Data transformation for neural network models in water resources applications

Gavin J. Bowden, Graeme C. Dandy and Holger R. Maier

ABSTRACT

A step that should be considered when developing artificial neural network (ANN) models for water resources applications is the selection of an appropriate transformation of the data. In general, the primary motivations for data transformation are: (1) to scale the data so as to be commensurate with the transfer function in the output layer; (2) to standardise each of the variables; (3) to provide a suitable initialization of the ANN; and (4) to modify the distribution of the input variables to provide a better mapping to the outputs. In this paper, five different transformations are investigated in an attempt to improve the ANN's forecasting ability. These are: linear transformation, logarithmic transformation, histogram equalization, seasonal transformation and a transformation to normality. A case study is presented in which each of the ANN models developed using the different transformation techniques is used to forecast salinity in the River Murray at Murray Bridge (South Australia) 14 days in advance. When tested on a validation set from July 1992 to March 1998, the model developed using the linear transformation resulted in the lowest root mean squared forecasting error. This finding further strengthens the claim that the probability distribution of the data does not need to be known to develop effective ANN models. No improvement in the ANN model's forecasting ability was made using the logarithmic, seasonal and normality transformations. The model developed using histogram equalization produced good results for data within the training domain but was not robust on new patterns outside of the calibration range.

Key words | artificial neural networks, data transformation, forecasting, salinity modelling, water quality

Gavin J. Bowden (corresponding author)
Graeme C. Dandy
Holger R. Maier
Centre for Applied Modelling in Water Engineering,
School of Civil and Environmental Engineering,
The University of Adelaide,
Adelaide 5005,
Australia
Tel: +61 8 8303 5451;
Fax: +61 8 8303 4359;
E-mail: *gbowden@civeng.adelaide.edu.au*

INTRODUCTION

In recent times, there has been a rapid increase in the number of applications of artificial neural networks (ANNs) to the prediction/forecasting of water resources variables. In a review of 43 papers on the use of ANNs for the modeling of water resources variables, Maier & Dandy (2000) found that data transformations were rarely performed. In only 18 of the 43 papers were the data scaled to a range commensurate with the transfer function in the output layer using a linear transformation. In addition to this, the probability distribution of the data was not considered in any of the papers.

In the past, it has commonly been perceived in the literature that data used by ANN models do not need to be transformed. However, more recently it has been suggested that certain transformations may improve the performance of ANN models. Shi (2000) described three broad classes of data transformation. These include: (1) linear transformation; (2) statistical standardization; and (3) mathematical functions. Linear transformation is by far the most widely employed data transformation technique in ANN applications. The dataset is usually scaled to the range [0,1] or [−1,1] by using the original data range as a scalar. The objective of linear transformation is to ensure that all variables receive equal attention during the training process and that the variables are scaled in a way so as to be commensurate with the limits of the transfer functions used in the output layer. For a multilayer perceptron (MLP), it is more useful to scale the data to the range $[-1,1]$ rather than $[0,1]$. This is because the hidden nodes in a MLP each define a hyperplane and the connection weights from the input layer to the hidden layer determine the orientation of the hyperplane and the bias determines the distance of the hyperplane from the origin. When the network is initialised, it is usual to set the bias terms as small random numbers and hence, the hyperplanes pass close to the origin. Therefore, if the data are not centered around the origin, the hyperplanes may fail to pass through the data cloud and, with such a poor initialization, local minima are likely to occur (Sarle 1997).

In statistical standardization, the computation involves subtracting a measure of location, such as the mean and then dividing by a measure of scale, such as the standard deviation. As mentioned previously, any scaling that sets the measure of central tendency to zero will be beneficial during the initialization of a MLP.

Mathematical function transformation applies a mathematical function to the data, for example, taking the logarithm of the data to stabilize the seasonality and variance (Faraway & Chatfield 1998). In statistical models, non-linear mathematical transforms, such as taking the square root or logarithm of the data, are widely used to transform the data to approximate a Gaussian distribution to minimize the effect of extreme values.

Recently, Shi (2000) proposed a new type of transformation, called distribution transformation, for transforming the inputs to an ANN model. This method transforms a stream of random data distributed on any range to uniformly distributed data points on [0,1]. Since ANNs are only useful for interpolation purposes, by transforming the input data to uniformity, a continuous and smooth mapping of the input variables to the output can be achieved. Distribution transformation requires that a distribution be fitted to each of the input variables. By using the relationship between the probability distribution function (PDF) and the cumulative distribution function (CDF), any distribution in the range can be transformed to a uniform distribution on [0,1] (Shi 2000).

In general, the primary motivations for data transformation in ANN models are to scale the data in order to be commensurate with the transfer function in the output layer, to standardize each of the variables, to provide a suitable initialization of the ANN and to modify the distribution of the input variables to provide a better mapping to the outputs. Most traditional statistical models also require that the data are normally distributed before the model coefficients can be estimated efficiently and, if this is not the case, suitable transformations to normality need to be found (Maier & Dandy 2000). Burke (1991) suggested that ANNs overcome this problem, as the probability distribution of the input data does not need to be known. However, there has been some confusion in the literature over this issue. For example, as pointed out by Fortin *et al.* (1997), if the Mean Squared Error (MSE) is used as the objective function in training the ANN, this corresponds to a maximum likelihood estimation only under the hypothesis of normal (or at least symmetrical) random shocks. In linear time series models, such as the mixed autoregressive–moving average (ARMA) model of order (p,q) (Equation 1), it is apparent that the data must be transformed to normality if the random shock component, e_t, is to be normally distributed:

$$\hat{x}_t = \sum_{k=1}^{p} a_k x_{t-k} + \sum_{m=1}^{q} b_m e_{t-m} + e_t \tag{1}$$

where $\hat{x}_t$ is the time series data, a_k and b_m are the autoregressive and moving average coefficients, respectively, and e_t is a random noise process with a mean of zero and variance σ^2. However, in non-linear models such as ANNs, it is not apparent that the data need to be transformed to normality if the random shock component, e_t, is to be normally distributed. For example, consider an ANN model with input variables $x_{t-1}, x_{t-2}, \ldots, x_{t-p}$, one hidden layer consisting of H hidden layer nodes and one output node:

$$\hat{x}_t = \phi_o \left\{ \bar{\omega}_0 + \sum_{h=1}^{H} \bar{\omega}_h \phi_h \left(\beta_0 + \sum_{j=1}^{p} \beta_{jh} x_{t-j} \right) \right\} + e_t \tag{2}$$

where ω_0 and β_0 denote the bias weights from the constant input to the output and hidden layers respectively, ω_h denotes the weights from the hidden layer to the output layer, β_{jh} denotes the weights from the input layer to the hidden layer and the errors e_t have a mean zero, variance

σ^2 and are independent across training cases. The two functions ϕ_h and ϕ_o denote the transfer functions used at the hidden and output layers, respectively. It is apparent from Equation (2) that transforming the input and output variables to normality will not necessarily guarantee a random shock component that is normally distributed. To the authors' knowledge, the effect of transforming the ANN's inputs and outputs to normality has not been investigated in the literature.

In the statistical literature, transformations or differencing is often used to transform a non-stationary process into a stationary process. It is not clear whether such a transformation would improve the results of an ANN model when the data are non-stationary. In their review paper, Maier & Dandy (2000) found that the issue of stationarity is largely ignored in papers on the application of ANNs to water resources variables.

Faraway & Chatfield (1998) developed ANN models for the well-known set of airline data and considered the effect of removing the seasonal component on the ANN's forecasting ability. Two alternative approaches were considered. In the first approach, the linear trend was removed from the data and the seasonal trend was removed by subtracting the monthly averages (model 2). In the second approach, first-order and seasonal differencing were applied to the logarithms of the data (model 3). Neither model 2 nor 3 were able to improve upon the forecasting ability of the ANN model developed using the raw data (model 1). Furthermore, the use of differencing to remove the seasonality may not be desirable as it can produce a forecast variance that increases without bound as the forecasting period increases (Stedinger 1996 personal communication).

The aim of this paper is to investigate the effect of different transformations on the performance of an ANN model for forecasting salinity within a river system. The transformations that will be investigated include:

1. Linear transformation. This is by far the most commonly employed data transformation. The distribution of the raw data is not altered but, rather, the data are rescaled to a range that is commensurate with the output layer transfer function.
2. Logarithmic transformation. This transformation is commonly used for hydrological data that are truncated at zero and have positive skewness. Taking the logarithm of the data also converts a multiplicative seasonal relationship to additive.
3. Histogram equalization transformation. The method of distribution transformation (Shi 2000) is dependent on fitting a PDF to each of the input variables at an acceptable level of significance. If the fit is poor, the transformed series will not be uniformly distributed when the CDF is used to transform the data. In many cases it is not possible to fit a distribution to the random input data at an acceptable level of significance. Therefore in this paper it is proposed to use a discrete version of distribution transformation that utilizes the histogram. This transformation is known as histogram equalization (Looney 1997) and is outlined below.
4. Seasonal standardization. In this transformation, the deterministic seasonality is removed by subtracting a seasonally varying mean and dividing by a seasonally varying standard deviation. The ANN model is then developed on the seasonally standardized data.
5. Transformation to normality. It is unclear whether transforming the input and output data to normality will improve the forecasting ability of an ANN model. Therefore, to determine the effect on the model's generalization ability, a normalizing transformation of the inputs and outputs has also been investigated.

CASE STUDY: FORECASTING SALINITY AT MURRAY BRIDGE

The real-world case study used to demonstrate the effect of different data transformation techniques is that of forecasting salinity in the River Murray at Murray Bridge, South Australia, 14 days in advance. Maier & Dandy (1996) have previously developed ANN models for this case study: hence it provides a good benchmark for testing the data transformation techniques. As input variables, Maier & Dandy (1996) used daily salinity, flow and river level data at various locations in the river for the period 1 December 1986 to 30 June 1992. Data from this period and at the same locations were also used in this study.

MODEL DEVELOPMENT

In this study, feedforward MLPs trained with the backpropagation algorithm were developed using the commercially available software package NeuralWorks Professional II/Plus (NeuralWare 1998). Unless stated otherwise, the default software parameters were used since the focus is on evaluating the data transformation techniques rather than studying the effect of varying the network's parameters. The default values were determined using the experience gained from developing backpropagation networks for a variety of applications (NeuralWare 1998).

Data division

In this paper, the main objective is to compare different data transformation techniques. To provide a fair comparison between the different models, it is important that all other modeling factors are held constant and that the models are tested and validated on data that are statistically representative of the data used in the training process. This provides the most rigorous test of a model's performance based on the data transformation method, since other sources of poor performance, such as attempting to validate the model on data outside the range used in training, are effectively eliminated.

A genetic algorithm (GA) was used to divide the data so as to minimize the statistical difference (as measured by the mean and standard deviation) between training, testing and validation data sets. Since the GA data division technique allows training, testing and validation sets to be selected that are statistically representative of the same population, a fair comparison of the data transformation techniques can be made whilst providing the most rigorous test of each method. Of the 2,005 data samples available in this case study, 1604 records (80%) were used for calibration and 401 records (20%) were used for validation. The 1604 records in the calibration set were further divided into 1283 training records (80%) and 321 testing records (20%).

The GA used for data division sorts the samples into training, testing and validation sets by using a set of random numbers. The decision variable governing the arrangement of the data samples is a random number seed, chosen to be in the range [1, 100,000]. This range was selected to provide a reasonable size search space. The GA string therefore consists of a single integer between 1 and 100,000. The random number seed controls the generation of a random sequence of numbers. The random number sequence is placed alongside the data samples and the contiguous block of data is sorted using these random numbers. In so doing, the data samples are arranged into subsets and the objective function is evaluated. The first 1283 samples are placed in the training set, the next 321 samples in the test set and the next 401 samples in the validation set. Penalty constraints are added to ensure that the maximum and minimum values of each input and output variable are included in the training set, rather than in the testing or validation sets. Training the ANN model on the extreme range of values available removes the need for the network to extrapolate and helps to ensure the best possible ANN model, given the available data.

To determine the 'fitness' of each solution an objective function is required. In this application, a suitable objective function to minimize is the sum of the absolute difference in mean and standard deviation values between each pair of the three subsets. A full description of the use of the GA for dividing data into statistically similar subsets is provided in Bowden *et al.* (2002).

Data transformation

Linear transformation

A linear transformation is simple and widely used and is typically performed by using the original data range to rescale the series to a range that is commensurate with the output transfer function (Equation 3). It should be noted that the inputs and outputs are scaled individually:

$$X^T(n,i) = \left[x(n,i) \left(\frac{x^T_{\text{high}} - x^T_{\text{low}}}{x_{\max} - x_{\min}} \right) \right] + \left(\frac{x_{\max} x^T_{\text{low}} - x_{\min} x^T_{\text{high}}}{x_{\max} - x_{\min}} \right) \quad (3)$$

where $X^T\,(n,i)$ is the ith data point of the nth transformed data series, $x(n,i)$ is the ith data point of the nth original data series, $x_{\max}$ and $x_{\min}$ are the maximum and minimum values of the original data range and x^T_{high} and x^T_{low} are the new maximum and minimum values for the transformed data series and are dependent on the transfer function in the output layer. For example, as the outputs for the hyperbolic tangent transfer function are between -1 and 1, the data are generally scaled in the range -0.9 to 0.9 or -0.8 to 0.8. For MLPs, the values are not usually scaled to the extreme limits of the transfer function, as the size of the weight updates may become extremely small with flatspots occurring during training (Maier & Dandy 2000).

Logarithmic transformation

Logarithmic transformations are commonly used in applications of hydrological modeling and are useful for time series data that are characterized by a distribution with an extended right hand tail. In such instances, a logarithmic transformation is commonly used to compress the distribution of the variable (Masters 1993). A logarithmic transformation converts multiplicative relationships to additive, which is believed to simplify and improve network training (Masters 1995).

Histogram equalization (Looney 1997)

This transformation is applicable when the original data series contain spacings that are disproportionate and do not increase or decrease monotonically across the interval. Histogram equalization transformation is non-monotonically non-linear and approximately equalizes the number of data points in each subinterval of equal length.

The first step in this procedure is to perform a linear transformation on each of the n inputs such that they are all scaled to fall into [0,1]. For each input, the range [0,1] is then partitioned into P subintervals $\{I_1, \ldots, I_P\}$ of equal length and the function $h(p)$ is assigned to the proportion of values in the pth interval. This process yields a histogram, defined by

$$h(p) = \frac{N_p}{Q} \quad (4)$$

where N_P is the number of $x(n,i)$ values that belong to the pth interval I_P and Q is the total number of data points in the series. To then approximately equalize the number of values in each subinterval, a histogram equalization transformation H_n is performed on each of the n component values. The histogram equalization transformation is given by

$$H_n(x) = \left(\frac{1}{Q} \right) \sum \{N_k : k \le p \text{ and } x \in Ip\} = \sum\nolimits_{(r=1,\text{p};x \in I_p)} h(r) \quad (5)$$

Each value of x between 0 and 1 is remapped into the sum of histogram values for all subintervals up to and including the one in which x lies. The resulting transformed series is uniformly distributed and will also be between 0 and 1 since

$$\frac{N_1}{Q} + \ldots + \frac{N_p}{Q} = 1 \quad (6)$$

Linear transformations are then applied to the inputs and outputs, in order to scale each series to a range that is commensurate with the output layer transfer function. It is important to point out that the aim of this transformation is to allow the ANN to perform a better mapping of the inputs to the output. Consequently, only the inputs are transformed. In addition, the output cannot be transformed using this method as it is not possible to back-transform the data into their original values since the transformation discretizes the data.

Seasonal standardization

Given that the case study investigated in this paper represents a real-world hydrological process, it is intuitive to treat the non-stationarity resulting from the seasonality in the data as a deterministic, rather than a statistical, phenomenon. Therefore, transformation rather than differencing is to be used to produce a stationary time series in the application investigated.

The first step in removing the seasonal component of the data is to fit a separate Fourier series to the mean and standard deviation of each time series. Using the seasonally varying mean and standard deviation, each time series can then be standardized by using the following equation:

$$X^T(t) = \frac{x(t) - \mu_s(t)}{\sigma_s(t)} \tag{7}$$

where $X^T(t)$ is the deseasonalized transformed variable, $x(t)$ is the raw data at time t, $\mu_s(t)$ is the function for the seasonally varying mean and $\sigma_s(t)$ is the function for the seasonally varying standard deviation. Both $\mu_s(t)$ and $\sigma_s(t)$ are fitted by functions of the form

$$f(t) = a_0 + \sum_{j=1}^{M} a_j \sin(\mathrm{j}w_0 t + \phi_j) \tag{8}$$

where the coefficients a_0, a_j, w_0 and $\boldsymbol{\phi}_j$ are found using the MS Excel Solver add-in and the number of sine curves M needed to accurately approximate $f(t)$ is found by adding sine curves until a significant percentage ($>99\%$) of the variance is captured by the fitted function. The resulting deseasonalized data set can then be used to develop ANN models. To convert the transformed output from the ANN model back into real-world values, the inverse of (7) is used.

Transformation to normality

Inverse transformation is commonly used in Monte Carlo Simulation to generate random distributions from the uniform distribution on [0,1]. In this paper, a new two-step transformation to normality is proposed which combines the histogram equalization transformation with the inverse transformation. The first step is to use histogram equalization to transform each of the inputs and outputs to uniformity on [0,1]. The second step is to then use a functional approximation of the inverse transformation to transform each input and output series to normality. In this paper we have used the functional approximation of the inverse transformation proposed by Beasley & Springer (1977). Finally, a linear transformation is applied to ensure that the data are scaled to a range suitable for the transfer function in the output layer.

This two-step numerical procedure has the disadvantage that the data are discretized in the histogram equalization step and therefore the ANN model's output cannot be transformed back to the exact real-world values. However, by using the raw and transformed values in the training set, it is possible to back-transform the ANN model's output via linear interpolation. An analytical transformation to normality would have been better if a successful one could have been found, but analytical transformations were found to be unsuccessful for the data used in this case study.

Determination of model inputs

Maier & Dandy (1996) found that the ANN models trained on the input set shown in Table 1 performed the best for this case study. The inputs used include forecast values of flow at Overland Corner and water level at Lock 1 Lower (i.e. the inputs with negative lags). Consequently, these 51 inputs were used for the ANN modeling. Maier & Dandy (1996) provide a detailed description of how these inputs were determined.

Determination of network architecture

Only one hidden layer is required to approximate any continuous function, given that sufficient degrees of freedom (i.e. connection weights) are provided (Cybenko 1989). Therefore, one hidden layer was utilised in this study. Empirical trials conducted by Maier & Dandy (1998) determined that 30 hidden layer nodes provided optimal performance for this case study. Consequently, a network with 51 nodes in the input layer, 30 hidden layer nodes and 1 node in the output layer was used for each of the models developed in this study.

Model validation and performance measures

To ensure that overtraining did not occur (i.e. when the network performs well on the training data, but poorly on

Table 1 | Summary of model inputs

Variable	Location	Acronym	Lags (days)	Total no.
Salinity	Murray Bridge	MBS	1, 3, . . ., 11	6
Salinity	Mannum	MAS	1, 3, . . ., 15	8
Salinity	Morgan	MOS	1, 3, . . ., 15	8
Salinity	Waikerie	WAS	1, 2, . . ., 5	5
Salinity	Loxton	LOS	1, 2, . . ., 5	5
Flow	Overland Corner	OCF	− 19, − 17, . . ., 7	14
Level	Lock 1 Lower	L1LL	− 3, − 1, . . ., 5	5
	Total number of inputs			51

independent test data), cross-validation was used as the stopping criterion. In this approach, a test set is used to determine the ANN's generalization ability. The test set root mean squared error (RMSE) is calculated every 1,000 iterations and the network with the best test results is saved during the run. After 200,000 iterations with no further improvement in the test set results, training is stopped and the network that performed best on the test set is used as the final model. Since this procedure uses the test data in the calibration phase, an independent validation set was used for all of the data transformation methods investigated in order to assess the true generalization ability of the model. The RMSE was used as the performance measure as it places greater emphasis on larger forecasting errors.

The ANN models developed using each transformation technique could be deployed in a real-world forecasting scenario. In such a case, it is likely that, in time, the models would encounter data outside of the calibration range. The model's robustness would directly depend on how accurately it could produce forecasts for such uncharacteristic data. To investigate the robustness of the ANN models developed using each transformation, a second independent validation set was used, consisting of daily data from the period 15 July 1992 to 13 March 1998. This second validation set is the same set used in Bowden *et al.* (2002) and was shown to contain regions of data outside of the calibration range. The models developed using the five transformation techniques were used to obtain 14-day forecasts for this second validation set.

RESULTS AND DISCUSSION

The RMSEs for the linear, logarithmic, histogram equalization seasonal and normality transformations are summarised in Table 2. It can be seen that the models developed using the linear and histogram equalization transformations performed significantly better than those developed using the logarithmic, seasonal and normality transformations for the training, testing and validation sets. These results are consistent with Faraway & Chatfield (1998) whose results indicated no improvement when a logarithmic transformation was used, and that predictive ability deteriorated when the seasonality was removed from the data. To investigate why the logarithmic, seasonal and normality transformations produced larger forecasting errors, sensitivity analyses were conducted. As part of the sensitivity analyses, each of the inputs is increased by 5% in turn and the change in the output caused by the change in the input is calculated. The sensitivity of each input is given by

Table 2 | RMSEs of five different transformations for the 14-day salinity forecasts at Murray Bridge

Data set	Linear transformation RMSE (EC units)	Logarithmic transformation RMSE (EC units)	Histogram equalization RMSE (EC units)	Seasonal transformation RMSE (EC units)	Transformation to normality RMSE (EC units)
Training set	32.5	54.0	24.8	54.1	54.6
Testing set	31.2	55.8	32.6	54.3	56.4
Validation set	36.5	54.2	28.3	53.4	56.6
Second validation set	54.7	74.6	61.1	115.7	89.5
Second validation set with uncharacteristic data removed	32.7	52.3	45.9	59.2	60.4

$$\text{Sensitivity} = \frac{\%\ \text{change in output}}{\%\ \text{change in input}} \times 100 \qquad (10)$$

The sensitivity analyses were performed for the models developed using the logarithmic, seasonal and normality transformations (Figures 1a–c) in order to determine the strength of the relationship between the output variable and the input variables. For comparison, a sensitivity analysis was performed for the model developed using linear transformation (Figure 1d).

In the sensitivity plots it is apparent that the most significant input for all four models is Waikerie salinity at lag 1. The second most significant variable for the logarithmic, seasonal and normality transformations is the salinity at Mannum at lag 1. For the linear transformation, the second most significant variable is the level at Lock 1 Lower forecast 3 days ahead. The most notable difference in the sensitivity plots is that the model developed using linear transformation assigned a much greater significance to the flow and river level input variables. The inability of the models developed using the logarithmic, seasonal and normality transformations to make use of the information contained in the flow and river level data may have resulted in the higher forecasting errors. One reason this may have occured for the model developed using a logarithmic transformation arises from the compressing nature of this transform. Table 3 shows the training set maximum and minimum values and the ratio of max/min for each of the variables used in this case study. It is evident that the ratio of max/min is relatively low for the salinity variables, ranging from 4.0–5.8. However, the ratio of max/min for the flow and river level data is 62.5 and 10.6, respectively. Consequently, when the logarithmic transformation is applied to these latter variables, they undergo a much greater compression and this could have distorted the information originally contained in the flow and level data.

Figure 2(a) shows the actual values of the flow at Overland Corner used in the training, testing and validation sets and the seasonal cycle fitted to the mean of these data. It can be seen that the effect of flow is largely accounted for by the seasonal cycle in its mean. So the flow itself may not be significant for a seasonal model.

The best result on the independent validation set came from the model developed using the histogram equalization transformation. This result is in accordance with Shi (2000) who found that transforming the input data to uniformity provided a smooth and continuous mapping of the input variables to the output variable and performed better than the model developed using linear transformation.

Diagnostic checks were performed on each of the five ANN models developed by examining the error residuals ($\hat{e}_t$). In general, these errors are caused by the random shock (noise) component in the data (e_t) and the inability of the model to perfectly predict the deterministic

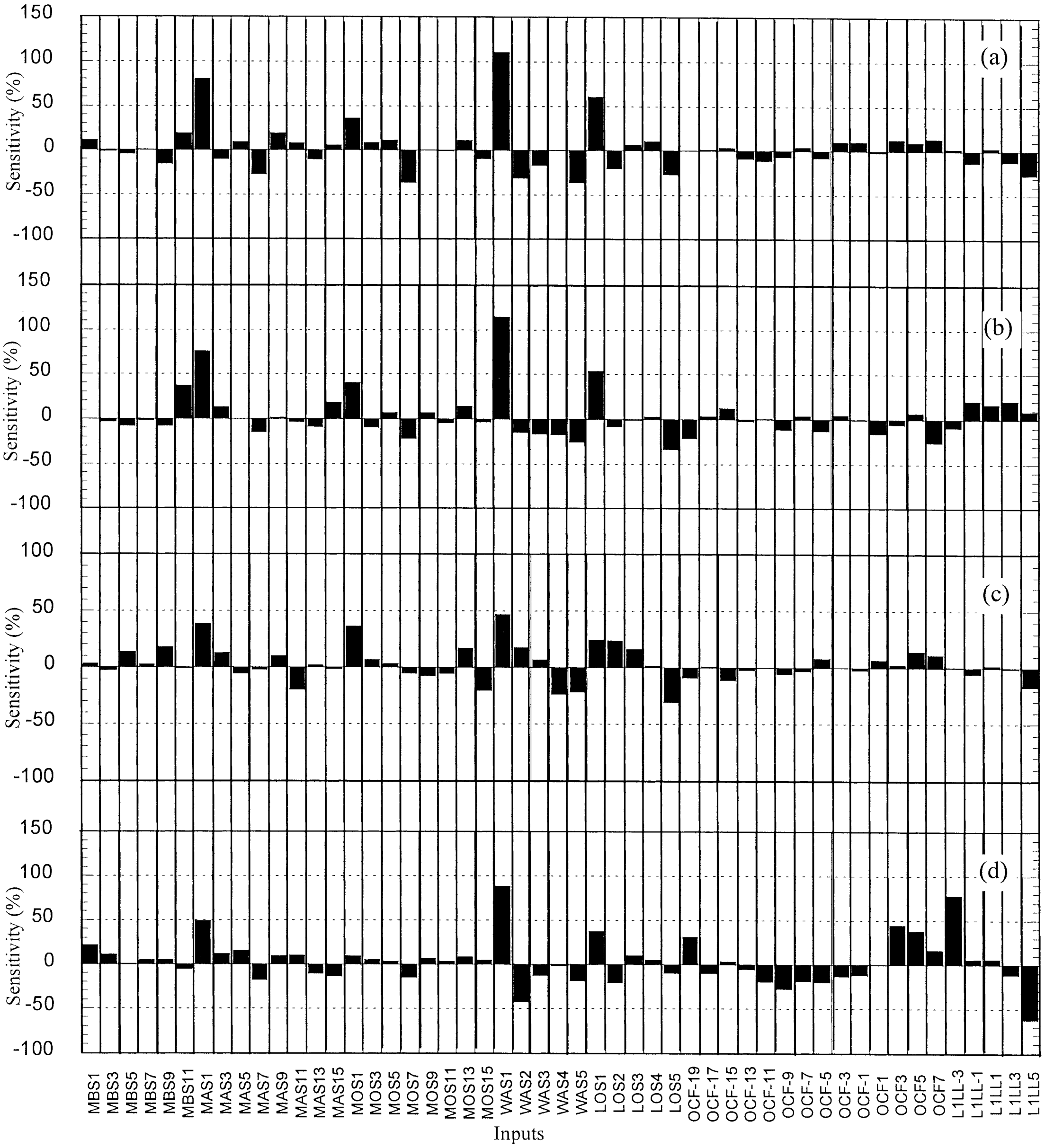

Figure 1 | Sensitivity of salinity forecasts to the 51 input variables for the model developed using (a) logarithmic transformation, (b) seasonal transformation, (c) transformation to normality and (d) linear transformation. Input variable abbreviations are given in Table 1.

components of the data. Consequently, $\hat{e}_t$ is only an estimate of the true random error e_t. If the model fits the data well, then these residuals should satisfy the following assumptions:

1. the expected value of $\hat{e}_t$ is zero,
2. the variance of $\hat{e}_t$ is a constant σ^2,
3. the errors are statistically independent of each other, and
4. the errors are normally distributed.

For each of the five transformations, the assumptions were tested by plotting histograms of the residuals (Figures 3a–e), by plotting the standardized residual

Table 3 | Maximum, minimum and ratio of max/min for each variable

Variable	Maximum	Minimum	Ratio (max/min)
MBS	1,116	261	4.3
MAS	1,075	253	4.2
MOS	1,061	183	5.8
WAS	1,021	247	4.1
LOS	907	225	4.0
OCF	110,618	1769	62.5
L1LL	5.3	0.5	10.6

versus the predicted response (Figures 4a–e) and by plotting the autocorrelation function of the residuals (Figures 5a–e). The model residuals were obtained by concatenating the training, testing and validation data sets and then chronologically ordering these data to obtain the original multivariate time series data. Each of the five models were then used to obtain forecasts for these data and the residuals were calculated.

From the histogram plots (Figures 3a–e) it can be seen that the residuals for all models were approximately normally distributed, satisfying assumption 4, with a mean of approximately zero, satisfying assumption 1. Figures 4(a–e) show that, for each of the five models, the 2,005 data points appear to be scattered randomly and largely contained within a parallel band with approximately 95% falling between -2 and 2, thereby satisfying assumption 2. In Figures 5(a–e) the 95% confidence limits for the autocorrelations are shown as estimated from statistically independent residuals. If most of the autocorrelation plot falls inside of the 95% limits, then the assumption that the residuals are statistically independent is not inconsistent with the data (assumption 2). This appears to approximately hold true for the linear and histogram equalization transformations (Figures 5a, c). However, the residuals from the models developed using

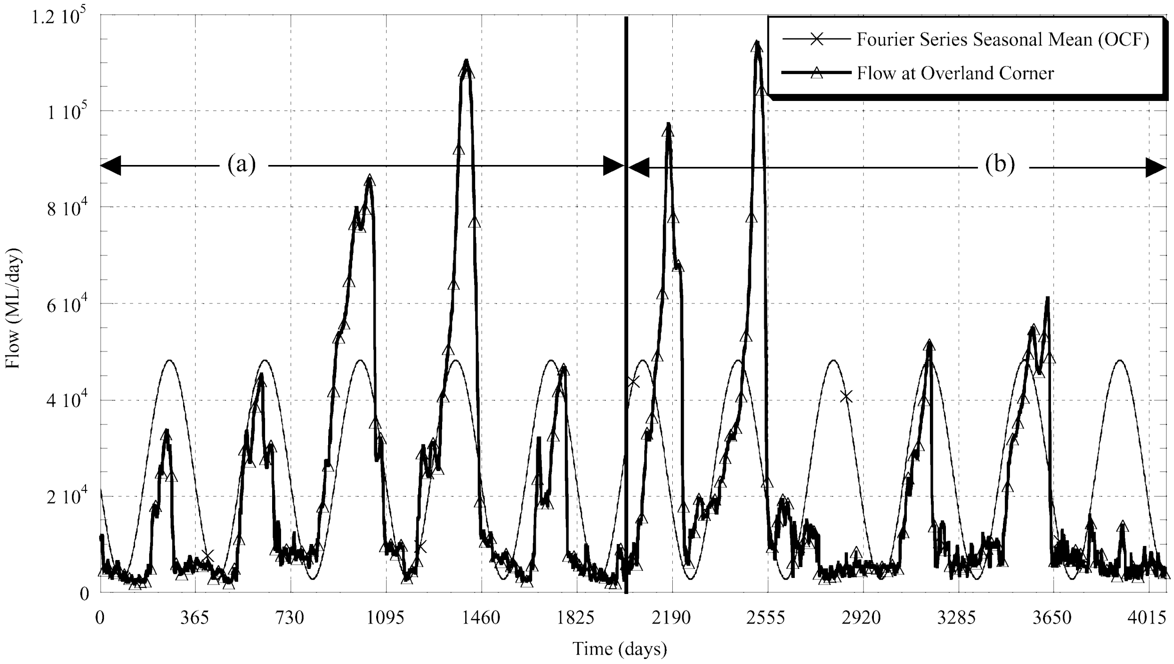

Figure 2 | Flow at Overland Corner and Fourier series seasonal mean (OCF) for (a) training, testing and validation data (December 1986–June 1992) and (b) second validation data (July 1992–March 1998).

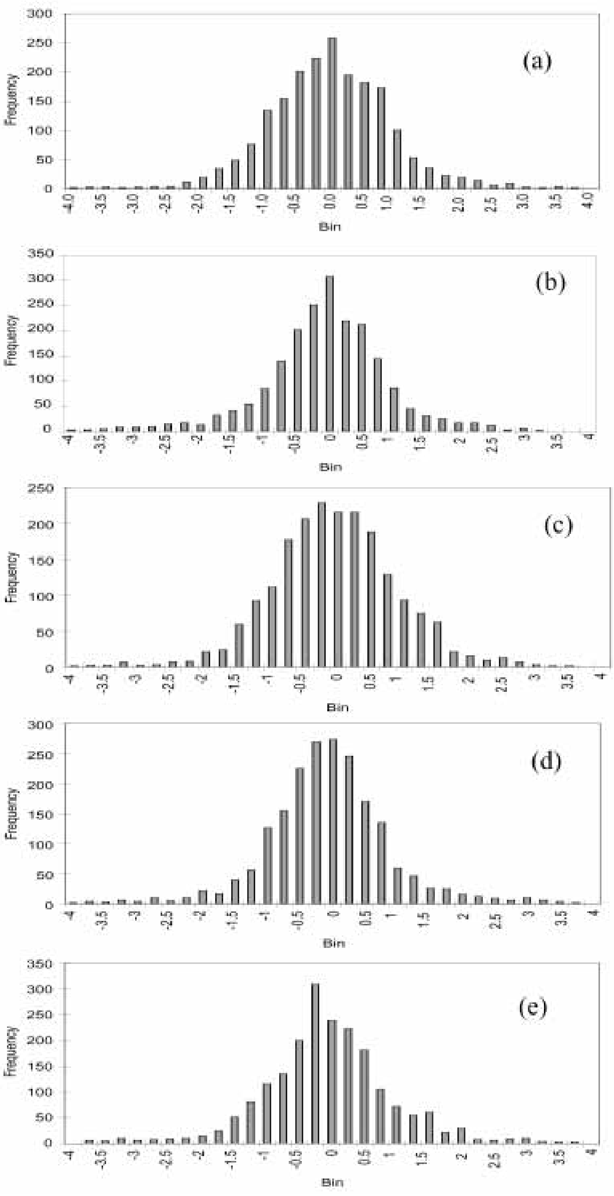

Figure 3 | Histograms of ANN residuals. (a) Linear transformation, (b) logarithmic transformation, (c) histogram equalization transformation, (d) seasonal transformation and (e) transformation to normality.

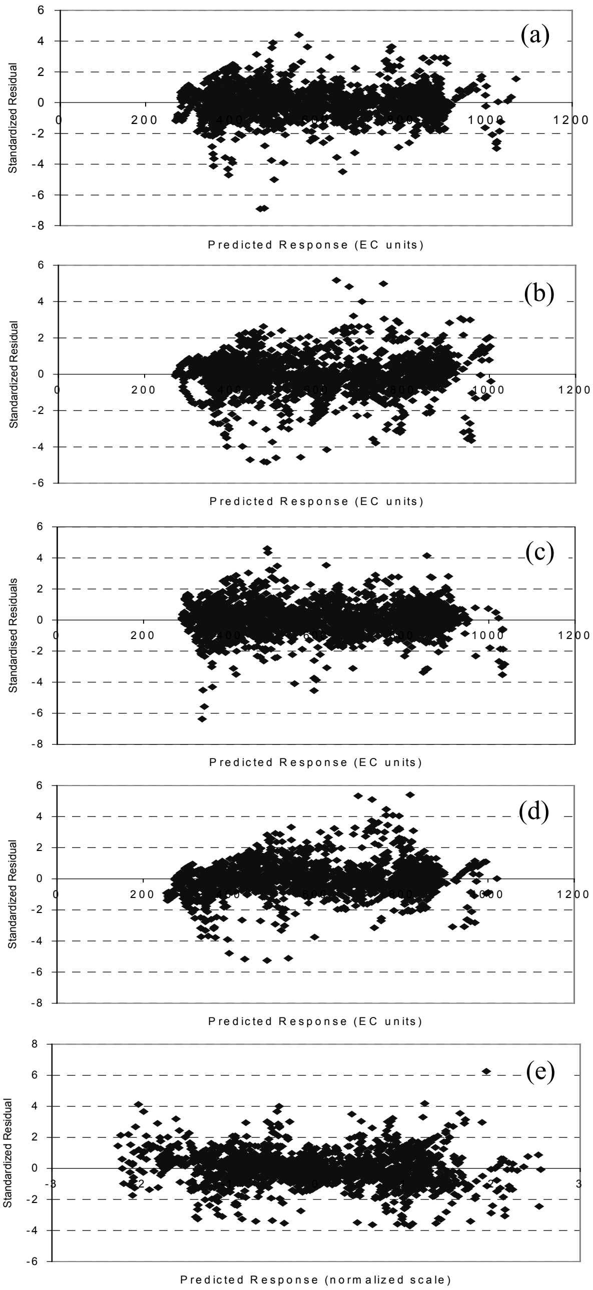

Figure 4 | Standardized residual versus predicted response (2,005 data points). (a) Linear transformation, (b) logarithmic transformation, (c) histogram equalization transformation, (d) seasonal transformation and (e) transformation to normality.

the normal, log and seasonal transformations appear to violate the statistical independence assumption.

It is interesting to note that the residuals from the model developed using the linear transformation satisfy assumptions 1–4 above. Therefore, the objective function used to calibrate the ANN model (mean square error) is justified as it maximizes the likelihood of the model under the hypothesis of normal random shocks. If the random shock component were known to be skewed,

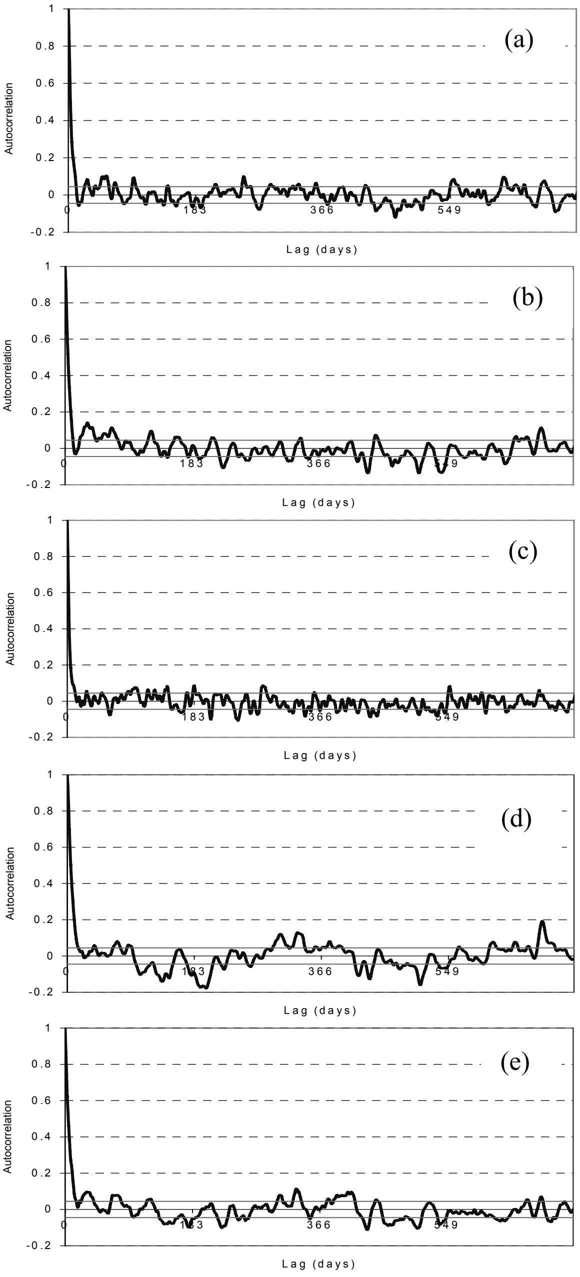

Figure 5 | Autocorrelation plots of ANN residuals. (a) Linear transformation, (b) logarithmic transformation, (c) histogram equalization transformation, (d) seasonal transformation and (e) transformation to normality.

then the objective function would need to be adjusted accordingly in order to obtain maximum likelihood estimates of the model parameters (Fortin *et al.* 1997).

In Table 2, the RMSEs for the second validation set are shown, and it is apparent that these are significantly higher than the training set RMSEs for all models. This is expected due to the presence of uncharacteristic data, i.e. new patterns that the models have not been trained on. The model developed using linear transformation appears to be the most robust, as indicated by the lowest forecasting error on the second validation set. It is surprising that the model developed using histogram equalization performed significantly worse on this set. This indicates that, although this model was able to learn the relationships in the training, testing and validation data, it was not robust when dealing with uncharacteristic data in the second validation set.

The model that gave the highest RMSE on the second validation set was the ANN developed using the seasonally transformed data. The second validation set forecasts for this model and the Fourier series for the seasonally varying mean (MBS) are shown in Figure 6. It is important to note that the Fourier function for the seasonally varying mean was developed using the training data. However, it is apparent that the seasonality in the second validation set does not follow the same fluctuations and this is highlighted by the seasonal mean moving out of phase with the actual salinity time series. This is especially noticeable in Regions 1 and 2 indicated in Figure 6. In both of these regions, the actual time series did not follow the typical seasonal cycle and, consequently, the resulting forecasts underestimated the true values. Figure 2(b) shows a plot of flow at Overland Corner and the Fourier series for the seasonally varying mean (OCF) for the second validation set data. It is evident that the high salinity events in Regions 1 and 2 of Figure 6 correspond to uncharacteristic low flow events in Figure 2(b). This highlights the need for periodic refitting of the Fourier series and retraining of the ANN model.

To diagnose regions of poor performance resulting from uncharacteristic data, Bowden *et al.* (2002) proposed a Self-Organizing Map (SOM) diagnosis procedure. In this procedure, the calibration data are clustered with the validation set using a SOM. By inspecting the resulting clusters of data, it is possible to discern uncharacteristic data, i.e. the validation data that form clusters on the Kohonen grid that contain no patterns from the training

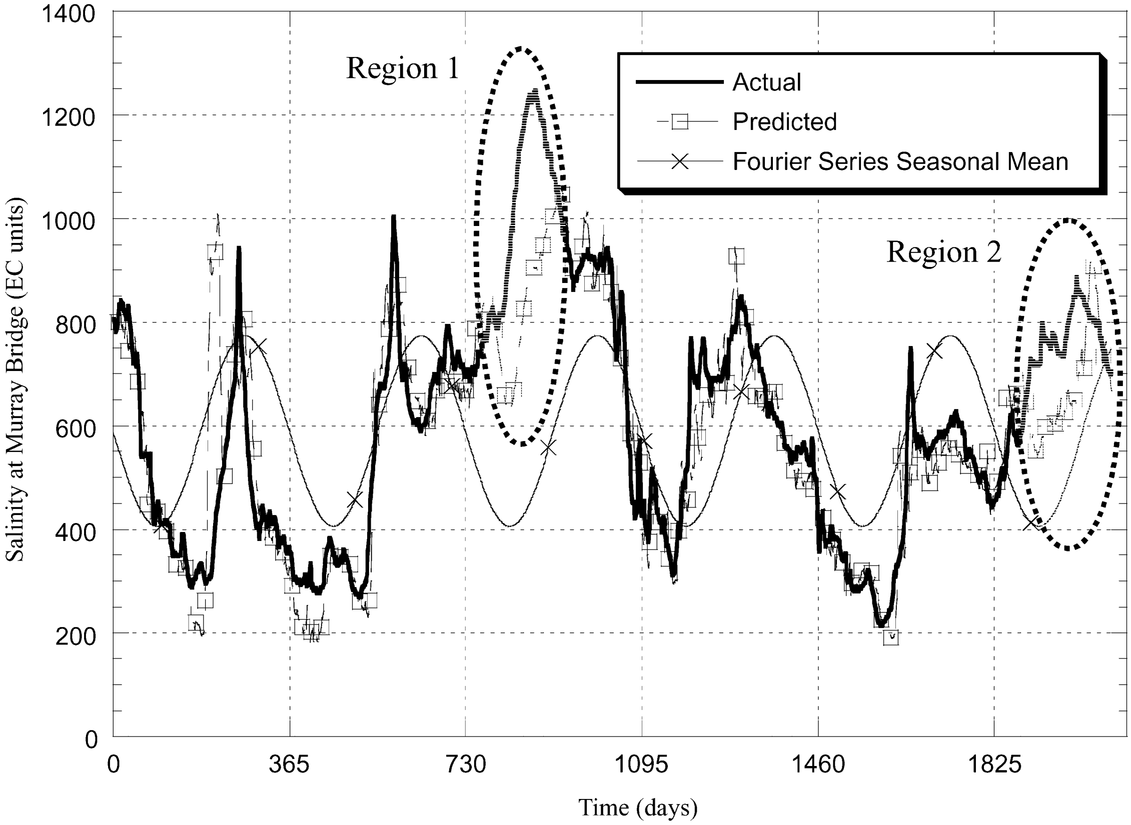

Figure 6 | Second validation set 14-day forecasts for the model developed using seasonally transformed data (July 1992–March 1998). The Fourier series seasonal mean (MBS) is also shown for the salinity at Murray Bridge.

set. The data in such clusters can be considered to lie outside of the training domain and, since ANNs are unable to extrapolate beyond the training range (Flood & Kartam 1994), poor generalization ability can be expected on these data. The SOM diagnosis procedure was conducted using the second validation set data in this study and the uncharacteristic data points were removed from the set. The models developed using the five transformations were then used to obtain forecasts for this truncated data set. The results of this experiment are shown in Table 2 and, as expected, the RMSEs were reduced quite considerably for all models. The models developed using the linear, logarithmic, seasonal and normality transformations all produced forecast errors that were very similar to their errors on the training set. The model developed using the histogram equalization data produced a forecast error that was still higher than its training error. However, in general, these results show the effectiveness of the SOM diagnosis procedure in determining the range of applicability of the ANN models.

CONCLUSIONS

In this paper, the effect of five methods for transforming data for use with ANNs in water resources applications was investigated. As ANN models are data-driven, an 'optimal' transformation can—if at all—only be defined relative to the specific application for which the ANN model is being developed. However, some general findings have come out of this study and it is hoped that, in time, results from similar studies will begin to define when such transformations may or may not be useful in water resources applications of ANN models. Taking a logarithmic transformation of the data, removing the seasonality from the data and transforming the inputs and outputs to normality were all found to give significantly larger forecasting errors than a simple linear transformation of the data. It is believed that these transformations distorted the original relationships between variables in a way that was not beneficial to the ANN learning. The model developed using data transformed by the histogram equalization method was found to perform well on data within the training domain but was not robust when applied to new data patterns. The model developed using a linear transformation gave the best results overall as this model proved the most robust on new data patterns whilst still giving a relatively low RMSE on the training, testing and validation data sets. These findings reinforce the popular belief that, when using ANNs (particularly backpropagation ANNs, which fit the data without assuming any functional form), one does not have to say that the data should be distributed in any particular way for the approach to be used. Furthermore, an analysis of the residuals produced by the linear transformation ANN model showed that the hypothesis of normal (or at least symmetrical) random shocks was valid. Consequently, the use of the mean squared error as the objective function to calibrate this model is justified as it maximizes the likelihood of the model under this assumption.

A second, independent validation set was used to test the models developed using the five transformations. This set contained data that were different to the training patterns. When these uncharacteristic data were removed from the second validation set, the RMSEs were reduced to a value similar to the training RMSE for all models except the model developed using the histogram equalization transformation.

REFERENCES

Beasley, J. D. & Springer, S. G. 1977 Algorithm AS 111: the percentage points of the normal distribution. *Appl. Stat.* **26** (1), 118–121.

Bowden, G. J., Maier, H. R. & Dandy, G. C. 2002 Optimal division of data for neural network models in water resources applications. *Wat. Res. Res.* **38** (2), 2-1–2-11 (10.1029/2001WR000266).

Burke, L. I. 1991 Introduction to artificial neural systems for pattern recognition. *Comput. Oper. Res.* **18** (2), 211–220.

Cybenko, G. 1989 Approximation by superpositions of a sigmoidal function. *Math. Control Signals Syst.* **2**, 203–314.

Faraway, J. & Chatfield, C. 1998 Time series forecasting with neural networks: a comparative study using the airline data. *Appl. Statist.* **47** (2), 231–250.

Flood, I. & Kartam, N. 1994 Neural networks in civil engineering. I: Principles and understanding. *J. Comput. Civil Engng.* **8** (2), 131–148.

Fortin, V., Ouarda, T. B. M. J. & Bobée, B. 1997 Comment on 'The use of artificial neural networks for the prediction of water quality parameters' by H. R. Maier & G. C. Dandy. *Wat. Res. Res.* **33** (10), 2423–2424.

Looney, C. G. 1997 *Pattern Recognition using Neural Networks.* Oxford University Press, New York.

Maier, H. R. & Dandy, G. C. 1996 The use of artificial neural networks for the prediction of water quality parameters. *Wat. Res. Res.* **32** (4), 1013–1022.

Maier, H. R. & Dandy, G. C. 1998 The effect of internal parameters and geometry on the performance of back-propagation neural networks: an empirical study. *Environ. Modell. Software* **13**, 193–209.

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

Masters, T. 1993 *Practical Neural Network Recipes in C + +.* Academic Press, San Diego.

Masters, T. 1995 *Neural, Novel and Hybrid Algorithms for Time Series Prediction.* John Wiley and Sons, New York.

NeuralWare 1998 *Neural Computing: A Technology Handbook for NeuralWorks Professional II/PLUS and NeuralWorks Explorer.* Aspen Technology Inc., USA.

Sarle, W. S. 1997 Neural network FAQ, periodic posting to the Usenet newsgroup comp.ai.neural-nets. URL: ftp://ftp.sas.com/pub/neural/FAQ.html.

Shi, J. J. 2000 Reducing prediction error by transforming input data for neural networks. *J. Comput. Civil Engng.* **14** (2), 109–116.