5th Conference on Nuclear and Particle Physics
19 - 23 Nov. 2005 Cairo, Egypt

EG0700178

MULTISCALE ANALYSIS OF A FUNCTION BY NEURAL NET WORKS ELEMENTARY DERIVATIVES FUNCTIONS

A.CHIKHI, L. AIT GOUGAM, and F. CHAFA

Theoretical Physics Laboratory, Faculty of Sciences/ Physics, USTHB, B. P. 32, El- Alia, Algiers, Algeria

Abstract:

Recently, the wavelet network has been introduced as a special neural network supported by the wavelet theory . Such networks constitute a tool for function approximation problems as it has been already proved in reference . Our present work deals with this model, treating a multiscale analysis of a function. We have then used a linear expansion of a given function in wavelets, neglecting the usual translation parameters. We investigate two training operations. The first one consists on an optimization of the output synaptic layer, the second one, optimizing the output function with respect to scale parameters. We notice a temporary merging of the scale parameters leading to some interesting results : new elementary derivatives units emerge, representing a new elementary task, which is the derivative of the output task.

Keywords : neural networks, wavelets, function approximation.

I-INTRODUCTION

Neural Networks are known to be universal approximators and have been proved and studied by many authors [1]. The model establishes a link between neural methods and wavelet decomposition theory. In particular, the wavelet network has been introduced as a special feed for ward neural network supported by wavelet theory [2]. These wavelet networks use wavelet functions as response functions rather than the usual sigmoids or steps. In this paper, we treat a multiscale analysis of a function conveniently chosen. We mainly use scale parameters, neglecting the usual translation parameters in the function expansion. We investigate two training operations. The first one consists on optimizing the output synaptic weights and the second one on optimizing hidden inside the elementary tasks, " the scale parameters".

The present paper is organized as follows. In the next section, we present the model and the architecture of the neural networks. In section III, we present the two training operations, optimizing both the output synaptic weights and the scale parameters. Section IV is devoted to the presentation of our results obtained using a well known function. A short discussion concludes this paper.

II-MODEL AND ARCHITECTURE

Consider an input X which must be processed into an output (a task) F(X). Consider now neural elementary units which receive the same input X. Each unit returns an output f which depends on two parameters: the translation parameter b and the scale one λ. Output synaptic weights $\omega(b, \lambda)$ linearly regroup these elementary outputs into a global output F(X). We have then the expansion [3]

$$F(X) = \int db\, d\lambda\, \omega(b, \lambda) f((X - b)/\lambda) \qquad (1)$$

As used in Ref. [1], we discretise equation (1) with N units and neglect the translation parameter. We obtain

$$F_{app}(X) = \sum_{i=1}^{N} \omega(\lambda_i) f(X / \lambda_i) \qquad (2)$$

III-TRAINING

III.1- Output Weights

To define the best "F_{app}", we must minimize the square norm of the error ($F-F_{app}$), first in terms of the output weights "ω_i" and second of the scale parameters λ_i. In terms of the ω_i's, this consists in solving the equation

$$\frac{\partial}{\partial \omega_i}\left(\left\langle F - F_{app} \middle| F - F_{app} \right\rangle\right) = 0 \qquad (3)$$

Using equation (2) and adopting short notation, we obtain

$$\frac{\partial}{\partial \omega_i}\left(\langle F|F\rangle - 2\sum_i \omega_i \langle f_i|F\rangle + \sum_{j,k=1}^{N} \omega_j \langle f_j|f_k\rangle \omega_k\right) = 0, \quad i = 1,\dots,N \qquad (4)$$

The solution of the above equation is given by

$$\omega_i = \sum_{j=1}^{N} (g^{-1})_{ij} \langle f_j|F\rangle, \quad i = 1,\dots,N \qquad (5)$$

where g is the matrix with elements $g_{jk} = \langle f_j | f_k \rangle$

III.2- Elementary Tasks

Let now minimize the square norm $\varepsilon = \left\langle F - F_{app} \middle| F - F_{app} \right\rangle$ in terms of scale parameters λ_i. As demonstrated in Ref. [4], we need only the derivatives of f_i with respect to their scales λ_i and there is no need to use those of ω_i. We have thus

$$\frac{\partial \varepsilon}{\partial \lambda_j} = \frac{2\omega_j}{\lambda_j^2}\left\langle X f'(X/\lambda_j) \middle| F - F_{app} \right\rangle = 0, \quad j = 1,\dots,N \qquad (6)$$

where f' is the derivative of the elementary task.

IV- DISCUSSION OF RESULTS

For our numerical applications, we consider the implementation of the function "the target task" given by

$$F(X)=\frac{1}{10}e^{-X/10}(\frac{2}{3}\tanh[4(X-3/2)]-4\tanh[4(X-10)])$$

As a wavelet function, we choose a "Mexican hat" for the elementary task of a formal neuron given by

$$f(x)=(1-x^2)e^{-x^2/2}$$

We start the learning process by minimizing the square norm of the error in terms of the synaptic weights ω_i, by satisfying equation (5). Let take N=5. We start with the initial values 1/4, 1/2, 1, 2, 4 for the λ_i's as it is often found in the literature that a traditional sequence $\lambda_i \cong 2^{i-(N+1/2)}$ is a good choice for a start. We then use the values of ω_i to find the new values of the scale parameters minimizing the error ε.

After around 75 steps, a saturation of $\|F_{app}\|^2$ occurs as illustrated in figure (1). A close comparison between F and F_{app} is provided by figure (2). It is worth noticing that F_{app} fits well the function F. On the other hand, we illustrate in figure (3) the evolution of $Log_2(\lambda_i)$. During learning, we notice that while the saturation is confirmed, the convergence of the λ_i's turns out to be slower. After n= 500 , the values of the λ_i's and ω_i's read as (0.25, 0.9452, 0.9699, 3.1825, 10.8625) and (-0.0029, 1.8742, -1.8684, -0.0411, 0.3832) respectively. The evolution of λ_i exhibits a very interesting and unexpected result. A merging between λ_2 and λ_3 occurs. The equivalent representations $\omega_i f_i + \omega_j f_j = (\omega_i + \omega_j)(f_i + f_j)/2 + (\omega_i - \omega_j)(f_i - f_j)/2$ can be used. As $\lambda_2 \cong \lambda_3$, it is the same for f_2 and f_3 and thus the functional basis contains a new elementary response $\frac{\partial f}{\partial \lambda}$. These new neural units which spontaneously emerge are "derivative elementary tasks" and may represent a new task $\frac{\partial F}{\partial \lambda}$.

V- CONCLUSION

To conclude, we have investigated neural networks whose response functions are wavelets rather than traditional steps or sigmoids. We have examined the universal approximation property which shows that the wavelet model may approximate any function. We have then noticed the occurrence of a very important result: the emergence of derivative elementary tasks allowing us to approximate simultaneously a function along with its first order derivative.

REFERENCES

[1] K. Hornik, M. Stinchcombe, and H. White, Multilayer Feed forward Networks are Universal Approximators, Neural Networks 2 , 359 (1989).

[2] Y. Oussar and G. Dreyfus, Initialisation by selection for wavelet Network Training, Neurocomputing 34, 131 (2000).

[3] S. Mallat, A Theory for multiresolution signal decomposition: the wavelet transform, IEEE Trans. Pattern Anal. Machine 11, 674 (1989).

[4] B. G. Giraud and A. Touzeau, Elementary Derivative Tasks and Neural Net Multiscale Analysis of tasks, Phys. Rev. E 65, 016109 (2001).

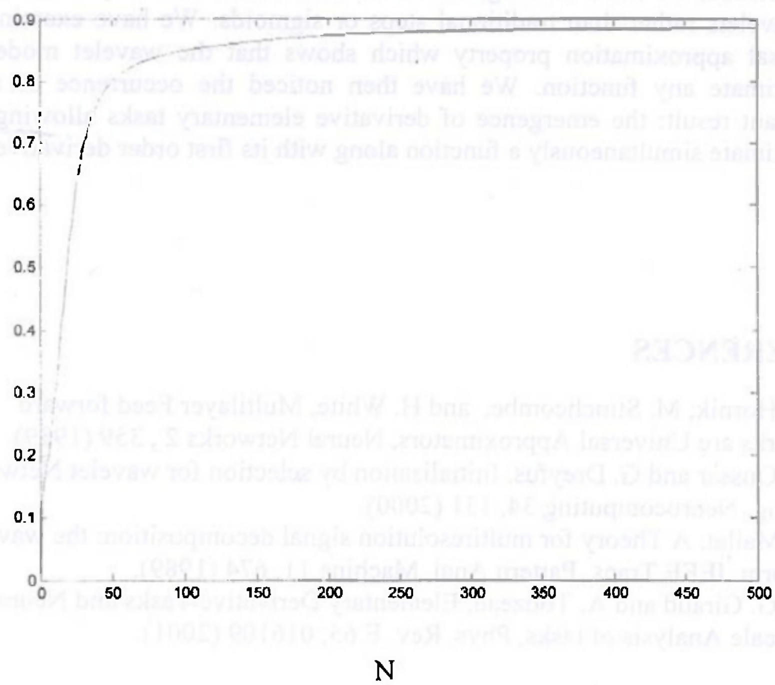

Figure 1: Plot of $|F_{app}|$ versus N.

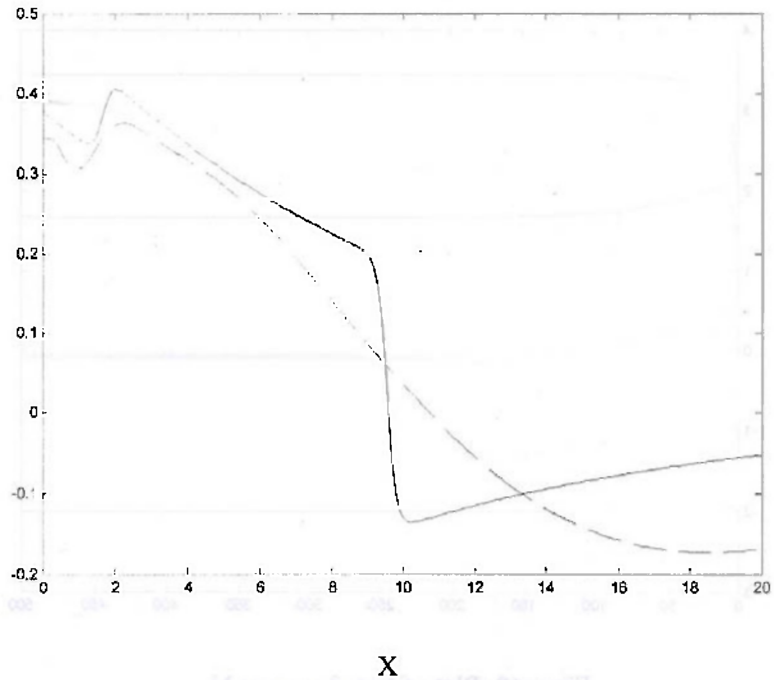

Figure 2: Plot of F versus X.

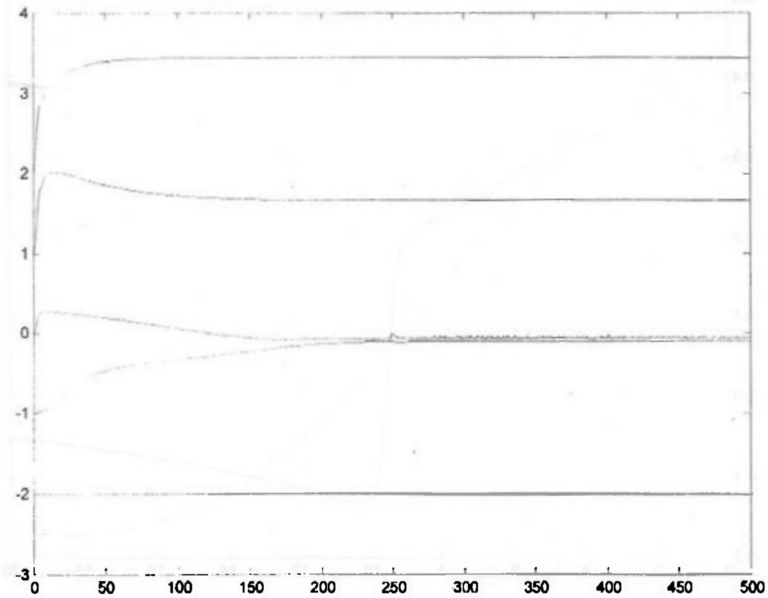

Figure 3: Plot of $Log_2\lambda_i$ versus N.