Mathematics

https://www.aimspress.com/journal/Math

AIMS Mathematics, 9(11): 31679–31695.
DOI: 10.3934/math.20241523
Received: 26 July 2024
Revised: 21 October 2024
Accepted: 25 October 2024
Published: 07 November 2024

Research article

Rate of approximaton by some neural network operators

Bing Jiang*

College of Artificial Intelligence and Big Data, Hefei University, Hefei, Anhui 230601, China

* **Correspondence:** Email: jiangbing@hfuu.edu.cn; Tel: 16622808944.

Abstract: First, we construct a new type of feedforward neural network operators on finite intervals, and give the pointwise and global estimates of approximation by the new operators. The new operator can approximate the continuous functions with a very good rate, which can not be obtained by polynomial approximation. Second, we construct a new type of feedforward neural network operator on infinite intervals and estimate the rate of approximation by the new operators. Finally, we investigate the weighted approximation properties of the new operators on infinite intervals and show that our new neural networks are dense in a very wide class of functional spaces. Thus, we demonstrate that approximation by feedforward neural networks has some better properties than approximation by polynomials on infinite intervals.

Keywords: feed-forward neural networks; approximation rate; finite intervals; infinite intervals
Mathematics Subject Classification: 41A25, 41A30, 47A58

1. Introduction

Feed-forward neural networks (FNNs) have been investigated extensively because of their universal approximation capabilities on compact input sets. Many authors have dealt with the FNNS with one hidden layer, which can be mathematically expressed as

$$N_n(x) = \sum_{j=0}^{n} c_j \sigma\left(\left\langle a_j \cdot x\right\rangle + b_j\right),\ x \in \mathbb{R}^s, s \in \mathbb{N},$$

where for $0 \le j \le n, b_j \in \mathbb{R}$ are the thresholds, $a_j \in \mathbb{R}^s$ are the connection weights, $c_j \in \mathbb{R}$ are the coefficients, $\left\langle a_j \cdot x\right\rangle$ is the inner product of a_j and x, and σ is the activation function. As we know, FNNS are universal approximators. Theoretically, any continuous function defined on a compact set can be approximated to any desired degree of accuracy by increasing the number of hidden neurons. A lot of results concerning the existence of an approximation and determining the number of neurons required to guarantee that all functions (belong to a certain class) can be approximated to the prescribed degree of accuracy have been achieved by many mathematicians (see [1–6]). The complexity problem of neural networks is to study the relationship between the topological structure of neural networks and their approximation ability. Among them, research on the quantitative estimates of approximation is particularly important (see [7–11]).

In many fundamental network models, the activation function σ is usually taken to be a sigmoidal function. In other words, σ satisfies the conditions

$$\lim_{x\to+\infty} \sigma(x) = 1, \quad \lim_{x\to-\infty} \sigma(x) = 0. \tag{1.1}$$

Many neural network operators with different activation functions in both univariate and multivariate settings are studied (see [12–16]). In [4], the authors investigated the FNN with activation function $g_j : \mathbb{R}^d \to \mathbb{R}$, defined by

$$g_j(x) := \frac{e^{-A\rho\left(x,x_j\right)}}{\sum_{i=0}^{n} e^{-A\rho(x,x_i)}}, \, j = 0, 1, 2, \cdots,$$

where $x_0, x_1, \cdots, x_n$ are the data in $\mathbb{R}^d$, $\rho(a, b)$ denotes the Euclidean distance between the points a and b in $\mathbb{R}^d$, and $A > 0$ is a parameter. In this case, $N_n(x)$ can be regarded as a FNN with four layers: the first layer is the input layer x; the second layer is the processing layer for computing values $\rho(x, x_i), i = 0, 1, \cdots, n$, between input x and the prototypical input points x_i, and it is the input of the third layer that contains $n + 1$ neuron, $g_j(x)$ is the activation function of the j-th neuron; the fourth layer is the output layer $N_n(x)$. Although $g_j(x)$ not sigmoidal, they have some better properties than the usual sigomidal functions. For example, (i) $0 < g_j(x) \le 1, j = 0, 1, \cdots n$; (ii) $\sum_{j=0}^{n} g_j(x) = 1, \ x \in \mathbb{R}^d$.

In [4], Cao, Zhang, and Xu constructed a class of neural networks $N_a(x, A)$ with activation functions $g_j(x)$ on a finite interval $[a, b]$ as follows:

$$N_a(x, A) := \sum_{j=0}^{n} f(x_j) \frac{e^{-A(n)\left|x-x_j\right|}}{\sum_{i=0}^{n} e^{-A(n)|x-x_i|}}, \tag{1.2}$$

where $x_j = a + \frac{b-a}{n} j, j = 0, 1, \cdots, n$, and $A(n)$ is a parameter depending on n. For the approximation rate of operator $N_a(x, A)$, they established the following results in [4].

Theorem 1.1. *Let f be a continuous function on $[a, b]$. Then there exists $A^* > 0$, such that*

$$|f(x) - N_a(x, A)| \le 2\omega\left(f, \frac{b-a}{n}\right) + 2\, \|f\|_{[a,b]} Mne^{-n}, \tag{1.3}$$

for $A(n) > A^*$ and for all $x \in [a, b]$, where $\|f\|_{[a,b]} := \max_{x\in[a,b]} |f(x)|$, and $\omega(f, t)$ is the modulus of continuity of f on $[a, b]$, that is,

$$\omega(f, t) := \sup_{|h|\le t} \max_{x,x+h\in[a,b]} |f(x + h) - f(x)|.$$

Throughout the paper, $\|f\|_I$ denotes the uniform norm of f on the interval I, C denotes an absolute constant, and $C_{t_1,\cdots,t_s}$ a positive constant only depending on the parameter(s) $t_1, t_2, \cdots, t_s$, which may be different in different occurrences.

To approximate the functions defined on the infinite interval $\mathbb{R} := (-\infty, +\infty)$, Cao, Li, et al. [5] investigated the approximation by neural networks with activation functions $g_j(x)$ on $\mathbb{R}$. In fact, they obtained the following:

Theorem 1.2. *Let f be a continuous function on $\mathbb{R}$ satisfying* $\lim_{x\to\infty} f(x) = B_1$, $\lim_{x\to-\infty} f(x) = B_2$, *where B_1 and B_2 are two constants. For any $r > 0$, define*

$$\mathrm{g}(x) := \sum_{j=0}^{n} f(x_j) \frac{e^{-A\left|x-x_j\right|}}{\sum_{i=0}^{n} e^{-A|x-x_i|}}, \tag{1.4}$$

where $x_j = -M + 2Mj/n$, $j = 0, 1, \cdots, n$, $A = \frac{n^2}{2M}$, and M is a positive number depending on $f$and r. Then,

$$\|f - \mathrm{g}\|_{\mathbb{R}} \leq 2n \|f\|_{\mathbb{R}} \exp(-n) + \omega\left(f, \frac{2M}{n}\right) + 2r.$$

For the operator $N_a(x, A)$ defined in (1.2), we see that it depends on the parameters $A(n)$ and A^* strictly. Therefore, we should calculate the values $A(n)$ and A^* before constructing the operator, while the calculations are usually not obvious. Similarly, to construct the operator $\mathrm{g}(x)$ in (1.4), we should estimate the values M first, which depends on r and f. On the other hand, generally speaking, we have $M \to +\infty$, when $r \to 0$. Hence, to assure the convergence of the approximation, the number of neurons n must tend to infinity a rate not too slow (it should hold that $M = o(n)$). To overcome the above problem in the constructions of the neural network operators, we will introduce two new classes of operators defined on finite intervals and infinite intervals, respectively. Also, the approximation rates by the new operators will be given.

The present paper is organized as follows: In Section 2, we construct a new class of neural networks on finite intervals and give pointwise and global estimates of the approximation (see (2.2) and (2.3)) which is better than (1.3) and cannot be achieved by the usual approximation by polynomials. In Section 3, we construct a new class of neural networks on finite intervals and give its rate of approximation. Furthermore, we investigated the weighted approximation of operators on the infinite intervals and show that our new neural networks are dense in a very wide class of functional spaces. Thus, we demonstrate that approximation by feedforward neural networks has some better properties than approximation by polynomials on infinite intervals.

2. Neural network operators on finite intervals

For convenience, we take the interval to be $[-1, 1]$. Let $x = x(\theta)$ be a function from $[0, 1]$ to $[-1, 1]$ defined by

$$x = x(\theta) := \begin{cases} (2\theta)^p - 1, & \theta \in \left[0, \frac{1}{2}\right], \\ -(2 - 2\theta)^p + 1, & \theta \in \left(\frac{1}{2}, 1\right], \end{cases}$$

where $p \geq 1$ is a given number.

Set $X = \left\{x_k = x\left(\frac{k}{n}\right) : k = 0, 1, 2, \cdots, n\right\}$. Define

$$N_p(f, X, x) = \sum_{k=0}^{n} f(x_k) \frac{e^{-n^p|x-x_k|}}{\sum_{i=0}^{n} e^{-n^p|x-x_i|}}. \tag{2.1}$$

See Figure 1, for the structural diagram of the neural network $N_p(f, X, x)$.

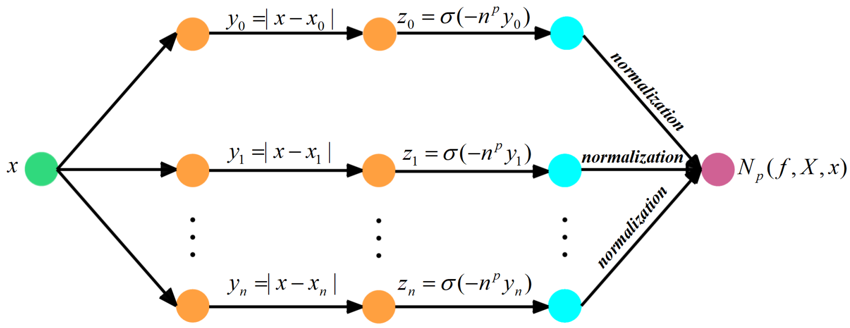

Figure 1. The structural diagram of the neural network $N_p(f, X, x)$.

Theorem 2.1. *If $f \in C([-1, 1])$, $p \geq 1$, then, we have*

$$\left|N_p(f, X, x) - f(x)\right| \leq C_p \omega\left(f, \frac{\delta_n(x)}{n}\right), \tag{2.2}$$

and

$$\left\|N_p(f, X) - f\right\|_{[-1,1]} \leq C_p \omega_\psi\left(f, \frac{1}{n}\right), \tag{2.3}$$

where $\delta_n(x) = \frac{1}{n^{p-1}} + \psi(x)$, $\psi(x) = \left(1 - x^2\right)^{\frac{p-1}{p}}$ and $\omega_\psi(f, t)$ is the Ditzian-Totik type modulus defined by

$$\omega_\psi(f, t) := \sup_{0<h\leq t} \sup_{-1\leq x\leq 1} \left|f\left(x + \frac{h}{2}\psi(x)\right) - f\left(x - \frac{h}{2}\psi(x)\right)\right|.$$

Remark 2.1. *(1) The operator defined in (2.1) only depends on the values $f(x_k)$, $1 \leq k \leq n$, and the number of neurons n.*

(2) The approximation rate achieved in (2.2) cannot be obtained by approximation of polynomials. In fact, Gopengauz proved in [17] that there exist continuous functions f on $[-1, 1]$ for which there are no algebraic polynomials P_n of degree less or equal n such that

$$|f(x) - P_n(x)| = O\left(\omega\left(f, \frac{\sqrt{1-x^2}}{n}\varepsilon(1 - x^2) + \frac{\delta(n^{-1})}{n^2}\right)\right),$$

for all integers n and $x \in I$, where $\varepsilon(u) \downarrow 0$ and $\delta(u) \downarrow 0$, when $u \to 0$.

(3) The approximation rate obtained in Theorem 2.1 is sharper than that in Theorem 1.1. For example, from (2.2), we see that we have a better approximation rate when x is nearer to the endpoints, when $p > 1$. In fact, if $x \in [-1, x_1] \cup [x_{n-1}, 1]$, we have

$$\left|N_p(f, X, x) - f(x)\right| \leq C_p \omega\left(f, \frac{1}{n^p}\right).$$

(4) Since $\omega_{\psi}(f,t) \leq \omega(f,t)$, we see that (2.3) is also an improvement of Theorem 1.1 on the approximation rate.

Proof of (2.2). We need the following inequalities (2.4)–(2.8) (see [18]): For $x \in \left[x_{j-1}, x_j\right], j = 2, 3, \cdots, n-1$, it holds that

$$|x - x_k| \leq C_p \frac{(|j-k|+1)^p}{n} \psi(u), \ u \in [x, x_k] \text{ or } u \in [x_k, x]. \tag{2.4}$$

For $x \in [-1, x_1]$, it holds that

$$|x - x_k| \leq C_p \frac{(k+1)^p}{n^p}, \ k = 0, 1, \cdots, n. \tag{2.5}$$

For $x \in [x_{n-1}, 1]$, it holds that

$$|x - x_k| \leq C_p \frac{(n-k+1)^p}{n^p}, \ k = 0, 1, \cdots, n. \tag{2.6}$$

Denote by $x_j, 0 \leq j \leq n$, the closest node to x, then

$$\left|x - x_j\right| \leq C_p \frac{\psi(x)}{n}, \tag{2.7}$$

$$|x - x_k| \geq C_p \frac{|j-k|}{n} \psi(x), \ k \neq j. \tag{2.8}$$

We prove (2.2) by considering the following two cases.

Case 1. $x \in [-1, x_1] \cup [x_{n-1}, 1]$. We only treat the subcase $x \in [-1, x_1]$, when $x \in [x_{n-1}, 1]$, it can be done similarly. It is obvious that

$$\begin{aligned} \left|N_p(f,X,x) - f(x)\right| &\leq \sum_{k=0,1} |f(x) - f(x_k)| \frac{e^{-n^p|x-x_k|}}{\sum_{i=0}^n e^{-n^p|x-x_i|}} + \sum_{k=2}^n |f(x) - f(x_k)| \frac{e^{-n^p|x-x_k|}}{\sum_{i=0}^n e^{-n^p|x-x_i|}} \\ &=: I_1(x) + I_2(x). \end{aligned} \tag{2.9}$$

By (2.5), we have

$$I_1(x) \leq |f(x) - f(x_1)| + |f(x) - f(x_2)| \leq C_p \omega\left(f, \frac{2^p}{n^p}\right) \leq C_p \omega\left(f, \frac{1}{n^p}\right), \tag{2.10}$$

where in the last inequality, we used the following well-known property of modulus of continuity:

$$\omega(f, \lambda t) \leq (\lambda + 1)\, \omega(f,t), \ \lambda > 0. \tag{2.11}$$

By (2.4) and (2.8), we observe that for $k \geq 2$,

$$\begin{aligned} \frac{e^{-n^p|x-x_k|}}{\sum_{i=0}^n e^{-n^p|x-x_i|}} &\leq e^{-n^p(|x-x_k|-|x-x_1|)} \leq e^{-n^p(|x_1-x_k|)} \\ &\leq e^{-C_p n^p\left(\frac{k}{n}\psi(x_1)\right)} \leq e^{-C_p k}. \end{aligned} \tag{2.12}$$

Now, by (2.4) and (2.11),

$$\begin{aligned} I_2(x) &\le \sum_{k=2}^{n} \omega(f, |x-x_k|)\, e^{-C_p k} \\ &\le C_p \omega\left(f, \frac{1}{n^p}\right) \sum_{k=2}^{n} (1 + n^p |x-x_k|)\, e^{-C_p k} \\ &\le C_p \omega\left(f, \frac{1}{n^p}\right) \sum_{k=2}^{n} \left(1 + C_p n^p \frac{k^p}{n} \psi(x_1)\right) e^{-C_p k} \\ &\le C_p \omega\left(f, \frac{1}{n^p}\right) \sum_{k=2}^{n} k^p e^{-C_p k} \\ &\le C_p \omega\left(f, \frac{1}{n^p}\right). \end{aligned} \tag{2.13}$$

By (2.9), (2.10), and (2.13), we verify that (2.2) holds for $x \in [-1, x_1]$.

Case 2. $x \in (x_1, x_{n-1})$, say $x \in \left[x_{j-1}, x_j\right]$, $j = 1, 2, \cdots, n-1$. Then

$$\begin{aligned} \left|N_p(f, X, x) - f(x)\right| &\le \left(\left|f(x) - f\left(x_j\right)\right| + \left|f(x) - f\left(x_{j-1}\right)\right|\right) \\ &\quad + \sum_{k=1}^{j-2} |f(x) - f(x_k)| \frac{e^{-n^p|x-x_k|}}{\sum_{i=0}^{n} e^{-n^p|x-x_i|}} \\ &\quad + \sum_{k=j+1}^{n} |f(x) - f(x_k)| \frac{e^{-n^p|x-x_k|}}{\sum_{i=0}^{n} e^{-n^p|x-x_i|}} \\ &=: I_3(x) + I_4(x) + I_5(x). \end{aligned} \tag{2.14}$$

By (2.4), (2.7), and (2.11), it is easy to deduce that

$$I_3(x) \le C_p \omega\left(f, \frac{\delta_n(x)}{n}\right). \tag{2.15}$$

By (2.8), we have for $k = 1, 2, \cdots, j-2$, that

$$\begin{aligned} \frac{e^{-n^p|x-x_k|}}{\sum_{i=0}^{n} e^{-n^p|x-x_i|}} &\le e^{-n^p\left(|x-x_k| - |x-x_{j-1}|\right)} \le e^{-n^p\left(|x_{j-1}-x_k|\right)} \\ &\le e^{-C_p n^p\left(\frac{j-k}{n}\psi\left(x_{j-1}\right)\right)} \le e^{-C_p|j-k|}. \end{aligned} \tag{2.16}$$

Therefore, by (2.11) and (2.4),

$$\begin{aligned} I_4(x) &\le \sum_{k=1}^{j-2} \omega(f, |x-x_k|)\, e^{-C_p|j-k|} \\ &\le \omega\left(f, \frac{\delta_n(x)}{n}\right) \sum_{k=1}^{j-2} \left(1 + \frac{n|x-x_k|}{\delta_n(x)}\right) e^{-C_p|j-k|} \end{aligned}$$

$$
\begin{aligned}
&\le C_p\omega\left(f, \frac{\delta_n(x)}{n}\right)\sum_{k=1}^{j-2}\left(1+\frac{n^{\frac{(|j-k|+1)^p}{n}}\psi(x)}{\psi(x)}\right)e^{-C_p|j-k|}\\
&\le C_p\omega\left(f, \frac{\delta_n(x)}{n}\right)\sum_{k=1}^{j-2}|j-k|^p\, e^{-C_p|j-k|}\\
&\le C_p\omega\left(f, \frac{\delta_n(x)}{n}\right). \qquad (2.17)
\end{aligned}
$$

Analogues to (2.16), we have for $k \ge j+2$ that

$$
\frac{e^{-n^p|x-x_k|}}{\sum_{i=0}^n e^{-n^p|x-x_i|}} \le e^{-n^p\left(|x_j-x_k|\right)} \le e^{-C_p|j-k|}. \qquad (2.18)
$$

Therefore, by (2.4), we can deduce that

$$
I_5(x) \le C_p\omega\left(f, \frac{\delta_n(x)}{n}\right) \qquad (2.19)
$$

in a similar way to that of (2.17).

Now, by (2.14), (2.15), (2.17), and (2.19), we conclude that (2.2) holds for $x \in (x_1, x_{n-1})$. □

Proof of (2.3). From (2.12), (2.16) and (2.18), we actually have for $x \in \left[x_{j-1}, x_j\right]$, $j = 1, 2, \cdots, n$ and $k = 0, 1, \cdots, n$, that

$$
\frac{e^{-n^p|x-x_k|}}{\sum_{i=0}^n e^{-n^p|x-x_i|}} \le e^{-C_p|j-k|}. \qquad (2.20)
$$

Setting $g(\theta) = f(x(\theta))$, then

$$
\begin{aligned}
\left|N_p(f,X,x) - f(x)\right| &\le \sum_{k=0}^n |f(x) - f(x_k)|\,\frac{e^{-n^p|x-x_k|}}{\sum_{i=0}^n e^{-n^p|x-x_i|}}\\
&\le \sum_{k=0}^n \omega\left(g, \frac{|j-k|+1}{n}\right)e^{-C_p|j-k|}\\
&\le C\omega\left(g, \frac{1}{n}\right)\sum_{k=0}^n (|j-k|+1)\,e^{-C_p|j-k|}\\
&\le C_p\omega\left(g, \frac{1}{n}\right).
\end{aligned}
$$

By [19],

$$
\begin{aligned}
K(f,t) &= \inf_{\|g'\|<\infty, h\in AC_{[0,1]}}\left\{\|f-g\| + t\|g'\|\right\},\\
K_\psi(f,t) &= \inf_{\|\psi g'\|<\infty, h\in AC_{[0,1]}}\left\{\|f-g\| + t\|\psi g'\|\right\},
\end{aligned}
$$

we have

$$
\omega\left(g, \frac{1}{n}\right) \sim K\left(g, n^{-1}\right)
$$

$$\begin{aligned}
&\sim \inf\left\{\|g-h\|+n^{-1}\|h'\|:h\in AC_{[0,1]}\right\}\\
&\sim \inf\left\{\|f-h\|+n^{-1}\|\psi h'\|:h\in AC_{[0,1]}\right\}\\
&\sim K_{\psi}\left(f,n^{-1}\right).
\end{aligned}$$

Thus (2.3) follows from the following equivalent relation (see [19])

$$K_{\psi}\left(f,n^{-1}\right)\sim\omega_{\psi}\left(f,n^{-1}\right).$$

□

3. Neural network operators on infinite intervals

In this section, we set

$$x_k:=\frac{k}{m},k=0,\pm1,\pm2,\cdots,\pm n, \tag{3.1}$$

where $m=m(n)>0$ is a parameter depending on n. To ensure the point system is dense on finite intervals and the whole real axis $\mathbb{R}$ is filled up with the nodes, we should assume that $m\to\infty$ and $m=o(n)$ when $n\to\infty$. Now, we construct a new neural network operator on $C(\mathbb{R})$ as follows:

$$g(x):=\sum_{k=-n}^{n}f(x_k)\frac{e^{-m|x-x_k|}}{\sum_{i=-n}^{n}e^{-m|x-x_i|}}. \tag{3.2}$$

Obviously, the way to construct $g(x)$ is more direct than $\mathfrak{g}(x)$ defined in (1.4). In fact, $\mathfrak{g}(x)$ only depends on the values $f(x_k),k=0,\pm1,\pm2,\cdots,\pm n$, the number of neurons and the parameter m.

Theorem 3.1. *Let f be a continuous function on $\mathbb{R}$ satisfying* $\lim_{x\to\infty}f(x)=B_1$, $\lim_{x\to-\infty}f(x)=B_2$, *where $B_1$and B_2 are two constants. Then,*

$$\|f-g\|_{\mathbb{R}}\le C\left(\omega\left(f,\frac{1}{m}\right)+\varepsilon_f\left(\frac{n}{m}\right)+e^{-\left(1-\frac{1}{m}\right)n}\|f\|_{\mathbb{R}}\right), \tag{3.3}$$

where

$$\varepsilon_f(t):=\max\left(\max_{x\ge t}|f(x)-B_1|,\max_{x\le -t}|f(x)-B_2|\right),\ t>0.$$

Proof. If $x>\frac{n}{m}$, then

$$\begin{aligned}
|f(x)-g(x)| &\le |f(x)-B_1|+\sum_{k=-n}^{n}|f(x_k)-B_1|\frac{e^{-m|x-x_k|}}{\sum_{i=-n}^{n}e^{-m|x-x_i|}}\\
&\le \varepsilon_f(x)+\sum_{|k|\ge n/m}|f(x_k)-B_1|\frac{e^{-m|x-x_k|}}{\sum_{i=-n}^{n}e^{-m|x-x_i|}}+\sum_{|k|<n/m}|f(x_k)-B_1|e^{-m(|x-x_k|-|x-x_n|)}\\
&\le \varepsilon_f(x)+\varepsilon_f(n/m)+2\|f\|_{\mathbb{R}}\sum_{|k|<n/m}e^{-(n-k)}\\
&\le 2\varepsilon_f(n/m)+Ce^{-\left(1-\frac{1}{m}\right)n}\|f\|_{\mathbb{R}},
\end{aligned}$$

which implies (3.3) holds for $x>\frac{n}{m}$.

Similarly, we see that the same estimation holds for $x < -\frac{n}{m}$.

Now, we consider the case when $x \in \left[-\frac{n}{m}, \frac{n}{m}\right]$, say, $x \in \left[x_{j-1}, x_j\right]$, $j = 0, \pm 1, \pm 2, \cdots, \pm n$. For any $-n \le k \le n$, set $x_{j^*} = x_j$ for $k > j$ or $k = j-1$, and $x_{j^*} = x_{j-1}$ for $k < j-1$ or $k = j$. In other words, we take x_{j^*} such that $\left|x_{j^*} - x_k\right| = \min\left\{\left|x_j - x_k\right|, \left|x_{j-1} - x_k\right|\right\}$. Then

$$\begin{aligned} \frac{e^{-m|x-x_k|}}{\sum_{i=-n}^{n} e^{-m|x-x_i|}} &\le e^{-m\left(|x-x_k|-\left|x-x_{j^*}\right|\right)} \\ &\le e^{-m\left(\left|x_{j^*}-x_k\right|\right)} \\ &\le Ce^{-|j-k|}. \end{aligned} \tag{3.4}$$

Therefore, by (2.11), we have

$$\begin{aligned} |f(x) - g(x)| &\le \sum_{k=-n}^{n} |f(x_k) - f(x)| \frac{e^{-m|x-x_k|}}{\sum_{i=-n}^{n} e^{-m|x-x_i|}} \\ &\le C \sum_{k=-n}^{n} \omega(f, |x - x_k|)\, e^{-|j-k|} \\ &\le C\omega\left(f, \frac{1}{m}\right) \sum_{k=-n}^{n} (m|x - x_k| + 1) e^{-|j-k|} \\ &\le C\omega\left(f, \frac{1}{m}\right) \sum_{k=-n}^{n} (|j - k| + 1)\, e^{-|j-k|} \\ &\le C\omega\left(f, \frac{1}{m}\right). \end{aligned}$$

Thus, we prove (3.3) for $|x| \le \frac{n}{m}$. □

Now, we consider the weighted approximation by neural networks on $\mathbb{R}$. Let $w(x) := e^{-Q(x)}$ be the weight function satisfying the following conditions:

(i) $Q(x)$ is even, $\lim_{x\to\infty} Q(x) = \infty$;

(ii) $Q'(x) \ge 0$ is either strictly monotone increasing or bounded in $[a, \infty)$;

(iii) In the case where $Q'(x)$ is strictly monotone increasing in $[a, \infty)$, $Q'(x+1) \le AQ'(x)$ ($x \in [a, \infty), A > 0$ independent of x).

In connection with these weights, for each $t \in [a, \infty)$, we define t^* as

(a) the unique solution of $tQ'(t^*) = 1$ if $Q'(x) \uparrow \infty$ as $x \uparrow \infty$,

(b) ∞ if $Q'(x)$ is bounded as $x \to \infty$.

Example 3.1. *Functions like* $Q(x) = \log^{\beta}(1 + |x|)$, $Q(x) = (1 + |x|)^{\beta}$ $(\beta > 0)$, *and* $Q(x) = e^{c|x|}$ $(c > 0)$ *are weights satisfying conditions (i)–(iii).*

In what follows, we always assume that the weight function $w(x) = e^{-Q(x)}$ satisfies conditions (i)–(iii). Define

$$C_w(\mathbb{R}) := \left\{f(x) : f \in C(\mathbb{R}),\ \lim_{|x|\to\infty} w(x) f(x) = 0\right\}.$$

For any $f \in C_w(\mathbb{R})$, define its modulus of continuity as follows:

$$\Omega(f,t)_w := \sup_{0<h\le t} \|w(x)(f(x+h/2)-f(x-h/2))\|_{[-h^*,h^*]}.$$

It is proved by Mastroianni and Szabados [20] that

$$\Omega(f,\lambda t)_w \le C_w(\lambda+1)\,\Omega(f,t)_w. \tag{3.5}$$

For convenience, we write $\lambda_n = \frac{n}{m}$, or equivalently, $m = \frac{n}{\lambda_n}$. Then, the nodes in (3.1) can be rewritten as

$$x_k = \frac{\lambda_n k}{n},\ k = 0, \pm 1, \pm 2, \cdots, \pm n,$$

and the operator $g(x)$ as

$$g(x) := \sum_{k=-n}^{n} f(x_k)\frac{e^{-\frac{n}{\lambda_n}|x-x_k|}}{\sum_{i=-n}^{n} e^{-\frac{n}{\lambda_n}|x-x_i|}}.$$

Theorem 3.2. *If $f \in C_w(\mathbb{R})$, then*

$$\|w(f-g)\|_{\mathbb{R}} \le C\left(\Omega\left(f,\frac{\lambda_n}{n}\right)_w + \varepsilon_{wf}(\delta_n) + \frac{w(\lambda_n)}{w(\delta_n)} + \frac{\lambda_n^2}{n^2 w(\lambda_n)}\right), \tag{3.6}$$

where $\{\delta_n\}$ is any given sequence satisfying $0 < \delta_n < \lambda_n$, and

$$\varepsilon_{wf}(t) := \max_{|x|\ge t} |w(x) f(x)|.$$

Proof. By symmetry, it is sufficient to prove for $x \ge 0$. We prove the result by considering the following two cases.

Case 1. $0 \le x \le \lambda_n = x_n$, say $x \in [x_{j-1}, x_j]$, $j = 1, 2, \cdots, n$. Set

$$K_n(x) := \left\{k : |k| \le n, |x - x_k| \le \min\left\{\frac{2}{Q'\left(\frac{x+x_k}{2}\right)}, 1\right\}\right\}.$$

Then for $k \in K_n(x)$, it holds that (see [20])

$$\frac{w(x)}{w\left(\frac{x+x_k}{2}\right)} \le C_w. \tag{3.7}$$

By (3.4), (3.5), and (3.7), we have

$$\begin{aligned} & w(x)\sum_{k\in K_n(x)} |f(x)-f(x_k)|\frac{e^{-\frac{n}{\lambda_n}|x-x_k|}}{\sum_{i=-n}^{n} e^{-\frac{n}{\lambda_n}|x-x_i|}} \\ \le\ & C\max\frac{w(x)}{w\left(\frac{x+x_k}{2}\right)}\sum_{k\in K_n(x)} \Omega(f,|x-x_k|)_w\, e^{-|j-k|} \\ \le\ & C_w\Omega\left(f,\frac{\lambda_n}{n}\right)_w \sum_{k\in K_n(x)}\left(1+\frac{n|x-x_k|}{\lambda_n}\right)e^{-|j-k|} \end{aligned}$$

$$
\begin{aligned}
&\le C_w \Omega\left(f, \frac{\lambda_n}{n}\right)_w \sum_{k \in K_n(x)} (|j-k|+1)\, e^{-|j-k|} \\
&\le C_w \Omega\left(f, \frac{\lambda_n}{n}\right)_w . \qquad (3.8)
\end{aligned}
$$

On the other hand, by the fact that (see [20])

$$
w(x)\left(Q'\left(\frac{x+x_k}{2}\right)\right)^2 \le C_w,
$$

we have

$$
\begin{aligned}
& w(x) \sum_{k \notin K_n(x)} |f(x) - f(x_k)| \frac{e^{-\frac{n}{\lambda_n}|x-x_k|}}{\sum_{i=-n}^{n} e^{-\frac{n}{\lambda_n}|x-x_i|}} \\
\le\; & C \frac{w(x)\|wf\|}{w(\lambda_n)} \sum_{k \notin K_n(x)} \frac{(x-x_k)^2}{(x-x_k)^2} e^{-|j-k|} \\
\le\; & C \frac{w(x)\lambda_n^2 \|wf\|}{n^2 w(\lambda_n)} \sum_{k \notin K_n(x)} \left(Q'\left(\frac{x+x_k}{2}\right)\right)^2 (|j-k|+1)^2 e^{-|j-k|} \\
\le\; & C \frac{\lambda_n^2 \|wf\|}{n^2 w(\lambda_n)}. \qquad (3.9)
\end{aligned}
$$

From (3.8) and (3.9), we have

$$
|w(x)(f(x) - g(x))| \le C\left(\Omega\left(f, \frac{\lambda_n}{n}\right)_w + \frac{\lambda_n^2}{n^2 w(\lambda_n)}\right), \ |x| \le \lambda_n.
$$

Case 2. $x > \lambda_n$. For any $\delta_n < \lambda_n$, we have

$$
\begin{aligned}
|w(x)(f(x) - g(x))| &\le w(x) \sum_{k=-n}^{n} (|f(x)| + |f(x_k)|) \frac{e^{-\frac{n}{\lambda_n}|x-x_k|}}{\sum_{i=-n}^{n} e^{-\frac{n}{\lambda_n}|x-x_i|}} \\
&\le \varepsilon_{wf}(\lambda_n) + \|wf\|_{\mathbb{R}} \sum_{|k| \le n\delta_n/\lambda_n} \frac{w(x)}{w(x_k)} \frac{e^{-\frac{n}{\lambda_n}|x-x_k|}}{\sum_{i=-n}^{n} e^{-\frac{n}{\lambda_n}|x-x_i|}} \\
&\quad + \varepsilon_{wf}(\delta_n) \sum_{n\delta_n/\lambda_n < |k| \le n} \frac{w(x)}{w(x_k)} \frac{e^{-\frac{n}{\lambda_n}|x-x_k|}}{\sum_{i=-n}^{n} e^{-\frac{n}{\lambda_n}|x-x_i|}} \\
&\le \varepsilon_{wf}(\lambda_n) + C\left(\frac{w(\lambda_n)}{w(\delta_n)} + \varepsilon_{wf}(\delta_n)\right) \\
&\le C\left(\frac{w(\lambda_n)}{w(\delta_n)} + \varepsilon_{wf}(\delta_n)\right),
\end{aligned}
$$

which implies that (3.6) holds for $|x| > \lambda_n$. □

In a similar way to the corollary in Page 3 of [20], we have the following corollary of Theorem 3.2:

Corollary 3.1. *For any $f \in C_w(\mathbb{R})$, there exists $\lambda_n > \delta_n > 0$ such that*

$$
\lim_{n \to \infty} \|w(f-g)\| = 0.
$$

Remark 3.1. *From Corollary 3.1, we see that the neural networks of the form*

$$\mathfrak{g}(x) = \sum_{j=0}^{n} C_j \frac{e^{-A|x-x_j|}}{\sum_{i=0}^{n} e^{-A|x-x_i|}}$$

are always dense in the space $C_w(\mathbb{R})$ with respect to the weights $e^{-Q(x)}$ satisfying the conditions (i)–(iii). This is in sharp contrast with the weighted polynomial approximation, where the density condition is

$$\int_{-\infty}^{+\infty} \frac{Q(x)}{1+x^2} dx = \infty.$$

Therefore, our result shows that the neural networks are dense in a wider class of weighted functional spaces than polynomials. Thus, Corollary 3.1 implies that approximation by feedforward neural networks has some better properties than approximation by polynomials on infinite intervals.

4. Numerical example

In this section, we give some numerical experiments to demonstrate the validity of the obtained results. We take

$$f(x) = \begin{cases} (1+x)^{\frac{1}{3}}, & -1 \le x \le 0, \\ (1-x)^{\frac{1}{3}}, & 0 < x \le 1, \end{cases}$$

as the target function, Table 1 gives the approximate interpolation neural network operators $N_p(f, X, x)$. Figure 2 gives the error of $N_p(f, X, x)$ with different p, Figures 3–10 give the error of $g(x)$.

Table 1. Values of approximation error $\|N_p(f, X, x) - f\|_\infty$.

	$p = 1$	$p = 2$	$p = 3$	$p = 4$	$p = 5$
$n = 10$	0.0822	0.0549	0.0868	0.0749	0.017
$n = 20$	0.0653	0.0331	0.0459	0.0061	0
$n = 30$	0.0571	0.0242	0.0291	0.0021	0
$n = 40$	0.0518	0.0202	0.0209	0.0006	0
$n = 50$	0.0481	0.0161	0.0161	0	0

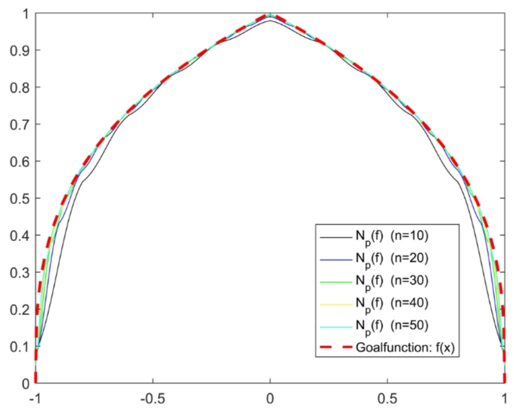

Figure 2. Errors of $N_p(f, X, x)$ with different p.

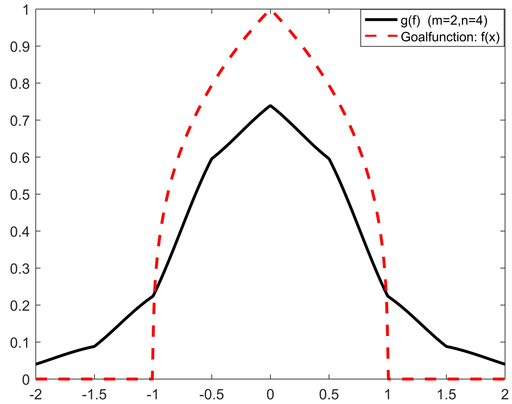

Figure 3. Errors of $g(x)$ with $m = 2, n = 4$.

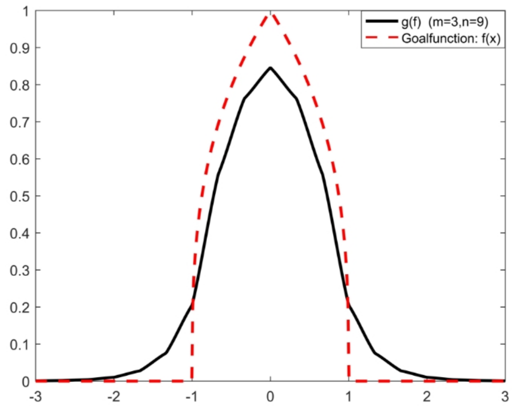

Figure 4. Errors of $g(x)$ with $m = 3, n = 9$.

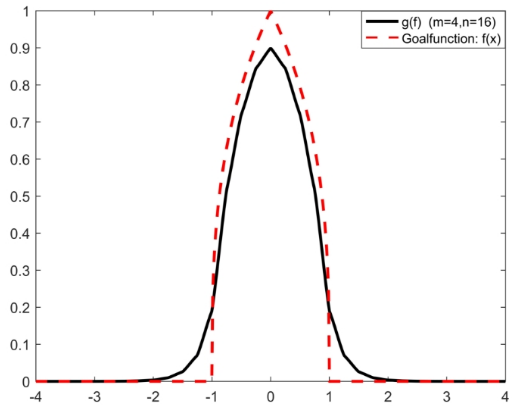

Figure 5. Errors of $g(x)$ with $m = 4, n = 16$.

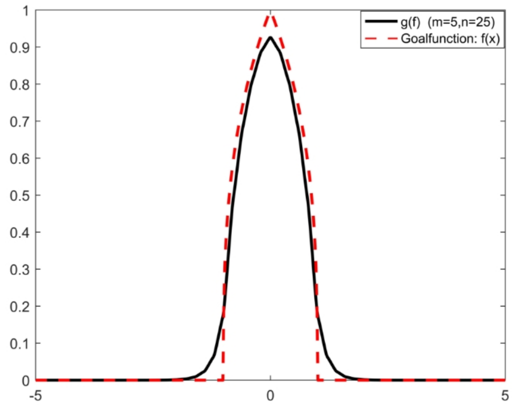

Figure 6. Errors of $g(x)$ with $m = 5, n = 25$.

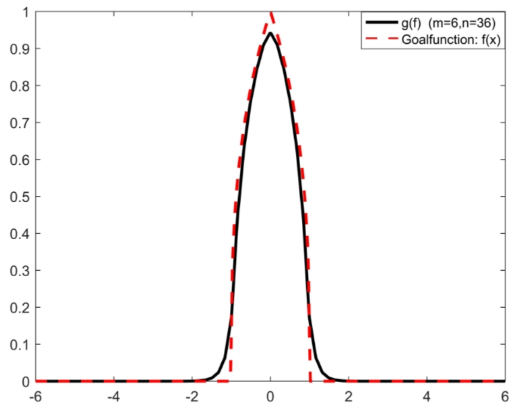

Figure 7. Errors of $g(x)$ with $m = 6, n = 36$.

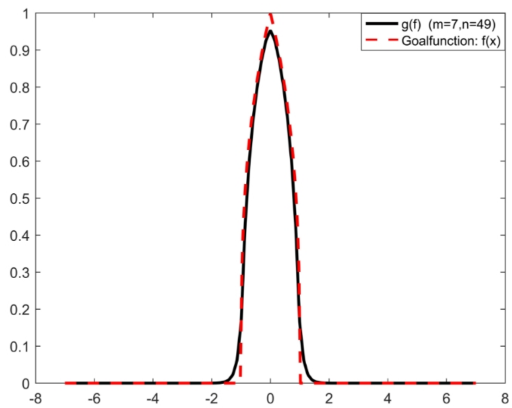

Figure 8. Errors of $g(x)$ with $m = 7, n = 49$.

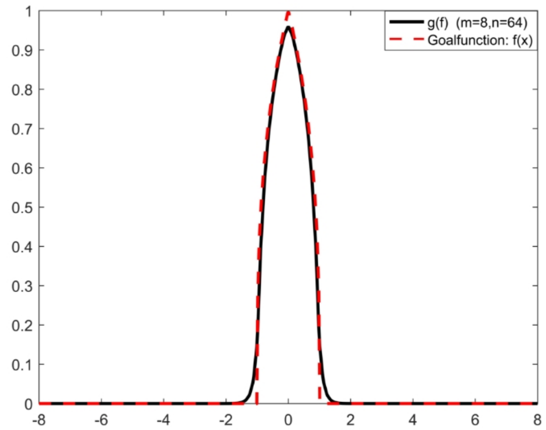

Figure 9. Errors of $g(x)$ with $m = 8, n = 64$.

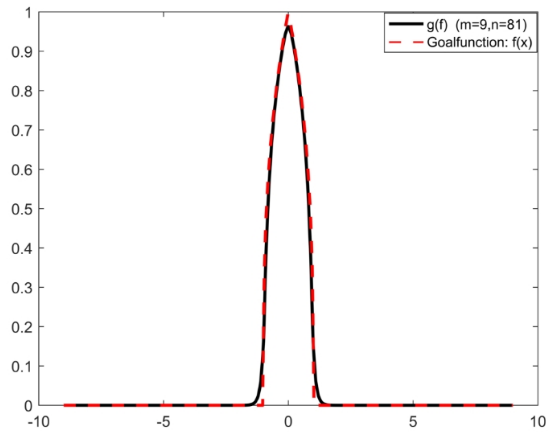

Figure 10. Errors of $g(x)$ with $m = 9, n = 81$.

5. Conclusions

In this paper, we constructs two kinds of feedforward neural network operators and estimates their approximation rate. One type of the operators can approximate the continuous functions with a very good rate which cannot be obtained by polynomial approximation. Another type of operators is constructed to investigate the weighted approximation properties by neural networks on infinite intervals. Thus, we demonstrate that approximation by feedforward neural networks has some better properties than approximation by polynomials on infinite intervals.

Acknowledgments

The work is supported by Hefei University Talent Research Fund Program (Grant No. 22040423015).

Conflict of interest

The author declares no conflicts of interest in this paper

References

1. G. A. Anastassiou, Univariate hyperbolic tangent neural network approximation, *Math. Comput. Model.*, **53** (2011), 1111–1132. https://doi.org/10.1016/j.mcm.2010.11.072
2. G. A. Anastassiou, Multivariate sigmoidal neural networks approximation, *Neural Netw.*, **24** (2011), 378–386. https://doi.org/10.1016/j.neunet.2011.01.003
3. F. L. Cao, T. F. Xie, Z. B. Xu, The estimate for approximation error of neural networks: A constructive approach, *Neurocomputing*, **71** (2008), 626–630. https://doi.org/10.1016/j.neucom.2007.07.024
4. F. L. Cao, Y. Q. Zhang, Z. R. He, Interpolation and rates of convergence for a class of neural networks, *Appl. Math. Model.*, **33** (2009), 1441–1456. https://doi.org/10.1016/j.apm.2008.02.009
5. F. L. Cao, Z. C. Li, J. W. Zhao, K. Lv, Approximation of functions defined on full axis of real by a class of neural networks: Density, complexity and constructive algorithm, *Chinese J. Comput.*, **35** (2012), 786–795. http://dx.doi.org/10.3724/SP.J.1016.2012.00786
6. Z. X. Chen, F. L. Cao, The approximation operators with sigmoidal functions, *Comput. Math. Appl.*, **58** (2009), 758–765. https://doi.org/10.1016/j.camwa.2009.05.001
7. D. X. Zhou, Universality of deep convolutional neural networks, *Appl. Comput. Harmon. Anal.*, **48** (2019), 787–794. https://doi.org/10.1016/j.acha.2019.06.004
8. C. K. Chui, S. B. Lin, B. Zhang, D. X. Zhou, Realization of spatial sparseness by deep ReLU nets with massive data, *IEEE Trans. Neural Netw. Learn. Syst.*, **33** (2022), 229–243. https://doi.org/10.1109/TNNLS.2020.3027613
9. X. Liu, Approximating smooth and sparse functions by deep neural networks: Optimal approximation rates and saturation, *J. Complexity*, **79** (2023), 101783. https://doi.org/10.1016/j.jco.2023.101783
10. D. X. Zhou, Theory of deep convolutional neural networks: Downsampling, *Neural Netw.*, **124** (2020), 319–327. https://doi.org/10.1016/j.neunet.2020.01.018
11. D. X. Zhou, Deep distributed convolutional neural networks: Universality, *Anal. Appl.*, **16** (2018), 895–919. https://doi.org/10.1142/s0219530518500124
12. G. S. Wang, D. S. Yu, L. M. Guan, Neural network interpolation operators of multivariate functions, *J. Comput. Appl. Math.*, **431** (2023), 115266. https://doi.org/10.1016/j.cam.2023.115266
13. D. S. Yu, Approximation by Neural networks with sigmoidal functions, *Acta. Math. Sin. English Ser.*, **29** (2013), 2013–2026. https://doi.org/10.1007/s10114-013-1730-2
14. D. S. Yu. Approximation by neural networks with sigmoidal functions, *Acta. Math. Sin. English Ser.*, **29** (2013), 2013–2026. https://doi.org/10.1007/s10114-013-1730-2

15. D. S. Yu, F. L. Cao, Construction and approximation rate for feedforward neural networks operators with sigmoidal functions, *J. Comput. Appl. Math.*, **453** (2025), 116150. https://doi.org/10.1016/j.cam.2024.116150
16. D. S. Yu, Y. Zhao, P. Zhou, Error estimates for the modified truncations of approximate approximation with Gaussian kernels, *Calcolo*, **50** (2013), 195–208. https://doi.org/10.1007/s10092-012-0064-2
17. I. E. Gopenguz, A theorem of A. F. Timan on the approximation of functions by polynomials on a finite segment, *Math. Notes Acad. Sci. USSR 1*, **1** (1967), 110–116. https://doi.org/10.1007/BF01268059
18. D. S. Yu, S. P. Zhou, Approximation by rational operators in L^p spaces, *Math. Nachr.*, **282** (2009), 1600–1618. https://doi.org/10.1002/mana.200610812
19. Z. Ditzian, V. Totik, *Moduli of smoothness*, New York: Springer, 1987. https://doi.org/10.1007/978-1-4612-4778-4
20. G. Mastroianni, J. Szabados, Balázs-Shepard operators on infinite intervals, II, *J. Approx. Theory*, **90** (1997), 1–8. https://doi.org/10.1006/jath.1996.3075