Function Approximation using Robust Wavelet Neural Networks*

Sheng-Tun Li[1] and Shu-Ching Chen[2]

[1] *Department of Information Management, National Kaohsiung First University of Science and Technology, Kaohsiung, Taiwan, ROC, Email: stli@ccms.nkfust.edu.tw*

[2] *School of Computer Science, Florida International University, Miami, FL 33199, USA, Email: chens@cs.fiu.edu*

Abstract

Wavelet neural networks (WNN) have recently attracted great interest, because of their advantages over radial basis function networks (RBFN) as they are universal approximators but achieve faster convergence and are capable of dealing with the so-called "curse of dimensionality." In addition, WNN are generalized RBFN. However, the generalization performance of WNN trained by least-squares approach deteriorates when outliers are present. In this paper, we propose a robust wavelet neural network based on the theory of robust regression for dealing with outliers in the framework of function approximation. By adaptively adjusting the number of training data involved during training, the efficiency loss in the presence of Gaussian noise is accommodated. Simulation results are demonstrated to validate the generalization ability and efficiency of the proposed network.

Keywords: Wavelet neural network, wavelet transform, outlier, least trimmed squares, function approximation.

1. Introduction

Function approximation involves estimating (approximating) the underlying relationship from a given finite input-output data set has been the fundamental problem for a variety of applications in pattern classification, data mining, signal reconstruction, and system identification [1, 5, 6, 8]. For instance, the task of pattern recognition is a function mapping whose objective is to assign each pattern in a feature space to a specific label in a class space. The problem of system identification is to estimate the underlying system characteristics using empirical input-output data from the system. In signal processing, it is desired to determine adaptively nonstationary system parameters through the input-output signals. Recently, feedforward neural networks such as multilayer perceptrons (MLP) and radial basis function networks (RBFN) have been widely used as an alternative approach to function approximation since they provide a generic black-box functional representation and have been shown to be capable of approximating any continuous function defined on a compact set in R^n with arbitrary accuracy [6]. Following the concept of locally supported basis functions such as RBFN, a class of wavelet neural networks (WNN) which originate from wavelet decomposition in signal processing has become more popular lately [2, 4, 8, 9, 10]. In addition to the salient feature of approximating any non-linear function, WNN outperforms MLP and RBFN due to its capability in dealing with the so-called "curse of dimensionality" and non-stationary signals and in faster convergence speed [3, 9]. It has also been shown that RBFN is a special case of WNN.

The task of training WNN involves estimating parameters in the network by minimizing some cost function, a measure reflecting the approximation quality performed by the network over the parameter space in the network. The least squares (LS) approach is the most popularly used in estimating the synaptic weights which provides optimal results if the underlying error distribution is Gaussian.

* This research was supported in part by NSC89-2213-E-327-007, Taiwan, ROC.

However, in real-world applications, the distribution of the noise is often unknown or highly disturbed and thereby more seriously contaminated data such as outliers may occur. Therefore, it is desirable to develop a robust learning rule insensitive to outliers.

In this paper, we propose a novel learning algorithm by applying least trimmed squares (LTS) method to increase the robustness of WNN. In contrast to most present robust learning rules, our approach neither is dependent upon any assumptions about error distribution nor is it necessary to estimate the error distribution. In order to compensate for the loss of efficiency, an adaptive version of the proposed learning rule is developed. The advantage of the proposed method is demonstrated by computer simulations. This paper is organized as follows. Section 2 presents the WNN framework used in this study. Section 3 illustrates the reason why a robust WNN is needed. We propose the robust WNN based on robust regression in Section 4. Section 5 demonstrates the simulation results on two function approximation problems. Section 6 concludes this paper.

2. Wavelet neural networks

Wavelets occur in family of functions and each is defined by dilation a_i which control the scaling parameter and translation t_i which controls the position of a single function, named the mother wavelet $\psi(\mathbf{x})$. Mapping functions to a time-frequency phase space, WNN can reflect the time-frequency properties of function more accurately than the RBFN. Given an n-element training set, the overall response of a WNN is:

$$\hat{y}(\mathbf{w}) = w_0 + \sum_{i=1}^{Np} w_i \psi_i\left(\frac{x - t_i}{a_i}\right),$$

where N_p is the number of wavelet nodes in the hidden layer and w_i is the synaptic weight of WNN. A WNN can be regarded as a function approximator which estimates an unknown functional mapping:

$$y = f(x) + \varepsilon,$$

where f is the regression function and the error term ε is a zero-mean random variable of disturbance.

There are a number of approaches for WNN construction (a brief survey is provided in [10]), we pay special attention on the model proposed by Zhang [10] due to its notable feature in dealing with the sparseness of training data. Following [10], constructing a WNN involves two stages: First, construct a wavelet library W of discretely dilated and translated versions of wavelet mother function ψ:

$$W = \left\{ \psi_i : \psi_i(x) = \alpha_i \psi(a_i(x - t_i)),\ \ \alpha_i = \left(\sum_{k=1}^{n} [\psi(a_i(x_k - t_i))]^2 \right)^{-\frac{1}{2}},\ \ i = 1, \cdots L \right\},$$

where x_k is the sampled input, and L is the number of wavelets in W. Then select the best M wavelets based on the training data from wavelet library W, in order to build the regression

$$f_M(x) = \sum_{i \in I} u_i \psi_i(x),$$

where I is an M-element subset of the index set $\{1, 2, \ldots, L\}$ and $M \leqq L$.

Secondly, to minimize the cost function

$$J(I) = \min_{u_i, i \in I} \frac{1}{n} \sum_{k=1}^{n} \left(y_k - \sum_{i \in I} u_i \psi_i(x_k) \right)^2,$$

Zhang derives two heuristic algorithms, namely, stepwise selection by orthogonalization for deciding appropriate wavelets in the hidden units and backward elimination for choosing the number of hidden units. The number of wavelets, M, is chosen as the minimum of the so-called Akaike's final prediction error criterion (FPE) [10]:

$$J_{\text{FPE}}(\hat{f}) = \frac{1 + n_{pa}/n}{1 - n_{pa}/n} \frac{1}{2n} \sum_{k=1}^{n} (\hat{f}(x_k) - y_k)^2$$

where n_{pa} is the number of parameters in the estimator.

After the initial WNN is constructed, it is further trained by the gradient descent algorithms like least mean squares (LMS) to minimize the mean-squared error:

$$J(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}(\mathbf{w}))^2,$$

where is the real output from a trained WNN at the fixed weight vector $\mathbf{w}$.

3. The need for robust wavelet neural networks

The LMS-based training approach provides an asymptotically optimal solution with minimum variance, which assumes that the error distribution is identically independent, and Gaussian. However, this assumption usually fails to hold in real-world application since either a priori information about the error distribution is generally unavailable, or the data are contaminated by non-Gaussion noise whereby some data points fall far outside of the majority of the data so that outliers are encountered.

Outlier may be introduced in different ways. For example, in computer vision, the outliers may be the result of clutter, large measurement errors, or impulse noise corrupting the data. In general, there are two kinds of outliers [7]; leverage points and vertical outlier. Leverage points result from contamination in the input space ***X*** due to some of the inputs **x** failing to obey the environmental probability rule $p(\mathbf{x})$. Since the outputs y are uncontaminated in the training set, the effects of horizontal outliers do not directly contribute to residuals. Contamination in the output space ***Y*** leads to vertical outliers due to the output y failing to obey the conditional probability rule $p(y|\mathbf{x})$. Such deviations in the output space directly contribute to residuals. Both anomalies may result in an aberrant and biased WNN since it is trained to fit these significant fluctuations by interpolation instead of approximating the underlying model in an attempt to compensate for outliers with least squared residuals. That is, WNN is greatly sensitive to the presence of outliers.

4. Robust wavelet neural networks

In order to enhance the robustness of WNN, the training procedure of the initial WNN is performed by the least trimmed squares (LTS) in robust regression. LTS is a famous robust estimator, which has been shown to have the highest possible breakdown point (≈50%) [5, 7]. The breakdown point of an estimator, without being confined to any assumed distribution of errors, gives a global measure of stability in terms of the fraction of outlying data it can tolerate. Instead of minimizing the sum of all squared residuals, the LTS estimator only considers the sum of the smallest order squared residuals up to the rank h,

$$\underset{\hat{w}}{\text{Minimize}} \sum_{i=1}^{h} (r^2)_{i:n} \,,$$

where $(r^2)_{1:n} \le \cdots \le (r^2)_{h:n} \le \cdots \le (r^2)_{n:n}$ are obtained by squaring the residuals first, and then ordering them. The principal difference between the LTS and the LS methods is that, at the true parameter vector **w**, those n-h outlying observations are left out of the cost function defined in Equation (7) since (at that **w**) they possess the largest residual. It follows that the WNN reaches its optimal break down point

$$\xi = \frac{\lfloor (n-p)/2 \rfloor + 1}{n},$$

when

$$h = \left\lfloor \frac{n}{2} \right\rfloor + \left\lfloor \frac{p+1}{2} \right\rfloor. \qquad (1)$$

(The notation $\lfloor q \rfloor$ stands for the largest integer less than or equal to q.)

The efficiency of the rule when small perturbations are encountered can be improved by adaptively adjusting the number of residuals contributing to weight updates h [5]. This is inspired by the observation that the more training data are involved in the beginning of training, the quicker the network can fit the underlying model. To accommodate both requirements of robustness against outliers of efficiency in the presence of Gaussian noise, we propose the specific value of h at time t according to the following rule:

$$h(t+1) = \lfloor \tau(t) n \rfloor + \lfloor (1-\tau(t))(p+1) \rfloor, \qquad (2)$$

where

$$\tau \in \left[\frac{1}{2}, 1 \right]$$

is a free parameter that determines the proportion of observation involved in error backpropagation and a function of the commonly used criterion normalized root-mean-squared error (NRMSE) on an uncontaminated test set V since it keeps track of the generalization ability of the work:

$$\tau(NRMSE_{test}) = 0.5\exp(-\frac{v}{NRMSE_{test}{}^2}) + 0.5$$

where v is a small positive real number, and

$$NRMSE_{test} = \sqrt{\frac{\sum_{i=1}^{n} (y_i - \hat{y})^2}{\sum_{i=1}^{n} (y_i - E[y])^2}} \,.$$

At the very beginning of training, $\tau \approx 1$ due to larger NRMSEs; if this were maintained constant, it would degenerate to the traditional WNN rule. On the other hand, if smaller NRMSEs are reached, τ goes to 1/2; a non-adaptive rule based on the classic LTS method is obtained.

The difference between Equations (1) and (2) is that Equation (2) is adaptive in tuning the parameter h, however Equation (1) fixed it always.

5. Experiments

In this section, we present two experimental results of the proposed robust WNN on approximating two functions. First, simulations on the 1-D function approximation $f(x)=0.5x\sin(x)+\cos^2(x)$ are conducted to validate the robustness of the proposed robust WNN. The input x is constructed by the uniform distribution on [-6 6], and the corresponding output y is functional of $y = f(x)$ and is artificially contaminated by stochastic errors according to the Cauchy distribution with location 0 and scale 0.05. The training and test data are composed of 100 points and 300 points, respectively (see Figure 1). Mexican Han wavelet,

$$\psi(\mathbf{x}) = (1-\mathbf{x}^T\mathbf{x})\exp(-\frac{1}{2}\mathbf{x}^T\mathbf{x})$$

is chosen as the mother wavelet for training WNN.

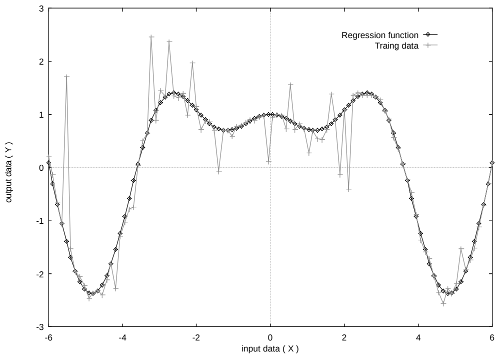

Figure 1. The regression function $f(x)=0.5x\sin(x)+\cos^2(x)$ and the contaminated data

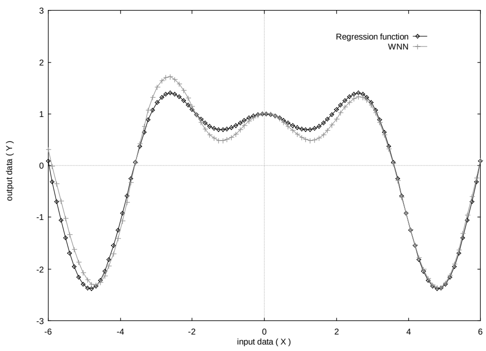

Figure 2. The reconstruction result by the conventional WNN

Figures 2 and 3 show the generalization results by Zhang's WNN and by the robust WNN after 500 and 300 training epochs, respectively. Both networks are powered with 10 hidden units, which are determined by Akaike's FPE criterion, and the parameters τ are fixed at 0.003 for impartiality. One can see that, in Figure 2, the curve shape predicted is pulled toward the larger outliers due to the interpolative character of WNN. On the other hand, in Figure 3, seven major outliers having larger residuals have been successful trimmed by the proposed method. The $NRMSE_{test}$ of the proposed WNN is 0.0641, comparing to 0.1327 the traditional WNN achieved. These results confirm that the proposed robust WNN is resistant to outliers.

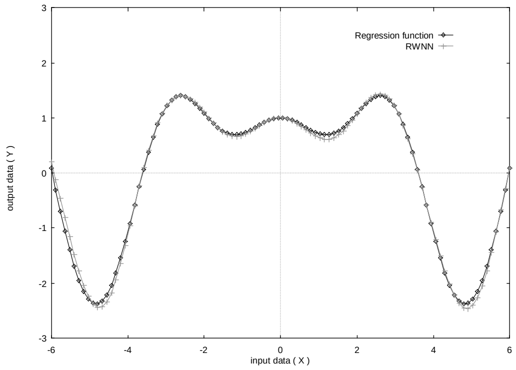

Figure 3. The reconstruction result by the proposed robust WNN

Next, the two-dimension function

$$f(x_1, x_2) = (x_1^2-x_2^2)\sin(5x_1)$$

is approximated to illustrate the validity of the proposed learning rule. The training set D and the test set V are constructed by evenly spaced 21×21 and 22×22 grid on [-1, 1] ×[-1,1]. The data set D is contaminated by stochastic errors according to the Cauchy distribution with scale 0.1, Figures 4 and 5 show the original surface and the highly contaminated surface for the corrupted training set D', respectively. The mother wavelet is chosen as follows:

$$\psi(\mathbf{x}) = (2-\mathbf{x}^T\mathbf{x})\exp(-\frac{1}{2}\mathbf{x}^T\mathbf{x})\ .$$

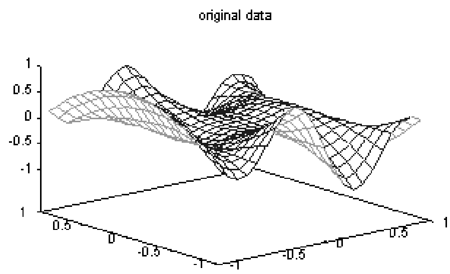

Figure 4. The original function: $f(x_1, x_2) = (x_1^2-x_2^2)\sin(5x_1)$

Based on the Akaike's FPE criterion, the number of hidden units is determined as 12. The parameter $v = 0.03$ is used again in the LTS-based method. The breakdown point of this network is 0.489, and it may tolerate 214 large outliers. The surface reconstructed by the conventional WNN, which reaches

$NRMSE_{test} = 0.291$ after 4000 epochs, is described in Figure 6, and the relatively residual surface is depicted in Figure 8. On the other hand, the proposed robust WNN reaches $NRMSE_{test} = 0.231$, and its reconstructed surface and residual surface are plotted in Figures 7 and 9. One notes that, from the two residual surfaces, the proposed approach makes the reconstructed surface smoother, i.e., the influence of outliers can be effectively filtered.

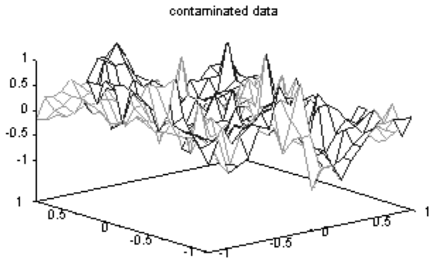

Figure 5. The surface contaminated by Cauchy-distributed errors with scale 0.1

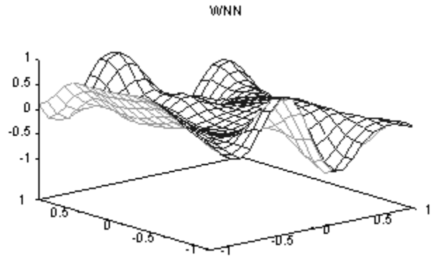

Figure 6. The reconstructed surface after 4000 training-epochs by the 2-12-1 WNN

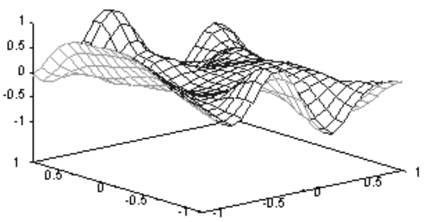

Figure 7. The reconstructed surface after 4000 training-epochs by the robust 2-12-1 WNN

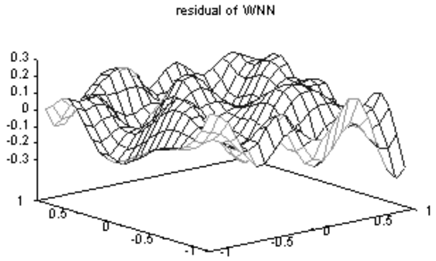

Figure 8. The residual surface by the 2-12-1 WNN

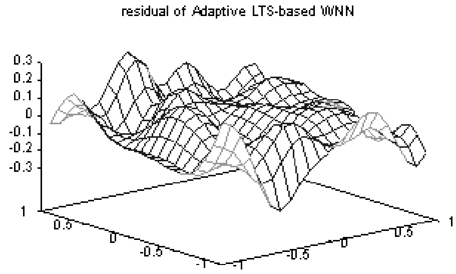

Figure 9. The residual surface by the robust 2-12-1 WNN

6. Conclusions

This paper presented a novel robust wavelet neural network for function approximation from a contaminated training set in which outliers or gross errors may occur. By appealing to the breakdown point approach in robust regression, neither *a priori* information about the error distribution nor estimating is required. An adaptive robust learning algorithm is also derived for improving the efficiency of the network. Simulation results demonstrate its superiority over the conventional WNN in function approximation from outlying data. Experimentation on real-world applications is undergoing.

7. Acknowledgments

The authors would like to thank Prof. S.-J. Chern for his guidance and Mr. S.-B. Hwang for his work in implementing the robust WNN.

8. References

[1] D. S. Chen and Ramesh C. Jain, "A Robust Back Propagation Learning

Algorithm for Function Approximation", IEEE Trans. Neural Networks, Vol. 5, May 1994.

[2] P. Cristea, R. Tuduce, and A. Cristea, "Time Series Prediction with Wavelet Neural Networks," Proceedings of IEEE Neural Network Applications in Electrical Engineering, 2000, pp. 5-10.

[3] B. Delyon, A. Juditsky, and A. Benveniste, "Accuracy Analysis for Wavelet Approximations," IEEE Trans. Neural Networs, Vol. 6, p332-348, 1995.

[4] D.W.C. Ho, P.-A. Zhang; J. Xu, "Fuzzy Wavelet Networks for Function Learning," Fuzzy Systems, IEEE Transactions on , Vol. 9, No. 1 , Feb 2001, pp. 200-211.

[5] S.-T. Li and E. Leiss, "On Noise-immune RBF Networks," in Radial Basis Function Neural Networks: Recent Developments in Theory and Applications, Editors: Howlett, R. J. and Jain, L.C., Springer Verlag, 2001, pp. 95-124.

[6] J.-N. Hwang, "Nonparametric Multivariate Density Estimation: A Comparative Study," IEEE Trans. Signal Processing, Vol. 42, p2795-2810, 1994.

[7] P. J. Rousseeuw and A. M. Leroy, Robust Regression and Outlier Detection, Wiley, 1987.

[8] H. L. Shashidhara, S. Lohani, and V. M. Gadre, "Function Learning using Wavelet Neural Networks," Proceedings of IEEE International Conference on Industrial Technology, Vol. 2, 2000, pp. 335-340.

[9] J. Zhang, G. G. Walter, Y. Miao, and W. N. W. Lee, "Wavelet Neural Networks for Function Learning," IEEE Trans. Signal Processing, Vol. 43, p1485-1497, June 1995.

[10] Q. Zhang,"Using Wavelet Network in Nonparametric Estimation", IEEE Trans. Neural Network, Vol. 8, pp.227-236, 1997.