REVIEW OF "NEW ERROR BOUNDS FOR DEEP RELU NETWORKS USING SPARSE GRIDS"*

DONGRUI SHEN

1. Introduction. Deep learning is based on approximations by deep networks. Deep networks are neural networks with one or more hidden layers. One hidden layer neural networks correspond to approximations f_N with N units of multivariate functions $f : \mathbb{R}^d \to \mathbb{R}$ of the form

$$f_N(\boldsymbol{x}) = \sum_{i=1}^{N} \alpha_i \sigma\left(\boldsymbol{w}_i^T \boldsymbol{x} + \theta_i\right), \quad \alpha_i, \theta_i \in \mathbb{R}, \boldsymbol{x}, \boldsymbol{w}_i \in \mathbb{R}^d, \tag{1.1}$$

for some activation function $\sigma : \mathbb{R} \to \mathbb{R}$. For *deep* networks, each unit of each layer performs an operation of the form $\sigma(\boldsymbol{\omega} \cdot \boldsymbol{x} + \theta)$. Deep *ReLU* networks use the activation function $\sigma(x) = \max(0, x)$. The *depth* of a neural network is defined as the number of hidden layers and the *size* is the total number of units.

Back to the late 1980s, it has been shown that any continuous functions can be approximated by *shallow* networks that use *sigmoid* functions as activation functions. A similar result for Borel measurable functions was also proved. These works provide the essential theoretical support for machine learning with neural networks. However, from a practical point of view, it is also important to consider how fast approximations by neural networks converge and how expensive the method is. For example, for a real valued function f in $\mathbb{R}^d$ and for some accuracy constant $\varepsilon > 0$, there exists a neural network f_N of size N that satisfies

$$\|f - f_N\| < \varepsilon \text{ with } N = \mathcal{O}(\varepsilon^{-\frac{d}{m}}),^{1} \tag{1.2}$$

for some norm $\|\cdot\|$, where m is the order of integrable or bounded derivatives. For large dimensions d, the size N increases at a geometric rate with d. Such a phenomenon is known as the "curse of dimensionality". Many results of the form (1.2) have been derived for shallow and deep networks; see Table 1a.

*Montanelli, Hadrien and Du, Qiang, 2019. New Error Bounds for Deep ReLU Networks Using Sparse Grids. SIAM journal on mathematics of data science, 1(1), pp.78–92. https://epubs.siam.org/doi/abs/10.1137/18M1189336?mobileUi=0

[1] The big O notation here means that there exists a $C > 0$ such that $N \le C\varepsilon^{-\frac{d}{m}}$ for sufficiently small ε, where C might depend on the dimension d.

2. Main results. Their paper aims to address the problem, why and when deep networks can lessen or break the curse of dimensionality. Unlike many related works that focus on a particular set of functions which have a very special structure (such as *compositional* or *polynomial*; see Table 1b), they consider functions in the Korobov spaces which is more general for high dimensional multivariate approximation. The Korobov spaces are defined by

$$X^{2,p}(\Omega) = \left\{ f \in L^p(\Omega) : f|_{\partial\Omega} = 0, D^k f \in L^p(\Omega), |\boldsymbol{k}|_\infty \leq 2 \right\}, \tag{2.1}$$

with norm $|f|_{2,\infty} = \left\| \frac{\partial^{2d} f}{\partial x_1^2 \ldots \partial x_d^2} \right\|_\infty$. By establishing a connection with sparse grids, they present new error estimates for which the curse of dimensionality is lessened; see Theorem 2.1 below.

THEOREM 2.1. *For any dimension d and $0 < \varepsilon < 1$, there is a deep ReLU network with d inputs $x_1, ..., x_d$ capable of expressing any function f in $X^{2,p}([0,1]^d)$ that satisfies $|f|_{2,\infty} \leq 1$ with accuracy ε, and has depth $\mathcal{O}(|\log_2 \varepsilon|(d-1))$ and size $\mathcal{O}(\varepsilon^{-\frac{1}{2}} |\log_2 \varepsilon|^{\frac{3}{2}(d-1)+1}(d-1))$.*

Compared with the result in Table 1a, the exponent d only affects logarithm factors $|\log_2 \varepsilon|$ instead of ε^{-1}. Their result shows that Deep *ReLU* networks can significantly lessen the effect of large dimensions d.

3. Sketch of proof. They use the following technique to prove the new error bounds. Show certain functions f can be approximated by sparse grids f_m to any prescribed accuracy ε, and so sparse grids f_M by neural networks f_N of size N. Together, the approximation error can be decomposed as

$$\|f - f_N\| \leq \|f - f_m\| + \|f_m - f_N\|, \tag{3.1}$$

for some norm $\|\cdot\|$.

3.1. Approximating functions in the Korobov spaces using sparse girds. To approximate functions of d variables $\boldsymbol{x} = (x_1, \ldots, x_d) \in [0,1]^d$, they introduce a tensor product construction. One can consider a family of grids $\Omega_{\boldsymbol{l}}$ of level $\boldsymbol{l} = (l_1, \ldots, l_d)$ with a grid size $\boldsymbol{h_l} = (2^{-l_1}, \ldots, 2^{-l_d})$ and $2^l - 1$ points $\boldsymbol{x_{i,l}} = \boldsymbol{i} \bigotimes \boldsymbol{h_l}$, $\boldsymbol{1} < \boldsymbol{i} < \boldsymbol{2^l} - \boldsymbol{1}$. For each $\Omega_{\boldsymbol{l}}$, one defines piecewise linear hat functions

$$\phi_{\boldsymbol{l},\boldsymbol{i}} = \prod_{j=1}^{d} \phi_{l_j,i_j}(x_j), \tag{3.2}$$

where $\phi(x_{l,i}) = \phi(\frac{x-x_{l,i}}{h_l})$ and $\phi(x) = \max(0, 1-|x|)$.

Consider a function spaces spanned by these functions $V_{\boldsymbol{l}} = \text{span}\{\phi_{\boldsymbol{l},\boldsymbol{i}} : \boldsymbol{1} \leq \boldsymbol{i} \leq \boldsymbol{2}^l - 1\}$ and the hierarchical increments space $W_{\boldsymbol{l}} = \text{span}\{\phi_{\boldsymbol{l},\boldsymbol{i}} : \boldsymbol{i} \in \boldsymbol{I}_l\}$, where $\boldsymbol{I_l} = \{\boldsymbol{i} \in \mathbb{N}^d : \boldsymbol{1} \leq \boldsymbol{i} \leq \boldsymbol{2}^l - 1, i_j \text{ odd for all } j\}$. These increment spaces satisfy the relation, $V_{\mathbf{m}} = \bigoplus_{\boldsymbol{1} \leq \boldsymbol{l} \leq \boldsymbol{m}} W_{\boldsymbol{l}}$.

Sparse grids are discretizations of $X^{2,p}(\Omega)$ defined by $V_m^{(1)} = \bigoplus_{1 \leq |\boldsymbol{l}|_1 \leq m+d-1} W_l$ and correspond to a number of grid points $M = \mathcal{O}\left(h_m^{-1} |\log_2 h_m|^{d-1}\right)$; see Figure 1 for a sparse grid in two dimensions. For any $f_m^{(1)} \in V_m^{(1)}$,

$$f_m^{(1)}(\boldsymbol{x}) = \sum_{|\boldsymbol{l}|_1 \leq m+d-1} \sum_{i \in \boldsymbol{I}_l} v_{\boldsymbol{l},\boldsymbol{i}} \phi_{\boldsymbol{l},\boldsymbol{i}}(\boldsymbol{x}), \tag{3.3}$$

where the hierarchical coefficients $v_{\boldsymbol{l},\boldsymbol{i}}$ depends on two order mixed derivatives of f. For any prescribed accuracy ε, $\left\|f - f_m^{(1)}\right\|_\infty = \varepsilon$ with $N = \mathcal{O}\left(\varepsilon^{-\frac{1}{2}} |\log_2 \varepsilon|^{\frac{3}{2}(d-1)}\right)$.

3.2. Approximating sparse girds by deep networks. The following proposition shows how deep networks can approximate multidimensional hat functions.

PROPOSITION 3.1. *For any dimension d and $0 < \varepsilon < 1$, there is a deep ReLU network with d inputs $x_1, \ldots, x_d$ that estimates the multiplication $\phi_{l,i}(\boldsymbol{x}) = \prod_{j=1}^{d} \phi_{l_j,i_j}(x_j)$ with accuracy ε, outputs 0 if one of the $\phi_{l_j,i_j}(x_j)$ is 0, and has depth $\mathcal{O}(|\log_2 \varepsilon| \log_2 d)$ and size $\mathcal{O}(|\log_2 \varepsilon|(d-1))$.*

Then, with the fact that functions in $X^{2,p}([0,1]^d)$ can be approximated by sparse grids $f_m \in V_m^{(1)}$, show that sparse grids can be represented by deep networks f_N using the approximated multiplication written as $\widetilde{\phi}_{\boldsymbol{l},\boldsymbol{i}}(\boldsymbol{x})$:

$$f_N(\boldsymbol{x}) = \sum_{|\boldsymbol{l}|_1 \leq m+d-1} \sum_{i \in \boldsymbol{I}_l} v_{\boldsymbol{l},\boldsymbol{i}} \widetilde{\phi}_{\boldsymbol{l},\boldsymbol{i}}(\boldsymbol{x}). \tag{3.4}$$

The corresponding network is shown in Figure 2.

4. Conclusion. Their proof is based on the ability of deep networks to approximate sparse grids via a binary tree structure (see Figure 2a). Their result provides an upper bound for the approximation complexity when the same network is used to approximate all functions in a given Korobov space, without taking advantage of special properties of the approximated functions. Yet it is pointed out that sparse grids they used are highly *anisotropic*: to be efficient, these require the functions being approximated to be aligned with the axes.

Table 1: Approximation results for different activation functions.

(a) Approximation results with the curse of dimensionality.

	Shallow	**Deep**
$\boldsymbol{\sigma} \in \boldsymbol{C^\infty}(\mathbb{R})$ **(not polynomial)**	$f \in W^{m,p}([-1,1]^d)$ depth 1, size $\mathcal{O}(\varepsilon^{-\frac{d}{m}})$ $\lVert\cdot\rVert_p$	-
$\boldsymbol{\sigma} \in \boldsymbol{C^\infty}(\mathbb{R})$ **(not polynomial)**	f analytic in E_ρ depth 1, size $\mathcal{O}(\lvert\log_\rho \varepsilon\rvert)$ $\lVert\cdot\rVert_p$	-
$\boldsymbol{\sigma}$ **ReLU**	$f \in W^{m,2}(B^d)$ depth 1, size $\mathcal{O}(\varepsilon^{-\frac{d}{m}})$ $\lVert\cdot\rVert_2$	$f \in W^{m,\infty}([0,1]^d)$ depth $\mathcal{O}(\lvert\log_2 \varepsilon\rvert)$, size $\mathcal{O}(\varepsilon^{-\frac{d}{m}})\lvert\log_2 \varepsilon\rvert$ $\lVert\cdot\rVert_\infty$

(b) Approximation results without the curse of dimensionality.

	Shallow	**Deep**
$\boldsymbol{\sigma} \in \boldsymbol{C^\infty}(\mathbb{R})$ **(not polynomial)**	$f \in W^{m,\infty}([-1,1]^d)$, compositional depth 1, size $\mathcal{O}(\varepsilon^{-\frac{d}{m}})$ $\lVert\cdot\rVert_\infty$	$f \in W^{m,\infty}([-1,1]^d)$, compositional depth $\log_2 d$, size $\mathcal{O}((d-1)\varepsilon^{-\frac{2}{m}})$ $\lVert\cdot\rVert_\infty$
$\boldsymbol{\sigma}$ **ReLU**	f Lipschitz, $[-1,1]^d$, compositional depth 1, size $\mathcal{O}(\varepsilon^{-d})$ $\lVert\cdot\rVert_\infty$	f Lipschitz, $[-1,1]^d$, compositional depth $\log_2 d$, size $\mathcal{O}((d-1)\varepsilon^{-w})$ $\lVert\cdot\rVert_\infty$

Fig. 1: Left: All subspaces $W_{\boldsymbol{l}}$ in two dimensions for $(l_1, l_2) \leq (3,3)$, and sparse and full grids $V_3^{(1)}$ and $V_3^{(\infty)}$. Right: A sparse grid in two dimensions.

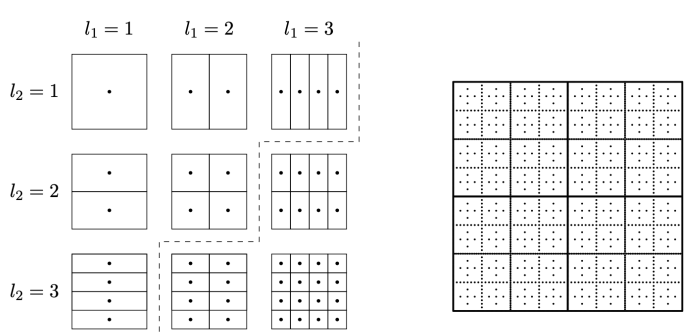
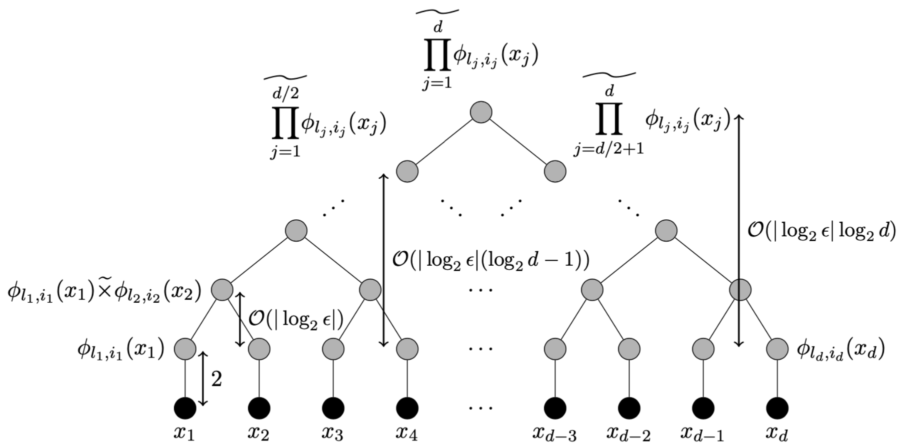


Fig. 2: The sparse grid based deep network.

(a) The network that implements the $(d-1)$ products in $\prod_{j=1}^{d} \phi_{l_j,i_j(x_j)}$ with a binary tree structure.



(b) The network consists of M subnetworks $S_1, S_2, \ldots, S_M$, which implement the multiplication, $\prod_{j=1}^{d} \phi_{l_j,i_j}(x_j)$.