Uniform Approximation Capabilities of Sum-of-Product and Sigma-Pi-Sigma Neural Networks

Jinling Long, Wei Wu*, and Dong Nan

Dept. Appl. Math., Dalian University of Technology, Dalian 116023, P.R. China
wuweiw@dlut.edu.cn

Abstract. Investigated in this paper are the uniform approximation capabilities of sum-of-product (**SOPNN**) and sigma-pi-sigma (**SPSNN**) neural networks. It is proved that the set of functions that are generated by an **SOPNN** with its activation function in $C(\mathbb{R})$ is dense in $C(\mathbb{K})$ for any compact $\mathbb{K} \in \mathbb{R}^N$, if and only if the activation function is not a polynomial. It is also shown that if the activation function of an **SPSNN** is in $C(\mathbb{R})$, then the functions generated by the **SPSNN** are dense in $C(\mathbb{K})$ if and only if the activation function is not a constant.

1 Introduction

There have been many methods for multivariate function approximation: polynomials, Fourier series, tensor products, wavelets, radial basis functions, ridge functions, etc. In this respect, a current trend is to use artificial neural networks to compute superpositions and linear combinations of simple univariate functions. One of the most important problems for neural networks is their approximation capability. This problem is related to the question that whether, or under what conditions, multivariate functions can be represented or approximated by superpositions of univariate functions. There have been many papers related to this topic: [1,2,7,8,9,11] for feedforward neural networks (**FNN**), [3,4,10,13,16,17] for radial basis function neural networks (**RBFNN**), and [5,15] for Sigma-Pi neural networks. **SOPNN** and **SPSNN** are introduced respectively in [14] and [12], and the aim of this paper is to show their uniform approximation capability.

SOPNN can approximate nonlinear mappings in a similar manner as the **FNN** and **RBF**. The output of **SOPNN** has the form $\sum_{m=1}^{M} \prod_{n=1}^{N} f_{mn}(x_n)$, where x_n's are the inputs, N is the number of inputs, and M is the number of the product terms. The function $f_{mn}(x_n)$ is supposed to have the form $\sum_k \omega_{mnk} B_{nk}(x_n)$, where $B_{nk}(\cdot)$ is a univariate basis function and ω_{mnk}'s are the weights. If $B_{nk}(\cdot)$ is a Gaussian function, the new neural network degenerates to a Gaussian function network. The learning algorithm and the novel performance

* Corresponding author, supported by the National Natural Science Foundation of China (10471017).

D. Liu et al. (Eds.): ISNN 2007, Part I, LNCS 4491, pp. 1110–1116, 2007.

in function approximation, prediction, classification and learning control are presented in [14]. For convenience of hardware implementation, artificial neural networks usually require the nonlinear basis functions (the activation functions) to have a similar structure. So we restrict the form of $B_{nk}(x_n)$ to $g(a_{nk}x_n+\theta_{nk})$ for a given univariate function $g \in C(\mathbb{R})$, and concentrate our attention to the set of functions in the form $\sum_{m=1}^{M}\prod_{n=1}^{N}\sum_{k=1}^{K_n} c_{mnk}g(a_{nk}x_n+\theta_{nk})$, where M and K_n are in $\mathbb{N}$ (natural numbers); c_{mnk}, a_{nk}, $\theta_{nk} \in \mathbb{R}$; $x = (x_1, \cdots, x_N) \in \mathbb{R}^N$. We will prove that this set of functions is dense in $C(\mathbb{K})$ if and only if g is not a polynomial.

We point out that our proof for the approximation capability of **SOPNN** uses some corresponding results obtained in [11] for **FNN**. This fact reveals to some extent the relationship between the two kinds of neural networks.

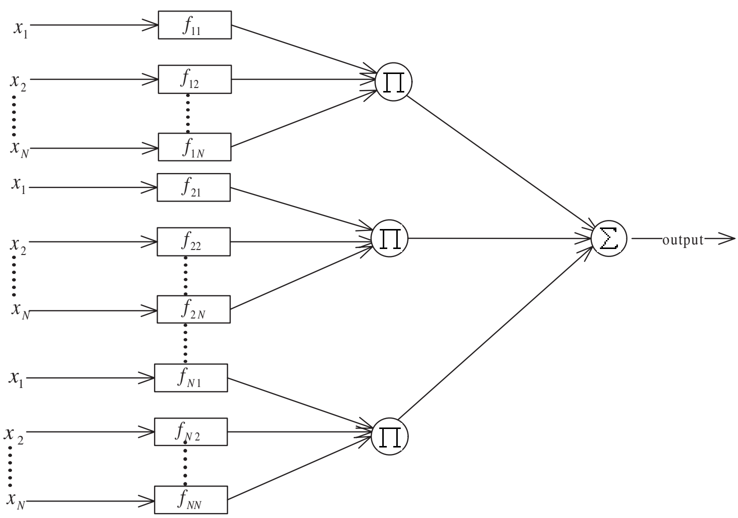

Fig. 1. Structure of sum-of-product neural network, $f_{mn}(x_n) = \sum_k \omega_{mnk}B_{nk}(x_n)$

The organization of this paper is as follows. The main result for the approximation capability of **SOPNN** is presented and proved in Section 2. In Section 3, the main result in Section 2 is generalized to deal with a sigma-pi-sigma neural network (**SPSNN**). Section 4 is devoted to a summary of results.

2 Approximation Capability of SOPNN

Lemma 1. ([6]: Weierstrass Approximation Theorem) *Let $\mathbb{K}$ be a compact set in $\mathbb{R}^N$. Then, the polynomials in N variables form a dense set in $C(\mathbb{K})$.*

Lemma 2. **([11])** *Let $f(t) \in C(\mathbb{R})$. The set of functions $\{\sum_{k=1}^{K} c_k f(\lambda_k \cdot x + \theta_k)\}$ is dense in $C(\mathbb{K})$ for any compact set $\mathbb{K}$ in $\mathbb{R}^N$, if and only if $f(t)$ is not a polynomial on $\mathbb{R}$, where c_k, $\theta_k \in \mathbb{R}$; x, $\lambda_k \in \mathbb{R}^N$; and $\lambda_k \cdot x$ denotes the inner product of λ_k and x.*

In the following, we show that for a continuous function to be qualified as an activation function in **SOPNN**, the necessary and sufficient condition is that it is not a polynomial. The next theorem is our main result on the approximation capacity of **SOPNN**.

Theorem 1. *Let $g(t) \in C(\mathbb{R})$. The family of the functions*

$$\left\{ \sum_{m=1}^{M} \prod_{n=1}^{N} \sum_{k=1}^{K_n} c_{mnk} g(a_{nk} x_n + \theta_{nk}) \;\middle|\; M, K_n \in \mathbb{N}; c_{mnk}, a_{nk}, \theta_{nk}, x_n \in \mathbb{R} \right\} \quad (1)$$

is dense in $C(\mathbb{K})$ for any compact set $\mathbb{K}$ in $\mathbb{R}^N$, if and only if $g(t)$ is not a polynomial on $\mathbb{R}$.

Proof. Sufficiency.
Since $\mathbb{K}$ is a compact set in $\mathbb{R}^N$, there exists a hypercube $\mathbb{H} = [a_1, b_1] \times \cdots \times [a_N, b_N]$, such that $\mathbb{K} \subset \mathbb{H}$.

Any $f(x) \in C(\mathbb{K})$ can be approximated by a multivariate polynomial thanks to Lemma 1, i.e., for any $0 < \varepsilon < 1$, there exists a multivariate polynomial $P(x) = \sum_{|i|=0}^{I} \alpha_i x^i$, such that

$$|f(x) - P(x)| < \frac{\varepsilon}{2}, \ \forall x \in \mathbb{K}, \quad (2)$$

where $x^i = x_1^{i_1} x_2^{i_2} \cdots x_N^{i_N}$, $i = (i_1, i_2, \cdots, i_N)$ is a multi-index, and $|i| = i_1 + i_2 + \cdots + i_N$.

Suppose $P(x)$ has $\widetilde{m}$ terms, $A = \max_{0 \le |i| \le I} \{|\alpha_i|, 1\}$, $B = \max_{1 \le j,\ k \le N} \{|a_j|, |b_k|, 1\}$, and $L = \max_{\substack{1 \le n \le N \\ 1 \le i_n \le I}} \left\{ \left| x_n^{i_n} \right|, a_n \le x_n \le b_n] \right\}$. We note that each component $x_n^{i_n}$ is a continuous function on $[a_n, b_n]$. Noting that g is not a polynomial and using Lemma 2, we can approximate $x_n^{i_n}$ by an **FNN** with g as its activation function in the following fashion: There exist $N_{i_n} \in \mathbb{N}$ and $c_k^{i_n}, \lambda_k^{i_n}, \theta_k^{i_n} \in \mathbb{R}$, such that

$$\left| x_n^{i_n} - \sum_{k=1}^{N_{i_n}} c_k^{i_n} g\left(\lambda_k^{i_n} x_n + \theta_k^{i_n}\right) \right| \le \frac{\varepsilon}{2^N A \widetilde{m} (L+1)^N B^I}, \forall x_n \in [a_n, b_n]. \quad (3)$$

Write $Q_n^{i_n}(x_n) = \sum_{k=1}^{N_{i_n}} c_k^{i_n} g\left(\lambda_k^{i_n} x_n + \theta_k^{i_n}\right)$. By Equation (3) we have that

$$\left| Q_n^{i_n}(x_n) \right| \le L + 1, \forall x_n \in [a_n, b_n], \quad (4)$$

and that

$$\left| x_n^{i_n} - Q_n^{i_n}(x_n) \right| \le \frac{\varepsilon}{2^N A \widetilde{m} (L+1)^N B^I}, \forall x = (x_1, x_2, \cdots, x_N) \in \mathbb{H}. \quad (5)$$

More generally, let us use an induction argument to establish the estimate

$$\left|x_1^{i_1}\cdots x_q^{i_q}-Q_1^{i_1}(x_1)\cdots Q_q^{i_q}(x_q)\right|\le\frac{\varepsilon}{2^{(N+1-q)}A\widetilde{m}(L+1)^{(N+1-q)}},\forall x\in\mathbb{H}. \quad (6)$$

Note that Equation (6) is already valid for $q=1$ due to (5). In the following, we assume that Equation (6) is valid for q with $1\le q\le N-1$, and we try to show that Equation (6) is also valid for $q+1$. To this end, we use the triangle inequality to get

$$\begin{aligned}&\left|x_1^{i_1}x_2^{i_2}\cdots x_{q+1}^{i_{q+1}}-Q_1^{i_1}(x_1)Q_2^{i_2}(x_2)\cdots Q_{q+1}^{i_{q+1}}(x_{q+1})\right|\\&\le\left|x_1^{i_1}\cdots x_q^{i_q}\right|\left|x_{q+1}^{i_{q+1}}-Q_{q+1}^{i_{q+1}}(x_{q+1})\right|\\&+\left|x_1^{i_1}\cdots x_q^{i_q}-Q_1^{i_1}(x_1)\cdots Q_q^{i_q}(x_q)\right|\left|Q_{q+1}^{i_{q+1}}(x_{q+1})\right|.\end{aligned}$$

It follows from Equations (3, 4, 6) and the above inequality that

$$\begin{aligned}&\left|x_1^{i_1}x_2^{i_2}\cdots x_{q+1}^{i_{q+1}}-Q_1^{i_1}(x_1)Q_2^{i_2}(x_2)\cdots Q_{q+1}^{i_{q+1}}(x_{q+1})\right|\\&\le\frac{\varepsilon}{2^{(N+1-(q+1))}A\widetilde{m}(L+1)^{(N+1-(q+1))}}.\end{aligned} \quad (7)$$

Here we have made the convention that $\prod_{t=q+2}^{N}(b_t-a_t)=1$ when $q=N-1$. So we have proved by induction that Equation (6) is valid for all $q=1,2,\cdots,N$.

In particular, by Equation (6) with $q=N$, we have

$$\left|x_1^{i_1}x_2^{i_2}\cdots x_N^{i_N}-Q_1^{i_1}(x_1)Q_2^{i_2}(x_2)\cdots Q_N^{i_N}(x_N)\right|\le\frac{\varepsilon}{2A\widetilde{m}},\forall x\in\mathbb{H}. \quad (8)$$

Set $Q(x)=\sum_{|i|=0}^{I}\alpha_iQ_1^{i_1}(x_1)Q_2^{i_2}(x_2)\cdots Q_N^{i_N}(x_N)$, then Equation (8) implies

$$\begin{aligned}|P(x)-Q(x)|&\le\sum_{|i|=0}^{I}|\alpha_i|\left|x_1^{i_1}x_2^{i_2}\cdots x_N^{i_N}-Q_1^{i_1}(x_1)Q_2^{i_2}(x_2)\cdots Q_N^{i_N}(x_N)\right|\\&\le\frac{\varepsilon}{2},\forall x\in\mathbb{H}.\end{aligned} \quad (9)$$

A combination of Equations (2) and (9) implies

$$|f(x)-Q(x)|\le|f(x)-P(x)|+|P(x)-Q(x)|<\frac{\varepsilon}{2}+\frac{\varepsilon}{2}=\varepsilon,\forall x\in\mathbb{K}. \quad (10)$$

Obviously $Q(x)\in\left\{\sum_{m=1}^{M}\prod_{n=1}^{N}\sum_{k=1}^{K_n}c_{mnk}g(a_{nk}x_n+\theta_{nk})\right\}$, and the estimate (10) shows that $C(\mathbb{K})\subseteq\overline{\left\{\sum_{m=1}^{M}\prod_{n=1}^{N}\sum_{k=1}^{K_n}c_{mnk}g(a_{nk}x_n+\theta_{nk})\right\}}$. Also note the apparent inclusion relation $\left\{\sum_{m=1}^{M}\prod_{n=1}^{N}\sum_{k=1}^{K_n}c_{mnk}g(a_{nk}x_n+\theta_{nk})\right\}\subseteq C(\mathbb{K})$. Then we see that $\overline{\left\{\sum_{m=1}^{M}\prod_{n=1}^{N}\sum_{k=1}^{K_n}c_{mnk}g(a_{nk}x_n+\theta_{nk})\right\}}=C(\mathbb{K})$.

Necessity.

Now let us assume $g(t)$ is a univariate polynomial with degree l. Then, all the functions in the form of (1) are polynomials with degrees at most l^N, which of course are not dense in $C(\mathbb{K})$. □

Remark 1. Theorem 1 describes an approximate representation of multivariate functions by univariate functions. It provides an answer to the question that what kinds of univariate functions are qualified to approximate multivariate functions. In our proof, $Q(x)$ as the approximant to a function in $C(\mathbb{K})$ is entirely constructed in terms of univariate functions $Q_n^{i_n}(x_n)$. The role of $Q_n^{i_n}(x_n)$ in the approximant $Q(x)$ is very much like the role of $x_n^{i_n}$ in the polynomial $P(x)$ as an approximant to a general nonlinear function.

3 Approximation Capability of SPSNN

A sigma-pi-sigma neural network (**SPSNN**) is proposed in [12]. Its output is

$$\sum_{m=1}^{M}\prod_{j=1}^{J_m}\sum_{n=1}^{N}\sum_{k=1}^{K_n}\omega_{mjnk}B_{jnk}(x_n), \tag{11}$$

where $x = (x_1, \cdots, x_N) \in \mathbb{R}^N$ is the input, ω_{mjnk}'s are the weights, $B_{jnk}(\cdot)$'s are univariate basis functions, and M, J_m and $K_n \in \mathbb{N}$. Similarly as before, we concentrate our attention to the family of functions of the form

$$\sum_{m=1}^{M}\prod_{j=1}^{J_m}\sum_{n=1}^{N}\sum_{k=1}^{K_n}c_{mjnk}g(a_{jnk}x_n + \theta_{jnk}),$$

where $c_{mjnk}, a_{jnk}, \theta_{jnk} \in \mathbb{R}$.

In this section, we show that for a continuous function to be qualified as an activation function in **SPSNN**, the necessary and sufficient condition is that it is not a constant. The next theorem is the other main result on the approximation capacity of **SPSNN**.

Theorem 2. *Suppose $g(t) \in C(\mathbb{R})$. The family of functions of $x = (x_1, \cdots, x_N)$*

$$\left\{ \sum_{m=1}^{M}\prod_{j=1}^{J_m}\sum_{n=1}^{N}\sum_{k=1}^{K_n}c_{mjnk}g(a_{jnk}x_n + \theta_{jnk}) \;\middle|\; M, K_n, J_m \in \mathbb{N}; c_{mjnk}, a_{jnk}, \theta_{jnk} \right\} \tag{12}$$

is dense in $C(\mathbb{K})$ for any compact set $\mathbb{K}$ in $\mathbb{R}^N$, if and only if $g(t)$ is not a constant function.

Proof. Sufficiency.

We first consider the case that $g(t)$ is not a polynomial. Note that the family of functions of form (1) is a subset of that of form (12), thus the latter must be dense in $C(\mathbb{K})$ for any compact set $\mathbb{K}$ in $\mathbb{R}^N$ according to Theorem 1.

Next, we turn to the other case that $g(t)$ is a polynomial but not a constant function on $\mathbb{R}$. We claim that $g(t)$ can generate the monomial t by translations, stretching and sums. To see this, let us assume that $g(t) = a_l t^l + a_{l-1} t^{l-1} + \cdots + a_0$ with $l \geq 1$ and $a_l \neq 0$. Then, $g(t+1) = a_l(t+1)^l + a_{l-1}(t+1)^{l-1} + \cdots + a_0$, and we have that $h_1(t) \equiv g(t+1) - g(t) = l a_l t^{l-1} + p_{(l-2)}(t)$, where $p_{(l-2)}(t)$ denotes a polynomial with degree equal to or less than $l-2$. We proceed to observe that $h_2(t) \equiv h_1(t+1) - h_1(t) = l(l-1)a_l t^{(l-2)} + p_{(l-3)}(t)$. And we can repeat this procedure to obtain $h_{l-1} = l(l-1)\cdots 2a_l t + p_0(t) = l(l-1)\cdots 2a_l t + b_0$, where b_0 is a constant. Let $b_1 = l(l-1)\cdots 2a_l$, then $h(t) \equiv \frac{1}{b_1}(h_{(l-1)}(t) - b_0) = t$. This confirms the claim. Next, we notice that the functions of form (12) are all polynomials and they constitute an algebra. On the other hand, it follows from the above claim that all the monomials $x_1, x_2, \cdots, x_N$ are members of the family (12). Thus, the family (12) must contains all the multivariate polynomials. Therefore, by Lemmas 1, the family (12) is also dense in $C(\mathbb{K})$ for any compact set $\mathbb{K}$ in $\mathbb{R}^N$ when $g(t)$ is a nonconstant polynomial.

Necessity.

Otherwise, if $g(t)$ is a constant function on $\mathbb{R}$, then the functions of form (12) are all constant functions on $\mathbb{R}^N$, which of course are not dense in $C(\mathbb{K})$. □

Remark 2. Theorem 2 gives the other approximate representation of multivariate functions by univariate functions. It provides an answer to the question that what kinds of univariate functions are qualified to approximate multivariate functions by the **SPSNN**. We can see that any nonconstant continuous function can be used as an activation function in the **SPSNN**.

4 Conclusion

The approximations of multivariate functions by sum-of-product and sigma-pi-sigma neural networks with a univariate activation function are investigated. This paper solves the problem of whether a function is qualified as an activation function in the two kinds of new structure neural networks. We have proved that all the functions generated by the **SOPNN** with its activation function in $C(\mathbb{R})$ are dense in $C(\mathbb{K})$, if and only if the activation function is not a polynomial. We also show that if the activation function of the **SPSNN** is in $C(\mathbb{R})$, then the functions generated by the **SPSNN** are dense in $C(\mathbb{K})$ if and only if the activation function is not a constant.

References

1. Attali, J.G., Pagès, G.: Approximations of Functions by a Multilayer Perceptron: a New approach. Neural Networks **10** (1997) 1069-1081
2. Chen, T.P., Chen, H., Liu, R.: Approximation Capability in by Multilayer Feedforward Networks and Related Problems. IEEE Transactions on Neural Networks **6** (1) (1995) 25-30

3. Chen, T.P., Chen, H.: Approximation Capability to Functions of Several Variables, Nonlinear Functionals and Operators by Radial Basis Function Neural Networks. IEEE Transactions on Neural Networks **6** (4) (1995) 904-910
4. Chen, T.P., Chen, H.: Universal Approximation Capability of RBF Neural Networks with Arbitrary Activation Functions and Its Application to Dynamical Systems. Circuits, Systems and Signal Processing **15** (5) (1996) 671-683
5. Chen, T.P., Wu, X.W.: Characteristics of Activation Function in Sigma-Pi Neural Networks. Journal of Fudan University **36** (6) (1997) 639-644
6. Cheney, W., Light, W.: A Course in Approximation Theory. Beijing: China Machine Press (2003)
7. Chui, C.K., Li, X.: Approximation by Ridge Functions and Neural Networks with One Hidden Layer. Journal of Approximation Theory **70** (1992) 131-141
8. Cybenko, G.: Approximation by Superpositions of Sigmoidal Functions. Mathematics of Control, Signals, and Systems **2** (1989) 303-314
9. Hornik, K.: Approximation Capabilities of Mutilayer Feedforward Networks. Neural Networks **4** (1991) 251-257
10. Jiang, C.H.: The Approximate Problem on Neural Network. Annual of Math (in Chinese) **19**A (1998) 295-300
11. Leshno, M., Lin, Y.V., Pinkus, A., Schocen, S.: Multilayer Feedforward Networks with a Non-Polynomial Activation Function Can Approximate Any Function. Neural Networks **6** (1993) 861-867
12. Li, C.K.: A Sigma-Pi-Sigma Neural Network (SPSNN). Neural Processing Letters **17** (2003) 1-19
13. Liao, Y., Fang, S., Nuttle, H.L.W.: Relaxed Conditions for Radial-Basis Function Networks to Be Universal Approximators. Neural Networks **16** (2003) 1019-1028
14. Lin, C.S., Li, C.K.: A Sum-of-Product Neural Network(SOPNN). Neurocomputing **30** (2000) 273-291
15. Luo, Y.H., Shen, S.Y.: L^p Approximation of Sigma-Pi Neural Networks. IEEE Transactions on Neural Networks **11** (6) (2000) 1485-1489
16. Park, J., Sandberg, I.W.: Universal Approximation Using Radial- Basis-Function Networks. Neural Computation **3** (1991) 246-257
17. Park, J., Sandberg, I.W.: Approximation and Radial-Basis-Function Networks. Neural Computation **5** (1993) 305-316