A Few Thoughts on Deep Network Approximation

Haizhao Yang
Department of Mathematics
Purdue University

Joint work with Zuowei Shen and Shijun Zhang
National University of Singapore

Mini-Symposium on Approximation Theory of Neural Networks
SIAM Annual Meeting
July 23rd, 2021

Deep neural networks

$$y = h(x; \theta) := T \circ \phi(x) := T \circ h^{(L)} \circ h^{(L-1)} \circ \cdots \circ h^{(1)}(x)$$

where

- $h^{(i)}(x) = \sigma({W^{(i)}}^T x + b^{(i)})$;
- $T(x) = V^T x$;
- $\theta = (W^{(1)}, \cdots, W^{(L)}, b^{(1)}, \cdots, b^{(L)}, V)$.

Deep Network Approximation

Goals

- Approximation error in terms of width and depth
- The curse of dimensionality exist? e.g., # parameters not $(\frac{1}{\epsilon})^d$
- Is exponential approximation rate available? e.g., # parameters $\log(\frac{1}{\epsilon})$

Literature Review

Functions spaces

- Continuous functions
- Smooth functions
- Functions with integral representations

ReLU DNNs, continuous functions $C([0,1]^d)$

ReLU; Fixed network width $O(N)$ and depth $O(L)$

- Nearly tight error rate $5\omega_f(8\sqrt{d}N^{-2/d}L^{-2/d})$ simultaneously in N and L with L^∞-norm. Shen, Y., and Zhang (CiCP, 2020)
- ω_f is the modulas of continuity
- Improved to a tight rate $O\left(\sqrt{d}\,\omega_f\left(\left(N^2L^2\log_3(N+2)\right)^{-1/d}\right)\right)$. Shen, Y., and Zhang (J Math Pures Appl, 2021)

Curse of dimensionality exists!

ReLU DNNs, smooth functions $C^s([0,1]^d)$

Does smoothness help?

ReLU; Fixed network width $O(N)$ and depth $O(L)$

- Nearly tight rate $85(s+1)^d 8^s \|f\|_{C^s([0,1]^d)} N^{-2s/d} L^{-2s/d}$ simultaneously in N and L with L^∞-norm
- Lu, Shen, Y., and Zhang (SIMA, 2021)

The curse of dimensionality exists if s is fixed.

DNNs with advanced activation function

Sine-ReLU; Fixed width $O(d)$, varying depth L

- $\exp(-c_{r,d}\sqrt{L})$ with L^∞-norm for $C^r([0,1]^d)$
- Root exponential approximation rate achieved
- Curse of dimensionality is not clear
- Yarotsky and Zhevnerchuk, NeurIPS 2020

Floor and ReLU activation, width $O(N)$ and depth $O(dL)$, $C([0,1]^d)$

- Error rate $\omega_f(\sqrt{d}N^{-\sqrt{L}}) + 2\omega_f(\sqrt{d})N^{-\sqrt{L}}$ with L^∞-norm
- NO curse of dimensionality for many continuous functions
- Root exponential approximation rate
- Merely based on the compositional structure of DNNs and depth is the key
- Shen, Y., and Zhang (Neural Computation, 2020)

DNNs with advanced activation function

Can width be as powerful as depth?

Floor, Sign, and 2^x activation, width $O(N)$ and depth 3, $C([0,1]^d)$

- Error rate $\omega_f(\sqrt{d}2^{-N}) + 2\omega_f(\sqrt{d})2^{-N}$ with L^∞-norm
- NO curse of dimensionality for many continuous functions
- Exponential approximation rate
- Merely based on the compositional structure of DNNs and width is the key
- Shen, Y., and Zhang (Neural Networks, 2021)

Further interpretation of our result

Explicit error bound

Floor, Sign, and 2^x activation, width $O(N)$ and depth 3, Hölder($[0,1]^d, \alpha, \lambda$)

- Error rate $3\lambda(2\sqrt{d})^\alpha 2^{-\alpha N}$ with L^∞-norm
- NO curse of dimensionality
- Exponential approximation rate
- Shen, Y., and Zhang (Neural Networks, 2021)

Key ideas of our approximation

For $\boldsymbol{x} \in Q_{\boldsymbol{\beta}}$:
$$\boldsymbol{x} \to \phi_1(\boldsymbol{x}) = \boldsymbol{\beta} \to \phi_2(\boldsymbol{\beta}) = k_{\boldsymbol{\beta}} \to \phi_3(k_{\boldsymbol{\beta}}) = f(\boldsymbol{x}_{\boldsymbol{\beta}}) \approx f(\boldsymbol{x})$$

- Piecewise constant approximation: $f(\boldsymbol{x}) \approx f_p(\boldsymbol{x}) \approx \phi_3 \circ \phi_2 \circ \phi_1(\boldsymbol{x})$
- 2^N pieces per dim and 2^{Nd} pieces with accuracy 2^{-N}
- Floor NN $\phi_1(\boldsymbol{x})$ s.t. $\phi_1(\boldsymbol{x}) = \boldsymbol{\beta}$ for $\boldsymbol{x} \in Q_{\boldsymbol{\beta}}$ and $\boldsymbol{\beta} \in \mathbb{Z}^d$.
- Linear NN ϕ_2 mapping $\boldsymbol{\beta}$ to an integer $k_{\boldsymbol{\beta}} \in \{1, \ldots, 2^{Nd}\}$
- Key difficulty: NN ϕ_3 of width $O(N)$ and depth $O(1)$ fitting 2^{Nd} samples in 1D with accuracy $O(2^{-N})$
- ReLU NN fails

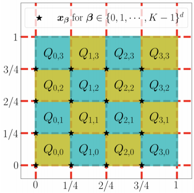

Figure: Uniform domain partitioning.

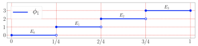

Figure: Floor function.

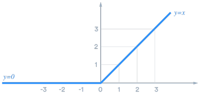

Figure: ReLU function.

Key ideas of our approximation

Binary representation and approximation

$\theta = \sum_{\ell=1}^{\infty} \theta_\ell 2^{-\ell}$ with $\theta_\ell \in \{0, 1\}$ is approximated by $\sum_{\ell=1}^{N} \theta_\ell 2^{-\ell}$ with an error 2^{-N}.

Bit extraction via a floor NN of width 2 and depth 1

$$\phi_k(\theta) := \lfloor 2^k \theta \rfloor - 2\lfloor 2^{k-1}\theta \rfloor = \theta_k$$

Bit extraction via a floor NN of width 2N and depth 1

Given $\theta = \sum_{\ell=1}^{\infty} \theta_\ell 2^{-\ell}$

$$\phi(\theta) := \begin{pmatrix} \phi_1(\theta) \\ \vdots \\ \phi_N(\theta) \end{pmatrix} = \begin{pmatrix} \theta_1 \\ \vdots \\ \theta_N \end{pmatrix} \in \mathbb{Z}^N$$

Key ideas of our approximation

Encoding K numbers to one number

- Extract bits $\{\theta_1^{(k)}, \ldots, \theta_N^{(k)}\}$ from $\theta^{(k)} = \sum_{\ell=1}^{\infty} \theta_\ell^{(k)} 2^{-\ell}$ for $k = 1, \ldots, K$
- sum up to get $a = \sum_{\ell=1}^{N} \theta_\ell^{(1)} 2^{-\ell} + \sum_{\ell=N+1}^{2N} \theta_\ell^{(2)} 2^{-\ell} + \cdots + \sum_{\ell=(K-1)N+1}^{KN} \theta_\ell^{(K)} 2^{-\ell}$

Decoding one number to get the k-th numbers

- Extract bits $\{\theta_1^{(k)}, \ldots, \theta_N^{(k)}\}$ from a via
$$\psi(k) := \phi(2^{(k-1)N} a - \lfloor 2^{(k-1)N} a \rfloor)$$
of width $O(N)$ and depth $O(1)$.
- sum up to get $\theta^{(k)} \approx \sum_{\ell=1}^{N} \theta_\ell^{(k)} 2^{-\ell} = [2^{-1}, \ldots, 2^{-N}]\psi(k) := \gamma(k)$,
- $\gamma(k)$ is an NN of width $O(N)$ and depth $O(1)$.

Key Lemma

There exists an NN γ of width $O(N)$ and depth $O(1)$ that can memorize arbitrary samples $\{(k, \theta^{(k)}\}_{k=1}^{K}$ with a precision 2^{-N}.

Key ideas of our approximation

For $\boldsymbol{x} \in Q_{\boldsymbol{\beta}}$:
$\boldsymbol{x} \to \phi_1(\boldsymbol{x}) = \boldsymbol{\beta} \to \phi_2(\boldsymbol{\beta}) = k_{\boldsymbol{\beta}} \to \phi_3(k_{\boldsymbol{\beta}}) = f(\boldsymbol{x}_{\boldsymbol{\beta}}) \approx f(\boldsymbol{x})$

- Piecewise constant approximation: $f(\boldsymbol{x}) \approx f_p(\boldsymbol{x}) \approx \phi_3 \circ \phi_2 \circ \phi_1(\boldsymbol{x})$
- 2^N pieces per dim and 2^{Nd} pieces with accuracy 2^{-N}
- Floor NN $\phi_1(\boldsymbol{x})$ s.t. $\phi_1(\boldsymbol{x}) = \boldsymbol{\beta}$ for $\boldsymbol{x} \in Q_{\boldsymbol{\beta}}$ and $\boldsymbol{\beta} \in \mathbb{Z}^d$.
- Linear NN ϕ_2 mapping $\boldsymbol{\beta}$ to an integer $k_{\boldsymbol{\beta}} \in \{1, \ldots, 2^{Nd}\}$
- Key difficulty: NN ϕ_3 of width $O(N)$ and depth $O(1)$ fitting 2^{Nd} samples in 1D with accuracy $O(2^{-N})$
- Key Lemma: There exists an NN γ of width $O(N)$ and depth $O(1)$ that can memorize arbitrary samples $\{(k, \theta^{(k)})\}_{k=1}^{K}$ with a precision 2^{-N}.

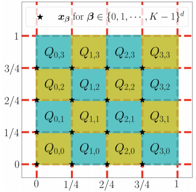

Figure: Uniform domain partitioning.

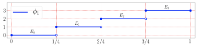

Figure: Floor function.

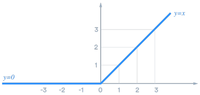

Figure: ReLU function.

Further interpretation of our result

Realistic consideration

- Constructive approximation requires f or exponentially many samples given
- Constructed parameters require high precision computation
- Floor and Sign are discontinuous functions leading to gradient vanishing

DNNs with advanced activation function

A continuous activation function without gradient vanishing

$$\sigma_1(x) = \left| x - 2\lfloor \tfrac{x+1}{2} \rfloor \right|,$$

$$\sigma_2(x) := \frac{x}{|x|+1},$$

$$\sigma(x) := \begin{cases} \sigma_1(x) & \text{for } x \in [0,\infty), \\ \sigma_2(x) & \text{for } x \in (-\infty, 0). \end{cases}$$

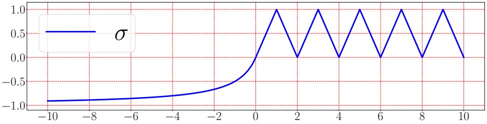

Figure: An illustration of σ on $[-10, 10]$.

DNNs with advanced activation function

Arbitrarily small error with a fixed number of neurons for $C([0,1]^d)$

- For any $\epsilon > 0$, there exists ϕ of width $36d(2d+1)$ and depth 11 s.t.
$$\|f(x) - \phi(x)\|_{L^\infty([0,1]^d)} \leq \epsilon$$
- Shen, Y., and Zhang (arXiv:2107.02397)

DNNs with advanced activation function

Exact representation with a fixed number of neurons for classification functions

- For any classification function $f(x)$ with K classes, there exists ϕ of width $36d(2d+1)$ and depth 12 s.t.

$$f(x) = \phi(x)$$

  on the supports of each class.
- Shen, Y., and Zhang (arXiv:2107.02397)

DNNs with advanced activation function

Two main ideas

- Kolmogorov-Arnold Superposition Theorem.

Theorem

$\forall f(\mathbf{x}) \in C([0,1]^d)$, there exist $\psi_p(x)$ and $\phi(x)$ in $C(\mathbb{R})$ and $b_{pq} \in \mathbb{R}$ s.t.

$$f(\mathbf{x}) = \sum_{q=1}^{2d+1} a_q \phi(\sum_{p=1}^{d} b_{pq}\psi_p(x_p)).$$

- NNs with width 36 and depth 5 is dense in $C([0,1])$ (Shen, Y., and Zhang (arXiv:2107.02397).

Acknowledgment

Collaborators

Qiang Du, Sean Hon, Jianfeng Lu, Hadrien Montanelli, Zuowei Shen, Shijun Zhang

Funding

National Science Foundation under the grant award 1945029