Kingston, G. B., H. R. Maier, and M. F. Lambert
Bayesian model selection applied to artificial neural networks used for water resources modeling
Water Resources Research, 2008; 44(4):04419-01-04419-12

DOI: 10.1029/2007WR006155

Bayesian model selection applied to artificial neural networks used for water resources modelling

by

Kingston G.B., Maier H.R. and Lambert M.F.

Water Resources Research

Citation:
Kingston G.B., Maier H.R. and Lambert M.F. (2008). "Bayesian model selection applied to artificial neural networks used for water resources modelling." *Water Resources Research*, 44, W04419, doi:10.1029/2007WR006155.

A Bayesian approach to artificial neural network model selection

Kingston, G. B., H. R. Maier and M. F. Lambert

Centre for Applied Modelling in Water Engineering, School of Civil and Environmental Engineering, The University of Adelaide, Adelaide SA 5005, Australia. E-Mail: gkingsto@civeng.adelaide.edu.au

Keywords: *Artificial neural networks, model selection, Bayes factors, Markov chain Monte Carlo*

EXTENDED ABSTRACT

Artificial neural networks (ANNs) have proven to be extremely valuable tools in the field of water resources engineering. However, one of the most difficult tasks in developing an ANN is determining the optimum level of complexity required to model a given problem, as there is no formal systematic model selection method. The generalisability of an ANN, which is defined by its predictive performance on the universe of possible data, can be significantly impaired if there are too few or too many hidden nodes in the network. Therefore, for an ANN to be a valuable prediction tool, it is important that some effort is made to optimise the number of hidden nodes.

This paper presents a Bayesian model selection (BMS) method for ANNs that provides an objective approach for comparing models of varying complexity in order to select the most appropriate ANN structure. Given a set of competing models $\mathcal{H}_1, \ldots, \mathcal{H}_H$, BMS is used to compare the posterior probability that each model $\mathcal{H}_i$ is the true data generating function, given a set of observed data $\mathbf{y}$. This probability is also known as the *evidence* of a model and the ratio of two competing models' evidence values, known as the *Bayes' factor*, can be used to rank the competing models in terms of the relative evidence in support of each model.

For ANNs (and other complex models), the evidence of a model $p(\mathcal{H}|\mathbf{y})$ is analytically intractable and, consequently, alternative methods are required to estimate these probabilities for the competing models. One such method involves the use of Markov chain Monte Carlo (MCMC) simulations from the posterior weight distribution $p(\mathbf{w}|\mathbf{y}, \mathcal{H})$ to approximate the evidence. It has already been shown that there are numerous benefits to estimating the posterior distribution of ANN weights with MCMC methods; therefore, the proposed BMS approach is based on such an approximation of $p(\mathbf{y}|\mathcal{H})$, as this only requires a simple additional step after sampling from $p(\mathbf{w}|\mathbf{y}, \mathcal{H})$. Furthermore, the weight distributions obtained from the MCMC simulation provide a useful check of the accuracy to the approximated Bayes' factors. A problem associated with the use of posterior simulations to estimate a model's evidence is that the approximation may be sensitive to factors associated with the MCMC simulation. Therefore, the proposed BMS method for ANNs incorporates a further check of the accuracy of the computed Bayes' factors by inspecting the marginal posterior distributions of the hidden-to-output layer weights, which indicate whether all of the hidden nodes in the model are necessary. The fact that this check is available is one of the greatest advantages of the proposed approach over conventional model selection methods, which do not provide such a test and instead rely on the modeller's subjective choice of selection criterion.

The aim of model selection is to enable generalisation to new cases. Therefore, in the case study presented in this paper, the performance of the proposed BMS method was assessed in comparison to the performance of conventional ANN selection methods on data outside the domain of the training data. This case study, which involves forecasting salinity concentrations in the River Murray at Murray Bridge, South Australia, 14 days in advance, was chosen as it had been shown previously that, if an ANN was trained on the first half of the available data, it would be required to extrapolate in some cases when applied to the second half of the available data set. In this case study, the proposed BMS framework for ANNs was shown to be more successful than conventional model selection methods in selecting an ANN that could approximate the relationship contained in the training data and generalise to new cases outside the domain of those used for training. The Bayes' factors calculated were useful for obtaining an initial guide to the most appropriate model; however, the final step involving inspection of marginal posterior hidden-to-output weight distributions was necessary for the final selection of the optimum number of hidden nodes. The model selected using the proposed BMS approach not only had the best generalisability, but was also more parsimonious than the models selected using conventional methods and required considerably less time for training.

1. INTRODUCTION

Over the past 10-15 years, artificial neural networks (ANNs) have proven to be extremely valuable tools in the field of water resources engineering (Maier and Dandy, 2000). These nonparametric empirical models require few assumptions about the system under study, making them well suited to modelling the poorly understood processes that occur within water resource systems. One of the main advantages of an ANN is its ability to capture and learn functional relationships contained within a sample of training (calibration) data, such that it can then generalise and make predictions about the population from which the data came. This generalisation ability, or generalisability, is measured by the predictive performance of the trained ANN on cases not contained in the training data set.

With an ANN, modelling the response of a system y, given a set of predictor variables $\mathbf{x}$, involves finding an appropriate relationship $y = f(\mathbf{x}, \mathbf{w})$, where $f(\cdot)$ is the function described by the ANN and $\mathbf{w}$ is a vector of connection and bias weights (free parameters) that characterise the data generating relationship. For the types of problems modelled by ANNs, the form of $f(\cdot)$ is generally not known; however, it is often complex and nonlinear, involving hundreds or even thousands of weights. Ideally, achieving good generalisability would involve selecting a network of optimal complexity, where optimality is defined as the smallest network that adequately captures the underlying relationship, and then estimating its weights from the training data. However, determining the optimal complexity is one of the most difficult tasks in designing an ANN, as there exists no systematic model selection method to ensure the optimal network will be chosen (Qi and Zhang, 2001).

The flexibility in ANN complexity selection primarily lies in selecting the appropriate number of hidden layer nodes, which determine the number of weights in the model. In doing this, a balance is required between having too few hidden nodes such that there are insufficient degrees of freedom to adequately capture the underlying relationship, and having too many hidden nodes such that the model fits to noise in the individual data points rather than the general trend underlying the data as a whole. The latter case is referred to as overfitting, which is often difficult to detect but can significantly impair the generalisability of an ANN. To prevent overfitting, cross validation during training is often used (Maier and Dandy, 2000); however, apart from being more susceptible to overfitting, large ANNs with many hidden nodes are inefficient to calibrate, the parameters and resulting predictions have a higher degree of associated uncertainty and it is more difficult to extract information about the modelled function from the parameters (Reed, 1993). Therefore, selection of the minimum number of necessary hidden nodes can be crucial to the performance of an ANN and its value as a prediction tool.

The most commonly used method for selecting the number of hidden layer nodes is by trial-and-error (Maier and Dandy, 2000), where a number of networks are trained, while the number of hidden nodes is systematically increased or decreased until the network with the best generalisability is found. The generalisability of an ANN can be estimated by evaluating its 'out-of-sample' performance on an independent test data set using some 'goodness of fit' measure, such as the root mean squared error (RMSE) or the coefficient of determination (r^2). However, this may not be practical if there are only limited available data, since the test data cannot be used for training. Furthermore, if the test data are not a representative subset, the evaluation may be biased. Alternatively, information criteria which measure 'in-sample' fit (i.e. fit to the training data) but penalise model complexity, such as Akaike's information criterion (AIC) or the Bayesian information criterion (BIC), can be used to estimate an ANN's generalisability. However, it has been suggested that these criteria overly penalise ANN complexity (Qi and Zhang, 2001). A limitation of both of these in-sample and out-of-sample model selection methods is that they assume a globally optimal solution has been found during training. In reality, ANNs may be sensitive to initial conditions and training parameters and can often become trapped in local minima (Reed, 1993). Therefore, evaluation of the model structure may be incorrectly biased by the weights obtained.

The aim of this paper is to present a model selection method for ANNs that provides an objective approach for comparing models of varying complexity in order to select the most appropriate ANN structure. The proposed framework is based on Bayesian methodology and Markov chain Monte Carlo (MCMC) sampling and uses sampled weight vectors from the posterior weight distribution to estimate the *evidence* in support of a given ANN structure. An advantage of the Bayesian approach is that it is based on the posterior weight distribution and, thus, is not reliant on finding a single optimum weight vector. Furthermore, the evidence of a model is evaluated using only the training data and therefore, there is no need to use an independent test set. A real-world water resources case study is used to demonstrate the proposed method, where it is compared with conventional model selection methods.

2. METHODS

2.1 Bayesian Methodology

The concept behind the Bayesian modelling framework is Bayes' theorem, which states that any prior beliefs regarding an uncertain quantity are updated, based on new information, to yield a posterior probability of the unknown quantity. In terms of an ANN, Bayes' theorem can be used to estimate the posterior distribution of the network weights $\mathbf{w} = \{w_1, \ldots, w_d\}$ given a set of N target data $\mathbf{y} = \{y_1, \ldots, y_N\}$ and an assumed model structure $\mathcal{H}$ as follows:

$$p(\mathbf{w}|\mathbf{y}, \mathcal{H}) = \frac{p(\mathbf{y}|\mathbf{w}, \mathcal{H})p(\mathbf{w}|\mathcal{H})}{p(\mathbf{y}|\mathcal{H}) = \int p(\mathbf{y}|\mathbf{w}, \mathcal{H})p(\mathbf{w}|\mathcal{H})d\mathbf{w}} \tag{1}$$

In this equation, $p(\mathbf{w}|\mathcal{H})$ is the *prior* distribution, which describes any knowledge of the weight values before observing the data; $p(\mathbf{y}|\mathbf{w}, \mathcal{H})$ is known as the *likelihood* function and is obtained by comparing the observed data $\mathbf{y}$ to the model outputs $\hat{\mathbf{y}}$. This is the function through which the prior knowledge of $\mathbf{w}$ is updated by the data. The denominator $p(\mathbf{y}|\mathcal{H})$ is a normalising constant known as the marginal likelihood, or *evidence*, of the model. When estimating the posterior of the weights, it is common to ignore this term, instead writing (1) as the proportionality $p(\mathbf{w}|\mathbf{y}, \mathcal{H}) \propto p(\mathbf{y}|\mathbf{w}, \mathcal{H})p(\mathbf{w}|\mathcal{H})$. However, when using Bayesian methods for model selection, the model evidence becomes very important.

2.2 Bayesian Model Selection (BMS)

Given a set of H competing models, Bayes' theorem can be rewritten to infer the posterior probability that each model $\mathcal{H}_i$, where $i = 1, \ldots, H$, is the "true" model of the system given the observed data, as follows:

$$p(\mathcal{H}_i|\mathbf{y}) = \frac{p(\mathbf{y}|\mathcal{H}_i)p(\mathcal{H}_i)}{p(\mathbf{y}) = \sum_{j=1}^{H} p(\mathbf{y}|\mathcal{H}_j)p(\mathcal{H}_j)} \tag{2}$$

where $p(\mathcal{H}_i)$ is the prior probability assigned to $\mathcal{H}_i$ and $p(\mathbf{y}|\mathcal{H}_i)$ is the evidence of the model, which is the denominator in (1). It is unlikely that any model will actually be the "true" model of the system; however, the Bayes' approach enables the relative merits of the competing models to be compared in an objective manner.

It is generally assumed that the prior probabilities assigned to the different models are approximately equal, as a model thought to be highly implausible would not even be considered in the comparison. Furthermore, even without a prior preference for simple models, the evidence of a model automatically favours simple theories, as discussed in MacKay (1995). Therefore, (2) can be simplified to:

$$p(\mathcal{H}_i|\mathbf{y}) = \frac{p(\mathbf{y}|\mathcal{H}_i)}{\sum_{j=1}^{H} p(\mathbf{y}|\mathcal{H}_j)} \propto p(\mathbf{y}|\mathcal{H}_i) \tag{3}$$

which states that the relative probabilities of the competing models can be compared based on their evidence. The ratio of two models' posterior probabilities is called the *Bayes' factor* BF, which, when assuming equal prior probabilities, is defined by:

$$BF_{2,1} = \frac{p(\mathcal{H}_2|\mathbf{y})}{p(\mathcal{H}_1|\mathbf{y})} = \frac{p(\mathbf{y}|\mathcal{H}_2)}{p(\mathbf{y}|\mathcal{H}_1)} \tag{4}$$

In order to interpret the information provided by $BF_{2,1}$ in terms of the evidence against model $\mathcal{H}_1$ in favour of model $\mathcal{H}_2$, Kass and Raftery (1995) suggest using the interpretive scale given in Table 1. The problem of BMS then becomes one of

Table 1. Bayes' factor interpretive scale

$2\log_e BF_{2,1}$	Evidence against $\mathcal{H}_1$
0 to 2	Weak
2 to 6	Positive
6 to 10	Strong
> 10	Very strong

estimating the evidence of each competing model and ranking the models according to their Bayes' factors. Nevertheless, this task is far from trivial. As shown in (1), the evidence can be evaluated by the integral $p(\mathbf{y}|\mathcal{H}) = \int p(\mathbf{y}|\mathbf{w}, \mathcal{H})p(\mathbf{w}|\mathcal{H})d\mathbf{w}$; however, except for the simplest of models, this integral is analytically intractable. Therefore, alternative methods are needed to estimate $p(\mathbf{y}|\mathcal{H})$.

2.3 Proposed BMS Framework

In order to estimate $p(\mathbf{y}|\mathcal{H})$, (1) can be rearranged as follows:

$$p(\mathbf{y}|\mathcal{H}) = \frac{p(\mathbf{y}|\mathbf{w}, \mathcal{H})p(\mathbf{w}|\mathcal{H})}{p(\mathbf{w}|\mathbf{y}, \mathcal{H})} \tag{5}$$

However, for ANNs (and other complex models), direct evaluation of this equation is impossible, as the posterior weight distribution $p(\mathbf{w}|\mathbf{y}, \mathcal{H})$ is analytically intractable. Recently, Markov chain Monte Carlo (MCMC) methods for simulating observations from posterior distributions have increased in popularity. As discussed in Kingston et al. (2005), there are numerous benefits to estimating the posterior distribution of ANN weights $p(\mathbf{w}|\mathbf{y}, \mathcal{H})$ with MCMC methods. Therefore, the proposed BMS approach is based on approximating $p(\mathbf{y}|\mathcal{H})$ using MCMC posterior simulations, as this only requires a simple additional step after sampling from $p(\mathbf{w}|\mathbf{y}, \mathcal{H})$.

Furthermore, the weight distributions obtained from the MCMC simulation may provide a useful check for the accuracy to the approximated Bayes' factors.

2.3.1 MCMC Sampling of the Posterior Weights

The first step in the proposed framework involves selecting an appropriate likelihood function and prior weight distribution. Assuming that the residuals between the observed data and the model outputs are normally and independently distributed with zero mean and constant variance σ^2, the likelihood function is given by:

$$p(\mathbf{y}|\mathbf{w},\sigma^2,\mathcal{H}) = \frac{1}{\sqrt{2\Pi\sigma^2}}\prod_{i=1}^{N}\exp\left\{-\frac{[y_i - f(\mathbf{x}_i,\mathbf{w})]^2}{2\sigma^2}\right\} \quad (6)$$

In this study, a wide uniform prior on the range [-100,100] was assumed for each weight in order to specify an equal probability of a weight taking on positive or negative values, but an otherwise lack of prior knowledge about the weights.

As the likelihood function given by (6) depends not only on the value of $\mathbf{w}$, but also on the value of the variance σ^2, a two-step MCMC procedure was used in this study to sample both $\mathbf{w}$ and σ^2 from the posterior distribution. This involved the use of the two simplest MCMC algorithms: the Gibbs sampler and the Metropolis algorithm. In the first step of this procedure, the variance parameter σ^2 is held constant while the weights $\mathbf{w}$ are sampled from the distribution:

$$p(\mathbf{w}|\sigma^2,\mathbf{y},\mathcal{H}) \propto p(\mathbf{y}|\mathbf{w},\sigma^2,\mathcal{H})p(\mathbf{w},\mathcal{H}) \quad (7)$$

using a Metropolis sampling step. As it is generally difficult to sample from $p(\mathbf{w}|\sigma^2,\mathbf{y},\mathcal{H})$ directly, the Metropolis algorithm makes use of a simpler, symmetrical distribution $Q(\mathbf{w}^*|\mathbf{w}_t)$ (often a multinormal distribution with mean $\mathbf{w}_t$), known as the 'proposal' distribution, to generate candidate weight vectors $\mathbf{w}^*$ based on the current weight vector $\mathbf{w}_t$, thus forming a random walk Markov chain within the weight space. An adaptive acceptance-rejection criterion is employed such that this sequence continually adapts to the posterior distribution of the weights. This works by only accepting the candidate weight state according to the probability α, given by:

$$\alpha(\mathbf{w}^*|\mathbf{w}_t) = \min\left[\frac{p(\mathbf{y}|\mathbf{w}^*,\mathcal{H})p(\mathbf{w}^*|\mathcal{H})}{p(\mathbf{y}|\mathbf{w}_t,\mathcal{H})p(\mathbf{w}_t|\mathcal{H})},1\right] \quad (8)$$

If $\mathbf{w}^*$ is accepted, $\mathbf{w}_{t+1}$ is set equal to $\mathbf{w}^*$, otherwise $\mathbf{w}_{t+1} = \mathbf{w}_t$ and the process is repeated. In this study, rather than using the straight Metropolis algorithm, a variation developed by Haario et al. (2001) called the adaptive Metropolis (AM) algorithm was used, as it has been found to have a number of advantages over other variants of the Metropolis algorithm in terms of efficiency and ease of use (Marshall et al., 2004). The AM algorithm was developed to overcome the problems experienced using the straight Metropolis algorithm associated with selecting an appropriate covariance for the proposal distribution. In this algorithm, the covariance of the proposal distribution is updated at each iteration based on all previous states of the weight vector, ensuring that information gained about the proposal distribution throughout the simulation is used to increase the efficiency of the algorithm and improve the convergence rate.

In the second step of the MCMC procedure, the weights are held constant while σ^2 is sampled from the full conditional distribution:

$$p(\sigma^2|\mathbf{w},\mathbf{y},\mathcal{H}) \propto p(\mathbf{y}|\mathbf{w},\sigma^2,\mathcal{H})p(\sigma^2) \quad (9)$$

using the Gibbs sampler. To enable straightforward sampling from $p(\sigma^2|\mathbf{w},\mathbf{y})$, a noninformative conjugate inverse chi-squared prior $\sigma^2 \sim \chi^{-2}(0.1, 0.01)$ was assumed. Given sufficient iterations, the sampled sequences should converge to a stationary distribution. From this point onwards, it can be considered that the sampled parameters are generated from the posterior distribution and can be used to a predictive distribution for each given input pattern.

2.3.2 Computation of Evidence

There are a number of methods available for approximating the evidence of a model using posterior simulations (see DiCiccio et al. (1997)). In this study, the framework proposed by Chib and Jeliazkov (2001) was used due to its simplicity and ease of programming. By taking the logarithm of (5) at some fixed point $\hat{\mathbf{w}}$, the following expression is obtained:

$$\begin{aligned}\log p(\mathbf{y}|\mathcal{H}) &= \log p(\mathbf{y}|\hat{\mathbf{w}},\mathcal{H}) + \log p(\hat{\mathbf{w}}|\mathcal{H}) \\ &\quad - \log p(\hat{\mathbf{w}}|\mathbf{y},\mathcal{H}) \end{aligned} \quad (10)$$

Thus, if $\hat{\mathbf{w}}$ is a sampled weight vector obtained using the above MCMC procedure, the only unknown in this equation is $\log p(\hat{\mathbf{w}}|\mathbf{y},\mathcal{H})$. Therefore, estimation of the evidence is reduced to estimating the posterior weight density at a single point $\hat{\mathbf{w}}$. Chib and Jeliazkov (2001) do this using the following equation:

$$p(\hat{\mathbf{w}}|\mathbf{y},\mathcal{H}) = \frac{K^{-1}\sum_{i=1}^{K}\alpha\left(\hat{\mathbf{w}}|\mathbf{w}^i\right)Q\left(\hat{\mathbf{w}}|\mathbf{w}^i\right)}{J^{-1}\sum_{j=1}^{J}\alpha\left(\mathbf{w}^j|\hat{\mathbf{w}}\right)} \quad (11)$$

where $\mathbf{w}^i$ are sampled draws from the posterior weight distribution, $\mathbf{w}^j$ are sampled draws from the proposal distribution $Q\left(\mathbf{w}^i|\hat{\mathbf{w}}\right)$ and $\alpha(\cdot)$ is given by (8). Chib and Jeliazkov (2001) note that while the choice of $\hat{\mathbf{w}}$ is arbitrary, for estimation efficiency it

is appropriate to choose a point that has high posterior density. Therefore, in this study, the median of the posterior distribution was chosen. Furthermore, they state that although J and K may be different, in practice they are set to be equal.

2.3.3 Checking Bayes Factors with Posterior Weight Distributions

There may be a number of problems associated with estimates of $p(\mathbf{y}|\mathcal{H})$ based on posterior simulations, for reasons discussed in DiCiccio et al. (1997). Therefore, in this framework, it is proposed that the Bayes' factors calculated using the approximated evidence values be used as a guide for model selection, but a final check of the model rankings be carried out using the posterior weight distributions. If the marginal posterior distribution of a hidden-to-output layer weight includes the value zero, this suggests that the associated hidden node may be pruned from the network without affecting model performance. If there are more than one hidden-to-output layer weights with marginal posterior distributions that include zero, scatter plots of pairs of these weights should be inspected to determine whether the joint distribution of the weights passes through the origin (0,0), which would indicate that both weights in the pair may be pruned.

3. CASE STUDY

The aim of model selection is to enable generalisation to new cases. In this case study, the performance of the proposed BMS method is compared to that of conventional ANN selection methods on data sampled from a different domain of the data-generating distribution than the training data. In other words, the model selection methods are considered with respect to *extrapolations* or *novel predictions*. The case study chosen for this was that of forecasting salinity concentrations in the River Murray at Murray Bridge, South Australia, 14 days in advance. This case study was also investigated by Bowden et al. (2002, 2005), who used approximately half of the available data (from 1 December 1986 to 30 June 1992) to develop an ANN, while the remaining data (from 1 July 1992 to 1 April 1998) were reserved to simulate a real-time forecasting situation using the developed ANN. By clustering the data, they identified that the data set used to perform the real-time simulation contained two regions of uncharacteristic data that were outside the domain of the data used to develop the model, thus it was known that the model would have to extrapolate in these regions.

Similar to Bowden et al. (2002, 2005), data from from 1 December 1986 to 30 June 1992 were used in this study for model development and data from 1 July 1992 to 1 April 1998 were used to evaluate the performance of the model selection methods. The same 13 model inputs used by Bowden et al. (2005) were also used in this study and included salinity, river level and flow data at various lags and locations in the river. The model development data (period from 1 December 1986 to 30 June 1992) were further divided into training (80%) and test (20%) data subsets, which were used for training the models and evaluating out-of-sample performance, respectively.

In the first part of the investigation, conventional ANN development and model selection methods were applied. This involved a using trial-and-error procedure to select the appropriate model structure, beginning with a network with two hidden nodes and successively increasing the number of hidden nodes in increments of two for each trial, until there was no significant improvement in in-sample performance. For each model, the hyperbolic tangent (tanh) activation function was used on the hidden layer nodes, while a linear activation function was used on the output layer. In order to decrease the networks' sensitivity to initial conditions, a genetic algorithm (GA) was used to train the models. Additionally, to prevent overfitting, the test set was used for cross validation during training. Once trained, the RMSE and r^2 criteria were used to measure out-of-sample performance on the test data, whereas AIC and BIC were used to evaluate in-sample performance, while penalising complexity, on the training data. In the second part of the investigation, the models investigated in the first part of the study were trained and compared using the proposed BMS method. For each model, the MCMC sampling algorithm was initialised with the weights obtained using the GA. The test data were not required using this method and, therefore, could have been added to the training data. However, in order to carry out a fair comparison of the model selection methods, this was not done in this study. Finally, the "optimal" models selected according to the various criteria were evaluated on the second period of data (from 1 July 1992 to 1 April 1998) to assess the ability of the different model selection approaches to select a model with the ability to generalise to novel cases.

4. RESULTS & DISCUSSION

The trial-and-error model selection approach resulted in 5 ANNs being developed, containing between 2 and 10 hidden nodes. Additional hidden nodes beyond 10 were considered not to significantly improve in-sample fit and were therefore not considered. The results obtained using the conventional model selection methods are shown in Table 2, where the "optimal" model selected according to each criterion is shown in bold font. The results in this

table demonstrate how conventional model selection methods can be inconclusive, as the in-sample performance measures (AIC and BIC) indicate that an 8 hidden node ANN is the best, whereas the out-of-sample performance measures (RMSE and r^2) indicate that a 10 hidden node ANN is optimal. As there is no general consensus on which is the better model selection criterion, it is difficult to determine which model to choose.

Table 2. Results obtained using conventional model selection methods

No. of hidden nodes	AIC	BIC	RMSE	r^2
2	12348	12507	46.46	0.936
4	12435	12748	38.79	0.955
6	11582	12050	39.99	0.955
8	**11319**	**11941**	41.89	0.949
10	11503	12278	**35.90**	**0.962**

The models developed under the BMS framework were ranked according to the computed Bayes' factors, as shown in Table 3, where the ranking of each model is given together with the log Bayes' factor calculated by comparison with the highest ranked model (i.e. the evidence against the model in favour of the Rank 1 model). It can be seen here that the strongest evidence was in favour of a 6 hidden node model. Apart from the 10 hidden node ANN, the rankings given in this table appear to be logical, with the overly simplistic and complex models being ranked the lowest. It was considered that the surprising result for the 10 hidden node model may have been the result of inappropriate convergence of the MCMC algorithm due to its large size; however, the accuracy of the calculated Bayes' factor was checked in the final step of the BMS approach, where the marginal posterior hidden-to-output layer weight distributions were inspected for the 4, 6 and 10 hidden node ANNs. These are shown in Figure 1 for the 6 hidden node model and, as it can be seen, the marginal posterior distributions for the weights exiting hidden nodes 1 and 2 both contained the value zero. A scatter plot of these weights (Figure 2) shows that the joint distribution of the weights passes through the origin, indicating that both weights may be pruned from the network. Seven of the marginal posterior hidden-to-output layer weight distributions for the 10 hidden node ANN included zero. When scatter plots of these weights were inspected, it was identified that 4 hidden nodes could be pruned as the joint distribution of weights exiting these nodes passed through the origin. None of the marginal posterior hidden-to-output layer weight distributions for the 4 hidden node ANN contained the value zero; therefore, the 4 hidden node model was selected as the optimal structure by the BMS method.

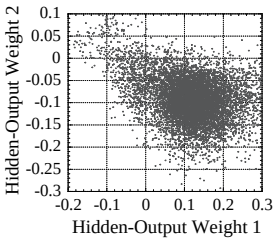

Figure 2. Scatter plot of joint weight distribution

Table 3. Bayes factor model rankings

Rank	No. of hidden nodes	$2\log_e BF_{Rank\,1,Rank\,i}$
1	**6**	–
2	4	11.84
3	10	14.89
4	2	21.63
5	8	33.82

Shown in Table 4 are the model performance results obtained on the second period of data using the "optimal" models as selected by the conventional in- and out-of-sample selection criteria and the proposed BMS approach. The RMSE and r^2 values were calculated based on the single-valued (deterministic) outputs obtained using the GA estimated weights, as well as the mean of the (Bayesian) predictive distribution obtained using the weight vectors sampled from the posterior distribution. For comparison, the results obtained by Bowden et al. (2005) using a 32 hidden node ANN are also presented. It is not surprising that these results are similar for all of the models, as cross validation was used to prevent overfitting in the deterministic case, and in the Bayesian case, the marginal posterior distributions of unnecessary weights often included zero, which effectively removed them from the network. However, overall, it can be seen that the 4 hidden node ANN trained and selected using the BMS method had the best generalisation ability on this new data set. Furthermore, this was the most parsimonious model (i.e. the simplest explanation of the system) and the time required for training was significantly less than that required for the larger models (65–74% less in the Bayesian case and 85–87% less in the deterministic case). The fact that the performance of the 4 hidden node ANN based on the posterior weight

Table 4. Performance of selected models on second period of data (from 1 July 1992 to 1 April 1998)

No. of hidden nodes	Bayesian		Deterministic	
	RMSE	r^2	RMSE	r^2
4	**78.14**	**0.885**	99.70	0.845
8	85.16	0.865	93.68	0.866
10	82.82	0.872	95.95	0.864
32 (Bowden et al., 2005)			95.0	–

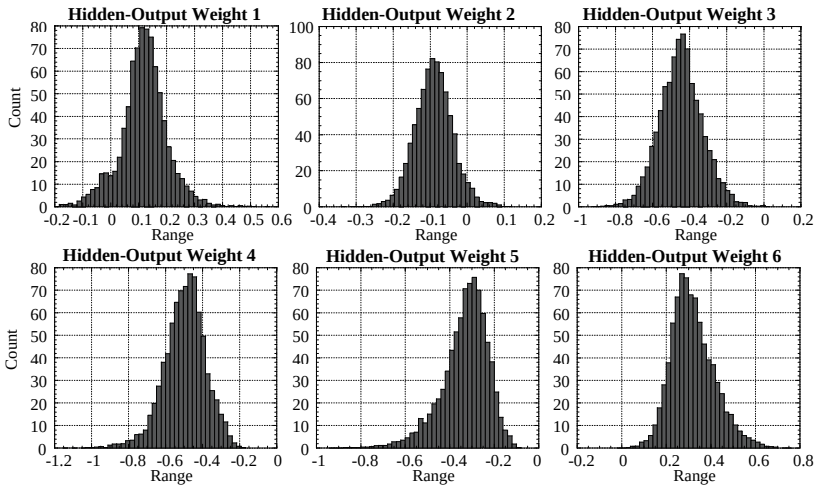

Figure 1. Marginal posterior hidden-output layer weight distributions

distribution was considerably better than that obtained using the deterministic weights, also highlights the benefits of the Bayesian weight estimation approach.

5. CONCLUSIONS

The BMS framework for ANNs presented in this paper was shown to be more successful than conventional model selection methods in selecting a parsimonious ANN that could approximate the relationship contained in the training data and generalise to new cases outside the domain of those used for training. While the Bayes' factors used in the approach provide a good initial guide for selecting the appropriate number of hidden nodes, the calculation of these values can be sensitive to factors such as convergence of the MCMC posterior simulation algorithm. Therefore, the final step in the proposed BMS framework, involving inspection of marginal posterior hidden-to-output weight distributions, is extremely important in selecting the appropriate model. The fact that this check is available is one of the greatest advantages of the proposed approach over conventional methods, which do not provide such a test and instead rely on the modeller's subjective choice of selection criterion.

6. REFERENCES

Bowden, G., H. Maier, and G. Dandy (2002), Optimal division of data for neural network models in water resources applications, *Water Resources Research*, 38(2), 1010.

Bowden, G. J., H. R. Maier, and G. C. Dandy (2005), Input determination for neural network models in water resources applications. Part 2. Case study: forecasting salinity in a river, *Journal of Hydrology*, 301(1-4), 93–107.

Chib, S., and I. Jeliazkov (2001), Marginal likelihood from the Metropolis-Hastings output, *Journal of the American Statistical Association*, 96(453), 270–281.

DiCiccio, T. J., R. E. Kass, A. Raftery, and L. Wasserman (1997), Computing bayes factors by combining simulation and asymptotic approximations, *Journal of the American Statistical Association*, *92*(439), 903–915.

Haario, H., E. Saksman, and J. Tamminen (2001), An adaptive Metropolis algorithm, *Bernoulli*, 7(2), 223–242.

Kass, R. E., and A. E. Raftery (1995), Bayes factors, *Journal of the American Statistical Association*, 90(430), 773–795.

Kingston, G. B., M. F. Lambert, and H. R. Maier (2005), Bayesian training of artificial neural networks used for water resources modeling, *Water Resources Research* (In press).

MacKay, D. J. C. (1995), Probable networks and plausible predictions - a review of practical Bayesian methods for supervised neural networks, *Network: Computation in Neural Systems*, 6(3), 469–505.

Maier, H. R., and G. C. Dandy (2000), Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications, *Environmental Modelling and Software*, 15(1), 101–124.

Marshall, L., D. Nott, and A. Sharma (2004), A comparative study of Markov chain Monte Carlo methods for conceptual rainfall-runoff modeling, *Water Resources Research*, 40(2), W02501.

Qi, M., and G. P. Zhang (2001), An investigation of model selection criteria for neural network time series forecasting, *European Journal of Operational Research*, 132(3), 666–680.

Reed, R. (1993), Pruning algorithms - a survey, *IEEE Transactions on Neural Networks*, 4(5), 740–747.