Journal of Engineering and Applied Sciences 13 (Special Issue 10): 8153-8158, 2018
ISSN: 1816-949X

Radial Basis Functions Neural Networks Convolution Approximation

Eman S. Bhaya and Omar A. Al-Sammak
College of Education for Pure Sciences, University of Babylon, AL Hilla, Babil, Iraq

Abstract: If we have a function defined on the real line we cannot approximate this function using a radial bases function neural networks to get an example. It mean we cannot find a radial base forward neural network to approximate a continuous function f. To make this possible we shall put some limits on f. In this study, we shall study approximation of functions in Lp spaces for p>1 defined on the real line using radial base neural network. The weights are fixed in the radial bases functions neural networks to have facilities in practical applications and prove direct theorem using radial basis function neural networks for functions in Lp spaces for p>1.

Key words: Neural network approximation, modulus of smoothness, Lp-spaces, best approximation, facilities, basis function

INTRODUCTION

In order to draw a meaningful picture in our minds for approximation by neural network and prepare the background for our reserach and motivate our results, we have to recall some definitions and results related to basic concepts of our reserach.

The origin point of artificial intelligence can be traced to 1930-1940's For a half century it has achieved very good achievements and it has little difficulties, we a table introduce for a brief history of the improvement of the artificial neural networks. The main aim of the artificial intelligence is to make a computer model to simulate the intelligent of the human brain and even animals brain and behavior. The main tasks of the artificial intelligence can be summarize in the following:

- Representing and storing knowledge
- Solving many kinds of problems with storing knowledge

A acquiring new knowledge at the system running. The above tasks need 50 years for developing and it has many and widely applications such as for expert systems, machine learning, logic reasoning natural language and theorem proving

First let us turn the light to the biological neuron. The brain is a complicated connected networks, consists of billions nerve cells (neurons). The human brain has 10^{10}-10^{11} neurons each one connected with 10^{3}-10^{5} other neurons. The neuron structure divided in to three parts as described in Fig. 1.

To one side of the soma, we find dendrites and to the other side, we find the axon. By dendrites of the axons,

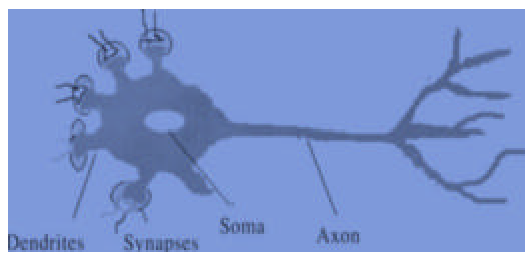

Fig. 1: Biological neuron

the neurons can be connected, the connection places of two neurons dendrites called synapses. The output places of the electrical transmission is the branches of the axon and the other dendrites of the neuron is the input places. The synapses make a weighted processes at the input signal. Then, each input signal undergoes a summation and nonlinear activation at the back of the soma of neurons. Under a condition for example, the intensity of the summeucl signals a certain level, it output signal. Then, this signal transferred to other neurons and then to the next processing. The deity of synapses in the neurons is not only the transformation of signals but it have experience memory function and can carry out weighted processing on the input signal according to memory. Let us now summarize, the differences between the processing of biological brain system and neumann architecture (Xingui and Shaohua, 1995). The biological brain does not have arithmetic unit, the neuron combines the function of the arithmetic unit. Some information are stored in the synapses, while some of the information are processed by numerous neurons.

The biological brain does not need a model and program for solving practical problems but it by learning, directly changes the connection weights what we called

Corresponding Author: Eman S. Bhaya, College of Education for Pure Sciences, University of Babylon, AL Hilla, Babil, Iraq

the synapses to get the knowledge for solving problems. The information (a processing object) processed by the biological brain is not completely certain and accurate but has obvious fuzziness and randomness. The processing object is either continuous or discrete quantum.

The biological brain using analog method, a digital/analog method or mixed in addition to the random method in processing the information. While computer use all those methods in addition to that its process is complex and non-reparable. The biological brain can a certain response to an activation in less than one second Let us describe, mathematical model of artificial neuron (Xingui and Shaohua, 1995) (Fig. 2).

x_i is the input signal to a neuron j, i = 1, 2, 3, ..., nw_{ij} is the connection weight between the ith neuron and the neuron j; θ_j is the activation threshold of the neuron j, f is the effect function. Y_j is the output of the neuron. We can relate the input and the output of the neuron by the following identity:

$$y_i = f\left(\sum_{i=1}^{n} w_{ij} x_i - \theta_J\right)$$

The simulation above has two disadvantages:

- There is no time delay between the inputs and the output information
- The outputs information only depends on the input information. And not on the earlier input in formation

Radial functions are a special class of function. Their characteristic feature is that their response decreases (or increases) monotonically with distance from a central point, for example, Funahashi (1989) and Hornik *et al.* (1989).

The centre, the distance scale and the precise shape of the radial function are parameters of the model all fixed if it is linear. A typical radial function is the Gaussian which in the case of a scalar input is:

$$h(x) = e^{\left(-\frac{(x-c)^2}{r^2}\right)}$$

Its parameters are its center c and its radius r . A Gaussian RBF monotonically decreases with distance from the Centre. In contrast, a multiquadric RBF which in the case of scalar input is:

$$h(x) = \frac{\sqrt{r^2 + (x-c)^2}}{r}$$

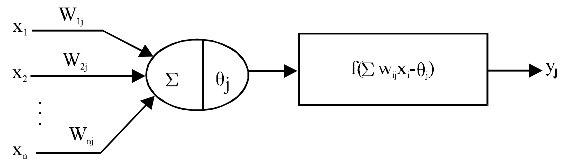

Fig. 2: Artificial neuron model

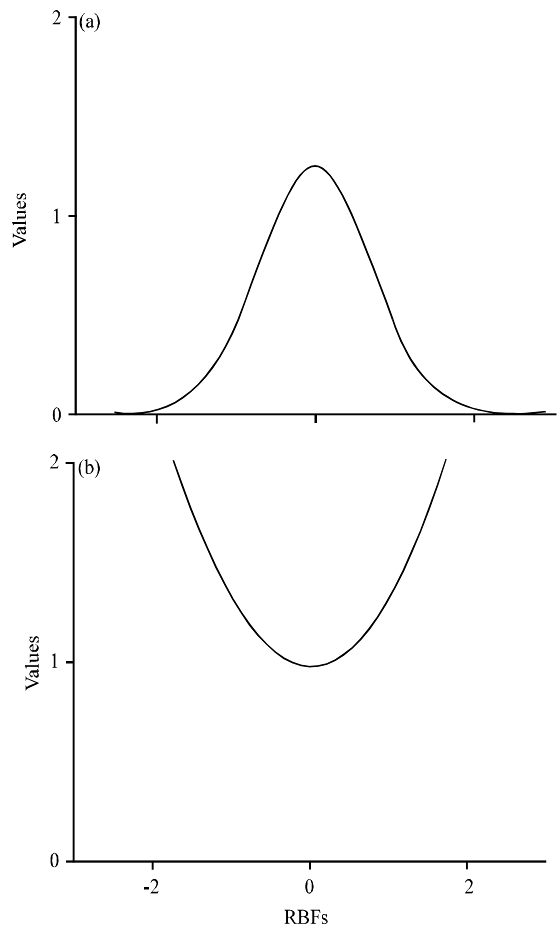

Fig. 3: a) Gaussian left and b) Multiquadric RBFs

more commonly used than multiquadric-type RBFs which have a global response. They are also more biologically plausible because their response is finite (Fig. 3a and b) a radial basis function network is an artificial neural network that uses radial basis functions as activation functions. The output of the network is a linear combination of radial basis functions of the inputs and neuron parameters. Radial basis function networks have many uses, including function approximation, time series prediction, classification and system control.

Radial functions are simply, a class of functions. In principle they could be employed in any sort of model (linear or nonlinear) and any sort of network (single layer or multi-layer) (Fig. 4).

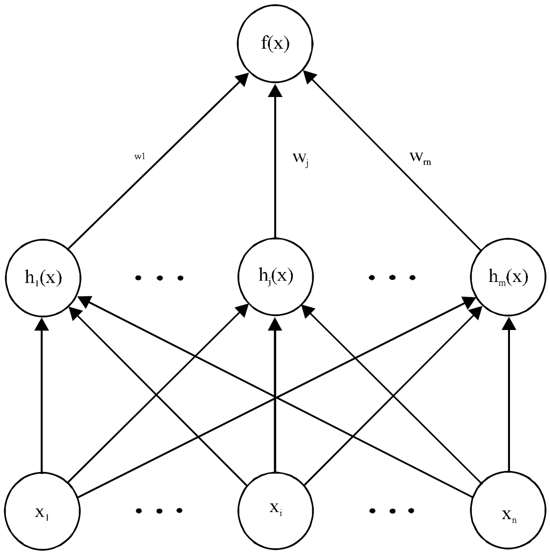

Fig. 4: The radial basis function network

Artificial neural network have many applications in various types of sciences fields and engineering for example. There are many papers introduced about the direct and inverse theorem for the approximation by neural networks called the upper and lower bounds of the rate of approximation. For example, Bhaya and Sammak (2016), Scarselli and Tsoi (1998), Hornik *et al.* (1989), Leshno *et al.* (1993) and Mhaskar and Micchelli (1992), we call the degree of the asymptotically identical upper and lower bounds, the essential rate of approximation. If we have a continuous function with multivariable and compact domain subset of R^d there exist a feed Forward Neural Networks (FNNs) as an approximation for it.

Artificial forward neural networks are nonlinear parametric expressions representing multivariate numerical functions. In connection with such paradigms there arise mainly three problems: a density problem, a complexity problem and an algorithmic problem. The density problem deals with the following question: which functions can be approximated and in particular can all members of a certain class of functions be approximated in a suitable sense. This problem was satisfactorily solved in the late 1980's (Bhaya and Sammak, 2016; Li, 2008; Hornik *et al.*, 1989). The forward neural network with 3 layers and d input units, 1 hidden and one output units can be written mathematically as:

$$N_n(x) = \sum_{i=1}^{n} c_i \sigma(\langle \omega_i, x \rangle + \theta_i),\ x \in R^d, d \geq 1$$

where, $1 \leq I \leq n$, $\theta_i \in R$ is the threshold:

$$\omega_i = (\omega_{i1}, \omega_{i2}, ..., \omega_{id})^T \in R^d$$

are connection weights of neuron i in the hidden layer with input neurons, $c_i \in R$ are the connection strength of neuron i with the output neuron and σ is the sigmoidal activation function used in the network. There are many papers introduced about approximation using radial basis function neural network function, for example, Bateman and Erdelyi (1974), Bhaya and Sammak (2016), Liu and Si (1994), Chui and Li (1992), Li (2008), Scarselli and Tsoi (1998), Funahashi (1989), Hornik *et al.* (1989), Leshno *et al.* (1993), Li (1998), Mhaskar and Micchelli (1992), Hahm and Hong (2004), Park and Sandberg (1991) and Kurkova (1995).

By Bhaya and Sammak (2016), Eman introduced an operator called modified Dunkl transform, then she used it to introduce a version of K-functional and a modulus of smoothness of function using these moduli Eman studied the regular neural network approximation.

By Liu and Si (1994), Binfan and Jennie showed that for any function have two continuous derivative, there exists a radial basis function neural network as a best approximation with distance equal to zero on $[0,1]^m$.

By Funahashi (1989) Li proved that for any multivariate function with its all derivatives we can find a radial basis function neural network as a simultaneous best approximation. Also, he proved that the degree of best approximation was inversely proportional to the number of hidden neurons.

MATERIALS AND METHODS

Approximation by rbf neural networks with a fixed weight in $L^o_p(K)$: If we have a function defined on the real line we cannot approximate this function using RBF neural network. By Chen and Chen (1995) you can find an example:

$$f(x) = x, x \in R \text{ and } |f(x)| \to \infty \text{ as } |x| \to \infty$$

It mean, we cannot find a radial base forwarded neural network to approximate a continuous function f . To make this possible we shall put some limits on f. In this study, we shall study approximation of functions in L_p spaces for $p \geq 1$ defined on the real line using real base neural network. $L^{\circ}_p(K)$ denote to the collection of all functions $f \in L_p(K)$ such that f(x) converge to 0 as $|x| \to \infty$.

Theorem a: Let σ be an RBF on R and given ∈>0, if $f\in L^*_p(K)$ then there exist a constants $\theta_i\in R$ and W, M, K positive integers satisfying:

$$\| f(x)-\sum_{i=0}^{2M^2-1}\Delta^K_{\frac{1}{m}}(f,x_i)\sigma(Wx-\theta_i)\|_p\leq c(p)\left(\in+\frac{2M^2-1}{(2M)^p}\omega_k(f,\frac{1}{M})_p\right)$$

Proof//: Since, $f\in L^*_p(K)$, so that, we can find a positive integer M, K satifying $|f(x)|<\in/2M$ for $|x|\geq K$. Let [-M, M] be a closed interval. Define a portion to the interval [-M, M], that divide it in to into $2M^2$ equal segments each of length 1/M.

$$-M = x_o < x_1 <, ..., < x_{2M^2} = M$$

and:

$$\theta_i = \frac{x_i + x_i + 1}{2}, (0\leq i\leq 2M^2-1)$$

Since, σ is an RBF then, we can find a positive integer:

$$L \ni |\sigma(x)| < \frac{1}{2M} \text{for} |x| \geq L$$

Then, we can choose a positive integer W such that W/2M>L. Now, let us define a neural network as :

$$N(x)=\sum_{i=0}^{2M^2-1}\Delta^K_{\frac{1}{M}}(f,x_i)\sigma(W(x-\theta_i)$$

Since $|x|\geq M$, so that, $|W(X-\theta_i)|\geq L$, therefore:

$$|\sigma(W(x-\theta_i)|<\frac{1}{2M}$$

when i = 1, 2, ..., $2M^2$-1. Which implies:

$$|f(x)-N(x)|\leq|f(x)|+|N(x)|$$

$$\leq\left|\frac{\in}{2}\right|+\left|\sum_{i=0}^{2M^2-1}\Delta^K_{\frac{1}{M}}(f,x_i)\sigma(W(x-\theta_i)\right|$$

$$\leq\frac{\in}{2}+\sum_{i=0}^{2M^2-1}\left|\frac{\Delta^{K(f,xi)}_{\frac{1}{M}}}{2M}\right|$$

Then:

$$|f(x)-N(x)|^p\leq\left(\frac{\in}{2}\right)^p+\frac{1}{(2M)^p}\sum_{I=0}^{2M^2-1}\left|\Delta^K_{\frac{1}{M}}(f,x_i)\right|^p$$

So:

$$\int_{-M}^{M}|f(x)-N(x)|^p dx\leq\int_{-M}^{M}\left(\frac{\in}{2}\right)^p dx+\frac{1}{(2M)^p}\int_{-M}^{M}\sum_{i=0}^{2M^2-1}\left|\Delta^K_{\frac{1}{M}}(f, x_i)\right|^p dx$$

Therefore:

$$\left(\int_{-M}^{M}|f(x)-N(x)|^p dx\right)^{\frac{1}{p}}\leq$$

$$\left(\int_{-M}^{M}\left(\frac{\in}{2}\right)^p dx\right)^{\frac{1}{p}}+\left(\frac{1}{(2M)^p}\int_{-M}^{M}\sum_{i=0}^{2M^2-1}\left|\Delta^K_{\frac{1}{M}}(f, x_i)\right|^p dx\right)^{\frac{1}{p}}$$

$$\leq\left(\int_{-M}^{M}\left(\frac{\in}{2}\right)^p dx\right)^{\frac{1}{p}}+\left(\frac{2M^2-1}{(2M)^p}\int_{-M}^{M}\left|\Delta^K_{\frac{1}{M}}(f, x_i)\right|^p dx\right)^{\frac{1}{p}}$$

Which implies:

$$\|f(x)-N(x)\|_p\leq c(M)\left(\varepsilon+\frac{2M^2-1}{(2M)^p}\left\|\Delta^K_{\frac{1}{M}}(f, x)\right\|_p\right)$$

by taking supermom for two sides of the inequality, we get:

$$\|f(x)-N(x)\|_p\leq c(M)\left(\in+\frac{2M^2-1}{(2M)^p}\omega_k\left(f,\frac{1}{M}\right)\right)$$

RESULTS AND DISCUSSION

Approximation by RBF neural networks using convolution: Using our theorem in the previous section and convolution, we introduce an RBF neural networks approximation of function direct theorem. The convolution of two functions f and g defined by:

$$(f*g)(x)=\int_R f(y)(x-y)dy \quad (1)$$

For a real x define:

$$f(x)=\begin{cases}de^{\frac{-1}{1-x^2}} & \text{if}|x|<1\\ 0 & \text{if}|x|\geq 1\end{cases} \quad (2)$$

we choose d to make the integration of F on R equal to 1. So, $F\in L^*_p\infty(K)$ where, $L_p\infty(K)$ referee to the space of functions in L^*_p and have any derivatives. k.

Let us define:

$$F_k(x) = kF(kx) \quad (3)$$

Therefore, the integration of F_k on R equal to 1 and $F_k(x) \in L^{\circ}_{p}\infty(K)$. Now using the convolution to get:

Theorem b: If $f \in L^{\circ}_p(R)$, $p \geq 1$. Then:

$$\|F_k * f - f\|_p \leq c(k)\omega_1(f, \delta)_p$$

Proof//: and since, $(k \leq M)$ k and any integer $x \in R$ for any:

$$\int_R F_k(y)dy = 1$$

we have:

$$\|F_k * f - f\|_{Lp(IR)} = \left\|\int_R F_k(y)f(x-y)dy - \int_R F_k(y)f(x)dy\right\|_{Lp(IR)}$$

using the property that:

$$F_k(x) = kF(x)$$

we have:

$$\|F_k * f - f\|_{Lp(IR)} = \left\|\int_R kF(ky)(f(x-y) - f(x)dy\right\|_{L_p(IR)}$$

If we assume:

$$z = \frac{ky}{\delta}, \delta > 0$$

then:

$$y = \frac{\delta z}{k}$$

and:

$$dy = dz$$

then:

$$\|Fk * f\|_{L_p(IR)} = \left\|\int_R kF(z)(f\left(x - \frac{z}{k}\right) - f(x)dz\right\|_{L_p(R)}$$

$$\leq \left\|\int_R kF(z)(f\left(x - \frac{z}{k}\right) - f(x)dz\right\|_{L_p(R)}$$

$$\leq \left\|\int_R F(z)\Delta^1_{\left|\frac{z}{k}\right|}(f, z)dz\right|_{Lp(IR)}$$

$$\leq c(p)\int_R F(z)dz\left\|\Delta^1_{\left|\frac{z}{k}\right|}f\right\| L_p(R)$$

$$\leq c(p)\omega_1(f, \delta)_p$$

Theorem c: If $f \in L_p[a, b]$ and σ is a RBF. Then, there exist a constant $y_i \in R$ and W, M, $K \in Z^+$ satisfying:

$$\left\|f(x) - B\sum_{i=0}^{2M^2-1}\Delta^K_{\frac{1}{M}}(f, x_i)\sigma(k(x-y_i)\right\|_p \leq$$

$$c(M, k)\left(\omega_1(f, \delta)_p + \frac{2M^2-1}{(2M)^p}\omega_k\left(f, \frac{1}{M}\right)\right)$$

Proof //: We construct the function $\tilde{f}$ on R as follows

$$\tilde{f}(x) = \begin{cases} f(a)x + (a-1)f(a) & \text{if } x \in [a-1, a] \\ f(x) & \text{if } x \in [a, b] \\ -f(b)x + (b+1)f(b) & \text{if } x \in [b, b+1] \\ 0 \quad \text{if } x \in (-\infty, a-1) \cup (b+1, \infty) \end{cases}$$

From theorem B, we have:

$$\|F_k * f - f\|_p \leq c(p)\omega_1(f, \delta)_p$$

We know that, since:

$$\int_R F_k(x-y)\tilde{f}(y)dy < \infty$$

$$\left(F_k * \tilde{f}\right)(x)$$

can be approximated by a Riemann sum. So that, for any natural k, we can find reals, M_k and y_i, c_i for i = 1, 2, ..., M_k satisfying:

$$\left\|(F_k * f)(x) - \sum_{i=1}^{M_k} c_i F_k(x-y_i)f(y_i)\right\|_p$$

$$\leq c(k)\omega 1(f, \delta)_p \quad (4)$$

From Eq. 2 and 3, we get:

$$F_k \in L^{\circ}_p$$

If we assume:

$$\sum_{i=1}^{Mk} c_i \tilde{f}(y_i) = B$$

our first theorem implies:

$$\left\|Fk(x-yi) - \sum_{i=0}^{2M^2-1}\Delta^K_{\frac{1}{M}}(f, x_i)\sigma(Wx - \theta_i)\right\|_p \leq$$

$$c(M)\left(\in+\frac{2M^2-1}{(2M)^p}\omega_k\left(f,\ \frac{1}{M}\right)\right) \quad (5)$$

Using theorem B, we get:

$$\|F_k*f-f\|_p \le c(k)\omega_1(f,\delta)_p$$

From above we have:

$$\left\|f(x)-\sum_{i=1}^{M_k}c_i\tilde{f}(y_i)\sum_{i=0}^{2M^2-1}\Delta_{\frac{1}{M}}^{K}(f,x_i)\sigma(W(x-y_i)\right\|_p$$

$$\le c(M,k)\left(\|F_k*f-f\|_p+\left\|F_k*f-\sum_{i=1}^{M_k}c_i F_k(x-y_i)\tilde{f}(y_i)\right\|_p\right)$$

$$\le c(M,k)\left(\omega_1(f,\delta)_p+\left(\in+\frac{2M^2-1}{(2M)^p}\omega_k\left(f,\frac{1}{M}\right)\right)\right)$$

which is true for any $\in>0$. So:

$$\left\|f(x)-B\sum_{i=0}^{2M^2-1}\Delta_{\frac{1}{M}}^{K}(f,x_i)\sigma(k(x-y_i)\right\|1\le$$

$$c(M,k)\left(\omega_1(f,\delta)_p+\frac{2M^2-1}{(2M)^p}\omega_k\left(f,\frac{1}{M}\right)\right)$$

CONCLUSION

We can not approximate a function in Lp spaces for p>1, using a radial bases function neurl network. If we put limits on the target function f in Lp, p>1, we can find a radial base neural network.

REFERENCES

Bateman, H. and A. Erdelyi, 1974. Higher Transcendental Functions. Vol. 2, McGraw-Hill, New York, USA.,.

Bhaya, E. and O.A. Sammak, 2016. Approximation by regular neural networks in terms of Dunkl transform. Res. J. Appl. Sci., 11: 933-941.

Chen, T.P. and H. Chen, 1995. Approximation capability to function of several variables, nonlinear functionals and operators by radial basis function neural networks. IEEE. Trans. Neural Netw., 6: 904-910.

Chui, C.K. and X. Li, 1992. Approximation by ridge functions and neural networks with one hidden layer. J. Approximation Theor., 70: 131-141.

Funahashi, K.I., 1989. On the approximate realization of continuous mappings by neural networks. Neural Networks, 2: 183-192.

Hahm, N. and B.I. Hong, 2004. An approximation by neural networks with a fixed weight. Comput. Math. Appl., 47: 1897-1903.

Hornik, K., M. Stinchcombe and H. White, 1989. Multilayer feedforward networks are universal approximators. Neural Networks, 2: 359-366.

Kurkova, V., 1995. Approximation of functions by perceptron networks with bounded number of hidden units. Neural Netw., 8: 745-750.

Leshno, M., V. Lin, A. Pinks and S. Schochen, 1993. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. Neural Netw., 6: 861-867.

Li, F., 2008. Function approximation by neural networks. Adv. Neural Netw., 5263: 384-390.

Li, X., 1998. On simultaneous approximations by radial basis function neural networks. Appl. Math. Comput., 95: 75-89.

Liu, B. and J. Si, 1994. The best approximation to C/sup 2/functions and its error bounds using regular-center Gaussian networks. IEEE. Trans. Neural Netw., 5: 845-847.

Mhaskar, H.N. and C.A. Micchelli, 1992. Approximation by superposition of sigmoidal and radial basis functions. Adv. Appl. Math., 13: 350-373.

Park, J. and I.W. Sandberg, 1991. Universal approximation using radial-basis-function networks. Neural Comput., 3: 246-257.

Scarselli, F. and A.C. Tsoi, 1998. Universal approximation using feed forward neural networks: A survey of some existing methods and some new results. Neural Netw., 11: 15-37.

Xingui, H. and X. Shaohua, 1995. Process Neural Networks, Theory and Applications. Springer, Berlin, Germany,.