International Journal of Applied Mathematics and Statistics,
Int. J. Appl. Math. Stat.; Vol. 54; Issue No. 3; Year 2016,
ISSN 0973-1377 (Print), ISSN 0973-7545 (Online)

Rainfall Prediction by Using Wavelet General Regression Neural Network

Budi Warsito[1], **Rahmat Gernowo**[2] **and Aris Sugiharto**[3]

[1]Department of Statistics, Diponegoro University
Tembalang, Semarang, Indonesia
budiwrst2@gmail.com

[2]Department of Physics, Diponegoro University
Tembalang, Semarang, Indonesia
rahmatgernowo@undip.ac.id

[3]Department of Informatics, Diponegoro University
Tembalang, Semarang, Indonesia
aris.sugiharto@gmail.com

ABSTRACT

In recent many years, several models have been developed to analyze and predict the rainfall. In this paper an attempt has been made to get an alternative model for rainfall prediction by combining two methods, the wavelet technique and Neural Network model. Wavelet transformation has become popular because of its ability to concurrently deal with both the spectral and the interim information contained within time series data. The wavelet decomposition used in this paper is Maximal Overlap Discrete Wavelet Transform (MODWT). In Neural Network layer, General Regression Neural Network (GRNN) is chosen to develop the hybrid model. The combination of MODWT and GRNN is called Wavelet General Regression Neural Network (WGRNN). The model developed is applied to the ten-daily rainfall data in two regions of Central Java, Indonesia.

Keywords: MODWT, GRNN, prediction, rainfall.

2010 Mathematics Subject Classification: 62M10, 62M45, 62P12.

1 Introduction

An accurate prediction of rainfall is one of the biggest challenges in the hydrological system, although many progress in weather forecasting in recent decades. Prediction rainfall is closely related to the agricultural sector, which contributes significantly to the nation's economy (Nayak et al., 2013). Neural network becomes an interesting approach in rainfall prediction because of their highly nonlinearity, flexibility and data driven learning in building models without any prior knowledge. Neural Network was a nonparametric model which employable in time series modeling. It does not require various assumptions in the residual. Therefore, the main focus considered on developing Neural Network is how to obtain residual as small as reasonably possible. Many researches have been found that this model have more accurate prediction than parametric model. A specific class of Neural Network which does not need a training procedure as in the back-propagation method is *General Regression Neural Network (GRNN)*. It has been used for a few applications in the water resources area, as in (Cigizoglu and Alp, 2004) and (Cigizoglu and Alp, 2006).

Many applications showed that the resolutions of Neural Network model have been influenced by determining input. Inexactitude of input selection or noise contained in input data, often make the weighting inaccuracy. Wavelet decomposition analysis has given new tool as an approaching of the problem. Wavelet transformation give a collection of wavelet coefficients. Calculation of wavelet coefficients can be done by using *Discrete Wavelet Transform (DWT)* proposed by (Mallat, 1998). Furthermore, (Yajnik and Mohan, 2009) used the DWT for approximation of a signal whereas (Nayak et al., 2013) made do this wavelet in predicting rainfall data. In (Renaud et al., 2003), wavelet has been applied in time series field. The decomposition method used to calculate wavelet coefficients was *Maximal Overlap Discrete Wavelet Transform (MODWT)*. The wavelet coefficients in various scale was determined by the level of the decomposition.

Several approaches have been proposed for combining Neural Network and wavelet decomposition. The model resulted from this technique is called Wavelet Neural Network (WNN). In a general way, there were two methods of combining the wavelet decomposition and Neural Network. The first is using wavelet as as activation function (Minu et al., 2010). In the second approximation, wavelet decomposition was used as *preprocessing*. Input model was decomposed according to the detail coefficients and approximation coefficients. Furthermore, processing is done by Neural Network and *post-processing* by the reconstruction (Al-Geelani et al., 2013). In this paper, the second method is used. The forming model is applied to predict rainfall in some regions in Central Java Indonesia. The regions chosen are the main rice plant regions. Wavelet decomposition method used in this paper is MODWT, whereas GRNN is the Neural Network architecture selected for processing unit. In the research before, (Kisi, 2011) has been develop WGRNN model, but the wavelet decomposition method used was DWT.

2 General Regression Neural Network

Essentially, GRNN operations is based on the theory of nonlinear regression where the estimate of the expected value of output is determined by the set of inputs. The output resulted by GRNN model can be either multivariate or univariate. In terms of the application of time series data, the output is univariate. Equation (1) summarize the logical of GRNN in nonlinear regression term (Specht, 1991):

$$E\left[y\left|X\right.\right]=\frac{\int\limits_{-\infty}^{\infty}yf\left(X,y\right)dy}{\int\limits_{-\infty}^{\infty}f\left(X,y\right)dy} \tag{2.1}$$

where y is the output predicted by GRNN, X is vector containing p predictor variables as input model $(X_1, X_2, \ldots, X_p)$, whereas $E[y|X]$ is the expected value of output y if given input X, and $f(X, y)$ is the probability density function of X and y.

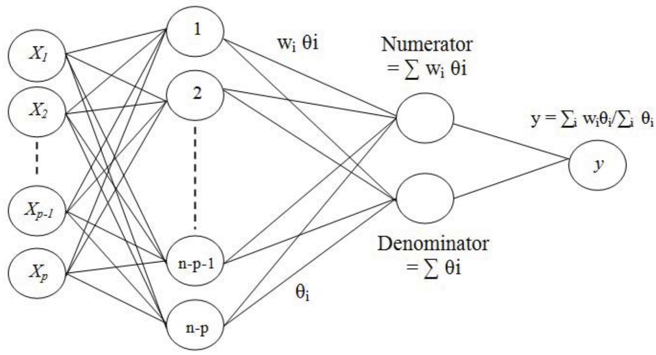

Figure 1: The design of GRNN in nonlinear regression term

The GRNN construction in nonlinear regression term is described in figure 1. The input variables are the vector $(X_1, X_2, \ldots, X_p)$, i.e the independent variables with p factors. The output resulted denoted by y. In time series term, the input variables become the past values, $X = (X_{t-1}, X_{t-2}, \ldots, X_{t-p})$. The order p is determined before. The output resulted X_t is the future prediction. The topology of GRNN proposed by (Specht, 1991) consist of four layers: input layer, pattern layer, summation layer and output layer. The task of the first layer is receive the information. There is no data processing in the layer. The first layer is fully joined to the second, the pattern layer. The total number of patterns equals to the number of input units in the first layer. The pattern neuron i derive data from input neuron and it count the output by the transfer function:

$$\theta_i = e^{-(X-U_i)'(X-U_i)\big/2\sigma^2} \tag{2.2}$$

where X is the input vector of predictor variables, U_i is training vector represented by pattern neuron i, and s is smoothing parameter. Each neuron in pattern layer is then resulting output θ_i. Output of the neuron pattern is then forwarded to the third layer called *summation layer*. In this layer, output of all of the patterns neuron is then added. There are two types of summation formed, simple arithmetic summation and weighted summation (Specht, 1991). In GRNN topology, there is separate processing unit which perform simple arithmetic summation and weighted summation, which is expressed in the following mathematical equations.

$$S_S = \sum_i \theta_i \tag{2.3}$$

$$S_W = \sum_i w_i \theta_i \tag{2.4}$$

Neuron which is formed from simple arithmetic summation is called *denominator* whereas neuron which is formed from weighted summation is called *numerator*. The two summations were obtained through learning process under supervision. The resulting sum by the summation neuron successively sent to the fourth layer, namely the *output layer*. The output neuron then formed the following division to get the regression output GRNN:

$$\hat{X}_t = \frac{S_W}{S_S} \tag{2.5}$$

The GRNN does not require an iterative training procedure as does the FFNN model. The local minimum problem was not faced in GRNN simulations (Kisi, 2011).

3 Wavelet Decomposition

Let X be an N dimensional vector whose elements are the real-valued time series $\{X_t : t = 0, \ldots, N-1\}$ and $\{W_n : n = 0, \ldots, N-1\}$ represent the DWT coefficients. It can be written that **W** $= \mathcal{W}X$, where **W** is a column vector of length $N = 2J$ whose nth element is the nth DWT coefficient W_n, and W is an $N \times N$ real-valued matrix defining the DWT and satisfying $\mathcal{W}^T\mathcal{W} = I_N$. The elements of the vector **W** is then decomposed into $J+1$ subvectors. The first J subvectors are denoted by $W_j, j = 1, \ldots, J$, and the jth such subvector contains all of the DWT coefficients for scale t_j. W_j is a column vector with $N/2j$ elements. The final subvector is denoted as V_j and contains just the scaling coefficient W_{N-1}. The vector **W** can be written as **W** $= \begin{bmatrix} W_1 & W_2 & \ldots & W_J & V_J \end{bmatrix}'$. By the orthonormality, i.e. $X = \mathcal{W}^T$**W** and $\parallel$ **W** $\parallel^2 = \parallel X \parallel^2$, X can be written as:

$$X = \sum_{j=1}^{J} \mathcal{W}_j^T W_j + \mathcal{V}_J^T V_J \tag{3.1}$$

where $\mathcal{W}_j$ and $\mathcal{V}_J$ are matrices by partitioning the rows of W commensurate with the partitioning of **W** into $W_1, \ldots, W_J$ and V_J. Thus the $(N/2) \times N$ matrix $\mathcal{W}_1$ is formed from the $n = 0$ up to $n = (N/2) - 1$ rows of W; the $(N/4) \times N$ matrix $\mathcal{W}_2$ is formed from the $n = N/2$ up to $n = (3N/4) - 1$ rows; and so forth, until the $1 \times N$ matrices W_J and V_J which are the last two rows of W.

Pyramid algorithm proposed by (Mallat, 1998), was an algorithm that allows W to be factored in terms of very sparse matrices using linear filtering. This algorithm make uses of a wavelet filter and a scaling filter. A filter $\{h_l : l = 0, ..., L-1\}$ of even width L (implying $h_0 = 0$ and $h_{L-1} \neq 0$) is called wavelet filter if

$$\sum_{l=0}^{L-1} h_l = 0 \quad and \quad \sum_{l=-\infty}^{\infty} h_l h_{l+2n} = \begin{cases} 1, & if \quad n = 0 \\ 0, & if \quad n \neq 0 \end{cases} \tag{3.2}$$

The scaling filter is defined in terms of the wavelet filter through

$$g_l \equiv (-1)^{l+1} h_{L-1-l}$$

This filter satisfies the conditions

$$\sum_{l=0}^{L-1} g_l h_{l+2n} = \begin{cases} 1, & if \quad n = 0 \\ 0, & if \quad n \neq 0 \end{cases} \qquad and \qquad \sum_{l=0}^{L-1} g_l h_{l+2n} = 0, for\ all\ n \tag{3.3}$$

Filtering by DWT cannot be done on any sample size, which cannot be expressed in the form $2J$ where J is a positive integer. (Percival and Walden, 2000) showed that wavelet decomposition called *Maximal Overlap Discrete Wavelet Transform* ($MODWT$), a modification version of Discrete Wavelet Transform (DWT), was well-defined in any sample size n. In $MODWT$, wavelet coefficients in each level is always the same, so it is more appropriate in time series modeling than DWT. The prediction of time series is modeled linearly based on wavelet coefficients resulted from the decomposition before. The model resulted can be constructed in this equation:

$$\hat{X}_t = \sum_{j=1}^{J} \sum_{k=1}^{A_j} (\hat{a}_{j,k} w_{j,t-k} + \hat{b}_{j,k} v_{j,t-k}) \tag{3.4}$$

The J symbol shows the level of decomposition, whereas A_j is the number of coefficients in each level. The next problem is how to determine the optimal lags be input of MODWT model in time series. If the number of lags in each level is the same, that is $A_j = A$ in each level j, so the number of variables be input are $2AJ$. If the scaling coefficient included in the model is just the last level, $(v_J, t-k)$ so the equation in (3.4) will be

$$\hat{X}_t = \sum_{j=1}^{J} \sum_{k=1}^{A_j} (\hat{a}_{j,k} w_{j,t-k}) + \sum_{k=1}^{A_{J+1}} \hat{a}_{J+1,k} v_{J,t-k} \tag{3.5}$$

In (Murtagh et al., 2004), detail coefficients $w_{j,t-k}$ and scaling coefficients $v_{J,t-k}$ resulted from MODWT transformation which important considered to predict the value in time t are $w_{j,t-2^j k}$ and $v_{J,t-2^J k}$. It can be defined by this equation:

$$\hat{X}_t = \sum_{j=1}^{J} \sum_{k=1}^{A_j} \hat{a}_{j,k} w_{j,t-2^j k} + \sum_{k=1}^{A_{J+1}} \hat{a}_{J+1,k} v_{J,t-2^J k} \tag{3.6}$$

In this case the decomposition level J and the number of coefficients in each level A_j were chosen by user.

4 Wavelet General Regression Neural Network

The Wavelet General Regression Neural Network (WGRNN) is a combination of wavelet decomposition and GRNN model. In this hybrid model, input of GRNN are the detail coefficients and scaling coefficients resulted from MODWT decomposition. In this case, we use the design explained in figure (1), but the input are the independent variables in equation (3.4) or is in equation (3.5). The MODWT decomposition is used to decompose the actual data at first up to desired level. The next step is to select the detail coefficients and scaling coefficients used as input. Lags of each coefficient are selected by multiscale technique proposed by citeRen.03. GRNN architecture is now applied for the selected input. By the GRNN procedure described before, the network is then work by using all of the four layer processing to obtain output.

$$\hat{X}_t = \frac{\tilde{S}_W}{\tilde{S}_S} \tag{4.1}$$

where

$$\tilde{S}_S = \sum_i \tilde{\theta}_i \tag{4.2}$$

$$\tilde{S}_W = \sum_i w_i \tilde{\theta}_i \tag{4.3}$$

and

$$\tilde{\theta}_i = e^{-(\tilde{X}-U_i)'(\tilde{X}-U_i)\big/2\sigma^2} \tag{4.4}$$

$\tilde{X} = (w_{j,t-2^j k}, v_{J,t-2^J k})$, i.e the detail coefficients and scaling coefficients of MODWT. The roadmap of developing WGRNN model is shown in figure (2).

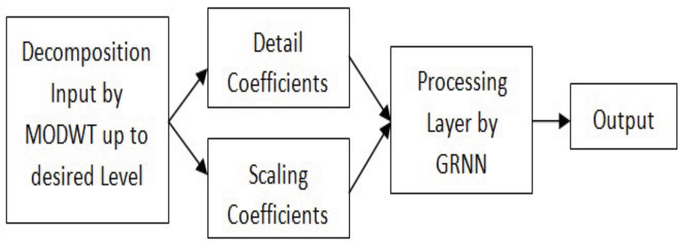

Figure 2: Roadmap of developing WGRNN model

In developing WGRNN model, the desired maximal level and the number of coefficients in each level are selected as appropriate as possible with the length of data. More the number is chosen, more the data is sliced. In GRNN processing, the input data is divided into two parts. The first is training data and the remaining is testing. The optimal parameter is obtained through the performance of the testing.

5 Rainfall Prediction

The data used in this paper are the ten-daily rainfall data in two regions of Central Java, Indonesia. The first is the rainfall data in ZOM 136 Cokrotulung, Musuk, Klaten and the second is in ZOM 146 Balepanjang, Tawangmangu, Karanganyar. All of the data are from January 2010 to December 2014. The length of the data is 180, respectively. The first 150 data as training and the last 30 data as testing. The first stage is decomposing data via MODWT. The model developed is it by (Murtagh et al., 2004) as in equation (3.6). In this step, the optimal level and the optimal number of coefficients in each level was obtained automatically by the computational programming constructed. Through the length of data reason, the maximal level chosen is 3 and the maximal number in each level is 6. All the computational programming in this section is done with Matlab routine, especially is wmtsa toolkit for Matlab. The MODWT decomposition resulting the optimal level and the number of coefficients in each level. The results of the both data analysis are showed in table (1).

Table 1: Input decomposition by MODWT

criteria	ZOM 136	ZOM 146
RMSE training	53.0248	103.6343
the optimal J	4	3
the optimal A_j	6	3
RMSE testing	87.5016	73.0038

The table (1) shows that, for the first data, ZOM 136, the optimal level is 4 and the number of coefficients in each level is 6. The total number of input resulted from this construction is 30. In the second data, ZOM 146, the optimal level is 3 and the number of coefficients in each level is 3. The total number of input resulted from this construction is 12. The second stage is using the input resulted before to processing in GRNN layer. In this layer, the optimal spread is obtained automatically by computational programming. The results of predict in-sample with training data and predict out-sample with testing data of WGRNN model is showed in table (2).

Table 2: Predict in-sample and predict out-sample of WGRNN model

criteria	ZOM 136	ZOM 146
RMSE training	27.8586	53.4430
RMSE testing	63.8004	81.0513
optimal spread	0.01	0.01

Table (2) shows that in the first data, WGRNN model can improve both in-sample prediction with training data and out-sample prediction with testing data. The proposed model can reduce the RMSE of in-sample prediction from 53.0248 up to 27.8586, it is about 47.46%. It also repair the RMSE of out-sample prediction from 87.5016 to 63.8004, about 27.09%. In the second data, WGRNN model just make a better in-sample prediction compared with MODWT model, but poor in out-sample prediction. The value of RMSE of in-sample prediction by MODWT is 103.6343, whereas RMSE of in-sample prediction resulted by WGRNN is 53.4430, it is go down about 48.43%. Unhappily, WGRNN is not success to obtain a magnificent out-sample prediction. The RMSE of out-sample prediction resulted is 81.0513. It is higher but not too bad than one by MODWT, which the value is 73.0038. The difference of the both values is about 9.93%.

Figure 3 and figure 4 show the model performance of both in-sample and out-sample prediction of WGRNN model in ZOM 136. In figure 3, it can be seen that although in some points have a wide difference with the actual, but broadly speaking that the in-sample prediction of WGRNN model is very accurate. The model output closely follows the true data. In each observation point, the difference of actual and the prediction is small enough. The error range of the model is small enough. Figure 4 shows the power of out-sample prediction of WGRNN model. It is the higher performance compared with one another. In some points, out-sample predictions give almost perfect forecasting. The predictions are very close to the actual. Although in some observations the out-sample prediction is still rather far from the native but overall, the forecasting of WGRNN model yields a resemble prediction with the actual.

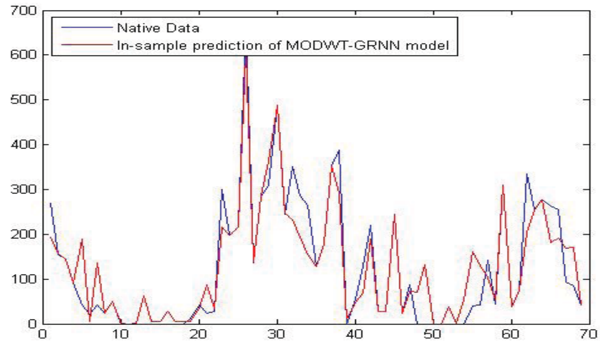

Figure 3: Predict in-sample of ZOM 136 by WGRNN

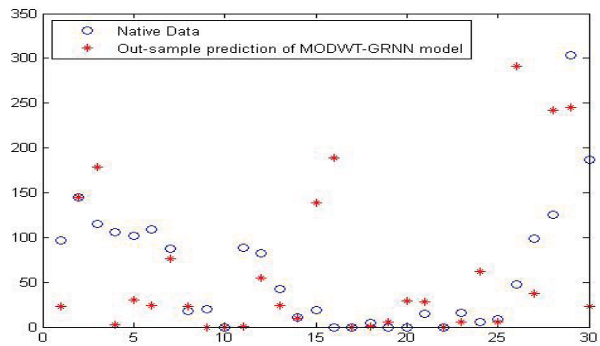

Figure 4: Predict out-sample of ZOM 136 by WGRNN

Figure 5 and figure 6 show the model performance of both in-sample and out-sample prediction of WGRNN model in ZOM 146. Figure 5 shows the power of in-sample prediction of WGRNN model. It make a wonderful result. The output is very closely follows the actual. In each observation point, from beginning to the end, the prediction has an ability to abreast of the actual. The error range of the model is very small, indicating that they all perform adequately. Figure 6 shows the out-sample prediction of WGRNN model. It is the higher performance compared with one another. In some points of forecasting, out-sample predictions give almost perfect forecasting but in some others the out-sample prediction is far from the actual. Overall, the forecasting of WGRNN model still yields a good prediction, because the pattern of the step ahead data is still can be followed by the forecasting.

6 Conclusion

The new design which combining wavelet decomposition and neural network in seasonal time series prediction has been developed. The wavelet decomposition exerted was MODWT,

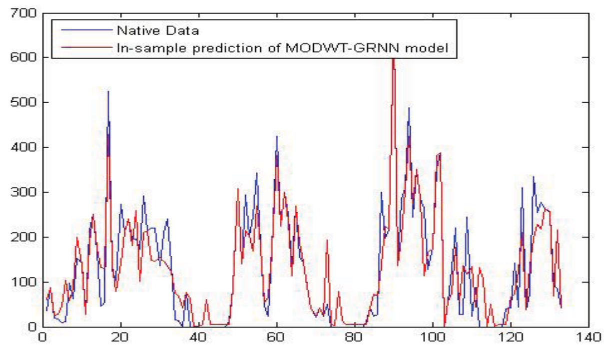

Figure 5: Predict in-sample of ZOM 146 with WGRNN model

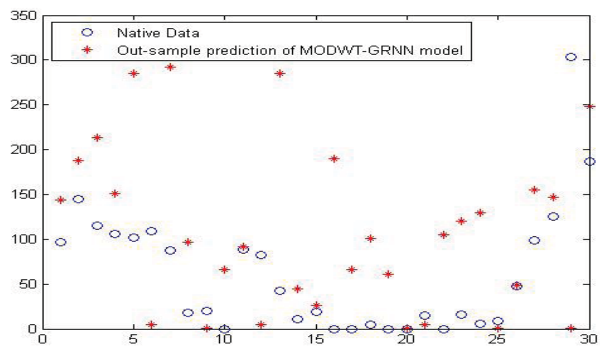

Figure 6: Predict out-sample of ZOM 146 with WGRNN model

whereas a class of neural network model used was GRNN. The proposed model was applied in rainfall prediction. The data used are the ten-daily rainfall data in two regions of Central Java, Indonesia. In the two data observed, the resulting model made an excellent accuracy in both in-sample predictions, better than one produced by just MODWT prediction. In ZOM 136, the out-sample prediction of the proposed model was still better result than MODWT, but on the contrary in ZOM 146. Overall, the proposed model can made into a good reference for modeling seasonal time series, especially for forecasting the rainfall data.

7 Acknowledgment

This work was supported by the Development and Application Research Grant, DIPA PNBP, Diponegoro University.

References

Al-Geelani, N.A., Piah, M.A.M., Adzis, Z. and Algeelani, M.A. 2013. Hybrid regrouping PSO based wavelet neural networks for characterization of acoustic signals due to surface discharges on H.V. glass insulators, *Journal Applied Soft Computing*, Vol 13, **12**: 4622–4632.

Cigizoglu, H.K. and Alp, M. 2004. Rainfall-Runoff Modelling Using Three Neural Network Methods, *Artificial Intelligence and Soft Computing - ICAISC, Lecture Notes in Computer Science*, Vol 3070, 166–171.

Cigizoglu, H.K. and Alp, M. 2006. Generalized regression neural network in modelling river sediment yield, *Journal Advances in Engineering Software*, Vol 37, **2**: 63–68.

Kisi, O. 2011. A Combined Generalized Regression Neural Network Wavelet Model for Monthly Streamflow Prediction, *KSCE Journal of Civil Engineering*, Vol 15, **8**: 1469–1479.

Kisi, O. 1998. A Wavelet Tour of Signal Processing, *Academic Press*, New York.

Minu, K. K., Lineesh, M. C. and John, C. J. 2010. Wavelet Neural Networks for Nonlinear Time Series Analysis, *Applied Mathematical Sciences*, Vol 4, **50**: 2485–2495.

Murtagh, F., Stark, J.L. and Renaud, O. 2010. On Neuro-Wavelet Modelling, *Decision Support System*, **37**: 475–484.

Nayak, P.C., Venkatesh, B., Krishna, B. and Jain, S.K. 2013. Rainfall-runoff modeling using conceptual, data driven, and wavelet based computing approach, *Journal of Hydrology*, **493**: 57–67.

Percival, D.,B. and Walden, A.,T. 2000. Wavelet Methods for Time Series Analysis, *Cambridge University Press*, Cambridge, United Kingdom.

Renaud, O., Starcx, J.L. and Murtagh, F. 2003. Prediction Based on a Multiscale Decomposition, *Int. Journal of Wavelets, Multiresolution and Information Processing*, Vol 1, **2**: 217–232.

Specht, D. F. 1991. A general regression neural network, *IEEE Trans. Neural Networks*, Vol 2, **6**: 568–576.

Yajnik, A. and Mohan, S.R. 2009. Approximation of a Signal Using Discrete Wavelets Transform, *International Journal of Artificial Intelligence*, Vol 2, **S09**.