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

EG0700177

NUMERICAL ANALYSIS OF DIFFERENT NEURAL TRANSFER FUNCTIONS USED FOR BEST APPROXIMATION

L. Ait Gougam , A. Chikhi, S. Biskri, and F. Chafa

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

Abstract :

It is widely recognised that the choice of transfer functions in neural networks is of crucial importance to their performance. In this paper, different neural transfer functions used for approximation are discussed. We begin with sigmoïdal functions used most often by different authors . At a second step, we use Gaussian functions as previously suggested in reference. Finally, we deal with a specified wavelet family. A comparison between the three cases cited above is made exhibiting therefore the advantages of each transfer function. The approached function improves as the dimension N of the elementary task basis increases.

Key words: Neural Networks, wavelets, approximation.

I-INTRODUCTION

Neural transfer functions are of crucial importance to the performance of neural networks. There is a growing understanding that the choice of transfer functions is at least as important as the network architecture and learning algorithm.

The first transfer functions proposed are "sigmoidal functions". It has been proved by different authors that they can approximate an arbitrary continuous function on a compact domain with arbitrary precision given sufficient number of neurons. Despite the fact that sigmoids are the most common used functions, there is no a priori reason why they should be optimal in all cases. For this purpose, several transfer functions have been proposed such as Gaussian functions, Lorentzian functions used by Giraud et al. [1] treated as simplified Gaussian ones.

In this paper, we investigate various transfer functions for neural networks. We examine, for this, the universal approximation theory by using three types of transfer functions. We begin with sigmoidal function. As a second step, we use a Gaussian one and finally we deal with a specified wavelet family. We make then a comparison between the three choices in that case of a well known function to approximate. In the next section, we present the different transfer functions used in this paper. In section III, we give a brief description of the network architecture by presenting the two training operations used to find the best approximation of the chosen function. Numerical applications and discussions are the subject of section IV. Finally, Section IV contains our conclusions.

II-NEURAL TRANSFER FUNCTIONS

The first neural networks models proposed were based on a step function as transfer function. The latter is also known as the Heaviside function. Later, this function was generalized to the sigmoidal functions leading to the graded response neurons used most often in the literature

$$S(I,s)=\frac{1}{1+e^{-I/s}}$$

where s determines the slope of the sigmoidal function around the linear part.
The second transfer function used in this paper is the well known one dimensional Gaussian given by

$$G(x,\sigma)=e^{-x^2/\sigma^2}$$

The wavelet decomposition theory coming from signal processing has been the subject of a complete investigation by different authors. They use different families of wavelets to study several problems of wavelet networks. In this paper, we make choice of a "Mexican hat" for the wavelet function to use as transfer function given by

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

III-NETWORK ARCHITECTURE

Consider an input X which must be processed into an output (a task) F(X). Consider now neural units which may be excitatory- inhibitory pairs of neurons providing a window- like elementary response. They may also be more complicated assemblies of neurons providing a more elaborate function such as sigmoidal function or a wavelet one. We denote by f(x) the response function of such a unit. We have then the following expansion [2]

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

where b and λ are two parameters. The integral above is most often reduced to a discrete sum. We have then

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

where we have used a seemingly poorer but simpler expansion which does not use the second parameter b. The central issue of any theory of neural nets is to find the values of the synaptic weights ω_i which are best suited for a given task. A training algorithm, optimizing the task is used. First in terms of the synaptic weights ω_i and second in terms of a parameter λ. For this, we minimize the square norm of the error ε given by

$$\varepsilon = \langle F - F_{app} | F - F_{app} \rangle \qquad (3)$$

in terms of ω_i and in terms of λ_i. We obtain the two equations [3]

$$\omega_i = \sum_{j=1}^{N} (g^{-1})_{ij} \langle f_j | F \rangle, \quad i = 1,\dots,N \qquad (4)$$

$$\frac{2\omega_j}{\lambda_j^2}\left\langle Xf'(X/\lambda_j)\middle| F-F_{app}\right\rangle = 0,\ \ j=1,\dots,N \qquad (5)$$

where g is the matrix with elements $g_{jk} = \langle f_j | f_k \rangle$ and f' the derivative of the elementary task.

IV- NUMERICAL APPLICATIONS AND DISCUSSIONS

Consider the implementation of the function given by

$$F(X) = (\tanh(x/10)+\cos(4x/10))/(4.8486)$$

We try to find the best approximation of this function using neural networks. We begin with a sigmoidal as a transfer function in the expansion of F_{app} given by equation (2). Figure 1 illustrates the comparison between the exact function F and its best approximation. It is worth to note that the parameter λ used in equation (2) in the case of sigmoidal functions is the slope s around the linear part of sigmoids. The two curves are close to each other in the beginning domain. Then F deviates from F_{app}. The comparison between F and its best approximation using Gaussian functions is provided by figure 2 after learning saturation. The figure exhibits an improvement of the best approximation of F(X). This is due to the form of Gaussian functions which is more appropriate than sigmoids to approximate the chosen function. Finally, in figure 3 we compare F with F_{app} expanded on the base of a specified wavelet family "the Mexican hat". We note a net improvement when compared with the two first cases and we can see that F_{app} fits well the chosen function to approximate.

V-CONCLUSION

In this paper, we investigated the universal approximation theory of neural networks made of "artificial neurons" whose response functions are: first, sigmoids, then gaussian and finally a specified family of wavelets. It is worth noticing that in the case of the chosen function, wavelets are more appropriate to use as "response function" to find the best approximation. None of the functions mentioned above is flexible enough to describe an arbitrarily shaped function and be optimal in all cases. It depends widely on the form of the function to approximate.

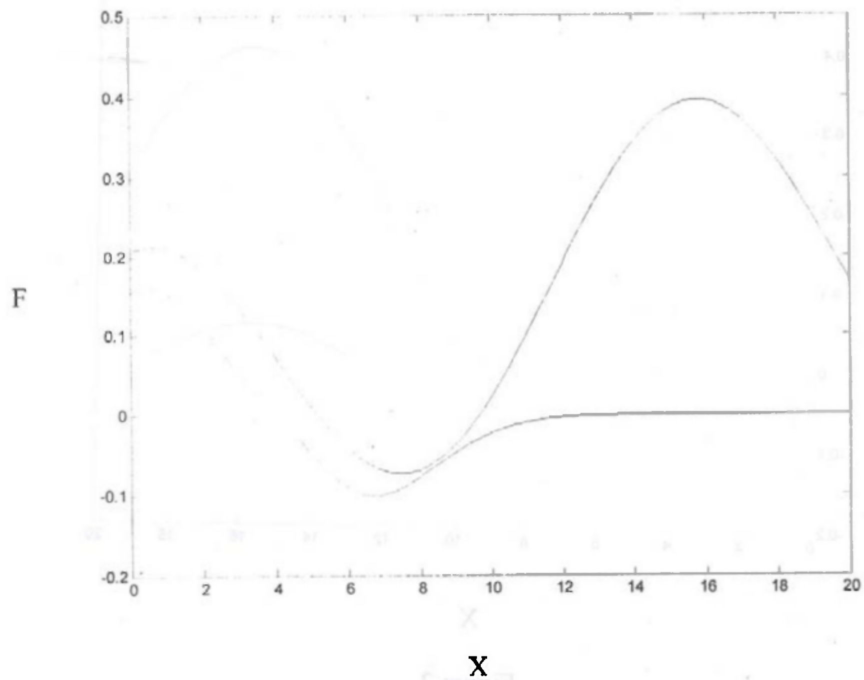

Figure 1

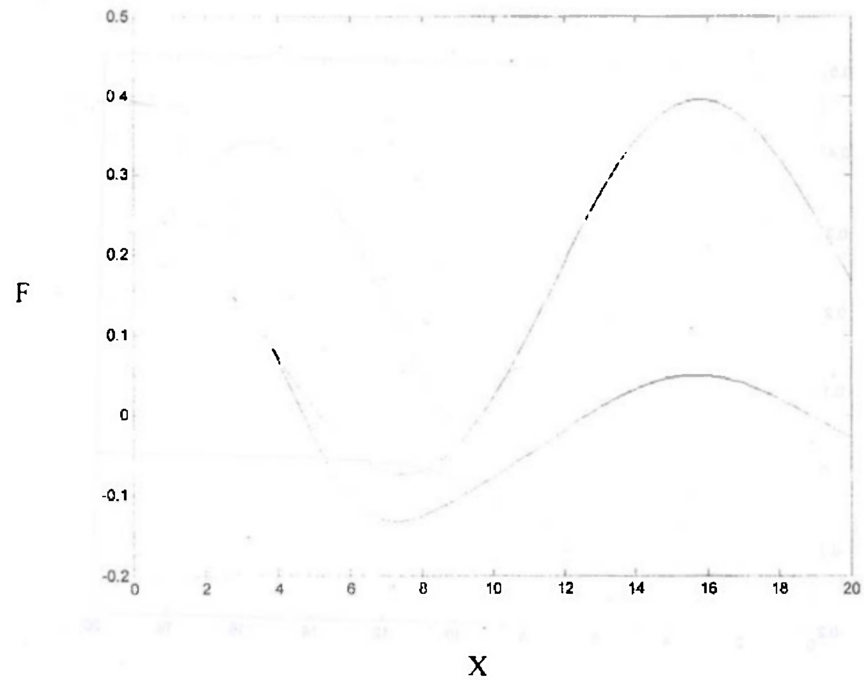

Figure 2

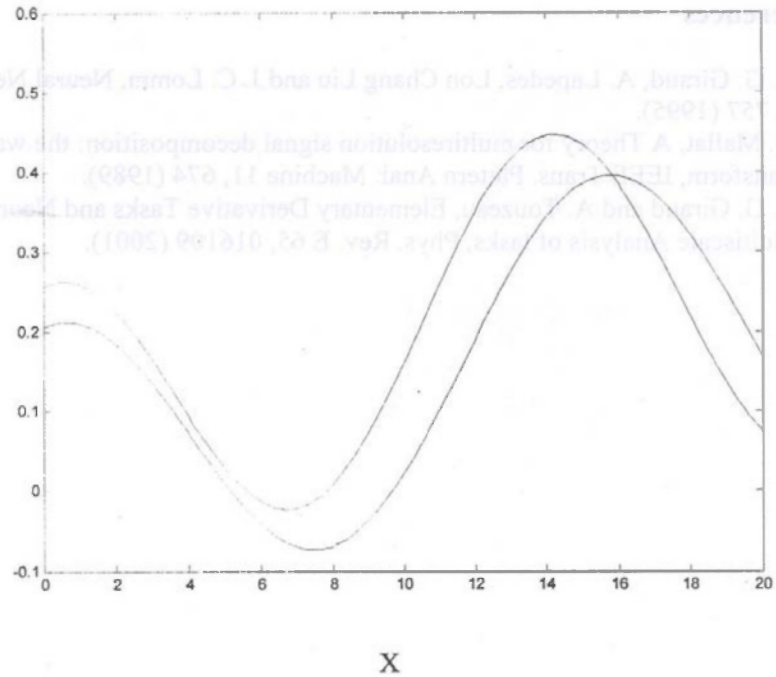

Figure 3

References

1- B. G. Giraud, A. Lapedes, Lon Chang Liu and J. C. Lomm, Neural Networks, 8, 757 (1995).
2- S. Mallat, A Theory for multiresolution signal decomposition: the wavelet transform, IEEE Trans. Pattern Anal. Machine 11, 674 (1989).
3- B. G. Giraud and A. Touzeau, Elementary Derivative Tasks and Neural Net Multiscale Analysis of tasks, Phys. Rev. E 65, 016109 (2001).