Honam Mathematical J. **35** (2013), No. 4, pp. 729–738
http://dx.doi.org/10.5831/HMJ.2013.35.4.729

THE CAPABILITY OF LOCALIZED NEURAL NETWORK APPROXIMATION

Nahmwoo Hahm† and Bum Il Hong*

Abstract. In this paper, we investigate a localized approximation of a continuously differentiable function by neural networks. To do this, we first approximate a continuously differentiable function by B-spline functions and then approximate B-spline functions by neural networks. Our proofs are constructive and we give numerical results to support our theory.

1. Introduction

Many mathematicians ([2], [4], [5], [6], [7], [9], [10], [11]) have been studied the neural network approximation in recent years. In [1], Chui, Li and Mhaskar pointed out the limitation of approximation by neural network with one hidden layer. Hahm and Hong [3] suggested a localized neural network approximation to cure this problem but the approximation algorithm was not constructive.

The motivation of localized approximation is explained as follows. Assume that we have n subintervals (not necessarily equal length) partitioning $[0,1]$. But for simplicity, let $x_i = \frac{i}{n}$ for $i = 0, 1, 2, \ldots, n$. Then $[0,1]$ is divided into n subintervals $\{[x_{i-1}, x_i] : i = 1, 2, \ldots, n\}$. If a target function f is continuously differentiable on $[0,1]$ and $\text{supp}(f) \subset [0,1]$, then f is mainly nonzero over at most $[l \cdot n] + 2$ subintervals, where l is the length of $\text{supp}(f)$ and $[\cdot]$ denotes the greatest integer function. Note that we may assume that f is nonzero over n subintervals when $[l \cdot n] + 2 \geq n$. Then we approximate f by neural networks locally with a small number of neurons on each subinterval where f is nonzero. If

Received October 10, 2013. Accepted October 16, 2013.
2010 Mathematics Subject Classification. 41A10, 41A24, 41A29.
Key words and phrases. localized approximation, neural network, B-spline.
†This research was supported by Incheon National University Research Fund, 2012.
*Corresponding author

f changes on a small part of the interval, this localized method has an advantage that we need to retrain neural networks related to that part only, not the whole interval.

Note that a neural network with one hidden layer is of the form

$$\sum_{i=1}^{n} c_i \sigma(a_i x + b_i) \tag{1.1}$$

where the weight a_i, the threshold b_i and c_i are real numbers for $i = 1, 2, \ldots, n$. The following functions are examples of activation functions.

$$\text{The squashing function} \ : \ \sigma(x) = (1 + e^{-x})^{-1} \tag{1.2}$$

$$\text{The Gaussian function} \ : \ \sigma(x) = e^{-x^2} \tag{1.3}$$

$$\text{Thin plate splines} \ : \ \sigma(x) = |x|^{2q-1}, \quad q \in \mathbb{N} \tag{1.4}$$

In [8], Kalman and Kwasny suggested the squashing function as an activation function since the squashing function was useful in hardware implementations of back propagation and related training algorithm. Thus we choose the squashing function as an activation function of localized neural networks.

2. Main results

Let $m \in \mathbb{N} \cup \{0\}$. We introduce B-spline of order m as Schumaker [12] suggested. The B-spline of order 0 is defined by

$$B_0(x) := \begin{cases} 0 & \text{if} \quad x < 0 \\ 1 & \text{if } 0 \le x < 1 \\ 0 & \text{if } 1 \le x. \end{cases} \tag{2.1}$$

Recursively, B_m is defined by

$$B_m(x) := \int_0^1 B_{m-1}(x - t)dt. \tag{2.2}$$

Then B_m has a compact support $[0, m+1]$ with $||B_m||_{\infty,[0,m+1]} \le 1$ and is $m - 1$ times continuously differentiable.

Let $C^{1,*}[0,1]$ be the set of all continuously differentiable functions f such that supp(f) is a proper subset of $[0,1]$. First, we approximate a function in $C^{1,*}[0,1]$ by B-spline functions.

Theorem 2.1. *Let $f \in C^{1,*}[0,1]$ and let $n \in \mathbb{N}$. For a given $m \in \mathbb{N}$, we define, for $x \in [0,1]$,*

$$T_{f,m,n}(x) := n \sum_{i=0}^{n-1} \left(\int_{\frac{i}{n}}^{\frac{i+1}{n}} f(s)ds \right) B_m(nx-k). \tag{2.3}$$

Then we get

$$||f - T_{f,m,n}||_{\infty,[0,1]} \le \frac{c_1}{n}, \tag{2.4}$$

where c_1 is a constant depending on f.

Proof. We extend f on $[-2,2]$ by $f(x)=0$ on $[-2,0]\cup[1,2]$. Then f is continuously differentiable on $[-2,2]$ and $|f(x)-f(y)| = |f^{'}(\eta)||x-y|$ for all $x, y \in [-2,2]$ by the Mean Value Theorem. Since $f^{'}(x)$ is continuous on $[-2,2]$, there exists a positive constant c_1 such that $|f^{'}(x)| \le c_1$ for all $x \in [-2,2]$. Thus

$$|f(x) - f(y)| \le \frac{c_1}{n} \tag{2.5}$$

for any $x, y \in [-2,2]$ with $|x-y| \le \frac{1}{n}$. By the extension of f, we have

$$\int_{\frac{i}{n}}^{\frac{i+1}{n}} f(s)ds = 0 \tag{2.6}$$

for $i = -m-1, -m, \ldots, -1$. In addition, by the property of B_m, we obtain that

$$\sum_{i=-m-1}^{n-1} B_m(nx-i) = 1. \tag{2.7}$$

For $x \in [0,1]$, we have, by (2.5), (2.6) and (2.7),

$$\begin{aligned} & |f(x) - T_{f,m,n}(x)| \\ = \; & \left| f(x) - n \sum_{i=0}^{n-1} \left(\int_{\frac{i}{n}}^{\frac{i+1}{n}} f(s)ds \right) B_m(nx-i) \right| \qquad (2.8) \\ = \; & \left| f(x) - n \sum_{i=-m-1}^{n-1} \left(\int_{\frac{i}{n}}^{\frac{i+1}{n}} f(s)ds \right) B_m(nx-i) \right| \\ \le \; & \left| n \sum_{i=-m-1}^{n-1} \left(\int_{\frac{i}{n}}^{\frac{i+1}{n}} |f(x) - f(s)|ds \right) B_m(nx-i) \right| \end{aligned}$$

$$\begin{aligned} &\leq \frac{c_1}{n} \sum_{i=-m-1}^{n-1} B_m(nx-i) \\ &= \frac{c_1}{n}. \end{aligned}$$

In all, we have

$$||f - T_{f,m,n}||_{\infty,[0,1]} \leq \frac{c_1}{n}. \tag{2.9}$$

Thus we complete the proof. □

Note that $B_m(x)$ is a piecewise polynomial of degree m. Hence

$$B_m(x) = \sum_{j=0}^{m} a_{i,j} x^j \tag{2.10}$$

for $x \in [\frac{i}{n}, \frac{i+1}{n}]$ and $i = 0, 1, 2, \ldots, n-1$. Since the squashing function $\sigma(x) = (1+e^{-x})^{-1}$ is a monotone function on $\mathbb{R}$, there exists $b \in \mathbb{R}$ such that $\sigma^{(n)}(b) \neq 0$ for all $n \in \mathbb{N}$ by the Baire's category theorem.

Theorem 2.2. *Let σ be the squashing function and let b be a point in $\mathbb{R}$ such that $\sigma^{(n)}(b) \neq 0$ for all $n \in \mathbb{N}$. Suppose that*

$$N_{m,h}(x) := \sum_{j=0}^{m} a_{i,j} \frac{1}{h^j \sigma^{(j)}(b)} \sum_{p=0}^{j} (-1)^{j-p} \binom{j}{p} \sigma(phx+b), \tag{2.11}$$

where $a_{i,j}$'s are the coefficients in (2.10), $h > 0$ and $j = 0, 1, \ldots, m$. Then, for any $\epsilon > 0$, the neural network $N_{f,m,n,h}$ defined by

$$N_{f,m,n,h}(x) := n \sum_{i=0}^{n-1} \left(\int_{\frac{i}{n}}^{\frac{i+1}{n}} f(s)ds \right) N_{m,h}(nx-i) \tag{2.12}$$

satisfies

$$||T_{f,m,n} - N_{f,m,n,h}||_{\infty,[0,1]} < \epsilon \tag{2.13}$$

for sufficiently small $h > 0$.

Proof. Since f is bounded on $[0, 1]$, we have

$$\left| n \int_{\frac{i}{n}}^{\frac{i+1}{n}} f(s)ds \right| \leq n \int_{\frac{i}{n}}^{\frac{i+1}{n}} M ds = M \tag{2.14}$$

for some $M > 0$ and $i = 0, 1, 2, \ldots, n-1$. By the divided difference formula, we have

$$||x - \frac{\sigma(hx+b) - \sigma(b)}{h\sigma'(b)}||_{\infty,[0,1]} = \mathcal{O}(h). \tag{2.15}$$

Inductively, we get

$$\tag{2.16} ||x^r - \frac{1}{h^r \sigma^{(r)}(b)} \sum_{q=0}^{r} (-1)^{r-q} \binom{r}{q} \sigma(qhx+b)||_{\infty,[0,1]} = \mathcal{O}(h)$$

for $r \in \mathbb{N}$. Thus we have

$$\tag{2.17} \begin{aligned} & ||T_{f,m,n} - N_{f,m,n,h}||_{\infty,[0,1]} \\ = \; & ||n \sum_{i=0}^{n-1} \left(\int_{\frac{i}{n}}^{\frac{i+1}{n}} f(s)ds \right) (B_m(n\cdot -i) - N_{m,h}(n\cdot -i))||_{\infty,[0,1]} \\ \leq \; & M|| \sum_{i=0}^{n-1} \big(B_m(n\cdot -i) - N_{m,h}(n\cdot -i)\big)||_{\infty,[0,1]} \\ = \; & \mathcal{O}(h) < \epsilon \end{aligned}$$

for sufficiently small $h > 0$. Thus we complete the proof. □

By Theorem 2.1 and Theorem 2.2, we obtain the following theorem that is the main result of this paper.

Theorem 2.3. *Let σ be the squashing function and let b be a point in $\mathbb{R}$ such that $\sigma^{(n)}(b) \neq 0$ for all $n \in \mathbb{N}$. Assume that $f \in C^{1,*}[0,1]$ and $m \in \mathbb{N}$. Then, for $h > 0$, there exists a neural network $N_{f,m,n,h}$ defined by*

$$\tag{2.18} N_{f,m,n,h}(x) := n \sum_{i=0}^{n-1} \left(\int_{\frac{i}{n}}^{\frac{i+1}{n}} f(s)ds \right) N_{m,h}(nx-i)$$

satisfies

$$\tag{2.19} ||f - N_{f,m,n,h}||_{\infty,[0,1]} \leq \frac{c_1}{n},$$

where c_1 is a constant depending on f.

Proof. Let $\epsilon > 0$ be given. By Theorem 2.1 and 2.2, we get

$$\tag{2.20} \begin{aligned} & ||f - N_{f,m,n,h}||_{\infty,[0,1]} \\ \leq \; & ||f - T_{f,m,n}||_{\infty,[0,1]} + ||T_{f,m,n} - N_{f,m,n,h}||_{\infty,[0,1]} \\ < \; & \frac{c_1}{n} + \epsilon, \end{aligned}$$

where c_1 is a constant depending on f. Since $\epsilon > 0$ is arbitrary, we complete the proof. □

3. Numerical Results

In this section, we give numerical results implemented by MATHEMATICA in order to justify our theory. We select

$$
(3.1) \qquad f(x) = \begin{cases} 0 & \text{if } 0 \le x < \frac{1}{3} \\ sin^2\big(3\pi(x - \frac{1}{3})\big) & \text{if } \frac{1}{3} \le x < \frac{2}{3} \\ 0 & \text{if } \frac{2}{3} \le x < 1 \end{cases}
$$

as a target function. Then f is continuously differentiable on $[0, 1]$ and $\text{supp}(f) = (1/3, 2/3)$. First, we approximate f by a linear combination of the B-splines function $B_2(x)$. A tedious calculation gives

$$
(3.2) \qquad B_2(x) = \begin{cases} 0 & \text{if } \quad x < 0 \\ \frac{x^2}{2} & \text{if } 0 \le x < 1 \\ \frac{1}{2}(-2x^2 + 6x - 3) & \text{if } 1 \le x < 2 \\ \frac{1}{2}(x^2 - 6x + 9) & \text{if } 2 \le x < 3 \\ 0 & \text{if } 3 \le x. \end{cases}
$$

By Theorem 2.1, we have

$$
(3.3) \qquad T_{2,n}(x) = n \sum_{i=0}^{n-1} \left(\int_{\frac{i}{n}}^{\frac{i+1}{n}} f(t)dt \right) B_2(nx - i)
$$

for $n \in \mathbb{N}$.

Figure 1 shows that the linear combination of B-spline function $T_{2,n}$ approximates the target function well when n is large as we expected in Theorem 2.1. In fact, our numerical computation shows that the maximum errors between the target function f and $T_{2,8}$, f and $T_{2,32}$, and f and $T_{2,128}$ are 0.725010, 0.281784 and 0.0734274, respectively. Thus Theorem 2.1 is justified numerically since the maximum errors decrease as n increases. Moreover, Theorem 2.1 theoretically shows that the order of accuracy is 1. The numerical order of accuracy is estimated by

$$
\text{order} = -\frac{\log \frac{\|f - T_{2,n_1}\|_{\infty,[0,1]}}{\|f - T_{2,n_2}\|_{\infty,[0,1]}}}{\log(n_1/n_2)},
$$

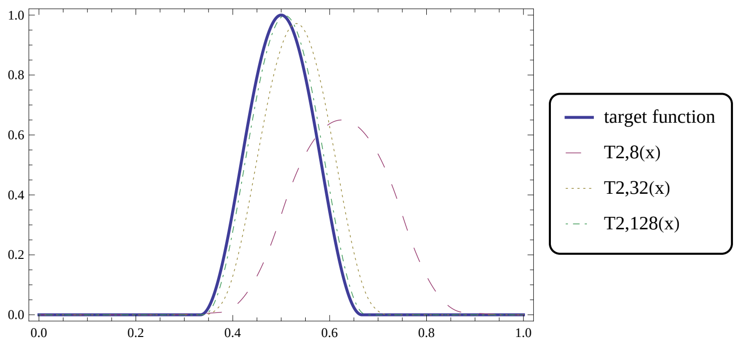

FIGURE 1. Target function and B-splines

where $n_i's$ are the number of subintervals of $[0, 1]$ increased by 4 in our case and the following table verifies that our numerical approximation errors tend to the order of accuracy 1 asymptotically.

n_i	Error	Order
8	0.725010	
32	0.281784	0.69
128	0.0734274	0.97

Now, we approximate B-spline function $B_2(x)$ in $T_{2,n}$ by neural networks with the squashing activation function. For $h > 0$, we replace x by

$$D_1(x) := \frac{\sigma(hx+1) - \sigma(1)}{h\sigma'(1)} \tag{3.4}$$

and replace x^2 by

$$D_2(x) := \frac{\sigma(2hx+1) - 2\sigma(hx+1) + \sigma(1)}{h^2\sigma''(1)}. \tag{3.5}$$

By Theorem 2.2, we construct a neural network as

$$N_{2,h}(x) = \begin{cases} 0 & \text{if} \quad x < 0 \\ \dfrac{D_2(x)}{2} & \text{if} \ 0 \le x < 1 \\ \dfrac{1}{2}(-2D_2(x) + 6D_1(x) - 3) & \text{if} \ 1 \le x < 2 \\ \dfrac{1}{2}(D_2(x) - 6D_1(x) + 9) & \text{if} \ 2 \le x < 3 \\ 0 & \text{if} \ 3 \le x. \end{cases} \tag{3.6}$$

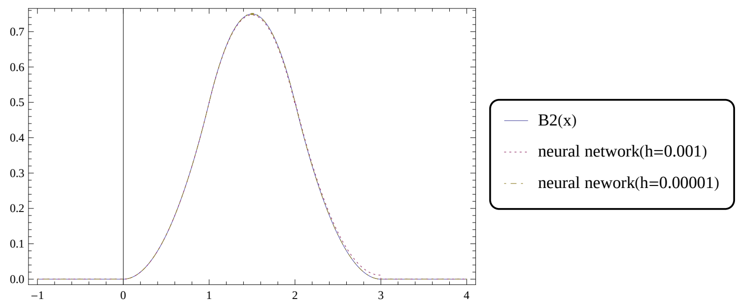

FIGURE 2. B-spline and neural networks

Figure 2 shows that neural networks approximate B_2 well when we use sufficiently small $h > 0$. In fact, numerical computation shows that the maximum errors between B_2 and $N_{2,0.001}$, and B_2 and $N_{2,0.00001}$ are 0.015545 and 0.000110858, respectively.

Finally, we approximate f numerically by neural networks $N_{f,2,n,h}$ in the cases of $N_{f,2,n,0.001}$ and $N_{f,2,n,0.00001}$ for $n = 8, 32, 128$.

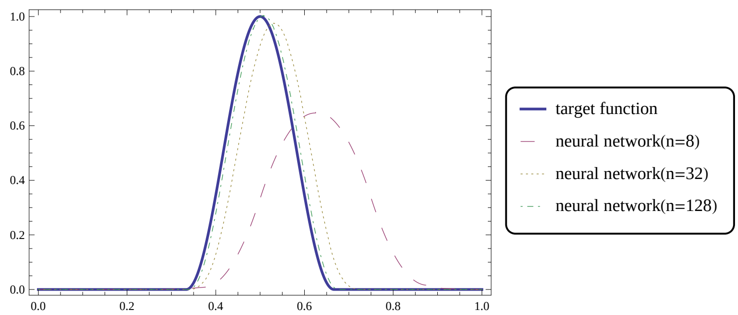

FIGURE 3. Target function and neural networks ($h = 0.001$)

Figure 3 and Figure 4 show that neural networks approximate f well when we use sufficiently small $h > 0$. Numerical computation shows that the maximum errors between the target function f and $N_{f,2,8,h}$, f and $N_{f,2,32,h}$, and f and $N_{f,2,128,h}$ are $0.725014, 0.286168$ and 0.0763111, respectively, when $h = 0.001$. Similarly, the maximum errors between the target function f and $N_{f,2,8,h}$, f and $N_{f,2,32,h}$, and f and $N_{f,2,128,h}$ are $0.725012, 0.282494$ and 0.0734536, respectively, when $h = 0.00001$. The following table verifies that our numerical approximation errors tend to the order of accuracy 1 asymptotically in the cases of $h_1 = 0.001$ and $h_2 = 0.00001$ as theoretically shown in Theorem 2.3.

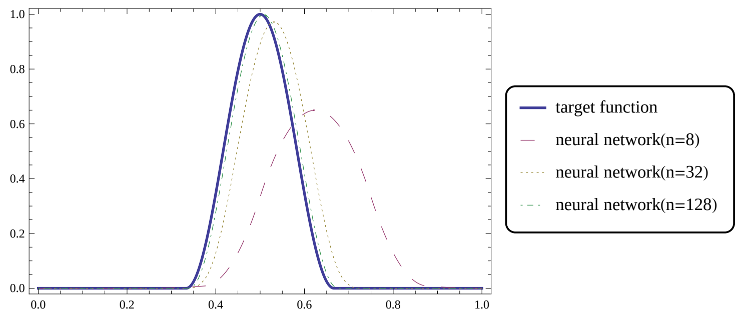

FIGURE 4. Target function and neural networks ($h = 0.00001$)

n_i	Error (h_1)	Order (h_1)	Error (h_2)	Order (h_2)
8	0.725014		0.725012	
32	0.286168	0.67	0.282494	0.68
128	0.0763111	0.95	0.0734536	0.97

References

[1] C. K. Chui, X. Li and H. N. Mhaskar, *Limitations of the approximation capabilities of neural networks with one hidden layer*, Adv. Comput. Math., **5** (1996), 233-243.

[2] B. Gao and Y. Xu, *Univariant approximation by superpositions of a sigmoidal function*, J. Math. Anal. Appl., **178** (1993), 221-226.

[3] N. Hahm and B. I. Hong, *Extension of localised approximation by neural networks*, Bull. Austral. Math. Soc., **59** (1999), 121-131.

[4] N. Hahm and B. I. Hong, *An approximation by neural networks with a fixed weight*, Comput. Math. Appl., **47** (2004), 1897-1903.

[5] N. Hahm and B. I. Hong, *Approximation order to a function in L_p space by generalized translation networks*, Honam Math. J. **28(1)** (2006), 125-133.

[6] N. Hahm and B. I. Hong, *A simultaneous neural network approximation with the squashing function*, Honam Math. J. **31(2)** (2009), 147-156.

[7] B. I. Hong and N. Hahm, *Approximation order to a function in $\bar{C}(\mathbb{R})$ by superposition of a sigmoidal function*, Appl. Math. Lett., **15** (2002), 591-597.

[8] B. L. Kalman and S. C. Kwasny, *Why Tanh : Choosing a sigmoidal function*, Int. Joint Conf. on Neural Networks **4** (1992), 578-581.

[9] M. Leshno, V. Lin, A. Pinkus and S. Schocken, *Multilayered feedforward networks with a nonpolynomial activation function can approximate any function*, Neural Networks, **6** (1993), 61-80.

[10] G. Lewicki and G. Marino, *Approximation of functions of finite variation by superpositions of a sigmoidal function*, Appl. Math. Lett. **17** (2004), 1147-1152.

[11] H. N. Mhaskar and N. Hahm, *Neural networks for functional approximation and system identification*, Neural Comput., **9** (1997), 143-159.

[12] L. L. Schumaker, *Spline Functions : Basic Theory*, Cambridge University Press, Cambridge, 2007.

Nahmwoo Hahm
Department of Mathematics, Incheon National University,
Incheon 406-772, Korea.
E-mail: nhahm@incheon.ac.kr

Bum Il Hong
Department of Applied Mathematics, Kyung Hee University,
Yongin 446-701, Korea.
E-mail: bihong@khu.ac.kr