Analysis of Neural Networks (3)
— Neural Network and Numerical PDEs

Jinchao Xu 许进超

Penn State University

xu@math.psu.edu   http://www.math.psu.edu/xu/

CUHK, October 22, 2021

NSF: DMS-1819157 and DMS-2111387

Analysis of Neural Network

Jinchao Xu

This lecture series will provide some mathematical understanding of neural networks and machine learning by studying their close relationship with classic numerical methods such as finite element and multigrid methods. Applications will be given to image classification and numerical partial differential equations

1. Finite Element Connection and Approximation Theory
   - ReLU neural networks $=$ linear finite elements
   - Largest function class that a stable neural network can approximate
   - Optimal approximation rates for popular neural networks
2. Multigrid and Image Classification
   - Linear separable sets and logistic regression
   - A model for feature extractions
   - Image classification by multigrid method
3. Neural Network and Numerical PDEs
   - Error analysis of neural network for numerical PDEs
   - Numerical quadrature and Rademacher complexity analysis
   - Training algorithm that achieves the best asymptotic convergence rate

Finite element: Piecewise linear functions

- Uniform grid $\mathcal{T}_h$

$$0 = x_0 < x_1 < \cdots < x_{N+1} = 1, \quad x_j = \frac{j}{N+1} \ (j = 0 : N+1).$$

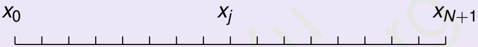
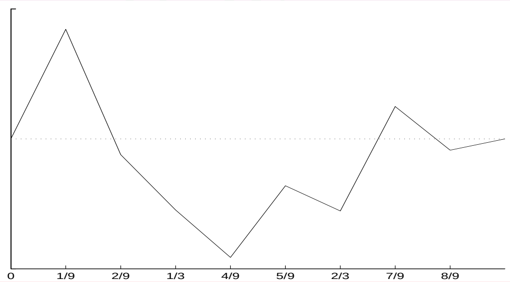


Figure: 1D uniform grid

- Linear finite element space

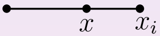
$$V_h = \{ v : \ v \text{ is continuous and piecewise linear w.r.t. } \mathcal{T}_h \}.$$

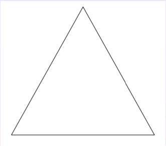


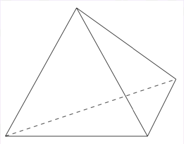
Finite element in multi-dimensions

$(k = 1)$

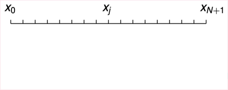
$w_1 x + b$ $\quad$ $w_1 x_1 + w_2 x_2 + b$ $\quad$ $w_1 x_1 + w_2 x_2 + w_3 x_3 + b$ $\quad \cdots$

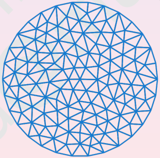


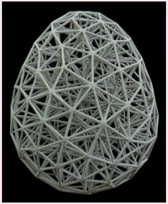




$\cdots$







$\cdots$

Lower bound for conforming elements

Theorem

Assume that V_h^k is a finite element of degree k on quasi-uniform mesh $\{\mathcal{T}_h\}$ of $\mathcal{O}(n)$ elements. Assume u is sufficiently smooth and not piecewise polynomials, then we have

$$c(u)n^{-\frac{k}{d}} \le \inf_{v_h \in V_h^k} \|u - v_h\|_{H^1(\Omega)} \le C(u)n^{-\frac{k}{d}}. \tag{1}$$

In general

$$\inf_{v_h \in V_h^k} \|u - v_h\|_{H_h^m(\Omega)} \eqsim n^{\frac{m-(k+1)}{d}}. \tag{2}$$

Ref: Q. Lin, H. Xie and J. Xu , Lower Bounds of the Discretization Error for Piecewise Polynomials, Math. Comp., 83, 1-13 (2014)

Model problem

(for any $d \geq 1$, $m \geq 1$)

Given $\Omega \subset \mathbb{R}^d$, consider a $2m$-th order elliptic problems

$$\sum_{|\alpha|=m} (-1)^m \partial^\alpha (a_\alpha(x)\, \partial^\alpha u) + u = f \quad \text{in } \Omega$$

Special cases:

$$-\Delta u = f \quad (m=1), \quad \Delta^2 u = f \quad (m=2).$$

Find $u \in V$ such that

$$J(u) = \min_{v \in V} J(v) \Longleftrightarrow a(u,v) = \langle f, v \rangle \quad \forall v \in V \tag{3}$$

where, for some $a_\alpha > 0$

$$a(u,v) := \sum_{|\alpha|=m} (a_\alpha \partial^\alpha u, \partial^\alpha v) + (u,v), \quad J(v) = \frac{1}{2} a(v,v) - (f,v)$$

and

$$V = \begin{cases} H^m(\Omega) & \textit{Neumann} \\ H_0^m(\Omega) & \textit{Dirichlet} \end{cases}$$

On the construction of conforming FEM

Question: For any $m, d \geq 1$, how to construct conforming finite element space

$$V_h \subset H^m(\Omega) \Longleftrightarrow V_h \subset C^{m-1}(\Omega)?$$

Answer: mostly open, especially when $m \geq 3, d \geq 3$ until recently (2021)

.

Theorem (Hu, Lin, & Wu 2021, ArXiv: 2103.14924))

For any $d \geq 1, m \geq 0$, a globally H^m finite element of degree $k \geq 2^d(m-1)+1$ can be constructed on any simplicial mesh with locally defined DOF.

- $k = 9$ and $\dim(P_9) = 220$ if $d = 3, m = 2$.

Challenges in classic finite element methods

1. High dimensional problems: linear element

$$c(u)n^{-\frac{1}{d}} \leq \|u - u_h\|_{H^1(\Omega)} \leq C(u)n^{-\frac{1}{d}}. \tag{4}$$

   - curse of dimensionality
2. High order PDEs
   - Conforming elements: very high order polynomials

σ-DNN: Linears, activation and composition

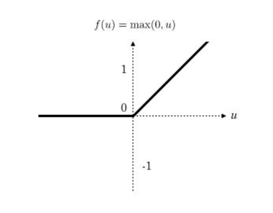
1. Start from a linear function
$$W^0x + b^0$$
2. Compose with the activation function:
$$x^{(1)} = \sigma(W^0x + b^0)$$
3. Compose with another linear function:
$$W^1x^{(1)} + b^1$$
4. Compose with the activation function:
$$x^{(2)} = \sigma(W^1x^{(1)} + b^1)$$
5. Compose with another linear function
$$f(x;\Theta) = W^2x^{(2)} + b^2$$
6. ...

Deep neural network functions with ℓ-hidden layers

$$\Sigma^{\sigma}_{n_{1:\ell}} = \{W^{\ell}x^{(\ell)} + b^{\ell},\ W^i \in \mathbb{R}^{n_i},\ b_i \in \mathbb{R}\}$$

ReLU-DNN and FEM

$$\Sigma^{k}_{n_{1:\ell}} = \Sigma^{ReLU^k}_{n_{1:\ell}}. \tag{5}$$



$$\textit{ReLU-DNN} = \Sigma^{1}_{n_{1:\ell}} = \textit{Linear FEM} \subset H^1(\Omega)$$
$$\textit{ReLU-DNN}^k = \Sigma^{k}_{n_{1:\ell}} = \textit{piecewise polynomials} \subset H^k(\Omega)$$

- Conforming piecewise polynomials of low order are trivial to construct using neural networks!

Stable Neural Network

- Consider approximation from the class

$$\Sigma_{n,M}^{\sigma} := \left\{ \sum_{i=1}^{n} a_i \sigma(\omega_i \cdot x + b_i),\ \omega_i \in \mathbb{R}^d,\ b_i \in \mathbb{R}, \sum_{i=1}^{n} |a_i| \le M \right\} \tag{6}$$

  of neural networks with ℓ^1-bounded outer coefficients.
- More generally for a dictionary $\mathbb{D} \subset H = L^2(\Omega)$, consider

$$\Sigma_{n,M}(\mathbb{D}) = \left\{ \sum_{i=1}^{n} a_i h_i,\ h_i \in \mathbb{D},\ \sum_{i=1}^{n} |a_i| \le M \right\} \tag{7}$$

- Let $M < \infty$ be fixed and consider approximation as $n \to \infty$.

An abstract approach: variation space

- Let $\mathbb{D} \subset X$ be collection of functions (called a dictionary)
  - Let $\mathbb{D}$ be symmetric, i.e. $f \in \mathbb{D} \to -f \in \mathbb{D}$
- Given $r > 0$, define

$$B_r(\mathbb{D}) := \overline{\left\{ \sum_{j=1}^{n} a_j h_j : \; n \in \mathbb{N}, \; h_j \in \mathbb{D}, \; \sum_{i=1}^{n} |a_i| \leq r \right\}}^{X} \tag{8}$$

We note that $B_r(\mathbb{D}) = rB_1(\mathbb{D})$ and

- $B_1(\mathbb{D})$ is the closed convex hull of $\mathbb{D}$

Variation spaces

- Define
$$\|f\|_{\mathcal{K}_1(\mathbb{D})} = \inf\{r > 0 : f \in B_r(\mathbb{D})\} \tag{9}$$
Namely,
$$f \approx \sum_{j=1}^{n} a_j h_j, \quad \sum_{i=1}^{n} |a_i| \leq r = \|f\|_{\mathcal{K}_1(\mathbb{D})} \tag{10}$$

- If $\mathbb{D}$ is bounded, the associated space, known as *variation space*
$$\mathcal{K}_1(\mathbb{D}) := \{f \in L^2(\Omega) : \ \|f\|_{\mathcal{K}_1(\mathbb{D})} < \infty\}$$
is a Banach space[1]

[1] **siegel2021some**.

Stable Dictionary Approximation Space

Theorem (Siegel & Xu 2021)

A function $f \in H = L^2(\Omega)$ can be approximated at all, i.e.

$$\lim_{n\to\infty} \inf_{f_n \in \Sigma_{n,M}(\mathbb{D})} \|f - f_n\|_H = 0, \tag{11}$$

for a fixed $M < \infty$ if and only if

$$f \in B_M(\mathbb{D}) \subset \mathcal{K}_1(\mathbb{D}).$$

Furthermore, if

$$\|\mathbb{D}\| \equiv \sup_{h\in\mathbb{D}} \|h\|_H < \infty$$

we have

$$\inf_{f_n \in \Sigma_{n,M}(\mathbb{D})} \|f - f_n\|_H \le n^{-\frac{1}{2}} \|\mathbb{D}\| \|f\|_{\mathcal{K}_1(\mathbb{D})}. \tag{12}$$

New Optimal Bounds[3]

Theorem

For $\mathbb{D} = \mathbb{P}_k^d$ for $k \geq 1$, we have

$$n^{-\frac{1}{2}-\frac{2k+1}{2d}-\varepsilon} \lesssim \sup_{f \in B_1(\mathbb{D})} \inf_{f_n \in \Sigma_{n,M}^k} \|f - f_n\|_{L^2(\Omega)} \lesssim n^{-\frac{1}{2}-\frac{2k+1}{2d}} \tag{13}$$

In comparison: optimal bound for finite elements[2]:

$$c(u) n^{-\frac{k}{d}} \leq \inf_{v_h \in V_h^k} \|u - v_h\|_{L^2(\Omega)} \leq C(u) n^{-\frac{k}{d}} = \mathcal{O}(h^k). \tag{14}$$

Earlier but nonoptimal results:

- Andrew R Barron(1993), Yuly Makovoz(1996), Jason M Klusowski & Andrew R Barron(2018), Weinan E & Chao Ma & Lei Wu(2019), Jinchao Xu(2021), Jonathan W. Siegel & Jinchao Xu(2021)

[2] **lin2014lower**.

[3] **siegel2021optimal**.

Removing the constraint that $\sum_{i=1}^n |a_i| \leq M$

Define

$$\Sigma_n^k := \left\{ \sum_{i=1}^n a_i \sigma_k(\omega_i \cdot x + b_i),\ \omega_i \in \mathbb{R}^d,\ b_i \in \mathbb{R}, a_i \in \mathbb{R} \right\}. \tag{15}$$

Then Σ_n^k has the following approximation property[4]

Theorem (Siegel and Xu)

$$\inf_{f_n \in \Sigma_n^k} \|f - f_n\| \lesssim \begin{cases} n^{-\frac{1}{2}} & \|f\|_{B^s(\Omega)} & \text{if} \quad s = \frac{1}{2} \\ n^{-(k+1)} \log n & \|f\|_{B^s(\Omega)} & \text{for some} \quad s > 1 \end{cases} \tag{16}$$

where

$$\|f\|_{B^s(\Omega))} = \inf_{f_e|_\Omega = f} \int_{\mathbb{R}^d} (1 + |\xi|)^s |\hat{f}_e(\xi)| d\xi. \tag{17}$$

- Improves result of Barron[5] by relaxing condition on f
- Shows that very high order approximation rates can be attained with sufficient smoothness
- Comparison with FEM:

$$\inf_{w \in \Sigma_n^k} \|u - w\| \approx \left\{ \inf_{v \in V_n^k} \|u - v\| \right\}^{d}.$$

[4] **siegel2020high**.

[5] **barron1993universal**.

Sparse-Grid Space versus Barron Space

($k = 1$, ReLU)

Basic estimates

- NN function space approximation

$$\|u - u_n\|_{0,\Omega} \lesssim n^{-1/2}\|u\|_{B^{\frac{1}{2}}(\Omega)}, \quad \|u - u_n\|_{1,\Omega} \lesssim n^{-1/2}\|u\|_{B^{\frac{3}{2}}(\Omega)}. \tag{18}$$

- Sparse-grid approximations
  1. Original (Zenger 1991):

$$\|u - u_n\|_{1,\Omega} \lesssim n^{-1}(\log n)^{d-1}|u|_{SG(\Omega)}. \tag{19}$$

where

$$|u|_{SG(\Omega)} = \|\partial^2_{x_1} \cdots \partial^2_{x_d} u\|_{0,\Omega}. \tag{20}$$

  2. More refined one (Bungartz and Griebel 2004):

$$\|u - u_n\|_{1,\Omega} \lesssim n^{-1}\|\partial^2_{x_1} \cdots \partial^2_{x_d} u\|_{\infty,\Omega}. \tag{21}$$

Imbedding relationship:

$$SG(\Omega) \subset B^{3/2-\epsilon}(\Omega) \subset B^{1/2}(\Omega) \tag{22}$$

Conclusion:

Shallow neural network function class seems to be potentially "better than" sparse-grid function class for approximation without "curse of dimensionality".

Model problem

(for any $d \geq 1$, $m \geq 1$)

Given $\Omega \subset \mathbb{R}^d$, consider a $2m$-th order elliptic problems

$$\sum_{|\alpha|=m} (-1)^m \partial^\alpha (a_\alpha(x)\, \partial^\alpha u) + u = f \quad \text{in } \Omega.$$

Special cases:

$$-\Delta u = f \quad (m = 1), \quad \Delta^2 u = f \quad (m = 2).$$

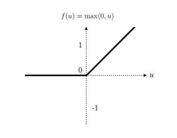
Conforming elements by neural network: $V_n^k \subset H^m(\Omega)$

Definition:

$$V_n^k = \left\{ \sum_{i=1}^{n} a_i (w_i x + b_i)_+^k, w_i \in \mathbb{R}^{1\times d}, a_i, b_i \in \mathbb{R}^1 \right\}$$

where

$$x_+ = \max(0, x) = \mathrm{ReLU}(x)$$

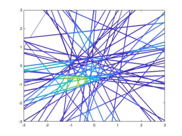


Properties:

1. Conforming for any $m, d \geq 1$ if $k \geq m$:

$$V_n^k \subset H^k(\Omega) \subset H^m(\Omega)$$

2. Piecewise polynomials of degree k in the following grids



Application to high order PDE in any dimension

Consider

$$\begin{cases} Lu = f & \text{in } \Omega, \\ B_N^k(u) = 0, & \text{on } \partial\Omega, \quad 0 \le k \le m-1. \end{cases} \tag{23}$$

$\Longleftrightarrow$ Find $u \in V = H^m(\Omega)$ such that

$$J(u) = \min_{v \in V} J(v) \tag{24}$$

where

$$J(v) = \frac{1}{2} \int_\Omega \sum_{|\alpha|=m} a_\alpha |\partial^\alpha v|^2 + v^2 dx - (f, v). \tag{25}$$

NN-FEM(FNM):[6] Find $u_n \in V_n^k$ as follows:

$$J(u_n) = \min_{v \in V_n^k} J(v). \tag{26}$$

Theorem:

$$\|u - u_n\|_a = \inf_{v_n \in V_n^k} \|u - v_n\|_a = \mathcal{O}(n^{m-(k+1)} \log n). \tag{27}$$

[6] **CiCP-28-1707**.

Superconvergence (?) property

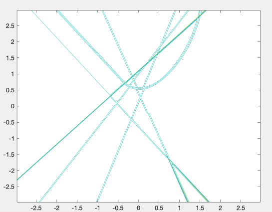
For $d = 2, m = 1$, consider

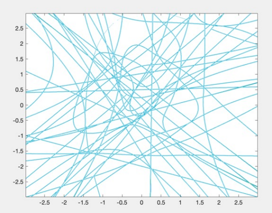
$$\Delta^2 u = f.$$

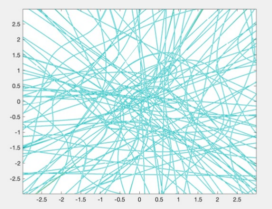
1. $k = 2$
   - Morley: $\|u - u_n\|_{2,h} = \mathcal{O}(h^{1}) = \mathcal{O}(n^{-\frac{1}{2}})$.
   - NN-FEM: $\|u - u_n\|_2 = \mathcal{O}(h^{2}) = \mathcal{O}(n^{-1})$.
2. $k = 5$
   - Argyris: $\|u - u_h\|_2 = \mathcal{O}(h^{4}) = \mathcal{O}(n^{-2})$.
   - NN-FEM: $\|u - u_n\|_2 = \mathcal{O}(h^{8}) = \mathcal{O}(n^{-4})$.

Properties of $[\mathrm{ReLU}]^k$-DNN_ℓ

1. Piecewise polynomials on "curved" elements
2. Best possible error estimate $\mathcal{O}(n^{m-(k+1)} \log n)$
3. If $k \geq 2$, we have spectral accuracy for smooth solution as ℓ increase[7].
4. Possible multi-scale adaptivity features (?):
   - local singularity.
   - global smoothness







[7] **li2019better**.

Issues:

- Discretization of the integral in $J(u)$, i.e. how do we evaluate

$$\int_\Omega |\nabla u(x)|^2 dx - \int_\Omega f(x)u(x)dx? \tag{28}$$

- Optimization of the discrete energy, i.e. how can we efficiently solve

$$\min J_N(u) \tag{29}$$

Convergence analysis when numerical quadratures are used? Existing works:

- M. Hutzenthaler, A. Jentzen, T. Kruse, T. A. Nguyen, and P. Wurstemberger, 2020;
- T. Luo and H. Yang, 2020;
- S. Mishra and T. K. Rusch, 2020; S. Mishra and R. Molinaro, 2020;
- J. Müller and M. Zeinhofer, 2020;
- Y. Shin, Z. Zhang, and G.E. Karniadakis, 2020;
- S. Lanthaler, S. Mishra, G.E. Karniadakis, 2021;
- J. Lu, Y. Lu and M. Wang, 2021;
- H. Son, J. Jang, W. Han, and H. J. Hwang, 2021;

Discretization of the Integral

There are two approaches for discretizing $J(u)$

- Sample points $x_1, ..., x_N$ uniformly at random from Ω and form

$$J_N(u) = \frac{1}{N}\sum_{i=1}^{N} |\nabla u(x_i)|^2 - f(x_i)u(x_i). \tag{30}$$

- Use a numerical quadrature rule such as Gaussian quadrature

$$J_N(u) = \sum_{i=1}^{N} a_i(|\nabla u(x_i)|^2 - f(x_i)u(x_i)). \tag{31}$$

Error analysis

Numerical quadrature: for any $g(x)$, $N = \frac{(k-1)d}{2}$

$$\left| \int_\Omega g(x)dx - |\Omega| \sum_{i=1}^{N} w_i g(x_i) \right| \lesssim N^{-\frac{r+1}{d}} \|g\|_{r,1}.$$

Challenges: how to bound

$$\|g\|_{r,1} \le ?, \quad \text{for} \quad g \in \Sigma_n^\sigma$$

OK if the following Bernstein or inverse inequality holds for $r > s$

$$\|v_n\|_r \lesssim n^\beta \|v_n\|_s, \quad \forall v_n \in \Sigma_n^k. \tag{32}$$

Many attempts have been made in existing literature

Bad news: Bernstein inequalty does not hold for NN

Given any $\epsilon > 0$, consider an NN function with 3 neurons:

$$u_3(x) = \mathrm{ReLU}(x - \frac{1}{2} + \epsilon) - 2\mathrm{ReLU}(x - \frac{1}{2}) + \mathrm{ReLU}(x - \frac{1}{2} - \epsilon), \quad \forall x \in (0, 1).$$

A direct calculation shows that

$$\int_0^1 |u_3'(x)|^2 dx = 2\epsilon \quad \text{and} \quad \int_0^1 |u_3(x)|^2 dx = \epsilon^2.$$

Therefore

$$|u_3|_{H^1} = \sqrt{\frac{2}{\epsilon}} \|u_3\|_{L^2}, \quad \forall \epsilon > 0$$

As a result, the following Bernstein inequality can not hold for any constant[8] $C(n)$

$$|v_n|_{H^1} \leq C(n) \|v_n\|_{L^2}, \quad \forall v_n \in \Sigma_n^\sigma$$

[8] **hong2021rademacher**.

Our approach

Development and analysis of stable neural network!

The use of $\mathcal{K}_1(\mathbb{D})$

- We consider the following variational form of Laplace's equation with Neumann boundary conditions

$$\min_{v \in H^1(\Omega)} J(v) := \int_\Omega |\nabla v(x)|^2 dx - \int_\Omega f(x)v(x)dx. \tag{33}$$

- We solve this problem by restricting

$$\min_{\|v\|_{\mathcal{K}_1(\mathbb{D})} \le M} J(v) := \int_\Omega |\nabla v(x)|^2 dx - \int_\Omega f(x)v(x)dx, \tag{34}$$

  for some M.

- With numerical quadrature

$$\min_{\|v\|_{\mathcal{K}_1(\mathbb{D})} \le M} J_N(v) \approx \int_\Omega |\nabla v(x)|^2 dx - \int_\Omega f(x)v(x)dx, \tag{35}$$

  for some M.

Uniform Bound on the Error

- When using numerical quadrature, we require the dictionary $\mathbb{D}$ to satisfy

$$|\mathbb{D}|_{W^{k,\infty}(\Omega)} := \sup_{d\in\mathbb{D}} \|d\|_{W^{k,\infty}(\Omega)} \leq C < \infty. \tag{36}$$

  This means that $\|u\|_{W^{k,\infty}(\Omega)} \leq C\|u\|_{\mathcal{K}_1(\mathbb{D})}$.

- So if we use r-th order quadrature, we will get[9]

$$|J_N(u) - J(u)| \lesssim N^{-\frac{r+1}{d}}, \tag{37}$$

  uniformly on $\{u : \|u\|_{\mathcal{K}_1(\mathbb{D})} \leq M\}$.

[9] **hong2021rademacher**.

Error estimates for numerical quadrature

Consider

$$\Sigma^k_{n,M} = \left\{ \sum_{i=1}^{n} a_i \sigma_k(\omega_i \cdot x + b_i) : \omega_i \in \mathbb{S}^{d-1},\ |b_i| \le 2,\ \sum_{i=1}^{n} |a_i| \le M \right\} \subset W^{k,\infty}(\Omega)$$

with

$$M = \|u\|_{\mathcal{K}_1(\mathbb{P}^d_k)}$$

Theorem

Let N be the number of quadrature points and

$$J_N(u_{n,N,M}) = \min_{v \in \Sigma^k_{n,M}} J_N(v). \tag{38}$$

If $N = \mathcal{O}(n^{\frac{d+1+2(k-m)}{k-1}})$, it holds that

$$\|u - u_{n,N,M}\|_a \lesssim n^{-\frac{1}{2} - \frac{2(k-m)+1}{2d}} \|u\|_{\mathcal{K}_1(\mathbb{P}^d_k)}.$$

Similarly for $u_{n,N,M} \in \Sigma^{\cos}_{n,M}$ with appropriate numerical quadrature, we have

$$\|u - u_{n,N,M}\|_a \lesssim n^{-s} \|u\|_{B^{s+m}(\Omega)}, \quad s > 0.$$

sketch of proof

1. For any conforming v,
$$|J_N(v) - J(v)| \le N^{-\frac{k-1}{d}} \|v\|_{k,2}. \tag{39}$$
2. Since Ω is bounded, $\omega_i \in \mathbb{S}^{d-1}$, $|b_i| \le 2$, $\sum_{i=1}^n |a_i| \le \|u\|_{\mathcal{K}_1(\mathbb{P}_k^d)}$,
$$\|u_{n,N,M}\|_{k,\infty} \lesssim \|u\|_{\mathcal{K}_1(\mathbb{P}_k^d)}.$$
3. For any n, there exists $v_{n,M} \in \Sigma_{n,M}^k$
$$\|u - v_{n,M}\|_a \lesssim n^{-\frac{1}{2}-\frac{2(k-m)+1}{2d}} \|u\|_{\mathcal{K}_1(\mathbb{P}_k^d)}. \tag{40}$$
4. Since $J_N(u_{n,N,M}) \le J_N(v_{n,M})$,
$$\begin{aligned}\frac{1}{2}\|u - u_{n,N,M}\|_a^2 =& J(u_{n,N,M}) - J(u)\\ \le& J(u_{n,N,M}) - J_N(u_{n,N,M}) + J_N(v_{n,M}) - J(v_{n,M}) + J(v_{n,M}) - J(u)\\ \lesssim& N^{-\frac{k-1}{d}} + \|v_{n,M} - u\|_a^2 \lesssim (N^{-\frac{k-1}{d}} + n^{-1-\frac{2(k-m)+1}{d}})\|u\|_{\mathcal{K}_1(\mathbb{P}_k^d)}^2.\end{aligned}$$
5. Choose $N = \mathcal{O}(n^{\frac{d+1+2(k-m)}{k-1}})$, then
$$\|u - u_{n,N,M}\|_a \lesssim n^{-\frac{1}{2}-\frac{2(k-m)+1}{2d}} \|u\|_{\mathcal{K}_1(\mathbb{P}_k^d)}.$$

Uniform Bound on the Error (cont.)

Lemma (Bartlett, 2002)

Let $\mathcal{F}$ be a set of functions. Then

$$\mathbb{E}_{x_1,\ldots,x_N\sim\mu} \sup_{h\in\mathcal{F}} \left| \frac{1}{N}\sum_{i=1}^{N} h(x_i) - \int_\Omega h(x)d\mu \right| \leq 2R_N(\mathcal{F}). \tag{41}$$

where $R_N(\mathcal{F})$ is the Rademacher complexity of function class $\mathcal{F}$ on Ω given by

$$R_N(\mathcal{F}) = \mathbb{E}_{x_1,\ldots,x_N}\mathbb{E}_{\xi_1,\ldots,\xi_N}\left(\sup_{f\in\mathcal{F}} \frac{1}{N}\sum_{i=1}^{N} \xi_i f(x_i)\right), \tag{42}$$

where x_i are drawn uniformly at random from Ω and ξ_i are uniformly random signs.

Uniform Bound on the Error (cont.)

The following results are obtained[10]

Lemma

$$R_N(\mathbb{D}), R_N(\nabla\mathbb{D}) \lesssim N^{-\frac{1}{2}}. \tag{43}$$

Theorem

Let

$$J_N(u_{n,M,N}) = \min_{v\in\Sigma_{n,M}} J_N(v). \tag{44}$$

Then

$$\mathbb{E}_{x_1,\ldots,x_N}\left(\sup_{\|u\|_{\mathcal{K}_1(\mathbb{D})}\leq M} |J_N(u) - J(u)|\right) \lesssim MN^{-\frac{1}{2}}, \tag{45}$$

and

$$\mathbb{E}_{x_1,\ldots,x_N}(\|u_{n,M,N} - u\|^2_{H^m(\Omega)}) \lesssim M\left(N^{-\frac{1}{2}} + Mn^{-1}\right). \tag{46}$$

[10] **hong2021rademacher**.

Challenge: SGD or Adam

For the ReLU^k shallow neural network, let

$$u_n = \arg \min_{v_n \in \Sigma_{n,M}(\mathbb{D})} \|v_n - u\| \tag{47}$$

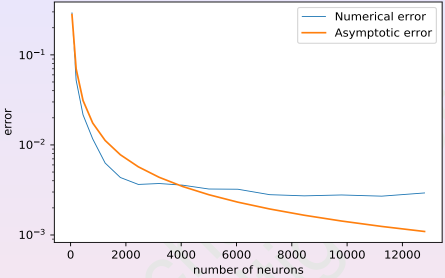
be the solution trained by SGD or Adam, then it is extremely difficult to observe

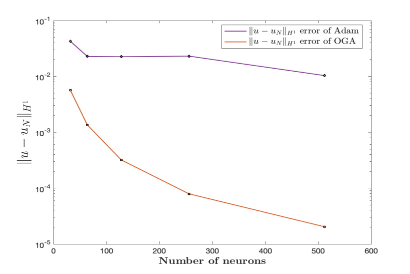
$$\|u - u_n\| \leq cn^{-\alpha} \tag{48}$$

for any $\alpha > 0$.
Question:

1. Can we numerically observe (48) for large n?

Non-convergence for Adam etc for large n





Our approach: Greedy Algorithms

- Orthogonal greedy algorithm[11]:

$$u_0 = 0, \; g_k = \arg\max_{g\in\mathbb{D}} \langle \nabla J_N(u_{k-1}), g\rangle, \; u_k = P_k u, \tag{49}$$

  where P_k: orthogonal projection onto span$\{g_1, ..., g_k\}$.

- Relaxed greedy algorithm

$$\begin{aligned} u_0 &= 0 \\ g_k &= \arg\max_{g\in\mathbb{D}} \langle \nabla J_N(u_{k-1}), g\rangle \\ u_k &= (1 - s_k)u_{k-1} - M s_k g. \end{aligned} \tag{50}$$

[11] **devore1996some**.

Convergence Rates of the Orthogonal Greedy Algorithm

Known convergence rates of the orthogonal greedy algorithm:

- Orthogonal greedy algorithm[12]: $O(n^{-\frac{1}{2}})$
- Similar convergence rates for the pure and relaxed greedy algorithms

Can any of these rates be improved for the dictionaries $\mathbb{P}_k^d$ or $\mathbb{F}_s^d$?

- Higher order approximation rates are possible!
- Can the orthogonal greedy algorithms attain them?

[12] **devore1996some**.

Convergence Rate of the Orthogonal Greedy Algorithm[13]

Theorem

Let the iterates f_n be given by the orthogonal greedy algorithm, where $f \in \mathcal{K}_1(\mathbb{P}_k^d)$. Then we have

$$\|u_n - u\| \lesssim n^{-\frac{1}{2}-\frac{2k+1}{2d}}. \tag{51}$$

- The orthogonal greedy algorithm can train optimal neural networks!

[13] **siegel2021improved**.

Optimization of the Discrete Energy: Greedy Algorithm

We solve the optimization problem

$$\min_{v \in \Sigma_{n,M}} J_N(v) \tag{52}$$

using the following relaxed greedy algorithm:

$$\begin{aligned} u_0 &= 0 \\ g_k &= \arg\max_{g \in \mathbb{D}} \langle \nabla J_N(u_{k-1}), g \rangle \\ u_k &= (1 - s_k)u_{k-1} - M s_k g. \end{aligned} \tag{53}$$

Theorem

$\|u_n\|_{\mathcal{K}_1(\mathbb{D})} \leq M$ *for all* k *and*

$$J_N(u_n) - \min_{v \in \Sigma_{n,M}} J_N(v) \lesssim \frac{1}{n}. \tag{54}$$

Main Theorem[14]

Theorem

Assume that the true solution $u \in \mathcal{K}_1(\mathbb{D})$ satisfies $\|u\|_{\mathcal{K}_1(\mathbb{D})} \leq M$ and let the numerical solution $u_{n,M,N} \in \Sigma_{n,M}(\mathbb{D})$ be obtained by the relaxed greedy algorithm for n steps. Then we have

$$\mathbb{E}_{x_1,\ldots,x_N}(J(u_{n,M,N}) - J(u)) \leq M\left[C_1(1 + \|f\|_{L^\infty(\Omega)})N^{-\frac{1}{2}} + C_2 M n^{-1}\right]. \tag{55}$$

and

$$\mathbb{E}_{x_1,\ldots,x_N}(\|u_{n,M,N} - u\|^2_{H^m(\Omega)}) \leq M\left[C_1' N^{-\frac{1}{2}} + C_2' M n^{-1}\right], \tag{56}$$

where C_1' and C_2' depend only upon the dictionary and the differential operator.

[14] **hong2021rademacher**.

Remarks on the relaxed greedy algorithm

Question: How to solve

$$g_k = \arg\max_{g \in \mathbb{D}} \langle \nabla J_N(u_{k-1}), g \rangle$$

- Feasible for small d ($d = 1, 2, 3$).
- A general approach proposed in book[15].
- Challenge if d is large.
- We do not know any other method.

[15] **gyorfi2002distribution**.

NN with OGA for Data fitting

Example (2D approximation)

Consider approximating the following 2D function

$$u(x,y) = \cos(2\pi x)\cos(2\pi y), \quad (x,y) \in (0,1)^2.$$

By fixing $\|\omega\| = 1$ and $b \in [-2,2]$, the convergence order of OGA is shown in Table below for ReLU^k neural networks. Theoretical order is shown in parenthesis.

n	$k = 1(n^{-1.25})$		$k = 2(n^{-1.75})$		$k = 3(n^{-2.25})$	
	$\|u - u_n\|_{L^2}$	order	$\|u - u_n\|_{L^2}$	order	$\|u - u_n\|_{L^2}$	order
2	4.969e-01	-	4.998e-01	-	4.976e-01	-
4	4.883e-01	0.025	4.992e-01	0.002	4.957e-01	0.006
8	2.423e-01	1.011	3.233e-01	0.627	4.193e-01	0.242
16	6.632e-02	1.869	4.911e-02	2.719	1.099e-01	1.932
32	2.206e-02	1.588	1.688e-02	1.541	8.075e-03	3.767
64	1.060e-02	1.058	4.156e-03	2.022	1.149e-03	2.813
128	4.284e-03	1.306	9.773e-04	2.088	2.185e-04	2.395
256	1.703e-03	1.331	2.622e-04	1.898	4.718e-05	2.211

Table: Convergence order of OGA with $ReLU^k$ activation function

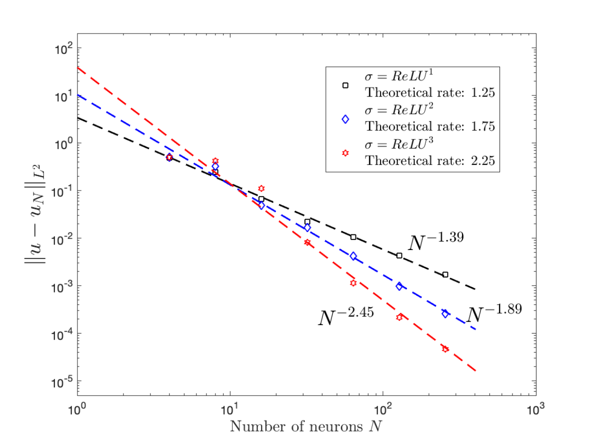


Figure: Convergence order of OGA with $ReLU^k$ activation function

NN with OGA for Numerical PDE

Example (1D elliptic equation)

Let us consider the 1D second order elliptic equation on $\Omega = (-1, 1)$:

$$-u'' + u = f, \text{ in } \Omega \tag{57}$$

$$\frac{\partial u}{\partial n} = 0, \text{ on } \partial\Omega. \tag{58}$$

with the source term $f = (1 + \pi^2) \cos(\pi x)$, then the analytical solution is $u(x) = \cos(\pi x)$. Let $\sigma = \mathrm{ReLU}^2$ and the convergence rates are predicted theoretically.

n	$\|u - u_n\|_{L^2}$	order	$\|u - u_n\|_{H^1}$	order
2	1.312179e+00	-	3.123769e+00	-
4	3.809296e-01	1.78	1.795590e+00	0.80
8	7.900097e-03	5.59	1.239320e-01	3.86
16	6.253874e-04	3.66	2.431156e-02	2.35
32	7.539756e-05	3.05	5.645258e-03	2.11
64	8.098691e-06	3.22	1.351523e-03	2.06
128	9.655067e-07	3.07	3.200813e-04	2.08
256	1.209074e-07	3.00	7.899931e-05	2.02

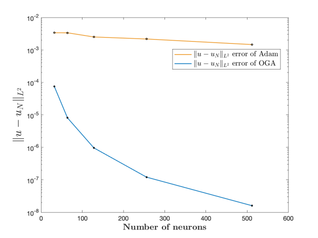
Table: L^2 and H^1 numerical error of the OGA solution

OGA v.s. SGD and Adam

We take 40000 samples and take the learning rates to be $1e-5$ and $1e-3$ for SGD and Adam, respectively. It is difficult to observe any convergence order from the solutions of SGD or Adam:

N	OGA		Adam		SGD	
	$\|u-u_N\|_{L2}$	$\|u-u_N\|_{H1}$	$\|u-u_N\|_{L2}$	$\|u-u_N\|_{H1}$	$\|u-u_N\|_{L2}$	$\|u-u_N\|_{H1}$
2	1.312e+00	3.124e+00	5.110e-01	7.913e-01	6.979e-01	1.959e+00
4	3.809e-01	1.796e+00	8.417e-03	1.378e-01	6.880e-01	1.940e+00
8	7.900e-03	1.239e-01	4.259e-03	5.364e-02	5.033e-01	1.136e+00
16	6.254e-04	2.431e-02	5.906e-03	9.033e-02	9.068e-02	4.124e-01
32	7.540e-05	5.645e-03	3.368e-03	4.244e-02	8.202e-02	3.969e-01
64	8.099e-06	1.352e-03	3.365e-03	2.279e-02	3.632e-02	2.273e-01
128	9.655e-07	3.201e-04	2.503e-03	2.250e-02	2.475e-02	2.427e-01
256	1.209e-07	7.900e-05	2.167e-03	2.307e-02	2.660e-02	2.417e-01
512	1.599e-08	2.033e-05	1.460e-03	1.034e-02	1.982e-02	2.306e-01

Table: Numerical results from different training methods.



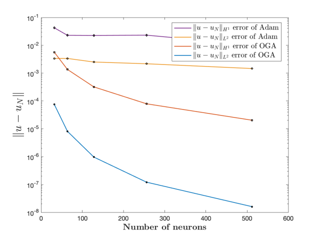


Figure: OGA vs Adam

NN with OGA for 4th order PDE

Example (1D 4th-order equation)

We consider the 4th-order equation $(-\Delta)^2 u + u = f$ on $(-1, 1)$ with

$$u(x) = (1 - x)^4 (1 + x)^4.$$

Take $\sigma = \mathrm{ReLU}^3$. The convergence orders with $\|\cdot\|_0$ and $\|\cdot\|_a$ errors are listed in the following table to confirm 4th order and 2nd order convergence, respectively.

n	$\|u - u_n\|_{L^2}$	order	$\|u - u_n\|_a$	order
2	8.762473e-01	-	6.062624e+00	-
4	9.891868e-01	-0.17	4.122220e+00	0.56
8	1.493237e-02	6.05	1.068900e+00	1.95
16	2.607811e-04	5.84	1.430653e-01	2.90
32	1.088935e-05	4.58	3.119824e-02	2.20
64	5.529557e-07	4.30	6.639040e-03	2.23
128	2.989261e-08	4.21	1.678663e-03	1.98
256	1.745507e-09	4.10	4.091674e-04	2.04

Table: The $\|\cdot\|_0$ and $\|\cdot\|_a$ errors of the OGA solution.

Numerical experiments of OGA

Example (2D 4th-order equation)

Next consider the $\|\cdot\|_0$ and $\|\cdot\|_a$ errors of OGA for 2D 4th-order equations on $\Omega = (-1, 1)^2$. Let the exact solution to be $u(x, y) = (x^2 - 1)^4(y^2 - 1)^4$ to satisfy the Neumann boundary conditions. We have

n	$\|u - u_n\|_{L^2}$	order	$\|u - u_n\|_a$	order
2	6.527642e-01	-	7.926637e+00	-
4	7.859126e-01	-0.27	7.592753e+00	0.06
8	9.906278e-01	-0.33	6.295085e+00	0.27
16	8.215047e-01	0.27	4.002859e+00	0.65
32	1.512860e-01	2.44	1.446132e+00	1.47
64	7.206241e-02	1.07	4.746744e-01	1.61
128	2.258788e-02	1.67	1.808527e-01	1.39
256	4.696294e-03	2.27	6.970084e-02	1.38

Table: The convergence order with $\|\cdot\|_0$ and $\|\cdot\|_a$ errors by OGA.

Nonlinear problem: 2D

Example (A nonlinear 2D example)

Consider the following nonlinear 2D equation $-\Delta u + u^3 + u = f$ on $(0,1)^2$ with $\partial u/\partial n = 0$. The analytical solution is $u = \cos(2\pi x)\cos(2\pi y)$ and the dictionary for RGA is taken as

$$\mathbb{D} = \{\sigma(w_1 x + w_2 y + b) | (w_1, w_2, b) \in [-20, 20]^3\}$$

where $\sigma(x)$ is the sigmoid function. The convergence is considered on the approximating space where $\|u\|_{\mathcal{K}_1(\mathbb{D})} \leq M = 15$.

n	$\|u - u_n\|_2$	order	$\|Du - Du_n\|_2$	order	$J(u_n) - J(u)$	order(n^{-1})
16	7.847118e-01	-	4.645084e+00	-	1.804723e+04	-
32	6.678914e-01	0.23	2.954645e+00	0.65	7.563223e+03	1.25
64	2.370456e-01	1.49	1.675239e+00	0.82	2.327894e+03	1.70
128	1.216064e-01	0.96	1.087479e+00	0.62	9.679782e+02	1.27
256	6.183769e-02	0.98	5.204851e-01	1.06	2.222200e+02	2.12
512	3.796748e-02	0.70	3.610805e-01	0.53	1.066532e+02	1.06
1024	2.687126e-02	0.50	2.110172e-01	0.77	3.661551e+01	1.54
2048	1.072196e-02	1.33	1.431628e-01	0.56	1.663444e+01	1.14

Table: Convergence order of RGA.

Nonlinear problem: 2D

Example (2D Poisson-Boltzmann equation)

Consider the PB equation $-\Delta u + \kappa \sinh(u) = f$ on the sphere $\{(x, y) | x^2 + y^2 \leq 4\}$ with $\partial u / \partial n = 0$. The energy functional for this problem is

$$\mathcal{J}(u) = \int_\Omega \left(\frac{1}{2} |\nabla u|^2 + \kappa \cosh(u) - fu \right) dx,$$

which is a strictly convex and coercive energy as long as $\kappa > 0$. We set $\kappa = 1$ and consider the radially symmetric solution

$$u(x, y) = \cos\left(\frac{\pi}{2} \sqrt{x^2 + y^2}\right).$$

n	$\|u - u_n\|_2$	order	$\|Du - Du_n\|_2$	order	$J(u_n) - J(u)$	order(n^{-1})
16	1.102900e+00	-	2.470451e+00	-	1.512249e+05	-
32	6.499420e-01	0.76	1.844984e+00	0.42	7.750018e+04	0.96
64	5.440200e-01	0.26	1.535850e+00	0.26	5.480476e+04	0.50
128	2.434633e-01	1.16	7.509427e-01	1.03	1.285071e+04	2.09
256	1.507433e-01	0.69	4.531679e-01	0.73	4.690009e+03	1.45
512	7.373288e-02	1.03	2.659362e-01	0.77	1.423244e+03	1.72
1024	2.948682e-02	1.32	1.646858e-01	0.69	5.362675e+02	1.41
2048	2.400795e-02	0.30	1.155901e-01	0.51	2.570714e+02	1.06

Table: Convergence order of RGA for the 2D Poisson-Boltzmann equation.

Concluding remarks

1. As piecewise linear function classes, “linear finite elements” = “ReLU-DNN”; but they have utterly different structures
2. For sufficiently “smooth” function, linear finite neuron method has no curse of dimensionality, where linear finite element method does
3. Finite neuron method has “superconvergence” properties
4. For high order PDEs, it is hard to construct conforming finite element method, but it is straightforward to construct conforming finite neuron method
5. Finite neuron methods for PDEs are very difficult/expensive to realize
   - SGD-Adam is hardly “convergent”!
   - Greedy algorithm can lead to asymptotic approximation
   - More efficient optimization algorithms need to be developed
6. Potentials of numerical PDE methods based on neural networks:
   - Theoretical interesting, practically challenging
   - Potential advantages for high-dimensional problem
   - Machine learning for numerical PDEs (with or without neural networks): a rich research field!

References

- J. Xu, The Finite Neuron Method and Convergence Analysis, Commun. Comput. Phys., 28, pp. 1707-1745, (2020).
- J. W. Siegel and J. Xu, "High-Order Approximation Rates for Neural Networks with ReLU^k Activation Functions." arXiv preprint arXiv:2012.07205 (2020).
- J. W. Siegel and J. Xu, "Approximation Rates for Neural Networks with General Activation Functions." Neural Networks (2020).
- J. W. Siegel and J. Xu, "Optimal Approximation Rates and Entropy Bounds for ReLU^k Networks." arXiv preprint arXiv:2101.12365 (2021)
- L. Ma, J. W. Siegel and J. Xu, "Uniform Approximation Rates for Neural Networks with ReLU^k Activation Functions." (2021)
- Q. Hong, J. W. Siegel, and J. Xu, "A Priori Error Analysis of the Finite Neuron Method with Deterministic and Monte Carlo Quadrature." (2021)

Thank you!