High-Dimensional Wavelet Modeling

Amitava Dhar and Rutvik Desai

January 26, 1996

Abstract

Wavelets can be thought of as a set of well-localized basis functions with very good approximation properties. The difficulty in applying wavelet approximation to high-dimensional data is that the number of basis functions increases exponentially with the number of dimensions, making the application of standard mathematical methods for determining coefficients diffic ult. We propose a modeling methodology that uses multidimensional cubic B-spline wavelets whose co efficients are determined by a nonlinear optimization procedure that combines simulated anneali ng with hill climbing.

1 Introduction

The wavelets can be viewed as an alternative to Fourier analysis for nonlinear modeling. The most important difference between Fourier analysis and wavelet analysis is probably in the nature of basis functions used: the Fourier basis functions are global while the wavelets basis functions are local. The local basis functions with certain other attractive approximation properties have made the wavelet analysis a better option for many applications such as signal processing, data compression, etc. [3].

There could be several ways of using wavelets for approximation in high dimensions, however, the proposed methodology adopts the technique suggested by Daubechies [3]. This technique generates the high dimensional wavelet basis functions from the product of the one dimensional truncated wavelet series representations. The procedure is analogous to that proposed

by Courant and Hilbert [1] for generating multidimensional orthonormal bases from one dimensional orthonormal basis functions.

The number of wavelet basis functions increase exponentially with the increase in number of dimensions or input variables. A nonlinear optimization technique may be required to determine the coefficients if the number of bases is very large. A technique such as simulated annealing coupled with a local optimization technique is powerful in the sense that the simulated annealing identifies a region on the error surface while the local optimization procedure finds the local minimum of that region very quickly. This improves the speed of convergence of the optimization procedure significantly when compared to a pure simulated annealing. Also, Desai and Patil [11] has shown that combining adaptive simulated annealing [6] with hill climbing [10] outperforms Parallel Genetic Algorithm and ASA on a variety of functions with different characteristics.

We will describe briefly about wavelets in the next section, which will be followed by the description of the proposed optimization procedure.

2 Wavelets

2.1 Introduction

As pointed out earlier, the wavelets have a significant advantage over Fourier analysis for modeling because of their good localization characteristics. A family of wavelets is defined in terms of translations and dilations of a single function, a "parent" wavelet $\psi(x)$:

$$\psi_{m,k}(x) = \sqrt{2^{-m}}\psi(2^{-m}x - k) \qquad \text{for} \quad (m,k) \in Z^2 \tag{1}$$

where m and k denote the resolution level and the location of a wavelet. In equation(1), the powers of two have been used to generate the family of wavelets, however, any other integer or fraction could also have been used [3].

It may be noted that if the parent wavelet $\psi(x)$ decays quickly, then linear combinations of functions from the family defined in equation(1) will retain this property and have good localization characteristics. Various wavelets have been developed by many researchers and the choice of wavelet may influence the prediction accuracy for a specific application.

The cubic spline wavelets have good localization characteristics. Although the cubic spline wavelets of the proposed algorithm are not completely orthogonal, the wavelets between the scales are orthogonal. Moreover, the cubic splines themselves being well-localized, the complete set of the basis functions may be considered as close-to-orthogonal. The cubic spline wavelets are, therefore, well-suited for nonlinear modeling.

A function needs to be expressed in terms of the wavelet components in order to perform modeling. The way in which this is accomplished is best described in the context of a multiresolution analysis.

2.2 Multiresolution Analysis

A multiresolution analysis of data can be performed by considering a sequence of approximation spaces V_m that satisfy certain properties, some of which are discussed below (a detailed description is available in [2]). The spaces V_m are related to each other as follows:

$$.... \quad \subset V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset V_{-2} \quad ... \tag{2}$$

There is a special function $\phi(x)$, called a scaling function, which lies in V_0. All the functions in V_0 are linear combinations of translations of this scaling function $\phi(x)$ and V_0 is "translation invariant":

$$\phi(x) \in V_0 \iff \phi(x-k) \in V_0, \qquad k \in Z \tag{3}$$

In fact, the scaling function $\phi(x)$ generates the entire sequence of V_m spaces in a sense. To be precise, if $\phi(x)$ is defined as

$$\phi_{m,k}(x) = \sqrt{2^{-m}}\,\phi(2^{-m}x - k) \qquad \text{for} \quad (m,k) \in Z^2 \tag{4}$$

then $\{\phi_{m,k} : \quad (m,k) \in Z^2\}$ is a basis for the space V_m. The scaling functions and the wavelets are linked by the following refinement equation:

$$\psi(x) = \sum_{-\infty}^{+\infty} c_k \phi_{-1,k}(x) \tag{5}$$

where c_ks are the coefficients. Note that ψ_k is in the space V_{-1} because $\phi_{-1,k}$s in the right hand side of the above equation are the bases for the space V_{-1}.

From the above-mentioned concept, we can write a wavelet series expression for a square-integrable function $f(x)$. If $F_m(x)$ denotes the best approximation of $f(x)$ in the space V_m, then we can write:

$$F_m(x) = \sum_{k=-\infty}^{+\infty} \alpha_{m,k}\phi_{m,k}(x) \tag{6}$$

One of the properties of a multiresolution hierarchy is that all the spaces are simply the scaled versions of each other:

$$F_m(x) \in V_m \iff F_{m-1}(2x) \in V_{m-1} \tag{7}$$

The difference between an approximation $F_m(x)$ in V_m and a finer approximation $F_{m-1}(x)$ in V_{m-1} can be found explicitly in terms of wavelets as follows:

$$F_{m-1}(x) - F_m(x) = \sum_{k=-\infty}^{+\infty} \delta_{m,k}\psi_{m,k}(x) \tag{8}$$

The above equation can be put together with equation(6) to yield a complete expression of the best approximation to $f(x)$ at the resolution level $(m-1)$:

$$F_{m-1}(x) = \sum_{k=-\infty}^{+\infty} \alpha_{m,k}\phi_{m,k}(x) + \sum_{k=-\infty}^{+\infty} \delta_{m,k}\psi_{m,k}(x) \tag{9}$$

2.3 Wavelets in Higher Dimensions

The one dimensional cubic B-spline scaling functions and wavelets at any scale m and location k have the following general expressions [8]:

$$\phi_{m,k}(x) = 0.690988\sqrt{2^{-m}}exp\{-1.5(2^{-m}x - k)^2\} \qquad \text{for} \quad (m,k) \in Z^2 \tag{10}$$

$$\psi_{m,k}(x) = 0.251477\sqrt{2^{-m}}cos\{2.570935\{2(2^{-m}x - k) - 1\}\} \times$$
$$exp\{-0.222759\{2(2^{-m}x - k) - 1\}^2\} \qquad \text{for} \quad (m,k) \in Z^2 \tag{11}$$

Multidimensional wavelets can be derived from one dimensional wavelets by the method proposed by Daubechies [3]. To start with, let us consider the

case of two input variables. The two-dimensional space $\mathbf{V_m}$ can be expressed as a tensor product of one-dimensional spaces V_{m1} and V_{m2}:

$$\mathbf{V}_m = V_{m1} \otimes V_{m2} = \overline{Span\{F(x_1, x_2) = f(x_1)g(x_2); f \in V_{m1}, g \in V_{m2}\}} \quad (12)$$

The two dimensional space $\mathbf{V}_m$ form a multiresolution ladder in $L^2(R^2)$. Since the spaces V_{m1} and V_{m2} can be represented by truncated wavelet series of equation(9), the tensor product of these two spaces is spanned by the basis functions generated by the product of two wavelet series. The concept can be readily extended to higher dimensions. For n-dimensional space (which means n number of input variable), the equation(9) can be modified as follows:

$$F_{m-1}(x_1, , x_n) = \prod_{i=1}^{n} [\sum_{k=-\infty}^{+\infty} \alpha_{i,m,k} \phi_{m,k}(x_i) + \sum_{k=-\infty}^{+\infty} \delta_{i,m,k} \psi_{m,k}(x_i)] \quad (13)$$

The above equation gives us the bases for modeling with wavelets in high dimenions. In the next section, we will describe briefly about how the bases are used to to model a discrete set of data points.

2.4 Modeling from Discrete Data Points

Many real-world problems are basically modeling from nonuniformly distributed discrete data points. Multidimensional wavelet modeling of discrete data points first requires scaling of input data. For simplicity, one can use same scaling range for all the dimensions. The scaling range will be larger for finer approximation and will always be between 0 and $(l-1)$ where $2l$ is the number of bases in one dimensional series representation. For example, one needs to scale the input data between 0 and 3 if a one dimensional wavelet series representation with eight basis functions is desired. The one dimensional equation then takes the following form:

$$f(x_i) = \sum_{k=0}^{3} \alpha_{i,m,k} \phi_{m,k}(x_i) + \sum_{k=0}^{3} \delta_{i,m,k} \psi_{m,k}(x_i) \quad (14)$$

Equation(13) can then be modified as follows to generate the multidimensional bases:

$$F_{m-1}(x_1, , x_n) = \prod_{i=1}^{n} [\sum_{k=0}^{3} \alpha_{i,m,k} \phi_{m,k}(x_i) + \sum_{k=0}^{3} \delta_{i,m,k} \psi_{m,k}(x_i)] \qquad (15)$$

The above equation can be conveniently rearranged or modified to design cost-function required for a specific application. Once the cost-function is formulated, the optimization can be carried out by the procedure described in the next section.

3 Simulated Annealing and SALO

3.1 Simulated Annealing

Simulated annealing [5] is an optimization technique with several advantages. It can process cost functions with arbitrary degrees of nonlinearities, discontinuities and stochasticity. Also, it takes care of arbitrary boundary conditions and constraints imposed on these cost functions. The methodology can be implemented quite easily and guarantees statistically optimal solution. It is derived from Monte Carlo methods in statistical mechanics and attempts to avoid local minima by taking non-locally optimal steps in the search space, when needed. The details of a conventional simulated annealing procedure is available in [4].

L. Ingber [6] proposed an improved method for annealing, which can use a significantly faster temperature annealing schedule than that for a conventional simulated annealing. This method is known as Very Fast Simulated Re-annealing (VFSR). We propose the use of a modified VFSR known as Adaptive Simulated Annealing (ASA) [7].

3.2 Local Optimization

While a local optimizer such as hill climber is an efficient method for optimization in simple, unimodal spaces it easily gets stuck in non-optimal regions in real world problems. The main reasons for the failure of the hill climber are (1) local optima (the foothill problem); (2) flat surfaces (the plateau problem) and/or (3) ridges [9]. The local optimization algorithm proposed in [10] attempts to tackle some of these problems while trying to maintain the efficiency by employing the following ideas: (1) Adjust the size

of the probing steps to suit the nature of the terrain, shrinking when probes do poorly and growing when probes do well. (2) Keep track of the directions of recent successes, so as to probe preferentially in the direction of most rapid descent. The outline of this algorithm is given below.

LOCAL-OPTIMIZE $(f, \vec{x})$
initialize $\vec{v}; \vec{u} \leftarrow \vec{0}$
while $\mid \vec{v} \mid \geq$ THRESHOLD
    $iter \leftarrow 0$
    while $f(\vec{x} + \vec{v}) > f(\vec{x})$ and $iter \leq$ MAXITER
        $\vec{v} \leftarrow$ RANDOM-VECTOR$(\vec{v})$
        $iter \leftarrow iter + 1$
    if $f(\vec{x} + \vec{v}) > f(\vec{x})$
        $\vec{v} \leftarrow \vec{v}/2$
    else if $iter = 0$
        $\vec{x} \leftarrow \vec{x} + \vec{v}$;
        $\vec{u} \leftarrow \vec{u} + \vec{v}$;
        $\vec{v} \leftarrow 2\vec{u}$;
    else if $f(\vec{x} + \vec{u} + \vec{v}) < f(\vec{x})$
        $\vec{x} \leftarrow \vec{x} + \vec{u} + \vec{v}$;
        $\vec{u} \leftarrow \vec{u} + \vec{v}$;
        $\vec{v} \leftarrow 2\vec{u}$;
    else
        $\vec{x} \leftarrow \vec{x} + \vec{v}$;
        $\vec{u} \leftarrow \vec{v}$;
        $\vec{v} \leftarrow 2\vec{v}$;
return $\vec{x}$

3.3 SALO

SALO (Simulated Annaling with Local Optimization) [11] algorithm attempts to combine the ability of the SA process to get out of local optima and the efficiency of the local optimizer in relatively "simple" regions of the space. From every point in the space generated by the SA process, we start the local optimizer and allow it to converge to the nearest local optimum. The value of the function at the local optimum is used as the evaluation

value for the original point, and an acceptance or rejection decision is made according to the metropolis criterion. A new state is generated from the locally optimal point, and the process is repeated. The basic high-level outline of the algorithm for minimization of a function $f(\vec{x})$ is given below. In this algorithm, $\vec{x}_{BSF}$ is the best $\vec{x}$ encountered so far. PERTURB($\vec{x}$) returns a point in the neighborhood of $\vec{x}$.

SALO (f)
INITIALIZE ($\vec{x}, t$)
$\vec{x}_{BSF} \leftarrow \vec{x}$;
do
  do
    $\vec{y} \leftarrow$ PERTURB($\vec{x}$);
    $\vec{y} \leftarrow$LOCAL-OPTIMIZE($f, \vec{y}$);
    $\Delta f_{xy} \leftarrow f(\vec{y}) - f(\vec{x})$;
    if (($\Delta f_{xy} \leq 0 \| ($RANDOM()<EXP($-\Delta f_{xy}/T(t)$))))
      $\vec{x} \leftarrow \vec{y}$;
      if ($f(\vec{x}) < f(\vec{x}_{BSF})) \vec{x}_{BSF} \leftarrow \vec{x}$;
  while (*equilibrium has not been reached*);
  INCREMENT(t);
while (*stop criterion has not been met*);
return $\vec{x}_{BSF}$;

We use ASA as the basic SA algorithm to which the local optimizer described above is added. In [11], SALO was shown to outperform a Parallel Genetic Algorithm and ASA by a significant margin on a variety of functions with different chracteristics.

4 Conclusions

The methodology to model with wavelet-based cost-function which is optimized by a combined simulated annealing - hill climbing algorithm needs to be applied to different problems and the results compared with that of other existing methods in order to be able to evaluate its approximating power. So far, the methodology has been applied to a fraud detection problem. The

preliminary results are encouraging to pursue further work on this topic.

References

[1] R. Courant and D. Hilbert, "Methods of Mathematical Physics," Vol. 1, *Interscience Publishers*, New York, 1955.

[2] C. K. Chui, "*An Introduction to Wavelets*," Academic Press Inc., San Diego, California, 1992.

[3] I. Daubechies, "Ten Lectures on Wavelets," *SIAM*, Philadelphia, Pennsylvania, 1992.

[4] S. Haykin, "Neural Networks," *Macmillan College Publishing Company, Inc.*, New York.

[5] S. Kirkpatrick, C. D. Gelatt and M. P. Vecchi, "Optimization by Simulated Annealing," *Science*, 220, pp. 671-680, 1983.

[6] L. Ingber, "Very Fast Simulated Re-Annealing," *Mathl. Comput. Modelling*, 12, pp. 967-973, 1989.

[7] L. Ingber, *Adaptive Simulated Annealing (ASA)* [ftp.caltech.edu: /pub/ingber/asa.Z], Software package documentation, 1995.

[8] M. Unser and A. Aldroubi, "Polynomial Splines and Wavelets- A Signal Processing Perspective," *Wavelets: A Tutorial in Theory and Applications*, Editor C. K. Chui, Academic Press Inc., San Diego, California.

[9] P. Winston, *Artificial Intelligence*, Addison-Wesley Publishing Company, Reading, MA, third edition, 1992.

[10] D. Yuret, "From Genetic Algorithms to Efficient Optimization," *Masters Thesis*, Dept. of Electrical Engineering, MIT, May 1994.

[11] R. Desai and R. Patil, "SALO: Combining simulated annealing and local optimization for efficient global optimization," *Technical Report LA-UR [95-2862]*, Los Alamos National Laboratory, 1995.