Bayesian wavelet networks for nonparametric regression

C C Holmes and B K Mallick

Department of Mathematics, Imperial College
180 Queen's Gate
London SW7 2BZ

September 1997

Abstract

Radial wavelet networks have recently been proposed as a method for nonparametric regression. In this paper we analyse their performance within a Bayesian framework. We derive probability distributions over both the dimension of the networks and the network coefficients by placing a prior on the degrees of freedom of the model. This process bypasses the need to test or select a finite number of networks during the modelling process. Predictions are formed by mixing over many models of varying dimension and parameterization. We show that the complexity of the models adapts to the complexity of the data and produces good results on a number of benchmark test series.

Keywords: Wavelets, radial basis functions, variable architecture selection, Bayesian neural networks, reversible jump Markov chain Monte Carlo, nonparametric regression.

1 Introduction

Wavelet networks have previously been studied in relation to nonparametric regression by Zhang (1997), Kugarajah and Zhang (1995), Zhang and Benveniste (1992) and Cao *et al* (1995). The networks are similar in structure to Radial Basis functions though motivated by wavelet theory (Chui, 1992). Zhang (1997) pays particular attention to a constructive algorithm used to build wavelet networks given a data set to be approximated. Here we adopt a complementary approach from a Bayesian perspective. Rather than specifying one particular architecture we define a probability distribution over the whole of wavelet network space. Using specially tailored Markov chain Monte Carlo (MCMC) algorithms we generate posteriors over this space. The posterior densities can be used to provide predictive distributions on model outputs and marginal distributions on say the models' dimension. This approach has previously been utilized in the field of Radial Basis Functions by Holmes and Mallick (1997).

The main contribution of this paper is to highlight a full Bayesian approach to wavelet network construction and show that the resulting models prove highly accurate at approximating a number of benchmark test series. We considerably extend the analysis of Holmes and Mallick (1997) to show how the Bayesian approach naturally controls model complexity through a Bayes Factor term that enters into the MCMC algorithm.

The paper is structured as follows. In section 2 we introduce wavelet networks from a classical and Bayesian perspective. Section 3 develops our Bayesian strategy and argues for a prior on the degrees of freedom exhibited by the network. In this section we describe the reversible jump MCMC algorithm used in our computations (Green, 1995). The reversible jump MCMC can be used to construct distributions when the dimension of the space is unknown. Results on some standard test sets are presented in section 4. We draw some conclusions on our work in section 5.

2 Wavelet Networks and Regression

Wavelet theory has proved remarkably popular in signal processing particularly in the areas of data compression and de-noising (Chui, 1992). Until recently however, the application to regression has not been widely studied.

This is probably because most of the emphasis has revolved around discrete wavelet analysis due to the fast algorithms and analytical solutions provided with this method. However, discrete wavelet analysis is generally only applicable to regularly spaced data of order 2^n (where n is an integer) and tend to be limited to the analysis of low dimensional data due to the exponential growth in storage terms with dimension. To overcome these problems Zhang and Benveniste (1992) advocated the use of continuous radial wavelet frames for nonparametric regression. We describe this method below.

Suppose we have a data set $\mathcal{D}$ that is partitioned into a variable of interest, $y \in \mathbf{R}$ and a group of covariates $x \in \mathbf{R}^d$. We assume that the following relationship exists

$$y_i = m(x_i) + \epsilon_i, \tag{1}$$

where the regression curve (surface) $m()$ is taken as the conditional mean function $m(x) = E[y \mid x]$ and ϵ is white noise (i.i.d) $\sim N(0, \sigma^2)$.

Then we can approximate $m()$ by

$$\hat{m}(x) = \sum_{j=1}^{K} w_j \psi^j(x), \tag{2}$$

where $\psi(x)$ represents a radial wavelet function in the frame F of $L_2(\mathbf{R}^d)$, and where

$$F = \{\psi_{m,n}(x) = \phi^{-1/2dn}\psi(\phi^{-n}x - m\tau) : n \in \mathbf{Z}, m \in \mathbf{Z}^d\}, \tag{3}$$

and ϕ and τ are dilation and translation step sizes.

A function ψ is permissible as a radial wavelet if the following inequality holds

$$C_\psi = (2\pi)^d \int_0^\infty \frac{\mid \eta(\zeta) \mid^2}{\zeta} d\zeta < \infty, \tag{4}$$

where $\eta(x)$ is the Fourier transform of the radial function $\psi(x)$. The Fourier transform is itself radial. By radial we refer to the property that the response is purely a function of the Euclidean norm of its argument. Throughout this paper we will use the 'Mexican hat' radial wavelet basis function given by

$$\psi(x) = (d - \|x\|^2) \exp(-\frac{\|x\|^2}{2}), \tag{5}$$

where d is the dimension of x and $\|x\|^2$ represents the Euclidean norm of the vector x. The wavelet networks are built up of dilations and translations of this function. We will allways include a global linear term in the approximations so that the full model can now be written as

$$\hat{m}(x) = \sum_{j=0}^{d} w_j x_j + \sum_{j=d+1}^{k} w_j \psi_j(D_j|x - \mu_j|), \tag{6}$$

where D_j and μ_j are the dilation and translation parameters of the j^{th} basis. See Zhang (1997) for further details.

Equation (2) represents a truncation of the infinite basis space defined by (3). If we consider potential basis functions in the frame as regressors, then the problem of which basis to truncate (or which to include) is analogous to the task of variable selection in regression theory. As such any of the techniques in this field can be used. Zhang (1997) argues persuasevely for the use of an initial screening process whereby a salient subset of potential bases are derived. This is followed by either backward elimination from the full subset or orthogonal stepwise construction to minimize some cost function, such as Akaike's AIC or a Generalised Cross Validation measure.

Here we provide a complementary approach based on Bayesian theory. We do not select one single network for the approximation in (2). Instead we define a probability distribution over the infinite basis space F.

Each network model is defined by the number, location and dilation of its k wavelet bases, $M_k = \{\mu_1, D_1, \ldots, \mu_k, D_k\}$ where we use k to index the class of models having k basis functions. Within each model structure we also define the output coefficients w. When making point predictions we integrate over the space $\{k, M_k, w\}$ to give the conditional expectation

$$E[Y \mid x, \mathcal{D}] = \sum_{k=0}^{\infty} \int \int \hat{m}(x, M_k, w) p(w \mid M_k, \mathcal{D}) p(M_k \mid \mathcal{D}) dM_k dw, \tag{7}$$

where $p(M_k \mid \mathcal{D})$ and $p(w \mid M_k, \mathcal{D})$ represent posterior probabilities given the training data $\mathcal{D}$. Draper (1995) presents the argument for accounting for model uncertainty using this form of model averaging.

The main computational effort in our method is the calculation of the posterior probabilities $p(\cdot \mid \cdot)$. This is complicated by the fact that the dimension of M and w is not fixed, relating to models of varying numbers of wavelet basis functions.

In the next section we briefly review Bayesian methods and describe the particular techniques that we apply to this problem.

3 Bayesian Methods

Bayesian theory provides a coherent framework for regression (Bernardo and Smith, 1994). All assumptions are made explicit and uncertainty in model parameters is accommodated in the computations. These methods have proved highly accurate and popular for data approximation, see MacKay (1992b) and Denison *et al* (1996) for some examples in nonparametric regression.

The growing interest in Bayesian methods comes partly from the increased accessibility to fast computing devices that are often necessary for the calculations of the integrals in (7). The posterior densities used in (7) are given as a normalized product of likelihood and prior

$$p(\pi \mid \mathcal{D}) = \frac{L(\mathcal{D} \mid \pi)p(\pi)}{p(\mathcal{D})}, \tag{8}$$

where $L(\mathcal{D} \mid \pi)$ is the likelihood function and $p(\pi)$ is some prior density on the parameterization of the full model space π.

The likelihood of our wavelet networks can be derived from (1) and by noting that the noise term takes a Normal distribution $N(0, \sigma^2)$. Hence we obtain the following log-likelihood of the observed data up to an additive constant

$$l(\mathcal{D} \mid M, w) = -n \log \sigma - \frac{1}{2\sigma^2} \sum_{i=1}^{n} [y_i - m(x_i, M, w)]^2. \tag{9}$$

The prior density in (8) captures subjective beliefs on the model space. Typically this might be a preference for simpler models with fewer parameters or for the parameters to take on small values[1]. Here we place a vague shrinkage prior on the output weights w in (6),

$$p(w) \sim N(w \mid 0, \lambda^{-1} I), \tag{10}$$

[1] These encompass the universal modelling heuristics of parsimony and smoothness

where λ is the precision (inverse variance) of the normal distribution. A large value for λ results in smoother model outputs. We also include a prior on the degrees of freedom of the network structure. The degrees of freedom is determined by the number of wavelet bases as well as their dilation and translation parameters. The degrees of freedom (DF) measures the potential of the model to fit the data set prior to the target values being known. The degrees of freedom of a Bayesian wavelet network is based on the most probable parameters and can be calculated as the trace of the smoothing matrix, $\mathrm{DF} = \mathrm{tr}(S)$. The smoothing matrix S is defined as

$$S = \mathbf{\Psi}(\mathbf{\Psi}^{\mathbf{T}}\mathbf{\Psi} + \lambda\mathbf{I})^{-1}\mathbf{\Psi}^{\mathbf{T}}, \tag{11}$$

where λ is the precision of the prior on w and $\mathbf{\Psi}$ is the $n \times k$ design matrix of outputs from the wavelet basis (hidden) layer and linear terms over the data set.

The trace of this matrix measures the flexibility of the model. It is equivalent to the sum of the derivatives of the model output with respect to the target data (Hastie and Tibshirani 1990),

$$\mathrm{DF} = \mathrm{tr}(S) = \sum_{i=1}^{n} \frac{dy_i}{d\hat{m}_i}, \tag{12}$$

where n is the number of data points.

In fact the calculation in (12) will underestimate the degrees of freedom in our model. It fails to take account that the basis layer output Ψ is to some extent dependent on the target series being approximated. This occurs as Ψ is adapted during the modelling process. This calculation does though provide a convenient approximation for use in the prior ratios that we use in section 3.1.

To express our beliefs on the degrees of freedom of the models we use a Gamma distribution with parameters α and β. The prior on the network space $p(M, w)$ can now be written as

$$\begin{aligned} p(M, w) &= p(D, \mu, k)p(w) \\ &= Ga(\mathrm{DF} \mid \alpha, \beta)N(w \mid 0, \lambda) \\ &\propto \mathrm{DF}^{\alpha-1}\exp(-\beta\mathrm{DF})\exp(-\lambda\|w\|^2). \end{aligned} \tag{13}$$

The prior on DF will penalize models with degrees of freedom above or below the mode[2], the extent of the penalty being driven by the variance.

[2]The mean of a Gamma distribution is α/β and the variance is α/β^2.

The attractiveness of this prior comes from its explicit nature. Setting a prior mean of $\alpha/\beta = 1$ indicates the belief that the data can be approximated by the average of the series. Setting $\alpha/\beta = d + 1$, where d is the dimension of the data, suggests a linear surface is preferred. Most people will have a feel for this kind of penalty. Prior subjective beliefs on coefficient values, such as shrinkage priors on output weights, are harder to quantify.

The prior on degrees of freedom encompasses the tradeoff that exists between the number of basis functions, the dilation parameters and the setting of the precision parameter λ for the output weights. Large values of λ will result in networks with few degrees of freedom even when there are many basis functions present. The interaction between $\{\lambda, \mu, D\}$ in determining the degrees of freedom is highlighted in Figures 1 and 2. In Figure 1 we plot the degrees of freedom calculated for three networks with different values of λ as we add basis functions at data points drawn randomly from a 50 point data set uniformly spaced on $(0, 5)$. The dilation parameters in all the bases were fixed at $D = 1.5$. Clearly, increasing the precision parameter λ decreases the degrees of freedom. With $\lambda = 0$ adding a basis function increases the DF by one. Now, using the network in Figure 1 for which $\lambda = 0.1$, we increase the dilation parameters in the basis functions to $D = 2.0$ and $D = 2.5$. The results are shown in Figure 2. Increasing D increases the degrees of freedom as the network can model sharper features of the data set.

Finally we must specify a prior term for the variance of the noise term in (1). This is given by a 'conjugate' inverse Gamma distribution. We will make this prior vague indicating our ignorance of the noise processes. Therefore the prior will be

$$p(\sigma^{-2}) = Ga(\sigma^{-2} \mid 10^{-3}, 10^{-3}). \tag{14}$$

Given a prior and a likelihood function we are now in a position to utilise equation (7) for predictions. Unfortunately these integrals are analytically intractable and as the number of wavelet basis functions is unknown *a priori* we also need to accommodate networks of varying dimension. This requires the use of approximating methods based on Markov chain Monte Carlo (MCMC) techniques (Tierney, 1994). These are described in the next section.

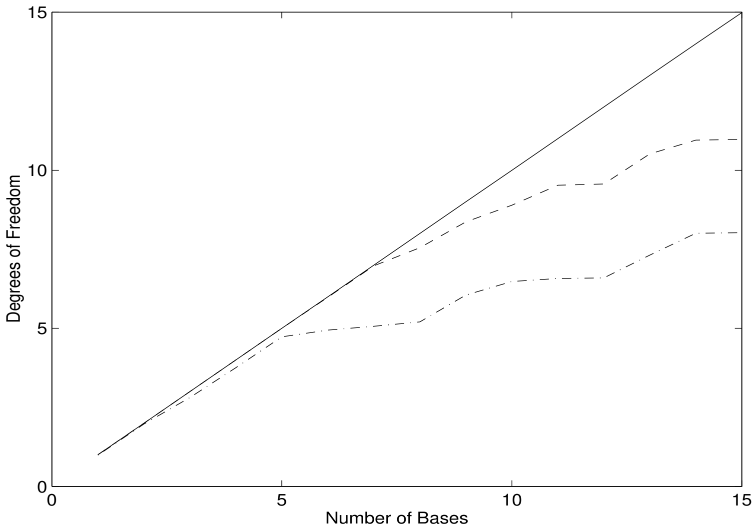

Figure 1: The Degrees of Freedom is plotted for three wavelet networks as basis functions are added. The data set consists of 50 data points uniformly spaced on $(0, 5)$. The location of the basis functions are the same for all three networks. Bases are added at data points drawn randomly from the data set. Each network differed only in the prior precisions λ. Solid line $\lambda = 0$, dashed line $\lambda = 0.0001$, dot dashed $\lambda = 0.1$. All basis functions in all three networks had dilation parameters set to 1.5.

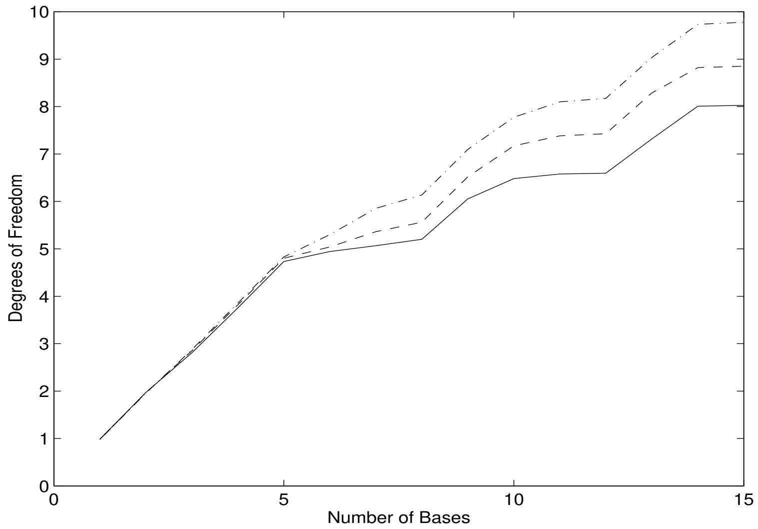

Figure 2: The Degrees of Freedom is plotted for three wavelet networks as basis functions are added. The data and basis locations are the same as that used in Figure 1. Each network differed only in the Dilation parameter of its basis functions: Solid line, $D = 1.5$; Dashed line, $D = 2.0$; Dot dashed, $D = 2.5$. The prior precision λ was fixed at $\lambda = 0.1$ for all three models.

3.1 Markov chain Monte Carlo Methods

MCMC methods are often used to approximate integrals of the form given in (7). They proceed by generating a Markov chain of samples, $(\pi_1, \pi_2, \ldots, \pi_N)$, whose members are drawn in direct proportion to the conditional probability $p(\pi \mid \mathcal{D})$. The integral in (7) can then be approximated by

$$I \approx \frac{1}{N - n_0} \sum_{t=n_0}^{N} m(\pi_t), \tag{15}$$

where N is the total number of samples generated (chain length) and n_0 is a 'burn in' period. The burn-in ensures that the Markov chain generating this sample has converged to the stationary distribution of interest, $p(\pi \mid \mathcal{D})$.

The task therefore is to generate samples of π in the correct proportions. This can be achieved as follows. Start the chain with a draw from the prior $p(\pi)$. Set $t = 0$. Propose a new sample, $\hat{\pi}_{t+1}$, to add to the chain based on the current value π_t and a randomly generated vector using a fixed (possibly symmetric) proposal distribution, say a Gaussian. This newly proposed state is then added to the chain with probability $Q()$ given by

$$Q(\pi_t, \hat{\pi}_{t+1}) = \min\left\{1, \frac{p(\hat{\pi}_{t+1} \mid \mathcal{D}) S(\hat{\pi}_{t+1}, \pi_t)}{p(\pi_t \mid \mathcal{D}) S(\pi_t, \hat{\pi}_{t+1})}\right\}, \tag{16}$$

where $p(\pi \mid \mathcal{D})$ is the posterior probability of π, and $S(\pi, \pi')$ is the probability of proposing a move from π to π'.

If the proposed state is not added to the chain then we add an extra copy of the current sample, $\pi_{t+1} = \pi_t$, and proceed to generate another proposal. Note that the proposal ratio, $S(\pi, \pi')/S(\pi', \pi)$, equals 1 if we use a symmetric proposal distribution. This particular MCMC method is known as the Metropolis-Hastings algorithm (Metropolis *et al*, 1953; Hastings, 1970).

We wish to integrate over models of differing dimension as well as over models of the same dimension. The reversible jump MCMC of Green (1995) is a MCMC techneque that can approximate integrals of this type. This is achieved by using a number of different move proposal types. At each iteration of the MCMC algorithm we select a proposal that attempts to do one of the following: change the dimension of the current state by adding or removing a basis function from the current model; dilate or shrink a wavelet basis within the current state by altering a dilation parameter D_i; shift the location of a wavelet basis μ_i.

The output vector of weights w are redrawn after each proposal by sampling from the full conditional distribution

$$p(w \mid \mathcal{D}, M) \sim N(w \mid (\Psi^T\Psi + \lambda I)^{-1}\Psi^T Y, \sigma^2(\Psi^T\Psi)), \tag{17}$$

where Ψ is the matrix of outputs from the wavelet (hidden) layer over the data set.

Having made the proposed changes to the network, a uniform random variable u is drawn from $U(0,1)$. If u is less than $Q()$ as given in (16) then the new proposed state is added to the chain otherwise a copy of the current network is retained.

The ratio in (16) can be written as

$$Q(\pi, \pi') = \min[1, (\text{model likelihood ratio})\times(\text{model prior ratio})\times(\text{proposal ratio})]. \tag{18}$$

It can easily be shown, see the Appendix, that using the above procedure leads to the following expression for $Q()$

$$Q(\pi, \pi') = \frac{p(\mathcal{D} \mid M')}{p(\mathcal{D} \mid M)} \times \frac{p(M')}{p(M)} \times \frac{S(M', M)}{S(M, M')}, \tag{19}$$

where $M = \{\mu, D\}$ as before. The prior ratio is taken from the Gamma prior on the degrees of freedom in (13). The ratio $p(\mathcal{D} \mid M')/p(\mathcal{D} \mid M)$ is known as the Bayes factor. The Bayes factor is the ratio of prior predictives for the two models. The prior predictive is the probability of observing the data before the data arrived, given the particular model. This term contains a natural "Occam" penalty against over complicated models. Complex models are able to fit the data better but they can also account for many different data sets. Hence the probability of observing the particular one being studied is reduced. The term $p(\mathcal{D} \mid M)$ contains a trade off between model complexity and fidelity to the data. MacKay (1992a) refers to this term as the "evidence" for the model. For further details and references see the review paper by Kass and Raftery (1995).

Using the above procedure we can generate many samples from the posterior $p(M, w \mid \mathcal{D})$. In the Appendix we list a pseudo-code algorithm of the above procedure. The chain of samples is used in (15) to approximate the integral in (7) when making predictions. In the next section we illustrate this method on two benchmark data sets.

4 Test Results

To illustrate the performance of the Bayesian wavelet networks we applied them to two tasks. The first is the forecasting of a chaotic time series. The second is to a surface smooth in 2-dimensions.

For the first example the Gamma prior on the degrees of freedom was set to a vague distribution, $p(\mathrm{DF}) \sim Ga(\mathrm{DF} \mid 10^{-3}, 10^{-3})$. This indicates that we have little prior knowledge of the degrees of freedom needed to fit the chaotic time series. The surface smooths in the second example are different though. Here we can visualize the data and make some assessment of the likely fit needed. For these examples we set $p(\mathrm{DF}) \sim Ga(\mathrm{DF} \mid 1, 1)$. This shows that we have some belief that the surfaces should be flat.

We sampled 10,000 networks using the algorithm described. The first 5,000 samples were discarded and of the remaining samples every third was used in our predictions to surpress correlations in the chain.

4.1 Mackey-Glass Chaotic Time Series

There has been considerable interest in recent years in the identification and prediction of chaotic time series (Farmer and Sidorowich, 1987; Casdagli, 1989; Lapedes and Farber, 1987; Cao *et al*, 1995). Conventional (linear) analysis fails to identify structure within these series as the global periodogram can appear flat indicating broad band noise. However, non-parametric methods however have proved successful in accurately modelling these series (Lillekjendlie *et al*, 1994). Most of the these models are based on Tarkens theorem of embedding, see Tarkens (1981). This theory states that for a dynamical system $x(t)$ of dimension d, a smooth mapping exists between $x(t)$ and the *embedding space* (formed by the lagging the system $\{x(t-\tau), x(t-2\tau), \ldots x(t-D\tau)\}$ for $D > 2d+1$ and almost all τ). For predictive purposes it is this smooth mapping that is the function to be approximated.

We generated a series from the Mackey-Glass equation. This equation was developed as a model for blood cell generation in leukemia patients (Mackey and Glass, 1977) and is given by

$$x(t+1) = 0.9x(t) + \frac{0.2x(t-\Delta)}{1 + x(t-\Delta)^{10}}, \tag{20}$$

Data Set	PVU
1	0.35
2	0.38
3	0.40

Table 1: Results for Mackey-Glass, $\Delta = 17$.

where Δ is a delay parameter and $x(t) = 0.9$, for $0 < t \leq \Delta$.

We generated a time series with $\Delta = 17$ and sampled every sixth point, $\tau = 6$. This forms the target series for our model. The embedding (input) dimension was set to 4. The regression equation given our model $m()$ can be written as

$$x(t) = m(x(t-6), x(t-12), x(t-18), x(t-24)).$$

Three training sets of 500 data points were constructed starting at $t = 3000$, $t = 6000$ and $t = 9000$. The three accompanying test sets were taken as the 500 points following the end of each training set.

The results are given in terms of the percentage of variance unexplained (PVU). The PVU is calculated using

$$\text{PVU} = \frac{E[\hat{m}(x) - m(x)]^2}{E[m(x) - \bar{m}]^2} \times 100\%, \tag{21}$$

where $\hat{m}(x)$ is the wavelet network prediction, $m(x)$ is the true value of the function and $\bar{m}$ is the mean of the true function over the test set.

The results are tabulated in Table 1. and a plot of 100 points from the test target series and prediction is shown in Figure 3. Clearly, the model has captured the underlying dynamics. The results in Table 1. can be compared with those listed in Lillekjendie *et al* (1994) in their review paper on chaotic time series. This is given in Table 2. On this data set the Bayesian Wavelet Network appears to outperform other techniques.

One advantage of using MCMC approaches is that marginal densities are readily obtained from the final chain. For instance in the above example we might be interested in the probability distribution for the number of wavelet bases used $p(k \mid \mathcal{D})$. This is shown in Figure 4. We can see that the most probable number of basis functions is 16.

Method	PVU
Multi-layer perceptron	1.0
Radial Basis Function network	1.1
Weighted linear map	1.3
Local Linear	3.3
Bayesian Wavelet Network (average)	0.4

Table 2: Comparative results for Mackey-Glass, $\Delta = 17$. The first four results are taken from Lillekjendie *et al* (1994). The figure for the wavelet network is the average error over the three data sets in Table 1.

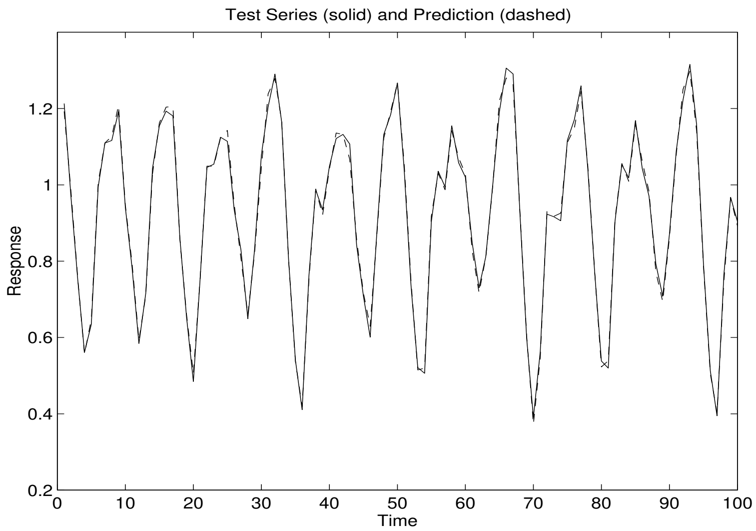

Figure 3: The Mackey-Glass time series (solid) with parameters $\Delta = 17$ and sampling rate $\tau = 6$ is shown alongside the model prediction (dashed).

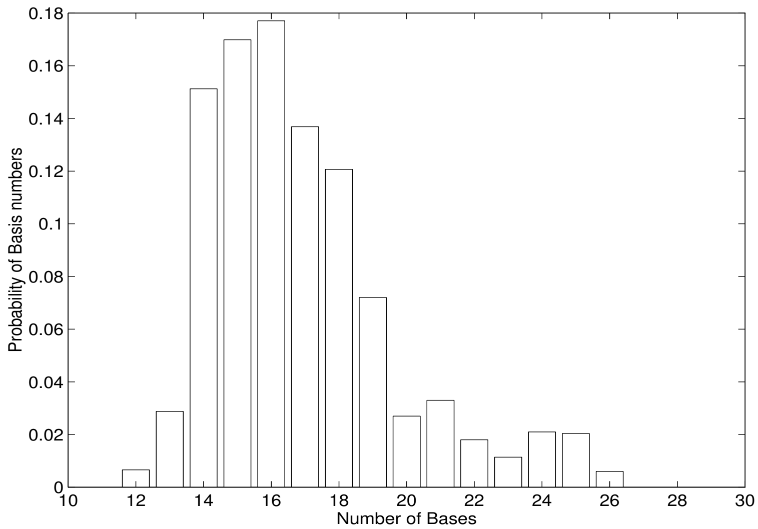

Figure 4: The marginalised probability of the number of wavelet basis for the first data set in the prediction of the Mackey-Glass time series with $\Delta = 17$.

We then generated another data set using the Mackey-Glass equation but now with the delay parameter set to $\Delta = 30$. Increasing the delay parameter is known to increase the difficulty of forecasting on the series as the information dimension increases (Casdagli, 1989). As before, three data sets are generated using a delay sampling rate $\tau = 6$ and taking the starting points at $t = 3000$, $t = 6000$ and $t = 9000$. This time the embedding space is six dimensional, $\{x(t-6), \ldots, x(t-36)\}$.

The results are shown in Table 3. and compared with other methods in Table 4. Again we see that the wavelet networks perform well when compared with other methods on this data set. If we examine the histogram of $p(k \mid \mathcal{D})$ in Figure 5. we see that the method tends to use more basis functions for this data set than for the previous one with $\Delta = 17$. This shows that the Bayesian approach can find the appropriate dimension to match the complexity of the data.

4.2 Surface Smooth

The previous example is slightly unusual in that no noise is present in the data set. This is atypical of real world examples where measurement noise is often

Data Set	PVU
1	0.50
2	0.65
3	0.67

Table 3: Results for Mackey-Glass, $\Delta = 30$.

Method	PVU
Multi-layer perceptron	3.16
Radial Basis Function network	2.51
Local Linear	5.75
Bayesian Wavelet Network (average)	0.61

Table 4: Comparative Results for Mackey-Glass, $\Delta = 30$. The first three results are taken from Casdagli (1989). The figure for the wavelet networks is the average error over the three data sets.

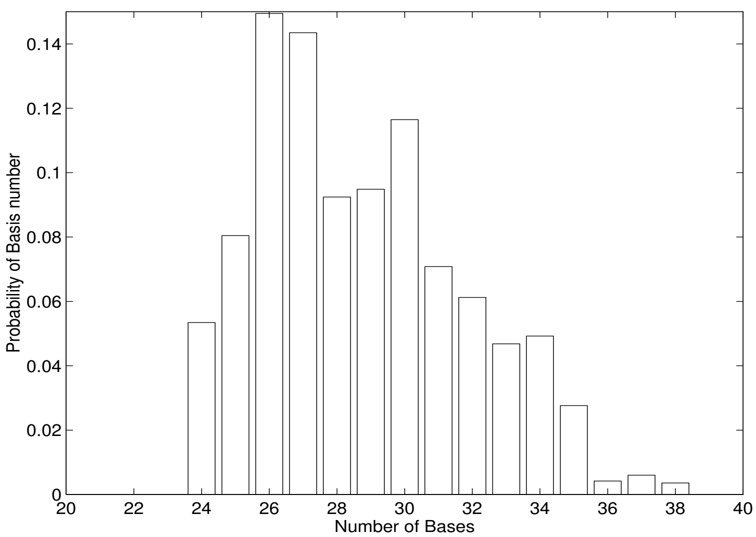

Figure 5: The marginalised probability of the number of wavelet basis for prediction of the Mackey-Glass time series with $\Delta = 30$.

present. In this section we analyse five test functions, in two dimensions, first described by Hwang *et al* (1994) in their study on multi-layer perceptrons and projection pursuit models. These data sets will be corrupted with white noise, $N(0, \sigma^2)$.

The five functions we examine are given below:

1. *Simple Interaction Function*

$$f^{(1)}(x_1, x_2) = 10.391[(x_1 - 0.4)(x_2 - 0.6) + 0.36]$$

2. *Radial Function*

$$\begin{aligned} f^{(2)}(x_1, x_2) &= 24.234[r^2(0.75 - r^2)] \\ &\quad \text{where } r^2 = (x_1 - 0.5)^2 + (x_2 - 0.5)^2 \end{aligned}$$

3. *Harmonic Function*

$$\begin{aligned} f^{(3)}(x_1, x_2) &= 42.659[0.1 + \hat{x}_1(0.05 + \hat{x}_1^4 - 10\hat{x}_1^2\hat{x}_2^2 + 5\hat{x}_2^4)] \\ &\quad \text{where } \hat{x}_1 = x_1 - 0.5, \text{ and } \hat{x}_2 = x_2 - 0.5 \end{aligned}$$

Method	Simple	Radial	Harmonic	Additive	Complex
Multi-layer perceptron	1.78	2.65	20.85	1.97	7.00
Projection Pursuit	0.76	2.29	1.04	2.10	4.27
Bayesian Wavelet Network	0.29	0.46	1.50	2.31	2.53
(most probable no. bases)	(23)	(28)	(65)	(68)	(53)

Table 5: Comparative Results for the five test functions in terms of PVU. The numbers in brackets give the most probable number of basis functions for each data set. This is the mode of $p(k \mid \mathcal{D})$. We can see that the mode changes with the complexity of the function to be approximated.

4. *Additive Function*

$$\begin{aligned} f^{(4)}(x_1, x_2) &= 1.3356\{1.5(1 - x_1) + e^{2x_1 - 1}\sin[3\pi(x_1 - 0.6)^2] \\ &\quad + e^{3(x_2 - 0.5)}\sin[4\pi(x_2 - 0.9)^2]\} \end{aligned}$$

5. *Complicated Interaction Function*

$$f^{(5)}(x_1, x_2) = 1.9\{1.35 + e^{x_1}\sin[13(x_1 - 0.6)^2]e^{-x_2}\sin(7x_2)\}$$

In order to compare our method with those studied by Hwang *et al* we follow their procedure. 225 data points were generated randomly on the unit square and the response, y, was calculated as the corrupted function $y_i = f(x_{1i}, x_{2i}) + \epsilon$ where $f()$ is the true test function and ϵ_i is Gaussian white noise drawn from a $N(0, 0.25^2)$ distribution. The test set comes from generating 10,000 data points on a 100 by 100 grid over the unit square, i.e. $[(1/200, 1/200),\ (3/200, 1/200), \ldots\ (199/200, 1/200), (1/200, 3/200), \ldots$ $(199/200, 199/200)]$.

The results are listed in terms of PVU alongside the most accurate models[3] studied by Hwang *et al* (1994). Under the PVU scores for the Bayesian wavelet network we list in brackets the size of the most probable networks in terms of number of basis functions used. Again we see that this approach adapts the size of models to fit the complexity of the data sets.

[3]As determined by average PVU over all five test functions.

5 Conclusions

Continuous wavelet networks provide a viable alternative to other methods of nonparametric regression. They help extend wavelet analysis into data sets of unequally spaced data in possibly high dimensions.

By taking a Bayesian approach to the analysis of wavelet networks we gain a number of advantages: Prior beliefs about the plausibility of various models can be incorporated within the computations; all assumptions are made explicit, and; distributions over predictions can be provided allowing confidence regions to be established and analysed.

The Bayesian wavelet networks appear to be very accurate on the limited number of data sets we have analysed. We have shown how the Bayesian framework incorporates an automatic 'Occam factor', elicited by the presence of the Bayes Factor, in the acceptance probability of the Markov chain Monte Carlo that we use to derive posterior probability distributions. This leads to models that adapt their complexity, and in this case dimension, to match the problem at hand.

Appendix 1

Suppose we have a wavelet network parameterised by (M, w, σ^2) where M contains the number of basis functions and the dilation and translation parameters $M = \{\mu, D\}$, w is a vector of output weights and σ^2 is the variance of the Gaussian noise term assumed to have corrupted the data $\mathcal{D} = \{Y, X\}$, where Y and X are dependent (output) and independent (input) variables respectively.

MCMC methods can be used to estimate the posterior probability of (M, w, σ^2) by drawing samples in the appropriate proportions, see section 3.1. To do this we need a proposal density $S(\{M, w, \sigma^2\}, \{M', w', \sigma'^2\})$ to take us from state $\{M, x, \sigma^2\}$ to $\{M', x', \sigma'^2\}$, which from now on we simply write this proposal as $S(M', w', \sigma'^2)$ and the reverse as $S(M, w, \sigma^2)$. This proposal term can be split to give

$$S(M', w', \sigma'^2) = S(M')S(w', \sigma'^2 \mid M').$$

Following section 3.1, the proposed move is now accepted with probability

$$\alpha = \min\left\{1, \frac{L(M', w', \sigma'^2)p(M', w', \sigma'^2)S(M)S(w, \sigma^2 \mid M)}{L(M, w, \sigma^2)p(M, w, \sigma^2)S(M')S(w', \sigma'^2 \mid M')}\right\}, \tag{22}$$

where $L(M, w, \sigma^2)$ is the likelihood function $p(\mathcal{D} \mid M, w, \sigma^2)$, and $p(M, w, \sigma^2)$ is the prior density.

Suppose we propose new values for w and σ^2 using the full conditional densities[4]

$$S(w, \sigma^2 \mid M, \mathcal{D}) = p(w, \sigma^2 \mid M, \mathcal{D}).$$

Now, writing

$$p(w, \sigma^2 \mid M, \mathcal{D}) = \frac{p(\mathcal{D} \mid w, \sigma^2, M)p(w, \sigma^2 \mid M)}{p(\mathcal{D} \mid M)}, \tag{23}$$

and substituting (23) into (22) and cancelling leaves

$$\begin{aligned} \alpha &= \min\left\{1, \frac{p(\mathcal{D} \mid M')}{p(\mathcal{D} \mid M)} \frac{p(M')}{p(M)} \frac{S(M)}{S(M')}\right\} \\ &= \min\{1, \text{Bayes Factor} \times \text{Prior Odds} \times \text{Proposal Odds}\}. \end{aligned} \tag{24}$$

We can derive an analytical solution to the Bayes Factor by noting that it is the ratio of prior predictives that appear as normalising constants in (23). These can be written as

$$p(\mathcal{D} \mid M) = \int\int p(\mathcal{D} \mid w, \sigma^2, M)p(w, \sigma^2 \mid M) dw d\sigma^2, \tag{25}$$

and assuming a conjugate Normal-Inverse Gamma prior so that

$$p(w, \sigma^{-2} \mid M) = N(w \mid 0, \lambda I)\text{Gamma}(\sigma^{-2} \mid \alpha, \beta),$$

then it is easily shown that an analytical solution to (25) exists namely

$$p(\mathcal{D} \mid M) = \frac{|V|^{1/2}\beta^{\alpha/2}\Gamma((\alpha+n)/2)}{|\lambda^{-1}I|^{1/2}\pi^{n/2}\Gamma(\alpha/2)} a^{-(\alpha+n)/2}, \tag{26}$$

[4]For the wavelet model assuming a normally distributed error term the full conditional density for w and σ^2 is simply a Normal-Inverse Gamma distribution (Bernardo and Smith, 1994)

where $|\cdot|$ is the determinant, n is the number of data points, $\Gamma()$ is the Gamma integral and

$$\begin{aligned} V &= (\Psi'\Psi + \lambda I)^{-1} \\ a &= \beta + Y'Y - m'V^{-1}m, \end{aligned}$$

and where Ψ is the $n \times k$ matrix of responses of the wavelet basis layer and linear terms over the data set $\mathcal{D}$ and

$$\begin{aligned} m &= (\Psi'\Psi + \lambda I)^{-1}(\Psi'Y) \\ &= V(\Psi'Y). \end{aligned}$$

The Bayes Factor is then given as a ratio of (25) for the proposed state, $\{M', w'\}$, and the current state, $\{M, w\}$ to give

$$\frac{p(\mathcal{D} \mid M')}{p(\mathcal{D} \mid M)} = \frac{|\lambda^{-1}I|^{1/2}|V'|^{1/2}}{|\lambda^{-1}I'|^{1/2}|V|^{1/2}} \left(\frac{a}{a'}\right)^{(\alpha+n)/2}. \tag{27}$$

Appendix 2.

The Bayesian wavelet modelling procedure using the reversible jump MCMC algorithm used can be written in pseudo-code as follows:

First set the proposal probabilities: a; r; d; t. Here a is the probability of proposing to add a basis function to the network, r is the probability of attempting to remove a basis function, d is the probability of proposing to alter the dilation parameters D in (6) and t is the probability of proposing to change the translation parameters μ in (6). For all our simulations we set $a = r = 0.1$, and $d = t = 0.4$.

Starting with one wavelet basis function

1. Draw the noise variance, σ^2, from its prior, Gamma$(\sigma^{-2} \mid 10^{-3}, 10^{-3})$.
2. Draw the output coefficients w in a Gibbs sampling step. This uses the full conditionals of w given the data, $p(w \mid \mathcal{D}) = N(w \mid (\Psi^T\Psi + \lambda I)^{-1}\Psi y, \sigma^2\Psi^T\Psi))$ where λ is the prior precision for w and Ψ is the $n \times k$ design matrix of outputs from the hidden layer of wavelet basis functions and polynomial terms. See Smith and Roberts (1993) for details of the Gibbs sampler.

3. Iterate until convergence is assumed.[5]

   (a) Draw a uniform random variable $u \sim U(0, 1)$
   (b) Propose the next state of the chain as follows
      i. If u is less than a then perform BIRTH step
      ii. Elseif u is less than $a + r$ then perform DEATH step
      iii. Elseif u is less than $a + r + d$ then perform DILATION step
      iv. Else perform TRANSLATION step.
   (c) Redraw the coefficients w, as before.
   (d) Draw a uniform random variable $u \sim U(0, 1)$
      i. If $u < Q()$, where $Q()$ is from (19), then accept the proposed state
      ii. Else set the next state to be the current state.
   (e) Draw the noise variance, σ^2, in a Gibbs sampling step using the full conditionals $\sigma^{-2} = \text{Gamma}(\sigma^{-2} \mid 10^{-3} + n/2, 10^{-3} + \psi/2)$, where n is the number of data points and

$$\begin{aligned} \psi &= Y'Y - m'Vm \\ V &= (\Psi'\Psi + \lambda I) \\ m &= V^{-1}(\Psi'Y), \end{aligned}$$

   and where Ψ is the hidden layer output as before.
   (f) Repeat

The DILATION, TRANSLATION, BIRTH and DEATH steps are simple. DILATION just selects a basis function at random and alters its dilation parameter from D_i to D_i' by

$$D_i' = \exp(\log(D_i) + \epsilon)$$

[5]This usually involves inspection of statistics from the Markov chain such as mean values of sampled parameters, see Tierney (1994) for details and heuristics.

where ϵ is drawn from $\epsilon \sim N(0, \sigma_d)$ and σ_d is determined from initial simulations[6] to give an acceptance rate of around 30%. The TRANSLATION step selects a basis function at random and resets its location to a data point drawn at random from the data set. BIRTH adds another basis function at a randomly selected point in the data set. DEATH just selects one basis at random and removes it.

The output of the algorithm is a Markov chain that has $p(M, w \mid \mathcal{D})$ as its stationary distribution. An initial portion of the chain is discarded to ensure convergence and then every m^{th} sample is used to make predictions using (6), (7) and (14). Taking every m^{th} sample reduces corelations in the chain, see Tierney (1994) for details.

[6]The jump proposals in a Metropolis-Hastings MCMC algorithm should be able to take the chain around the density of interest in a reasonable amount of time. However large values for the variance of the jump proposals will tend to find states that are rejected more often than ones with smaller values. Thus a trade off exists between traversing the state space quickly and maintaining a reasonable acceptance rate. Current heuristics suggest an acceptance rate of around 30% is desirable (Gelman *et al*, 1996).

References

Bernardo, J. and Smith, A. (1994) *Bayesian Theory.* Wiley

Cao, L., Hong, Y., Fang, H. and He. G. (1995) Predicting chaotic time series with wavelet networks. *Physica D* vol. 85, pp. 225-238.

Casdagli, M. (1989) Nonlinear prediction of chaotic time series. *Physica D*, vol. 35, pp. 335-356.

Chui, C. K., Ed., (1992) *Wavelets: A Tutorial in Theory and Applications.* New York: Academic.

Denison, D., Mallick, B., Smith, A. F. M. (1996) Bayesian M.A.R.S. *Tech. Report.* Department of Mathematics, Imperial College, London.

Draper, D. (1995) Assessment and Propogation of Model Uncertainty (with discussion). *J. Royal Stat. Soc.* B, vol. 57, pp. 45-98.

Falmer, J. D. and Sidorowich, J. J. (1987) Predicting Chaotic Time Series. *Physical Review Letters.* vol. 59, no. 8, pp. 845-848.

Gelman, A., Roberts, G. O. and Gilks, W. R. (1996) Efficient Metropolis Jumping Rules. In *Bayesian Statistics 5.* Bernardo, J. M., Berger, J. O., Dawid, A. P. and Smith, A. F. M., eds., Oxford University Press.

Green, P. J. (1995) Reversible Jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, vol. 82, pp. 711-732.

Hastie, T. J. and Tibshirani, R. J. (1990) *Generalised Additive Models.* London: Chapman & Hall.

Hastings, W. K. (1970) Monte Carlo sampling methods using Markov chains and their applications. *Biometrica*, vol. 57, pp. 97-109.

Holmes, C. and Mallick, B. (1997) Bayesian Radial Basis Functions of Variable Dimension. *Tech. Report.* Department of Mathematics, Imperial College, London.

Hwang, J., Lay, S., Maechler, R., Martin, D. and Schimert, J. (1994). Regression Modeling in Back-Propagation and Projection Pursuit Learning. *IEEE Trans. Neural Networks*, vol. 5, no. 3, pp. 342-353.

Kass, R. E. and Raftery, A. E. (1995) Bayes Factors *J. Am. Stat. Assoc.*, vol. 90, no. 430, pp. 773-795.

Kugarajah, T. and Zhang, Q. (1995) Multidimensional wavelet frames. *IEEE Trans. Neural Networks*, vol. 6, pp. 1552-1556.

Lapedes, A. and Farber, R. (1987) Nonlinear signal processing using neural networks: Prediction and signal modelling, *preprint*, Los Alamos.

Lillekjendlie, B., Kugiumtzis, D. and Christophersen, N. (1994) Chaotic time series Part II. System identification and prediction. *Modeling, identification and control*, vol. 15, no. 4, pp. 225-243.

MacKay, D. (1992a) Bayesian interpolation. *Neural Computation*, vol. 4, pp. 415-447.

MacKay, D. (1992b) A practical Bayesian framework for backpropogation networks. *Neural Computation*, vol. 4, pp. 448-472.

Mackey, M. C. and Glass, L. (1977) Oscillations and chaos in physiological control systems. *Science*, vol. 287.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. and Teller, E. (1953) Equations of state calculations by fast computing machines. *J. Chem. Pys.*, vol. 21, pp. 1087-1091.

Smith, A.F.M. and Roberts, G.O. (1993) Bayesian computation via Gibbs sampler and related Markov chain Monte Carlo methods. *J. R. Statist. Soc.* B, vol. 55, pp. 2-24.

Tarkens, F. (1981) Detecting strange attractors in turbulence. In *Dynamical Systems and Turbulence, Warwick 1990, Lecture Notes in Math. 898*, Springer, Berlin. pp. 366-381.

Tierney, L. (1994) Markov chains for exploring posterior distributions. *Ann. Statist.*, vol. 22, pp. 1701-1762.

Zhang, Q. (1997) Using Wavelet Network in Nonparametric Estimation. *IEEE Trans. Neural Networks*, vol. 8, no. 2, pp. 227-236.

Zhang, Q. and Benveniste, A. (1992) Wavelet Networks. *IEEE Trans. Neural Networks*, vol. 3, pp. 889-898.