Beta wavelet networks for function approximation

Wajdi BELLIL[1], Chokri BEN AMAR[2] and Adel M. ALIMI[2]

REGIM: Research Group on Intelligent Machines

[1] Department of Electrical Engineering, High Institute of Technological Studies of Gafsa, Tunisia

[2] Department of Electrical Engineering, ENIS, University of Sfax, Tunisia

E-mail: {Wajdi.bellil, chokri.benamar, adel.alimi}@ieee.org

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. In this paper we present a novel wavelet neural network, based on Beta wavelets, for 1-D and 2-D function approximation. Our purpose is to approximate an unknown function f: $R^n \rightarrow R$ from scattered samples (x_i; $y_i = f(x)$) i=1....n, where:

✓ we have little a priori knowledge on the unknown function f which lives in some infinite dimensional smooth function space,

✓ the function approximation process is performed iteratively: each new measure on the function (x_i; $f(x_i)$) is used to compute a new estimate $\hat{f}$ as an approximation of the function f.

Simulation results are demonstrated to validate the generalization ability and efficiency of the proposed Beta wavelet network.

1 Introduction

Combining the wavelet transform theory with the basic concept of neural networks [1-3], a new mapping network called wavelet neural network or wavenets (WNN) is proposed as an alternative to feedforward neural networks for approximating arbitrary nonlinear functions. Kreinovich proved in [14] that if we use a special type of neurons (wavelet neurons), then the resulting neural networks are *optimal* approximators in the following sense: as $\varepsilon \rightarrow 0$, the number of bits that is necessary to store the results of a 3-layer wavelet neural network approximation, increases slower than for any other approximation scheme.

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

$$\hat{y}(w) = w_0 + \sum_{i=1}^{N_p} w_i \Psi_i\left(\frac{x - t_i}{a_i}\right) \qquad (1)$$

where Np 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:

$$y = f(x) + \varepsilon \qquad (2)$$

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 to WNN construction (a brief survey is provided in [9-12]), we pay special attention on the model proposed by Zhang [1, 6, 8].

2 The Beta wavelet

The Beta function [13] is defined as:

if p>0, q>0, (p, q) $\in$ IN

$$\beta(x) = \begin{cases} \left(\frac{x - x_0}{x_c - x_0}\right)^p \left(\frac{x_1 - x}{x_1 - x_c}\right)^q & if\ x \in [x_0, x_1] \\ 0 & \text{else} \end{cases} \qquad (3)$$

where, $x_c = \dfrac{px_1 + qx_0}{p + q}$

2.1. The derivatives of Beta function

We proved in [4, 5] that all derivatives of Beta function $\in L^2(IR)$ and are of class C^∞. The general form of the n^{th} derivative of Beta function is:

$$\Psi_n(x) = \frac{d^n \beta(x)}{dx^n}$$

$$= \left[(-1)^n \frac{n!p}{(x - x_0)^{n+1}} + \frac{n!q}{(x_1 - x)^{n+1}}\right]\beta(x) + P_n(x)P_1\beta(x)$$

$$+ \sum C_n^i \left[(-1)^n \frac{(n-i)!\,p}{(x - x_0)^{n+1-i}} + \frac{(n-i)!\,q}{(x_1 - x)^{n+1-i}}\right]P_i(x)\beta(x) \qquad (4)$$

where: $P_1(x) = \dfrac{p}{x - x_0} - \dfrac{q}{x_1 - x}$

and $P_n(x) = (-1)^n \dfrac{n!p}{(x - x_0)^{n+1}} - \dfrac{n!q}{(x_1 - x)^{n+1}}$

The first (BW1), second (BW2) and third (BW3) derivatives of Beta wavelet are shown graphically in Figure 1.

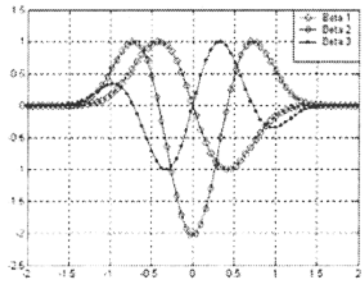

Fig.1. First, second and third derivatives of Beta function.

2.2. Proposition

if p = q , for all n ∈ IN and 0 < n < p the functions

$$\Psi_n(x) = \frac{d^n\beta(x)}{dx^n} \text{ are wavelets [4, 5].} \qquad (5)$$

3 Experiments

In this section, we present two experimental results of the proposed Beta Wavelet Neural Networks (BWNN) on approximating two functions using the *Stepwise selection by orthogonalization* training algorithm. First, simulations on the 1-D function approximation $f(x)=0.5x\sin(2x)+\cos^2(2x)$ are conducted to validate and compare the proposed BWNN with some other wavelets. The input x is constructed by the uniform distribution on [-2.5, 2.5], and the corresponding output y is functional of $y = f(x)$ and is artificially contaminated by random errors. The training and test data are composed of 50 points and 500 points, respectively. Beta wavelet is chosen as the mother wavelet for training network. Second, the two-dimension function:

$$f(x_1, x_2) = \frac{e^{-\frac{81}{16}[(x1-0.5)^2+(x2-0.5)^2]}}{3} \text{ is}$$

approximated to illustrate the robustness of the proposed wavelets family. The training set D contains 11x11 uniform spaced points, and 11x11 stochastic points. The test set V is constructed by evenly spaced 21x21 grid on [-1, 1]x[-1, 1].

3.1 1-D interpolation using the stepwise selection by orthogonalization algorithm

These results are given, using the Stepwise selection by orthogonalization algorithm, on a Neural Wavelet Networks using 9 wavelets, 4 levels decomposition, 500 iterations, 50 points for training and a uniform spaced points. $f(x)=0.5x\sin(2x)+\cos^2(2x)$.

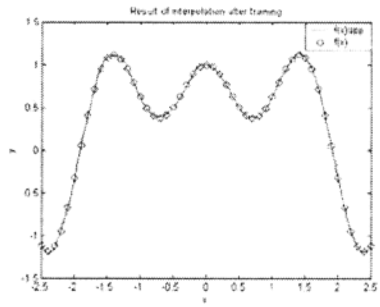

Fig.2. Result of interpolation by Mexican hat.

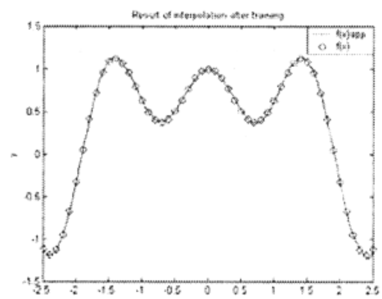

Fig.3. Result of interpolation by BW2.

Approximated functions are displayed in Figures 2 and 3. The Normalized Root Mean Square Error *NRMSE* of the Mexican hat WNN is 0.0138434 compared to 0.009344716 the BW2 WNN achieved. From these simulations we can deduce the efficiency of Beta wavelet in term of function interpolation. The table below gives the normalized square root mean square error and mean square error using traditional wavelets and Beta wavelet:

	Stepwise selection by orthogonalization algorithm	
Wavelets	NSRMSE	MSE (e-005)
Mexican hat	0.0138434	10.7061
Beta 1	0.0295078	48.6430
Beta 2	*0.00934716*	*4.88097*

Table.1 Comparison of NSRMSE and MSE for Beta wavelets and some others in term of 1-D approximation.

3.2 1-D interpolation of noisy data using the stepwise selection by orthogonalization algorithm

These results are given, using on a Neural Wavelet Network using 9 wavelets, 4 levels decomposition, 500 iterations, 50 points for training, uniform spaced points.

$$f(x)=0.5x\sin(2x)+\cos^2(2x)+\varepsilon(x) \qquad (6)$$

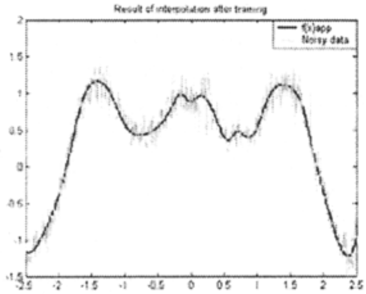

Fig.5. Result of interpolation of noisy data by Mexican hat wavelet.

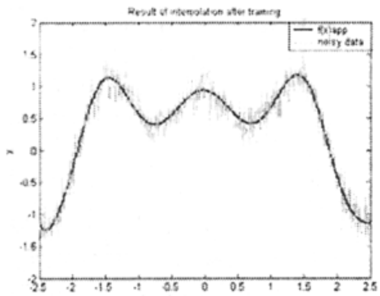

Fig.6. Result of interpolation of noisy data by BW2 wavelet.

We display in figure 5 the result of approximation of a noisy signal using Mexican hat WNN and in figure 6 the Beta WNN one. From these simulations we can see that Beta 2 WNN is more efficient than the Mexican hat wavelet on noisy data approximation.

3.3 2-D interpolation using the stepwise selection by orthogonalization algorithm

These results are given, using the Stepwise selection by orthogonalization algorithm on a Neural Wavelet Networks using 4 wavelets, 4 levels decomposition, 200 iterations, 11x11 points for training.

$$f(x_1\ x_2) = \frac{e^{-\frac{81}{16}[(x1-0.5)^2+(x2-0.5)^2]}}{3} \qquad (7)$$

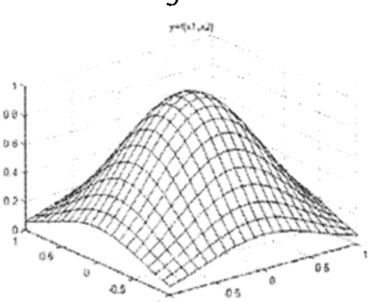

Fig.7. 2-D data to be interpolated.

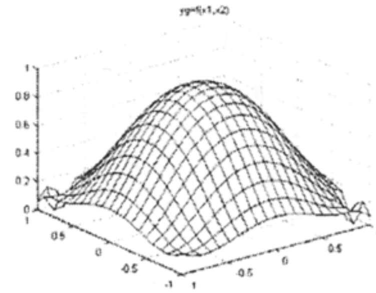

Fig.8. Result of 2-D interpolation by Mexican hat wavelet after training using uniform spaced input patterns.

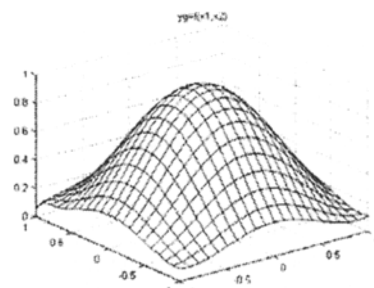

Fig.9. Result of 2-D interpolation by BW2 wavelet after training using uniform spaced input patterns.

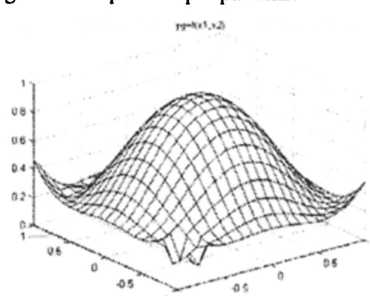

Fig.10. Result of 2-D interpolation by Mexican hat wavelet after training using randomly input patterns.

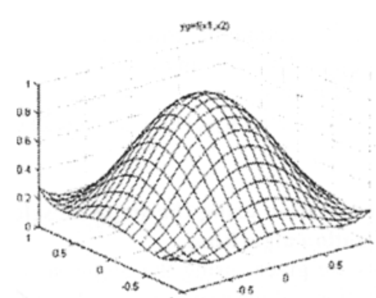

Fig.11. Result of 2-D interpolation by BW2 wavelet after training using randomly input patterns.

We display in Figure 7 the 2-D data we use in our tests, in figure 8 the result of 2-D interpolation using Mexican hat WNN on which we see some distortion in amplitude and at the edges. In figure 9 using Beta 2 WNN we reduce the amplitude distortion. In figure 10 we display the result of interpolation using Mexican hat WNN on which distortion becomes greater than its homolog using Beta 2 WNN displayed in figure 11.

4 Conclusion

We present two experimental results of the proposed Beta Wavelet Neural Networks (BWNN) on approximating two functions using the *Stepwise selection by orthogonalization* training algorithm. First, simulations on the 1-D function approximation on which we prove the superiority of Beta wavelets in term of NSRMSE. Second, the two-dimension function is approximated with the second derivative of Beta wavelet and the Mexican hat wavelet to illustrate the robustness of the proposed wavelets family. The training set D contains 11x11 uniform spaced points, and 11x11 random points, the test set V is constructed by evenly spaced 21x21 grid on [-1, 1]x[-1, 1]. So the new Beta wavelets family has the superiority of approximation in the 1-D and the 2-D case. This new wavelet family can be used to approximate volume using the 2-D 1-D 2-D technique.

Acknowledgment

The authors would like to acknowledge the financial support of this work by grants from the General Direction of Scientific Research and Technological Renovation (DGRSRT), Tunisia, under the ARUB program 01/UR/11/02.

References

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

[2] S.T. Li and E. Leiss, (2001) 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, pp. 95-124.

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

[4] W. Bellil, C. Ben Amar et M. A. Alimi, (2003) Beta Wavelet Based Image Compression, International Conference on Signal, System and Design, SSD03, Tunisia, Mars, vol. 1, pp. 77-82.

[5] W. Bellil, C. Ben Amar, M. Zaied and M. A. Alimi, (2004) La fonction Beta et ses derivees: vers une nouvelle famille d'ondelettes, First International Conference on Signal, System and Design, SCS'04, Tunisia, Mars, vol. 1, PP. 201-207.

[6] Q. Zhang and A. Benveniste, (1992) Wavelet Networks, IEEE Trans. on Neural Networks 3 (6),PP 889-898.

[7] J. Zhang, G. G. Walter, Y. Miao and W. N. Wayne Lee, (1995) Wavelet Neural Networks For Function Learning, IEEE Trans. on Signal Processing 43 (6), PP1485-1497.

[8] Q. Zhang (1997) Using Wavelet Network in Nonparametric Estimation. IEEE Trans. on Neural Networks, Vol. 8, No. 2, pp. 227-236.

[9] Y. Oussar, I. Rivals, L. Personnaz (1998) Training Wavelet Networks for Nonlinear Dynamic Input-Output Modeling Neurocomputing.

[10] C.S.Chang, Weihui Fu, Minjun Yi, (1998) Short term load forecasting using wavelet networks Engineering Intelligent Systems for Electrical Engineering and Communications 6, 217-23.

[11] J.Echauz, (1998) Strategies for Fast Training of Wavelet Neural Networks, 2nd International Symposium on Soft Computing for Industry, 3rd World Automation Congress, Anchorage, Alaska, May 10-14, 1-6.

[12] V.Kruger, Happe, A., Sommer, G (1999) Affine real-time face tracking using a wavelet network Proc. Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems, IEEE, 141-8.

[13] M. A. Alimi, (2000) The Beta System: Toward a Change in Our Use of Neuro-Fuzzy Systems. International Journal of Management, Invited Paper, pp. 15-19.

[14] V. Kreinovich, O. Sirisaengtaksin. S. Cabrera, (1992) Wavelets compress better than all other methods: a 1-dimensional theorem. University of Texas at El Paso, Computer Science Department, Technical Report.