Machine Learning with Physics

Scaling Law, Optimization and Minimax Optimality

Yiping Lu

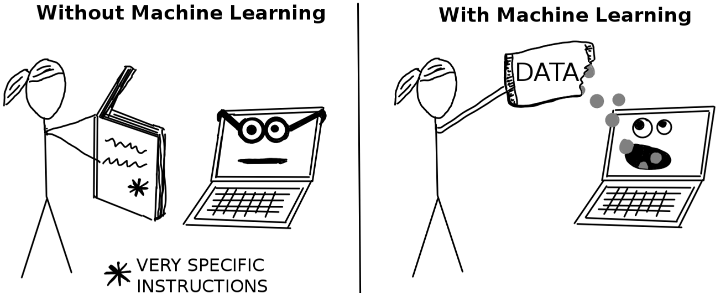
Institute for Computational and Mathematical Engineering
School Of Engineering
Stanford University

Joint work with Haoxuan Chen, Jianfeng Lu, Lexing Ying and Jose Blanchet.

Motivation 1

We can make Predictions from

- physics using PDEs/Structure Form
- data using Machine Learning



Motivation 2

Inverse Problem: What we can measure is not what we want to know! How to do machine learning?

- Stock price $\rightarrow$ drift
- Imaging: X-Ray, CT, Calderon problems
- Our work: "Inverse Game Theory": policy $\rightarrow$ utility (not included today)

How much data we need?

Questions Aim to Answer in This Talk

Satistical Limit. For a given PDE , how large the sample size are needed to reach a prescribed performance level?

Optimal Estimators. How complex the model are needed to reach the satistical limit?

Computational Power. How can we design an algorithm?

Questions Aim to Answer in This Talk

Satistical Limit. For a given PDE , how large the sample size are needed to reach a prescribed performance level?

Optimal Estimators. How complex the model are needed to reach the satistical limit?

Computational Power. How can we design an algorithm?

Questions Aim to Answer in This Talk

Satistical Limit. For a given PDE , how large the sample size are needed to reach a prescribed performance level?

Optimal Estimators. How complex the model are needed to reach the satistical limit?

Computational Power. How can we design an algorithm?

Answers by this Talk

Satistical Limit. Gradient value have more information

Optimal Estimators. PINN and **Modified** DRM are optimal

Computational Power. Sobolev Loss Accelerates Training

Answers by this Talk

Satistical Limit. Gradient value have more information

Optimal Estimators. PINN and **Modified** DRM are optimal

Computational Power. Sobolev Loss Accelerates Training

Answers by this Talk

Satistical Limit. Gradient value have more information

Optimal Estimators. PINN and **Modified** DRM are optimal

Computational Power. Sobolev Loss Accelerates Training

Insights for Empirical Users

- Deep Ritz Method **High** dimensional problem, **Smooth** problem
- PINN **Low** dimensional problem, **Non-smooth** problem

1. Problem Formulation

2. Lower Bound

3. Upper Bound
   - Empirical Risk Minimization
   - Gradient Descent

Problem Formulation

Problem Formulation

Static Schrödinger Equation

$$\begin{aligned} -\Delta u + Vu = f \quad & \text{in } \Omega, \\ u = 0 \quad & \text{on } \partial\Omega. \end{aligned} \tag{1}$$

What we observed:

- Random Samples in Domain: $\{x_i\}_{i=1}^n \sim \mathrm{Unif}(\Omega)$
- RHS Function Values: $\{f_i = f(x_i) + \eta_i\}_{i=1}^n$

What we want:

- An Esitmate of $\overline{u}$ in **Sobolev Norm**.

Lower Bound

General Lower Bound

Information Theoretical Lower Bound

Any Estimator H using $(X_i, f_i)_{i=1}^n$ can't do better than

$$\inf_{H} \sup_{u \in C^{\alpha}(\Omega)} \mathbb{E}\| H(\{X_i, f_i\}_{i=1,\cdots,n}) - u^* \|_{W_s^2} \gtrsim n^{-\frac{2\alpha-2s}{2\alpha-2t+d}},$$

For

- t-th order PDE
- Solution $u \in H^{\alpha}$
- Consider Convergence in H^s

Now:
PINN: H^2 norm
DRM: H^1 norm

Upper Bound

Problem Formulation

Strong form (residual minimization) → Physics Informed Neural Network/DGM

$$\mathcal{L}(u) := |(-\Delta + V)u - f|^2_{L^2(\Omega)}$$

Variational form → Deep Ritz Methods

$$u^* = \arg\min_{u \in H^1(\Omega)} \mathcal{E}(u) := \frac{1}{2}\int_\Omega \|\nabla u\|^2 + V\|u\|^2 u(x) - \int_\Omega fu(x)$$

Problem Formulation

Strong form (residual minimization) → Physics Informed Neural Network/DGM

$$\mathcal{L}(u) := |(-\Delta + V)u - f|^2_{L^2(\Omega)}$$

Variational form → Deep Ritz Methods

$$u^* = \arg\min_{u \in H^1(\Omega)} \mathcal{E}(u) := \frac{1}{2}\int_\Omega \|\nabla u\|^2 + V\|u\|^2 u(x) - \int_\Omega f u(x)$$

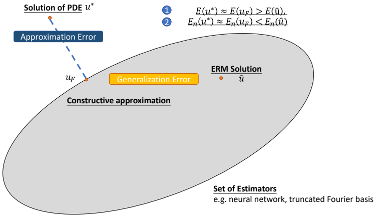
Further Question

Will different objective function gives different answers to **Statistical Efficiency**,**Optimization**?

Error Decomposition

If we

$$\mathbb{E}\left(\mathcal{E}(u_n)-\mathcal{E}(u^*)\right) \leqslant \underbrace{\mathbb{E}[\mathcal{E}(u_n)-\mathcal{E}_n(u_n)]}_{\Delta\mathcal{E}_{\textbf{gen}}} + \underbrace{\mathbb{E}[\mathcal{E}_n(u_{\mathcal{F}})]-\mathcal{E}(u_{\mathcal{F}})}_{\Delta\mathcal{E}_{\textbf{bias}}} + \underbrace{\mathcal{E}(u_{\mathcal{F}})-\mathcal{E}(u^*)}_{\Delta\mathcal{E}_{\textbf{approx}}}.$$



bias+variance decomposition:

$$\text{approximation} + \frac{\text{Complexity}}{\sqrt{n}}\ \text{bound}$$

But leads to sub-optimal results... [Shin et al 2020], [Lu et al 2021], [Duan et al 2021]

Motivating Example

Estimating the mean

Goal. Estimate $\theta = \mathbb{E}[X]$ via loss function $\frac{1}{2}(\theta - x)^2$

Empirical Solution of ℓ_2 loss: $\theta_n = \frac{1}{n}\sum_{i=1}^n x_i$, using chernoff bound we know $\theta_n - \theta = \sqrt{\frac{\sigma^2 \log \frac{1}{\delta}}{n}}$ w.h.p.

The generalization gap $L(\theta_n) - L(\theta^*) = \|\theta - \theta^*\|^2$ w.h.p

$$L(\theta_n) - L(\theta^*) = (\theta_n - \theta^*)^2 \leqslant C\frac{\sigma^2 \log \frac{1}{\delta}}{n}$$

A $O(\frac{1}{n})$ fast rate bound.

Motivating Example

Estimating the mean

Goal. Estimate $\theta = \mathbb{E}[X]$ via loss function $\frac{1}{2}(\theta - x)^2$

Empirical Solution of ℓ_2 loss: $\theta_n = \frac{1}{n}\sum_{i=1}^n x_i$, using chernoff bound we know $\theta_n - \theta = \sqrt{\frac{\sigma^2 \log \frac{1}{\delta}}{n}}$ w.h.p.

The generalization gap $L(\theta_n) - L(\theta^*) = \|\theta - \theta^*\|^2$ w.h.p

$$L(\theta_n) - L(\theta^*) = (\theta_n - \theta^*)^2 \leqslant C\frac{\sigma^2 \log \frac{1}{\delta}}{n}$$

A $O(\frac{1}{n})$ fast rate bound.

Observation 1: Fast rate via Localization

The variational form has some "strongly convex"

Lemma

Assume $0 < V_{\min} \leqslant V(x) \leqslant V_{\max}$ *for all* $x \in \Omega$

$$\frac{2}{\max(1, V_{\max})}(\mathcal{E}(u) - \mathcal{E}(u^*)) \leqslant \|u - u^*\|_{H^1(\Omega)}^2 \leqslant \frac{2}{\max(1, V_{\min})}(\mathcal{E}(u) - \mathcal{E}(u^*))$$

Can we have a $\frac{1}{n}$ fast rate generalization bound?

Local Rademacher Complexity

Local Rademacher Complexity

$$\psi(r) \geqslant \mathbb{E}R_n\{f \in \mathcal{F}, T(f) \leqslant r\}$$

The generalization bound: fix point solution of $\psi(r) = r$

$$\underbrace{\sqrt{\frac{r}{n}}}_{1/\sqrt{N}\text{ rate}} = r \to r = \frac{1}{n}$$

Key: increase speed according to r.

Local Rademacher Complexity

Local Rademacher Complexity

$$\psi(r) \geqslant \mathbb{E}R_n\{f \in \mathcal{F}, T(f) \leqslant r\}$$

The generalization bound: fix point solution of $\psi(r) = r$

$$\underbrace{\sqrt{\frac{r}{n}}}_{1/\sqrt{N}\text{ rate}} = r \rightarrow r = \frac{1}{n}$$

Key: increase speed according to r.

Is Fast Rate Optimal?

For PINN, Yes!. For DRM, No!

Upper Bounds			Lower Bound
Objective Function	Neural Network	Fourier Basis	
Deep Ritz	$n^{-\frac{2s-2}{d+2s-2}} \log n$	$n^{-\frac{2s-2}{d+2s-2}}$	$n^{-\frac{2s-2}{d+2s-4}}$
PINN	$n^{-\frac{2s-4}{d+2s-4}} \log n$	$n^{-\frac{2s-4}{d+2s-4}}$	$n^{-\frac{2s-4}{d+2s-4}}$

Table: Upper bounds and lower bounds Fast Rate achieved.

Why?

A Fourier Basis View

Solving a simple PDE $\Delta u = f$ using Fourier Basis.

Estimator 1

First Estimate f then solve u, $f_z = \frac{1}{n}\sum f(x_i)\phi_z(x_i)$, then $u = \sum \frac{1}{\|z\|^2} f_z \phi_z(x)$

Estimator 2

Plug $u = \sum u_z \phi_z(x)$ into the Deep Ritz Objective function

$$\frac{1}{n}\sum_{i=1}^{n}\left(\sum_z u_z \nabla \phi_z(x_i)\right)^2 + \sum_z u_z \phi_z(x_i) f(x_i)$$

A Fourier Basis View

Solving a simple PDE $\Delta u = f$ using Fourier Basis.

Estimator 1

First Estimate f then solve u, $f_z = \frac{1}{n}\sum f(x_i)\phi_z(x_i)$, then $u = \sum \frac{1}{\|z\|^2} f_z \phi_z(x)$

Estimator 2

Plug $u = \sum u_z \phi_z(x)$ into the Deep Ritz Objective function

$$\frac{1}{n}\sum_{i=1}^{n}\left(\sum_z u_z \nabla\phi_z(x_i)\right)^2 + \sum_z u_z\phi_z(x_i) f(x_i)$$

A Fourier Basis View

Solving a simple PDE $\Delta u = f$ using Fourier Basis.

Estimator 1

First Estimate f then solve u, $f_z = \frac{1}{n}\sum f(x_i)\phi_z(x_i)$, then $u = \sum \frac{1}{\|z\|^2} f_z \phi_z(x)$

Estimator 2

Plug $u = \sum u_z \phi_z(x)$ into the Deep Ritz Objective function

$$\frac{1}{n}\sum_{i=1}^{n}\left(\sum_z u_z \nabla\phi_z(x_i)\right)^2 + \sum_z u_z \phi_z(x_i) f(x_i)$$

Estimator1 is Optimal

Consider estimating in H_{-1} norm using Fourier Basis up to Z, *i.e.* $\mathcal{Z} := \{z \in \mathbb{N}^d \| \|z\|_\infty \leqslant Z\}$.

- **Bias**:

$$\| \sum_{\|z\|_\infty > Z} f_z \phi_z \|_{H^{-1}}^2 \leqslant C \sum_{\|z\|_\infty > Z} f_z^2 z^{-2} \leqslant \|z\|^{-2(s-1)} \|f\|_{H_{\alpha-2}}^2$$

- **Variance**:

$$\mathbb{E}\|f - f\|_{H_{-1}}^2 \leqslant \mathbb{E} \sum_{\|z\|_\infty \leqslant Z} (f_z - f_z)^2 \|\phi_z\|_{H_{-1}}^2 \leqslant \sum_{\|z\|_\infty \leqslant Z} |z|^{-1} \mathrm{Var}(f_z)$$

Final bound: $Z^{-2(s-1)} + \frac{Z^{d-2}}{n}$

Difference Between Estimator1 and 2

- **Estimator 1:** The Fourier coefficient of the solution of Estimator 1 is

$$\mathrm{u}_{1,z} = \operatorname{diag}\left(\|z\|_2^2\right)^{-1}_{\|z\|_\infty \leqslant Z} f_z. \tag{2}$$

- **Estimator 2:** The Fourier coefficient of the solution of Estimator 2 is

$$\mathrm{u}_{2,z} = \underbrace{\left(\frac{1}{n}\sum_{i=1}^{n} \nabla\phi_i(x_i)\nabla\phi_j(x_i)\right)^{-1}_{\|i\|_\infty \leqslant Z, \|j\|_\infty \leqslant Z}}_{\text{empirical Gram Matrix } A} f_z, \tag{3}$$

Thus $\|u_1 - u_2\|^2_{H_1} \propto \|((\mathbb{E}A) - A)\|^2_{\mathrm{H}} \propto \frac{Z^d}{n}$.

Difference Between Estimator1 and 2

- **Estimator 1:** The Fourier coefficient of the solution of Estimator 1 is

$$\mathrm{u}_{1,z} = \mathrm{diag}\left(\|z\|_2^2\right)^{-1}_{\|z\|_\infty \leqslant Z} f_z. \tag{2}$$

- **Estimator 2:** The Fourier coefficient of the solution of Estimator 2 is

$$\mathrm{u}_{2,z} = \underbrace{\left(\frac{1}{n}\sum_{i=1}^{n} \nabla\phi_i(x_i)\nabla\phi_j(x_i)\right)^{-1}_{\|i\|_\infty \leqslant Z, \|j\|_\infty \leqslant Z}}_{\text{empirical Gram Matrix } A} f_z, \tag{3}$$

Thus $\|u_1 - u_2\|_{H_1}^2 \propto \|((\mathbb{E}A) - A)\|_{\mathrm{H}}^2 \propto \frac{Z^d}{n}$.

How Much Gradient We Need?

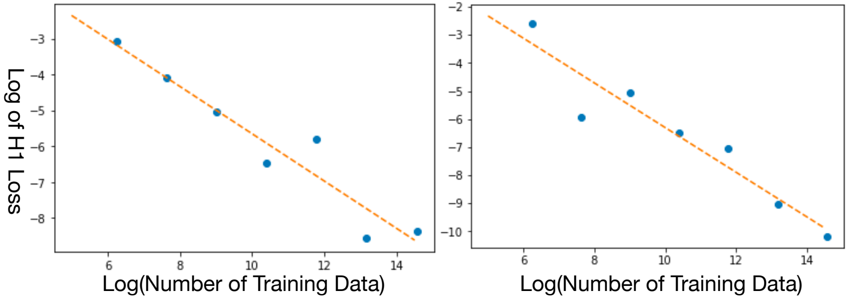
We Introduce the Modified DRM

$$\mathcal{E}_{N,n}^{\mathrm{MDRM}}(u) = \underbrace{\frac{1}{N}\sum_{j=1}^{N}\left[|\Omega|\cdot\frac{1}{2}\|\nabla u(X_j')\|^2\right]}_{\text{Sample More Gradients}} + \frac{1}{n}\sum_{j=1}^{n}\left[|\Omega|\cdot\left(\frac{1}{2}V(X_j)|u(X_j)|^2 - f_j u(X_j)\right)\right] \tag{4}$$

Thus Variance: $\frac{\xi^d}{N} < \frac{\xi^{d-2}}{n} \simeq \xi^{-2(s-1)} \Rightarrow \xi \simeq n^{\frac{1}{d+2s-4}}$ and

$$\frac{N}{n} = \xi^2 = n^{\frac{2}{d+2s-4}}$$

Experiment



	(a) Deep Ritz Methods	(b) Modified Deep Ritz Methods
Theory	$\frac{2s-2}{d+2s-2} = 0.75$	$\frac{2s-2}{d+2s-4} = 1$
Empirical	0.6595	0.7953
R2 Score	0.91	0.89

Summarize in One Table...

Upper Bounds			Lower Bound
Objective Function	Neural Network	Fourier Basis	
Deep Ritz	$n^{-\frac{2s-2}{d+2s-2}} \log n$	$n^{-\frac{2s-2}{d+2s-2}}$	$n^{-\frac{2s-2}{d+2s-4}}$
Modified Deep Ritz	$n^{-\frac{2s-2}{d+2s-2}} \log n$	$n^{-\frac{2s-2}{d+2s-4}}$	$n^{-\frac{2s-2}{d+2s-4}}$
PINN	$n^{-\frac{2s-4}{d+2s-4}} \log n$	$n^{-\frac{2s-4}{d+2s-4}}$	$n^{-\frac{2s-4}{d+2s-4}}$

Table: Upper bounds and lower bounds we achieve in this paper and previous work. The upper bound colored in red indicates that the convergence rate matches the min-max lower bound.

Observation 3: Tigher Local Rademacher

Local Rademacher Complexity

$$\psi(r) \geqslant \mathbb{E} R_n\{f \in \mathcal{F}, \underbrace{T(f) \leqslant r}_{\text{loss function}}\}$$

- For nonparametric estimation: ℓ_2 Norm
- For Solving PDE: Sobolev Norm

Can Tigher Norm leads to Tigher Bound?

- Fourier Basis Yes DNN No

Gradient Descent

Why you select Ritz form in the first paper

Me

minimizing $\int(\Delta u)^2$ is crazy to me due to the condition number of $\Delta^\top \Delta$

Lexing

Gradient Descent

Why you select Ritz form
in the first paper

Me

minimizing $\int(\Delta u)^2$ is crazy to me
due to the condition number of $\Delta^\top \Delta$

Lexing

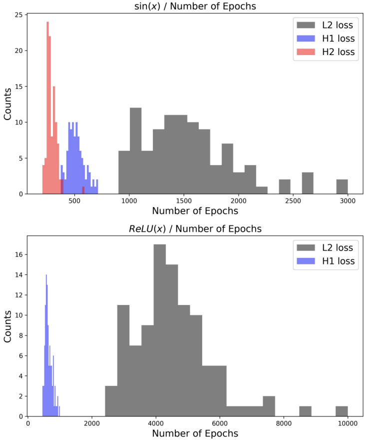
Gradient Descent

Why you select Ritz form in the first paper

Me

minimizing $\int(\Delta u)^2$ is crazy to me due to the condition number of $\Delta^\top\Delta$

Lexing



(Stochastic) Gradient Descent

Let's consider $\Delta u = f$ via minimizing $\frac{1}{2}\langle f, \mathcal{A}_1 f\rangle - \langle u, \mathcal{A}_2 f\rangle$

- **Deep Ritz Methods**. $\mathcal{A}_1 = \Delta, \mathcal{A}_2 = Id$
- **PINN**. $\mathcal{A}_1 = \Delta^2, \mathcal{A}_2 = \Delta$

We consider parameterize f using kernel regression $f(x) = \langle \theta, K_x\rangle$. Then we apply a stochastic gradient descent and get

$$\theta_{t+1} = \theta_t - \eta(\langle \theta, \mathcal{A}_1 K_{x_i}\rangle K_{x_i} - f_i \mathcal{A}_2 K_{x_i})$$

(Stochastic) Gradient Descent

Let's consider $\Delta u = f$ via minimizing $\frac{1}{2}\langle f, \mathcal{A}_1 f\rangle - \langle u, \mathcal{A}_2 f\rangle$

- **Deep Ritz Methods**. $\mathcal{A}_1 = \Delta, \mathcal{A}_2 = Id$
- **PINN**. $\mathcal{A}_1 = \Delta^2, \mathcal{A}_2 = \Delta$

We consider parameterize f using kernel regression $f(x) = \langle \theta, K_x\rangle$. Then we apply a stochastic gradient descent and get

$$\theta_{t+1} = \theta_t - \eta(\langle \theta, \mathcal{A}_1 K_{x_i}\rangle K_{x_i} - f_i \mathcal{A}_2 K_{x_i})$$

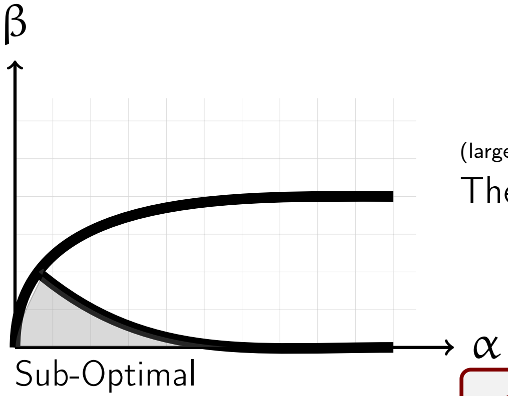
Setting: Sobolev Learning Rate

We can formulate the Sobolev Norm as $[H^\alpha]$ norm as

$$\|\sum_{i\geqslant 1} a_i \mu_i^{\alpha/2} e_i\|_{[H]^\alpha} := \left(\sum_{i\geqslant 1} a_i^2\right)^2$$

- The evaulation Sobolev norm can be different as the training Sobolev norm. We consider convergence rate in H^γ norm.

First Result: Three Regime



Can be concluded into Three Regimes

- β:function smoothness
- α:kernel smoothness

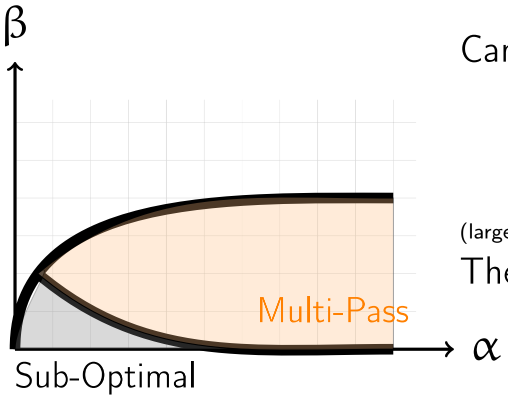
(larger, smoother)

The first Regime:

- Suboptimal, concentration error of $\frac{1}{n} K_x \otimes K_x \to \Sigma$ dominates

Similar to the modified DRM!

First Result: Three Regime



Can be concluded into Three Regimes

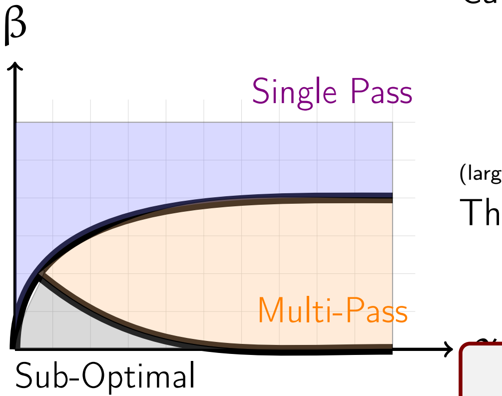
- β:function smoothness
- α:kernel smoothness

(larger, smoother)

The second regime:

- Constant Lr, Multipass

First Result: Three Regime



Can be concluded into Three Regimes

- β:function smoothness
- α:kernel smoothness

(larger, smoother)

The third regime:

- Small Lr, Single Pass

Online Learning

Lower Bound

Recall

$$\inf_{H} \sup_{u \in C^{\alpha}(\Omega)} \mathbb{E}\| H(\{X_i, f_i\}_{i=1,\cdots,n}) - u^* \|_{W_s^2} \gtrsim n^{-\frac{2\alpha - 2s}{2\alpha - 2t + d}},$$

and translate it into kernel setting

$$\| f_\lambda - f \|^2_{[H]^\gamma} \leqslant n^{-\frac{(\beta-\gamma)\alpha}{\beta\alpha + 2(p-q)+1}}$$

They matches for

- $\alpha = 1/d$
- $\beta = 2\alpha, \gamma = 2s$
- $(q - p) = t$ (p,q: eigen decay of $\mathcal{A}_1, \mathcal{A}_2$)

Upper Bound

We can achieve infomration theortical optimal rate $n^{-\frac{(\beta-\gamma)\alpha}{\beta\alpha+2(p-q)+1}}$ via Bias-Variance Tradeoff.

- Train Longer, Bias Smaller.
- Train Longer, Bias Larger.

Convergence time

The convergence time will equal to the optimal selection of λ

Iteration Time

$$\lambda = n^{\frac{\alpha+\mathbf{p}}{\beta\alpha+2(\mathbf{p}\text{-}\mathbf{q})+1}}$$

- Independent of γ.
- $(\mathbf{p}\text{-}\mathbf{q})$ is from the equation.
- $\mathbf{p}$ the only thing effects!

DRM Vs PINN

Recall Iteration time $\lambda = n^{\frac{\alpha+p}{\beta\alpha+2(p-q)+1}}$. To compare DRM and PINN, we should fix $p-q$ and then consider the dependency of iteration time on p.

- Denominator do nothing with p
- Numerator
  - $p < 0, \alpha > 0$, differential operator helps to balance the condition number of the kernel operator. PINN is faster.
  - $\alpha + p > 0$ means activation function should be smooth for NTK

DRM Vs PINN

Recall Iteration time $\lambda = n^{\frac{\alpha+p}{\beta\alpha+2(p-q)+1}}$. To compare DRM and PINN, we should fix $p-q$ and then consider the dependency of iteration time on p.

- Denominator do nothing with p
- Numerator
  - $p < 0, \alpha > 0$, differential operator helps to balance the condition number of the kernel operator. PINN is faster.
  - $\alpha + p > 0$ means activation function should be smooth for NTK

DRM Vs PINN

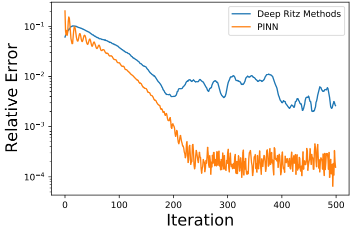


Figure: $\sum_{i=1}^{d} \sin(2\pi x)$

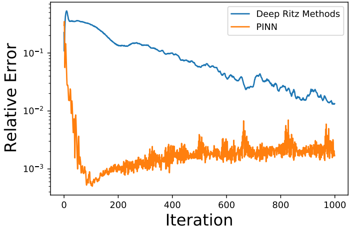


Figure: $\sum_{i=1}^{d} \sin(4\pi x)$

Variance of Integral by Parts

$$\mathbb{E}_{\mathbb{P}_n(x,y)} \frac{1}{2} \langle u, K_x \otimes \mathcal{A}_1 K_x u \rangle - y \langle u, \mathcal{A}_2 K_x \rangle$$

We considered the dynamic

$$\theta_t = \theta_{t-1} + \gamma \frac{1}{n} \sum_{i=1}^{n} \left(y_i \mathcal{A}_2 K_{x_i} - \underbrace{\langle \theta_{t-1}, \mathcal{A}_1 K_{x_i} \rangle_{\mathcal{H}} K_{x_i}}_{\text{not } \left(\left\langle \theta_{t-1}, \mathcal{A}_1 K_{x_i} \right\rangle_{\mathcal{H}} K_{x_i} + \left\langle \theta_{t-1}, K_{x_i} \right\rangle_{\mathcal{H}} \mathcal{A}_1 K_{x_i} \right)} \right)$$

for the variance of integral by parts may dominated.

Main Message

- Deep Ritz Method **High** dimensional problem, **Smooth** problem
- PINN **Low** dimensional problem, **Non-smooth** problem

Take Home Message

- Non-parametric statistics view of numerical PDE solver
- Gives us new constraints to design objective functions to be statistical/information theortical optimal
- sparsity of the weight is not a good measurement of the complexity of gradients, we need to find new measure
- GD analysis sugguest Sobolev Training

Reference

- Lu Y, Chen H, Lu J, et al. Machine Learning For Elliptic PDEs: Fast Rate Generalization Bound, Neural Scaling Law and Minimax Optimality. ICLR 2021.
- Lu Y, Jose B, et al. Sobolev Acceleration and Statistical Optimality for Learning Elliptic Equations, submitted.

Thank you for listening! and Questions?

Yiping Lu

yplu@stanford.edu.cn
https://web.stanford.edu/~yplu/