and software for them is currently available. Furthermore, the "smoothing" process given here and implied by the wavelet analysis can be simply described with minimum reference to the underlying wavelet family.

We want to model the random variable Y as a function of the variable x. For each x_k we observe $Y = y_k$ for $k = 0, \cdots N$. We assume $x_{k+1} = x_k + 1$ throughout. The goal is to use the points $\{(x_k, y_k)\}$ to generate a "smooth" function of the variable x useful for predicting values of the variable Y. It is convenient to assume that there is an unknown function f such that

$$y_k = f(x_k) + \epsilon_k,$$

where the $\{\epsilon_k\}$ are uncorrelated random variables each with mean zero and common variance σ^2. Usually some weak restrictions are placed on the behavior of f but we will ignore these for the moment. We describe an estimator of f as a prediction function for Y.

An initial prediction function can be produced by choosing a smooth function and interpolating the data points with it. However, such a prediction function may be unnecessarily bumpy due to the noise ϵ_k in y_k. One reasonable approach is to replace the y values in the original data set by a smoother set before interpolating the points. We will describe a simple method for the smoothing that is implied by the D_4 wavelet analysis . The resulting estimator will be seen to be a nonparametric "window" smoother whose window width varies with x. We illustrate the usefulness of a variable window width in the following example.

For the data in Figure 1, the apparent underlying function f seems to have bumps along with a linear trend. Many nonparametric regression estimators either oversmooth the bumps or undersmooth the constant portion because they are fixed-width "window" smoothers. A "window" smoother replaces the y value of a data point by a function (usually a weighted linear combination) of the y values of the data point and its neighbor points in a surrounding "window". Naturally, the window width controls which neighbor points are allowed to influence the smoothing for the point. A small window width allows one to trace an abrupt swing of the apparent underlying curve f but some points in Figure 1 are bettered smoothed with larger window widths. In fact, a method based on wavelet analysis can be used to smooth data sets like this in a manner that automatically prescribes small window widths for the points near bumps and large window widths for other points where the apparent underlying f values seem to be smoother.

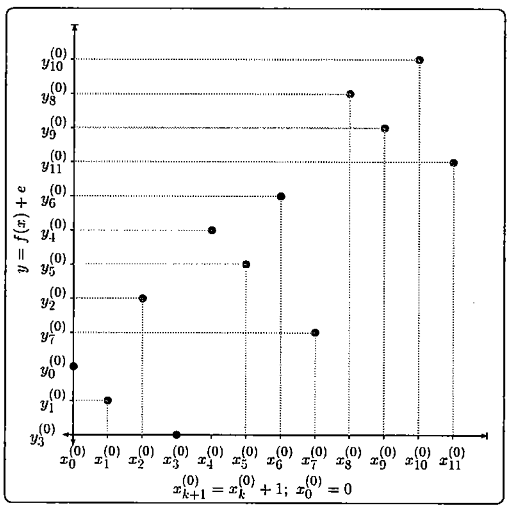

Figure 1: Original Noisy Data to be fit.

2 An Overview of the D_4 Data Smoothing

This section gives an overview of the smoothing process that replaces the y values in the original set of points by "smoother" ones. These adjusted points can then be interpolated to generate a prediction function for Y. Boundary considerations are ignored for ease of discussion throughtout the article.

The smoothing process is divided into a "decomposition" phase followed by a "reconstruction" phase. (These refer to the same phases in wavelet multiresolution analysis.) Although we do not present them, fast algorithms are available for all these steps.

In the decomposition phase, one creates a sequence of sets of "prediction" points and a sequence of sets of "discrepancies". (In wavelet multiresolution analysis, the y values of the prediction points are unnormalized scaling function coefficients and the "discrepancies" are unnormalized wavelet coefficients multiplied by the value $(\sqrt{3}-2)$.) We sketch the method for creating these sets at each stage of the sequence in the following paragraph.

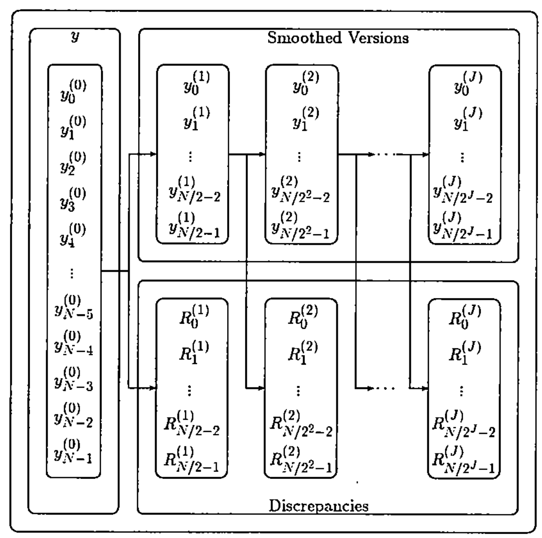

Figure 2: Decomposition Process.

At stage zero of the decomposition phase, the stage zero prediction function is formed by straight-line interpolation, i.e. by drawing straight lines to connect neighbor points. The original data set is the stage zero set of prediction points. At each succeeding stage j of the decomposition phase, the prediction points from the previous stage $(j-1)$ are replaced after using them to create a smaller set of smoother points, the stage j prediction points. A new stage j prediction function is then formed by straight-line interpolation of the stage j prediction points. The "discrepancy" for the stage j prediction point is defined to be the difference along the y axis between the prediction function for stage $(j-1)$ and the given stage j prediction point. These form the set of stage j "discrepancies". The successive stages of the decomposition phase are illustrated in Figure 2.

¿From the decomposition phase one retains the entire sequence of sets of discrepancies but only the last set of predictions points in the sequence. These sets contain all the information in the original data set. In fact, there is enough information there to reverse the decomposition process and recreate the original data set, making it unnecessary to retain the original data or the entire sequence of sets of prediction points.

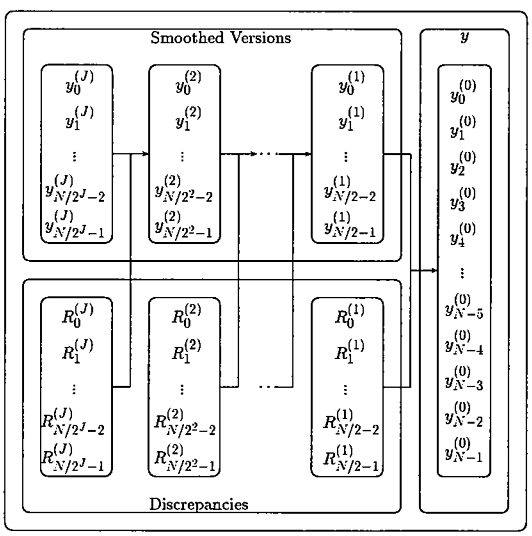

Figure 3: Reconstruction Process.

The second phase is called the reconstruction phase and is, in some sense, a reversal of the previous process. It is based on the fact that at any stage of the decomposition phase the current set of prediction points and its companion set of "discrepancies" can be used to reconstruct the set of prediction points for the previous stage. This corresponds to reversing the direction of the arrows in Figure 2, as in Figure 3.

Exact reconstruction would be pointless but a modification of the reconstruction process leads to improved estimators. It depends on the idea that in the decomposition phase the stage j "discrepancies" contain whatever was smoothed out from the stage $(j-1)$ prediction points in computing the stage j prediction points . Some noise is swept out into these "discrepancies". But they also contain parts of sharp local peaks or bumps in the underlying function f that were rounded off by the smoothing process. The important modification of the reconstruction process is the following: The "discrepancies" are adjusted in a way that saves the bumps but throws out the noise. Then these adjusted "discrepancies" are used instead of the original "discrepancies" in the formulas for the reconstruction process. Starting from the last stage of the decomposition phase, the use of the modified reconstruction process produces "improved" prediction points for the previous stage. Repeated applications create a "reverse-order" sequence of sets of "improved" prediction points. The last set in the "reverse-order" sequence is a new version of the original data set. Using a smooth interpolating function with this new version produces a powerful smoother with variable window width.

Section 3 gives a simple geometric description of the local smoothing method that produces the new set of prediction points at each stage of the decomposition phase.

Section 4 gives a corresponding geometric interpretation for the new set of "discrepancies" generated at each stage of the decomposition phase.

Section 5 describes the reconstruction process. Enough information is given to suggest reasonable criteria for modifying the "discrepancies".

Section 6 ties the prediction points and corresponding "discrepancies" of the decomposition process to wavelet multiresolution analysis. A concluding section summarizes the results.

3 Geometric Interpretation of Smoothing in the Decomposition Phase

At each stage of the decomposition phase, a new set of smoothed prediction points is generated from the set for the previous stage. In this section we describe the set and give a geometric interpretation of the smoothing procedure that generates the new prediction points.

First we label the k^{th} prediction point in the stage j set as $(x_k^{(j)}, y_k^{(j)})$. We say that the original data points form the stage zero set of prediction points. The original data point (x_k, y_k) is now labeled $(x_k^{(0)}, y_k^{(0)})$, refer to Figure 1.

The local smoothing procedure produces the stage j set of smoothed prediction points from the the stage $(j-1)$ set.

Proposition 1 *Let $(x_k^{(j)}, y_k^{(j)})$ be the k^{th} prediction point for stage j, generated from the four prediction points*

$$(x_{2k+i}^{(j-1)}, y_{2k+i}^{(j-1)}), i = 0, \ldots, 3$$

in the stage $(j-1)$ set, shown in Figure 4.

Let $x_k^{(j)}$, the x value of the new point or location for $y_k^{(j)}$ the smoothed value, be defined

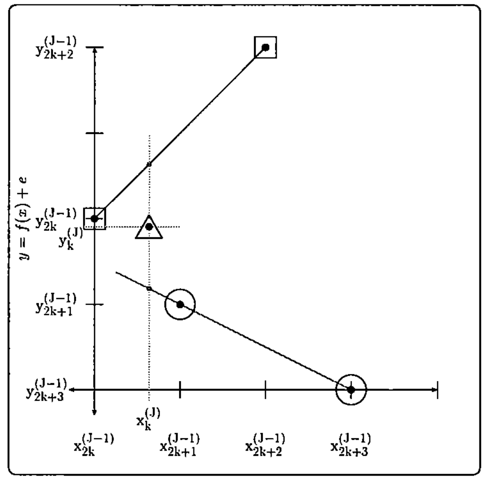

Figure 4: D4-Wavelet Geometric Approach.

as:

$$x_k^{(j)} = x_{2k}^{(j-1)} + 2^{j-1}\mu$$

where

$$\mu = (3 - \sqrt{3})/2,$$

which is about 0.63.

This definition for the x value of the new smoothed point has the property that if the four stage $(j-1)$ prediction points all lie on the same quadratic curve, then the new smoothed point also lies on that curve.

Let $m(e)_k^j$ be the slope, and $b(e)_k^j$ be the intercept, of the line fit through the two even points $(x_{2k}^{(j-1)}, y_{2k}^{(j-1)})$ and $(x_{2k+2}^{(j-1)}, y_{2k+2}^{(j-1)})$ labeled by squares in Figure 4, where,

$$m(e)_k^j = \frac{(y_{2k+2}^{(j-1)} - y_{2k}^{(j-1)})}{2^j},$$

and

$$b(e)_k^j = -m(e)_k^j x_{2k}^{(j-1)} + y_{2k}^{(j-1)}$$

Let $y_k^{(j)}(e)$ be the point lying on the even fit line at $x_k^{(j)}$, the x value of the new point as shown in Figure 4, therefore:

$$y_k^{(j)}(e) = m(e)_k^j x_k^{(j)} + b(e)_k^j$$

Likewise, let $m(o)_k^j$ be the slope, and $b(o)_k^j$ be the intercept, of the line fit through the two odd points $(x_{2k+1}^{(j-1)}, y_{2k+1}^{(j-1)})$ and $(x_{2k+3}^{(j-1)}, y_{2k+3}^{(j-1)})$. labeled by circles in Figure 4, where,

$$m(o)_k^j = \frac{(y_{2k+3}^{(j-1)} - y_{2k+1}^{(j-1)})}{2^j},$$

and

$$b(o)_k^j = -m(e)_k^j x_{2k+1}^{(j-1)} + y_{2k+1}^{(j-1)}$$

Let $y_k^{(j)}(o)$ be the point lying on the odd fit line at $x_k^{(j)}$, the x value of the new point as shown in Figure 4, therefore:

$$y_k^{(j)}(o) = m(o)_k^j x_k^{(j)} + b(o)_k^j$$

Then, the y value $y_k^{(j)}$ for the new smoothed point is the simple average of $y_k^{(j)}(e)$ and $y_k^{(j)}(o)$ values from the even and odd fit lines when x equals $x_k^{(j)}$.

$$y_k^{(j)} = \frac{y_k^{(j)}(e) + y_k^{(j)}(o)}{2}$$

An algebraic formula for the y value of the k^{th}prediction point of stage j based on the y values of prediction points from stage $(j-1)$ is

$$y_k^{(j)} = \frac{1}{2}\sum_{i=0}^{3} c_i y_{2k+i}^{(j-1)},$$

where

$$\begin{aligned} c_0 &= (1+\sqrt{3})/4 \quad (1)\\ c_1 &= (3+\sqrt{3})/4 \quad (2)\\ c_2 &= (3-\sqrt{3})/4 \quad (3)\\ c_3 &= (1-\sqrt{3})/4 \quad (4)\end{aligned}$$

(These $\{c_i\}$ are tied to the definition of the wavelet family D_4.)

The new point is labeled by a triangle in Figure 4.

One implication of the geometric view smoothing procedure just described can be seen immediately. If the four stage $(j-1)$ prediction points lie on the same straight line, then the new smoothed point will also lie on that line.

Proof 1 *Follows from algebraic operations.*

Here are some statistical implications of these definitions. The y value of the new stage j prediction point is "smoother" because its variance is one half the variance for the stage $(j-1)$ prediction points. Furthermore, if the y values for stage $(j-1)$ prediction points are uncorrelated, then the y values for stage j prediction points are uncorrelated. Thus if the underlying f function is locally linear or quadratic, then the new point will be a less noisy observation on that line or quadratic curve.

The spacing of the original data points $\{(x_k, y_k)\}$ satisfied $x_{k+1} = x_k + 1$. The distance between the points on the x axis for the j^{th} set of prediction points is also constant and equal to 2^j because the x value of the k^{th} point in the stage j prediction set is

$$x_k^{(j)} = 2^j(k+\mu) + (x_0 - \mu).$$

The distance is twice the corresponding distance for the $(j-1)^{st}$ set of prediction points.

In Figure 5 the stage $(j-1)$, prediction points are labeled with dots while the stage j prediction points are labeled with triangles. The thiner vertical lines cutting the x-axis are x values for the $(j-1)^{st}$ set while the thicker

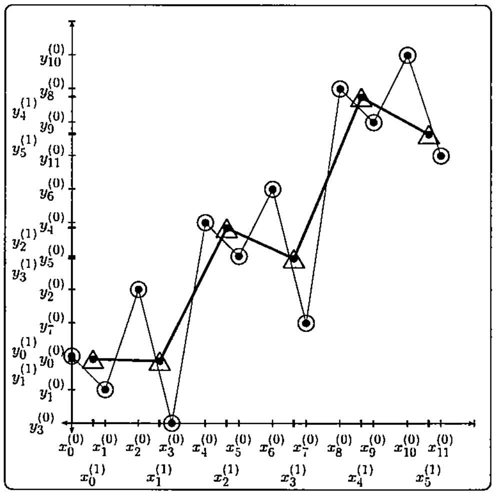

Figure 5: One-Step Smoothed Data.

vertical lines cutting the x-axis are x values for the j^{th} set. We say that the resolution of the j^{th} set of prediction points is growing coarser as j increases even though the prediction points may be getting smoother. Observe that none of the x values for the prediction set of one stage overlaps those of another stage since μ is irrational. The spacing implies that (ignoring boundary problems) there are half as many points in the jth set of prediction points as in the $(j-1)^{st}$ set. This implies an obvious limitation on the number of stages in the decomposition phase since the original set of data points is finite. The usual definitions for the number of stages are much smaller though and based on sample size considerations.

Now we describe the stage j prediction function that interpolates the stage j prediction points. The straightforward approach is to "connect the dots", i.e. draw straight lines to connect neighboring stage j prediction points as shown in Figure 5.

That is exactly the definition of the stage j prediction function. It will play an important role in the definition of the "discrepancies" in the next section. (An alternative "interpolating" function is used in wavelet analysis but it must be admitted that you won't be able to see the difference in the graphs of the two functions when the stage j prediction points have x values spaced close together.)

4 The "Discrepancies" of the Decomposition Phase

A significant feature of the decomposition phase is the computation of a measure of the difference between a stage j prediction point and the ones for the previous stage. This "discrepancy" will have an important role to play in the reconstruction phase when we examine the reversal of the smoothing process, i.e. going from stage j to the previous stage $(j-1)$. One cannot simply compare the y value for the k^{th} stage j prediction point to that of one of the stage $(j-1)$ prediction points since none of the stage $(j-1)$ points have the same x value, $x_k^{(j)}$, as the k^{th} stage j prediction point. But there is a reasonable alternative. Compare the y value for the stage j prediction point to the corresponding y value of the stage $(j-1)$ prediction function that interpolated the stage $(j-1)$ prediction points, i.e. look at the difference between the prediction function for the stage $(j-1)$ and the prediction function for the stage j when $x = x_k^{(j)}$. We define the "discrepancy" associated with $(x_k^{(j)}, y_k^{(j)})$, the k^{th} prediction point of stage j, to be

$$R_k^{(j)} = (1-\mu)y_{2k}^{(j-1)} + \mu y_{2k+1}^{(j-1)} - y_k^{(j)}.$$

The "discrepancy" $R_k^{(j)}$ has a simple geometric interpretation that is given in Figure 6. For stage j, the k^{th} prediction point $(x_k^{(j)}, y_k^{(j)})$ is labeled by a triangle in Figures 5 and 6. The two closest prediction points from stage $(j-1)$ are $(x_{2k}^{(j-1)}, y_{2k}^{(j-1)})$ and $(x_{2k+1}^{(j-1)}, y_{2k+1}^{(j-1)})$ which

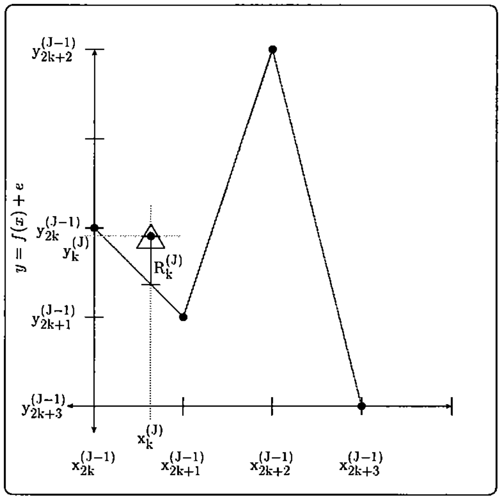

Figure 6: Discrepancies.

are labeled by circles in Figure 5. A straight line is drawn through these two stage $(j-1)$ prediction points. The "discrepancy" $R_k^{(j)}$ is found by subtracting the y value of the new stage j prediction point from the y value of the line at $x = x_k^{(j)}$. The geometric interpretation makes it clear that the "discrepancy" is the difference of the "connect the dots" prediction function for stage j and the corresponding one for stage $(j-1)$ at $x = x_k^{(j)}$ where $x_k^{(j)} = x_{2k}^{(j-1)} + 2^{j-1}\mu$. (In wavelet multiresolution analysis, the "discrepancy" is an unnormalized coefficient of the wavelet function multiplied by the constant $(\sqrt{3}-2)$.)

The definition of the "discrepancy" implies a number of interesting statistical properties. The variance of the noise in $R_k^{(j)}$ is equal to $2^{-j}(\sqrt{3}-2)^2$ times the variance of the noise in the y_k, the y values of the original data set. It can also be shown that the "discrepancies" for stage j are uncorrelated with one another and uncorrelated with the "discrepancies" for any of the other stages. The "discrepancies" for stage j are also uncorrelated with the y values of the stage j prediction set.

A number of properties are seen from the geometric interpretations of the "discrepancies" and prediction points. For instance, the "discrepancy" $R_k^{(j)}$ for $(x_k^{(j)}, y_k^{(j)})$ is zero if the four stage $(j-1)$ prediction points used to compute $y_k^{(j)}$ all lay on the same straight line. By defining the "discrepancy point" as $(x_k^{(j)}, R_k^{(j)})$, we can see in the next section how other patterns in the stage $(j-1)$ prediction points are reflected in the stage j discrepancy points.

The "discrepancies" $\{R_{k+1}^{(j)}\}$ for stage j of the decomposition phase are closely connected to the "difference" function for stage j. The "difference" function for stage j is defined to be the prediction function for stage $(j-1)$ minus the prediction function for stage j. We can see that the "difference" function based on the "connect the dots" prediction functions actually interpolates the discrepancy points. (This is essentially the case for the difference function based on the definition of the alternatives to the prediction functions given for the wavelet analysis.)

5 Reconstruction

At any stage j of the "smoothing" process, one can reverse the process to obtain the prediction points of the previous stage $(j-1)$ as shown in Figure 3. The y values of the stage $(j-1)$ prediction points can be computed from the stage j "discrepancies" and the y values for the stage j prediction points using the reconstruction formula

$$\begin{aligned} y_k^{(j-1)} &= \sum_i c_{k-2i} y_i^{(j)} \\ &\quad + (-1)^k \sum_i c_{3-k+2i} R_i^{(j)} / (\sqrt{3}-2) \end{aligned}$$

By adjusting or smoothing the "discrepancies" $\{R_k^{(j)}\}$ before employing them in the reconstruction formula above, one may systematically improve the stage $(j-1)$ prediction points. For instance, "small" $R_k^{(j)}$ may be set equal to zero under the assumption that they reflect noise while "large" $R_k^{(j)}$ may be preserved intact under the assumption that they reflect oversmoothing of bumps. Using these adjusted $R_k^{(j)}$ in the reconstruction formula above yields an "improved" set of stage $(j-1)$ prediction points superior to both the original set of stage $(j-1)$ prediction points and the original set of stage j prediction points. This is because the "improved" points are not oversmoothed yet the noise is reduced wherever possible. These "improved" stage $(j-1)$ prediction points can now be combined with similarly adjusted stage $(j-1)$ "discrepancies" in another application of the reconstruction formula to create an "improved" set of stage $(j-2)$ prediction points. Similar reasoning imples that it is superior to the original set of stage $(j-2)$ prediction points as well as the "improved" set of stage $(j-1)$ prediction points. Repeated applications of this process leads to a "reverse-order" sequence of sets of "improved" prediction points culminating in an improved version of the original stage zero set of prediction points. Because the original stage zero set of prediction points is just the original data set, we see that we have replaced it by a set better than or at least as good as any of the other sets of prediction points that we have created. We have not indicated at what stage j this whole reconstruction process starts but the considerations include the size of the data set and attendant boundary problems.

In developing criteria for adjusting or smoothing the $\{R_k^{(j)}\}$, it is important to understand the effect on these stage j "discrepancies" of underlying local relationships between the x and y values in the stage $(j-1)$ prediction points. Some "bumps" in the data can come from underlying local high degree polynomials. For example assume that for x values in a certain interval, the corresponding stage $(j-1)$ prediction points lie on a polynomial curve of degree m. Then the stage j "discrepancy" points with x values in that interval will lie on a polynomial curve of degree $(m-2)$, where a polynomial of degree less than or equal to (-1) is assumed to be the zero function. In such a case, if noise is also added to the y values of the stage $(j-1)$ prediction points, then the y values of the stage j "discrepancy" points also have added noise. If there are enough points with x values in the interval, then it is appropriate to "smooth" the $\{R_k^{(j)}\}$ to reduce noise yet retain the underlying $(m-2)$ degree polynomial curve before employing them in the reconstruction formula. The same techniques used to smooth the original data set $\{(x_k, y_k)\}$ can be applied to the "discrepancy" points. But then it may also be appropriate to smooth the "discrepancy" points of the "discrepancy" points and so on. There are actually fast algorithms to carry out such schemes. Entropy-based criteria for deciding which "discrepancies" to "smooth" before truncating or "shrinking" have been successfully employed [Daubechies, 1992, Section 10.6].

6 The D_4 Multiresolution Analysis

In this section we connect the prediction points and the "discrepancies" of the earlier sections to D_4 multiresolution analysis described in the previous article.

For an unknown function f, it is assumed (as previously) that

$$y_k = f(x_k) + \epsilon_k$$

where the $\{\epsilon_k\}$ are uncorrelated random variables with mean zero and common variance σ^2.

It is assumed that y_k is observed for known x_k with

$$x_k = k + x_0.$$

The scaling function ϕ for the wavelet family D_4 can be used to provide a sequence $\{F_{-j}\}$ of approximating functions to f. This F_{-j} is an alternative to the stage j prediction function of the previous sections and satisfies

$$F_{-j}(x) = \sum_k y_k^{(j)} \phi(2^{-j}x - k).$$

(For details see Note 1 in the Appendix.) In order to preserve the interpretation of its coefficients $y_k^{(j)}$ as function values for F_{-j}, the function $\phi(2^{-j} \cdot -k)$ is not normalized, i.e. multiplied by $2^{-j/2}$. The function F_{-j} essentially passes through the values $y_k^{(j)}$ at the x locations

$$x_k^{(j)} = 2^j(k + \mu),$$

where

$$\mu = (3 - \sqrt{3})/2.$$

This occurs because the support of ϕ is $[0, 3]$ and $\phi(\mu + i)$ is nearly zero for i equal to one or two while it is nearly one for i equal to zero. (The fact that these ϕ function values are "nearly" one or zero leads to the qualifying phrases "essentially interpolates" and the quotation marks that sometimes appear around the word "interpolating" in our discussion.)

By defining

$$x_k = k + \mu,$$

we have

$$x_k^{(j)} = 2^j(\mu + k),$$

and the function F_{-j} essentially interpolates the stage j prediction points. It is more common to assume that the x_k are integers but then F_0 would not pass through the original data points, even if they all fell on the same straight line. The last definition for the x_k would insure that F_0 reproduces the line exactly, even for interior points. (Using this last definition also implies that if the y_k are observations on an underlying function f with a local quadratic Taylor series approximation, then the F_{-j} function gives the correct orthogonal projection to the approximating space V_{-j} of the multiresolution analysis.)

We are interested in the difference between the two approximating functions and we define it as G_{-j} where

$$G_{-j}(x) = F_{-(j-1)}(x) - F_{-j}(x).$$

The G_{-j} may also be written in terms of the wavelet function ψ for the D_4 family. In particular,

$$G_{-j}(x) = \sum_k R_{k-1}^{(j)} \psi(2^{-j}x - k)/(\sqrt{3} - 2),$$

where the wavelet coefficients for stage j are

$$R_{k-1}^{(j)}/(\sqrt{3} - 2) = (1/2) \sum_i (-1)^i c_{1-i} y_{2k+i+2}^{(j-1)}.$$

(The values of the coefficients $\{c_l\}$ are given in Section 3.)

Note that ψ has support $[-1, 2]$ and $\psi(\mu - k)$ is nearly zero for k equal to zero or one. Furthermore, $\psi(\mu - 1)$ is nearly equal to K where

$$K = \sqrt{3} - 2.$$

This property implies that the function G_{-j} essentially interpolates the "discrepancy" points, $\{(x_k^{(j)}, R_k^{(j)})\}$. It is an alternative to the "difference" function described in the last paragraph of Section 4.

We can describe the influence of the noise in the $\{y_k\}$ on the statistical behavior of the coefficients of the scaling function and the coefficients of the wavelet function for each stage j. The j^{th} set of scaling function coefficients $\{y_k^{(j)}\}$ all have variance $\sigma^2/2^j$ and are uncorrelated. They are also uncorrelated with the l^{th} set of wavelet function coefficients $\{R_k^{(l)}/(\sqrt{3}-2)\}$, for all l such that $l \leq j$. Also the j^{th} set of wavelet function coefficients $\{R_k^{(j)}/(\sqrt{3}-2)\}$ have variance $\sigma^2/2^j$ and are uncorrelated with each other and with the coefficients in the l^{th} set $\{R_k^{(l)}/(\sqrt{3}-2)\}$, for all l such that$l \leq (j-1)$.

7 Conclusion

Properties of the prediction points and the "residual-like" discrepancies follow directly from their algebraic definition and at each stage they may be seen to be orthogonal linear transformations of the prediction points from the previous stage. Their geometric interpretation gives them an independent life from the wavelet multiresolution analysis that generated them. We hope that statisticians will gain insight into wavelet analysis from this example and see ways to use or improve it.

No new references are offered but two statisticians who have written on the topic and whose names are not mentioned in the previous article are R. Carmona at the University of California at Irvine and I. McKeague at Florida State University. General information about wavelets is available by e-mail in the Wavelet Digest recently edited by Wim Sweldens at the University of South Carolina. To subscribe, send e-mail to wavelet@math.scarolina.edu with "subscribe" as the subject.

8 Appendix

Note 1. Each approximating function F_{-j} is assumed to lie in the multiresolution analysis approximating space V_{-j}, a closed linear subspace of L_2. The functions in V_{-j} have the form

$$g_{-j}(x) = \sum_k \gamma_{-j,k}\phi(2^{-j}x - k)$$

where g_{-j} is in L_2. (The reader is cautioned that this subspace would be labeled V_j in the notation of Daubechies book while it is labeled V_{-j} in Chui's book.) These V_{-j} are the same closed subspaces described in the multiresolution analysis of the previous article. (The fact that the subspaces are closed was inadvertently omitted there.) For the wavelet family D_4, the scaling function ϕ is continuous and all the functions in V_{-j} continuous. Functions of the form given in the first equation of this note can be used to represent low degree polynomials such as constant functions and straight lines over the entire real line. But none of these nonzero polynomials are in L_2 and thus none of these polynomials are in V_{-j}. Notice also that these functions F_{-j} are not the f_{-j} of the previous article, where

$$f_{-j}(x) = \sum_k a_{-j,k}\phi(2^{-j}x - k).$$

Those f_{-j} are the projections of the function f to the subspaces V_{-j}. The $y_k^{(j)}$ are only approximations to the correct coefficients

$$a_{-j,k} = 2^{-j}\int f(y)\phi(2^{-j}y - k)dy$$

Note 2. In the representation of G_{-j}, the function $\psi(2^{-j}\cdot -k)$ is purposefully not normalized, i.e. multiplied by $2^{-j/2}$. This preserves the interpretation of $R_k^{(j)}$ as a "discrepancy". Notice also that these functions G_{-j} are not the difference functions described in the previous article, i.e. G_{-j} is not

$$\sum_k b_{-j,k}\psi(2^{-j}x - k),$$

the projection of the function f to the subspace W_{-j}. The value $R_{k-1}^{(j)}/(\sqrt{3}-2)$ is only an approximation to the correct coefficient

$$b_{-j,k} = 2^{-j} \int f(y)\psi(2^{-j}y - k)dy.$$

References

[1] I. Daubechies. *Ten Lectures on Wavelets.* CBMS-NSF Series in Applied Mathematics, SIAM.