Approximations by Deep Neural Networks with Rectified Power Units

Li Lin

Peking University

September 10, 2019

- x^2 can be approximated within any error $\varepsilon > 0$ by a ReLU network having the depth, the number of weights and computation units all of order $\mathcal{O}(\log \frac{1}{\varepsilon})$.
- Rectified power units (RePUs) are defined as $(s \in \mathbb{N})$

$$\sigma_s(x) = \begin{cases} x^s, & x \geq 0, \\ 0, & x < 0. \end{cases} \tag{1}$$

- σ_2:ReQU.
- σ_3:ReCU.

Some notations

Denote a neural network Φ with input of dimension d, number of layer L, by a matrix-vector sequence

$$\Phi = ((A_1, b_1), ..., (A_L, b_L)), \tag{2}$$

where $N_0 = d, N_1, ..., N_L \in \mathbb{N}$, A_k are $N_k \times N_{k-1}$ matrices, and $b_k \in \mathbb{R}^{N_k}$. If Φ is a neural network, and $\rho : \mathbb{R} \to \mathbb{R}$ is an arbitrary activation function, then define

$$R_\rho(\Phi) : \mathbb{R}^d \to \mathbb{R}^{N_L}, \quad R_\rho(\Phi)(\boldsymbol{x}) = \boldsymbol{x}_L, \tag{3}$$

where $R_\rho(\Phi)(\boldsymbol{x})$ is defined as

$$\begin{cases} \boldsymbol{x}_0 := \boldsymbol{x}, \\ \boldsymbol{x}_k := \rho(A_k \boldsymbol{x}_{k-1} + b_k), \quad k = 1, 2, ..., L-1, \\ \boldsymbol{x}_L := A_L \boldsymbol{x}_{L-1} + b_L. \end{cases} \tag{4}$$

- number of hidden layers: $L-1$
- number of nodes:$\sum_{k=1}^{L-1} N_k$
- number of nonzero wights: $\sum_{k=1}^{L}(|A_k|_0 + |b_k|_0)$

- The function x, x^2 and xy can be exactly represented with no approximation error using networks having just a few nodes and nonzero weights.

Lemma

For $\forall x, y \in \mathbb{R}$ the following identities hold:

$$x^2 = \beta_2^T \sigma_2(\omega_2 x), \tag{5}$$

$$x = \beta_1^T \sigma_2(\omega_1 x + \gamma_1), \tag{6}$$

$$xy = \beta_1^T \sigma_2(\omega_1 x + \gamma_1 y), \tag{7}$$

where

$$\beta_1 = \frac{1}{4}[1, 1, -1, -1]^T, \beta_2 = [1, 1]^T, \tag{8}$$

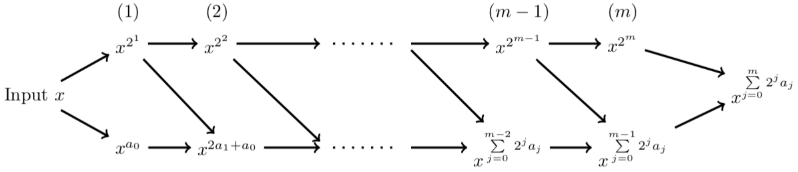
$$\omega_1 = [1, -1, 1, -1]^T, \omega_2 = [1, -1]^T, \gamma_1 = [1, -1, -1, 1]^T. \tag{9}$$

- The Realizations are not unique!

Optimal realizations of polynomials by deep ReQU netorks

Theorem

- *The monomial* $x^n, n \in \mathbb{N}$ *defined on* $\mathbb{R}$ *can be represented exactly by a* σ_2 *network. The number of network layers, number of nodes and number of weights required to realize* x^n *are at most* $\lfloor \log_2 n \rfloor + 2, 5\lfloor \log_2 n \rfloor + 5$ *and* $25\lfloor \log_2 n \rfloor + 14$*, respectively. Here* $\lfloor x \rfloor$ *represents the largest integer not exceeding* x *for* $x \in \mathbb{R}$*.*
- *For any* $n > 2$*,* x^n *can not be represented exactly by any ReQU network with only one hidden layer.*



Univariate polynomials

Theorem

If $f(x)$ is a polynomial of degree n on $\mathbb{R}$, then it can be represented exactly by a σ_2 neural network with $\lfloor \log_2 n \rfloor + 1$ hidden layers, and number of nodes and nonzero weights are both of order $\mathcal{O}(n)$. To be more precise, the number of nodes is bounded by $9n$, and number of nonzero weights is bounded by $61n$.

$$
\begin{aligned}
f(x) &= a_{15}x^{15} + a_{14}x^{14} + \cdots + a_8x^8 + a_7x^7 + a_6x^6 + \cdots + a_1x + a_0 \\
&= \underbrace{x^8}_{\xi_{3,0}} \underbrace{\left\{ \underbrace{x^4}_{\xi_{2,0}} \underbrace{\left[\underbrace{x^2}_{\xi_{1,0}} \underbrace{(a_{15}x + a_{14})}_{\xi_{1,8}} + \underbrace{(a_{13}x + a_{12})}_{\xi_{1,7}} \right]}_{\xi_{2,4}} + \underbrace{\left[x^2 \underbrace{(a_{11}x + a_{10})}_{\xi_{1,6}} + \underbrace{(a_9x + a_8)}_{\xi_{1,5}} \right]}_{\xi_{2,3}} \right\}}_{\xi_{3,2}} \\
&\quad + \underbrace{\left\{ x^4 \underbrace{\left[x^2 \underbrace{(a_7x + a_6)}_{\xi_{1,4}} + \underbrace{(a_5x + a_4)}_{\xi_{1,3}} \right]}_{\xi_{2,2}} + \underbrace{\left[x^2 \underbrace{(a_3x + a_2)}_{\xi_{1,2}} + \underbrace{(a_1x + a_0)}_{\xi_{1,1}} \right]}_{\xi_{2,1}} \right\}}_{\xi_{3,1}}.
\end{aligned}
$$

Some remarks

- Use a σ_2 network of scale $\mathcal{O}(\log_2 n)$ to represent x^n exactly.
- Any polynomial of degree less than n can be represented exactly by a σ_2 neural network with $\lfloor \log_2 n \rfloor + 1$ hidden layers, and no more than $\mathcal{O}(n)$ nonzero weights.

Some notations

Define Jacobi-weighted Sobolev space $B^m_{\alpha,\beta}(I)$ as

$$B^m_{\alpha,\beta}(I) := \{u : \partial^k_x u \in L^2_{\omega^{\alpha+k,\beta+k}}(I), 0 \leq k \leq m\}, m \in \mathbb{N}, \tag{10}$$

with norm

$$||f||_{B^m_{\alpha,\beta}} := (\sum_{k=0}^{m} ||\partial^k_x u||^p_{L^2_{\omega^{\alpha+k,\beta+k}}})^{\frac{1}{2}}. \tag{11}$$

The weight

$$\omega^{\alpha,\beta} = (1-x)^\alpha (1+x)^\beta, \quad \alpha, \beta > -1. \tag{12}$$

Define the $L^2_{\omega^{\alpha,\beta}}$-orthogonal projection $\pi^{\alpha,\beta}_N : L^2_{\omega_{\alpha,\beta}}(I) \to P_N$ as

$$(\pi^{\alpha,\beta}_N u - u, v)_{\omega^{\alpha,\beta}} = 0, \quad \forall v \in P_N. \tag{13}$$

Error bounds of approximating smooth functions

Theorem

Let $\alpha, \beta > -1$. For any $u \in B^m_{\alpha,\beta}(I)$, there exist a ReQU network Φ^u_N with $\lfloor \log_2 N \rfloor + 1$ hidden layers, $\mathcal{O}(N)$ nodes, and $\mathcal{O}(N)$ nonzero weights, satisfying the following estimate

- *if $0 \le l \le m \le N + 1$, we have*

$$||\partial^l_x (R_{\sigma_2}(\Phi^u_N) - u)||_{\omega^{\alpha+l,\beta+l}} \le c\sqrt{\frac{(N-m+1)!}{(N-l+1)!}}(N+m)^{\frac{(l-m)}{2}}||\partial^m_x u||_{\omega^{\alpha+m}} \tag{14}$$

- *if $m > N + 1$, we have*

$$||\partial^l_x (R_{\sigma_2}(\Phi^u_N) - u)||_{\omega^{\alpha+l,\beta_l}} \le c(2\pi N)^{-\frac{1}{4}}(\frac{\sqrt{\frac{e}{2}}}{N})^{N-l+1}||\partial^{N+1}_x u||_{\omega^{\alpha+N+1,\beta+N}} \tag{15}$$

where $c \approx 1$ for $N \gg 1$.

Theorem

For any given function $f(x) \in B^m_{\alpha,\beta}(I)$ *with norm less than* 1*, where* m *is either a fixed positive integer or infinity, there exists a ReQU network* Φ^f_ε *with number of layers* L*, number of nonzero weights* N *satisfying*

- *if* m *is a fixed positive integer, then* $L = \mathcal{O}(\frac{1}{m}\log_2(\frac{1}{\varepsilon}))$*, and* $N = \mathcal{O}(\varepsilon^{-\frac{1}{m}})$*;*
- *if* $m = \infty$*, i.e.* f *is analytic, then* $L = \mathcal{O}(\log_2(\log\frac{1}{\varepsilon}))$*, and* $N = \mathcal{O}(\frac{1}{\gamma}\log(\frac{1}{\varepsilon})), \gamma \approx \mathcal{O}(\log(\log\frac{1}{\varepsilon}))$*,*

can approximate f *within an error tolerance* ε*, i.e.*

$$||R_{\sigma_2}(\Phi_\varepsilon)^f - f||_{\omega^{\alpha,\beta}(I)} \leq \varepsilon. \tag{16}$$

- For a foxed m, or $N \gg m$,

$$||R_{\sigma_2}(\Psi_N^u) - u||_{\omega^{\alpha,\beta}(I)} \leq cN^{-m}||\partial_x^m u||_{\omega_{\alpha+m,\beta+m}}. \tag{17}$$

- For analytic function, by taking $m = \infty$

$$||R_{\sigma_2}(\Phi_N^u) - u||_{\omega^{\alpha,\beta}(I)} \leq c(2\pi N)^{-\frac{1}{4}}(\frac{\sqrt{e/2}}{N})^{N+1}||u||_{B_{\alpha,\beta}^{\infty}} \tag{18}$$

$$\leq c'e^{-\gamma N}||u||_{B_{\alpha,\beta}^{\infty}}$$

Approximation by using general rectified power units

Theorem

Regarding the problem of using $\sigma_s(x)(2 \leq s \in \mathbb{N})$ neural networks to exactly represent monomial $x^n, n \in \mathbb{N}$, we have the following results:

- *If $s = n$, the monomial x^n can be realized exactly using a σ_s networks having only 1 hidden layer with two nodes.*
- *If $1 \leq n < s$, the monomial x^n can be realized exactly using a σ_s networks having only 1 hidden layer with no more than $2s$ nodes.*
- *If $n > s \geq 2$, the monomial x^n can be realized exactly using a σ_s networks having only $\lfloor \log_s n \rfloor + 2$ hidden layer with no more than $(6s + 2)(\lfloor \log_s n \rfloor + 2)$ nodes, no more than $\mathcal{O}(25s^2 \lfloor \log_s n \rfloor)$ nonzero weights.*

$$
\begin{tikzcd}[column sep=small]
 & (1) & (2) & (3) & & (m-1) & (m) & (m+1) & \\
 & x^{s^1} \arrow[r] \arrow[dr] & x^{s^2} \arrow[r] \arrow[dr] & x^{s^3} \arrow[r] \arrow[dr] & \cdots & x^{s^{m-1}} \arrow[r] \arrow[dr] & x^{s^m} \arrow[r] & x^{s^m} \arrow[dr] & \\
\text{Input: } x \arrow[ur] \arrow[dr] & & (x^s)^{a_1} \arrow[dr] & (x^{s^2})^{a_2} \arrow[dr] & \cdots & (x^{s^{m-2}})^{a_{m-2}} \arrow[dr] & (x^{s^{m-1}})^{a_{m-1}} \arrow[dr] & & x^{\sum\limits_{j=0}^{m} s^j a_j} \\
 & x^{a_0} \arrow[r] & x^{a_0} \arrow[r] & x^{sa_1+a_0} \arrow[r] & \cdots & x^{\sum\limits_{j=0}^{m-3} s^j a_j} \arrow[r] & x^{\sum\limits_{j=0}^{m-2} s^j a_j} \arrow[r] & x^{\sum\limits_{j=0}^{m-1} s^j a_j} \arrow[ur] &
\end{tikzcd}
$$

Approximation of multivariate smooth functions

Theorem

If $f(x)$ is a multivariate polynomial with total degree n on $\mathbb{R}^d$, then there exists a σ_2 neural network having $d\lfloor \log_2 n \rfloor + d$ hidden layers with no more than $\mathcal{O}(C_d^{n+d})$ activation functions and nonzero weights, can represent f with no error.

$$f(x,y) = \sum_{i=0}^{n} \left(\sum_{j=0}^{n-i} a_{ij} y^j\right) x^i =: \sum_{i=0}^{n} a_i^y x^i, \ where \ a_i^y = \sum_{j=0}^{n-i} a_{ij} y^j \tag{19}$$

Theorem

For a polynomials f_N in a tensor product space $Q_N^d(I_1 \times \cdots I_d) := P_N(I_1) \otimes \cdots \otimes P_N(I_d)$, there exists a σ_2 network having $d\lfloor \log_2 N \rfloor + d$ hidden layers with no more than $\mathcal{O}(N^d)$ activation functions and nonzero weights, can represent f_N with no error.

Error bounds

For $u \in B^m_{\boldsymbol{\alpha},\boldsymbol{\beta}}(I^d)$, we have the following error estimate

$$||\pi_N^{\boldsymbol{\alpha},\boldsymbol{\beta}} u - u||_{L^2_{\omega^{\boldsymbol{\alpha},\boldsymbol{\beta}}}(I^d)} \leq cN^{-m}|u|_{B^m_{\boldsymbol{\alpha},\boldsymbol{\beta}}}, 1 \leq m \leq N. \tag{20}$$

Theorem

For any $u \in B^m_{\boldsymbol{\alpha},\boldsymbol{\beta}}(I^d)$, with $|u|_{B^m_{\boldsymbol{\alpha},\boldsymbol{\beta}}(I^d)} \leq 1$, there exists a σ_2 neural network Φ^u_ε having $\mathcal{O}(\frac{d}{m}\log_2\frac{1}{\varepsilon} + d)$ layers with no more than $\mathcal{O}(\varepsilon^{-\frac{d}{m}})$ nodes and nonzero weights, approximate u with $L^2_{\omega^{\boldsymbol{\alpha}},\boldsymbol{\beta}}(I^d)$ -error less than ε, i.e.

$$||R_{\sigma_2}(\Phi^u_\varepsilon) - u||_{L^2_{\omega^{\boldsymbol{\alpha}},\boldsymbol{\beta}}(I^d)} \leq \varepsilon. \tag{21}$$

Thanks!