Tensor networks and optimal sampling in physics informed machine learning

GDR Mascot-Num: Workshop on Physics Informed Learning

Robert Gruhlke, Charles Miranda, Anthony Nouy, **Philipp Trunschke**

December 5, 2023

Stationary diffusion

- Consider the random stationary diffusion equation

$$\begin{aligned} -\nabla_x \cdot (a(x,y)\nabla_x u(x,y)) &= f(x) \qquad \text{in } D \\ u(x,y) &= 0 \qquad \text{on } \partial D \end{aligned}$$

- $x \in D$ for a bounded Lipschitz domain $D \subseteq \mathbb{R}^d$
- $y \sim \rho$ for a measure ρ on the probability space (Ω, Σ, ρ)

The Dirichlet principle states that

$$u = \underset{v \in H_0^1(D) \otimes L^2(\rho)}{\arg\min} \int \frac{1}{2} a(x,y) \|\nabla_x v(x,y)\|_2^2 - f(x) v(x,y) \, \mathrm{d}x \, \mathrm{d}\rho(y)$$

Goal: A theory for physics-informed losses, using adaptivity and optimal sampling.
Disclaimer: No final numerical experiments.

Approximation by tensor networks

- Approximate u in a *model class* $\mathcal{M} \subseteq H_0^1(D) \otimes L^2(\rho)$ as

$$u_{\mathcal{M}} = \operatorname*{arg\,min}_{v \in \mathcal{M}} \int \frac{1}{2} a(x,y) \|\nabla_x v(x,y)\|_2^2 - f(x) v(x,y) \, \mathrm{d}x \, \mathrm{d}\rho(y)$$

- Choose $\mathcal{M}$ as a set of low-rank tree tensor networks.
- Tensor networks are multilinear approximations that can break the curse of dimensionality.
- They can be interpreted as a subclass of neural networks.
- **They are a popular tool in the numerics of parametric PDEs because**
  - the optimisation problem is practically solvable
  - refinement is possible and interpretable

The theory-to-practice gap

General setting

- Let ρ be a probability measure on $\mathcal{X}$.
- Let $\mathcal{H}$ be a Hilbert space with inner product

$$(v, w) := \int (L_x v)^\mathsf{T} (L_x w) \, \mathrm{d}\rho(x).$$

  - $L^2(\rho)$ corresponds to $L_x v := v(x)$.
  - $H_0^1(\rho)$ corresponds to $L_x v := \nabla v(x)$.
- For a model class $\mathcal{M} \subseteq \mathcal{H}$ consider

$$u_{\mathcal{M}} = \operatorname*{arg\,min}_{v \in \mathcal{M}} \mathcal{L}(v) \qquad \text{with} \qquad \mathcal{L}(v) := \int \ell(v; x) \, \mathrm{d}\rho(x).$$

Generalisation error bounds

- If $\mathcal{L}$ is replaced by a MC estimate $\mathcal{L}_n$ with sample size n,

$$u_{\mathcal{M},n} := \arg\min_{v \in \mathcal{M}} \mathcal{L}_n(v).$$

- This ensues a *generalisation error*.
- Suppose that $\mathcal{M}$ is compact.
- Suppose ℓ is bounded and $\ell(\bullet, x)$ is Lipschitz on $\mathcal{M}$ for all $x \in \mathcal{X}$.
- Then, at best,

$$\mathbb{E}[\mathcal{L}(u_{\mathcal{M},n})] \leq \mathcal{L}(u_{\mathcal{M}}) + \mathcal{O}(n^{-1/2}).$$

This is a slow convergence under strong assumptions.

Least squares setting

- Consider, initially,
$$\mathcal{L}(v) := \tfrac{1}{2}\|u - v\|^2$$
  - for the sake of simplicity,
  - as a model for locally L-smooth and strongly convex losses and
  - because we will use it later.
- Recall that
$$u_{\mathcal{M}} \in \operatorname*{arg\,min}_{v \in \mathcal{M}} \tfrac{1}{2}\|u - v\|^2 \qquad \text{and} \qquad u_{\mathcal{M},n} \in \operatorname*{arg\,min}_{v \in \mathcal{M}} \tfrac{1}{2}\|u - v\|_n^2.$$
- Specifically, let $w > 0$ satisfy $\int w^{-1} \,\mathrm{d}\rho = 1$ and $x_1, \dots, x_n \sim w^{-1}\rho$ be i.i.d. and let
$$\|u - v\|_n^2 := \frac{1}{n}\sum_{i=1}^{n} w(x_i)\|L_{x_i}(u - v)\|_2^2.$$

Conditions for solvability and stability

1. To obtain a valid solution, we want that

$$\|u - v\|_n^2 \leq \varepsilon \text{ implies } \|u - v\|^2 \leq (1 - \delta)^{-1}\varepsilon$$

for some $\delta \in (0, 1)$ and all $v \in \mathcal{M}$.

Conditions for solvability and stability

1. To obtain a valid solution, we want that

$$\|u - v\|_n^2 \leq \varepsilon \text{ implies } \|u - v\|^2 \leq (1 - \delta)^{-1}\varepsilon$$

for some $\delta \in (0, 1)$ and all $v \in \mathcal{M}$.

   - Otherwise, there exist spurious (global) empirical minima.

Conditions for solvability and stability

1. To obtain a valid solution, we want that
$$\|u - v\|_n^2 \leq \varepsilon \text{ implies } \|u - v\|^2 \leq (1 - \delta)^{-1}\varepsilon$$
for some $\delta \in (0, 1)$ and all $v \in \mathcal{M}$.
   - Otherwise, there exist spurious (global) empirical minima.
2. For numerical stability, we also want that
$$\|u - v\|^2 \leq \varepsilon \text{ implies } \|u - v\|_n^2 \leq (1 + \delta)\varepsilon$$
for some $\delta \in (0, 1)$ and all $v \in \mathcal{M}$.

Conditions for solvability and stability

1. To obtain a valid solution, we want that
$$\|u - v\|_n^2 \leq \varepsilon \text{ implies } \|u - v\|^2 \leq (1 - \delta)^{-1}\varepsilon$$
for some $\delta \in (0, 1)$ and all $v \in \mathcal{M}$.
   - Otherwise, there exist spurious (global) empirical minima.
2. For numerical stability, we also want that
$$\|u - v\|^2 \leq \varepsilon \text{ implies } \|u - v\|_n^2 \leq (1 + \delta)\varepsilon$$
for some $\delta \in (0, 1)$ and all $v \in \mathcal{M}$.
   - Otherwise, numerically minimising $\|\bullet\|_n$ might not yield a minimum for $\|\bullet\|$.

Conditions for solvability and stability

1. To obtain a valid solution, we want that
$$\|u - v\|_n^2 \leq \varepsilon \text{ implies } \|u - v\|^2 \leq (1-\delta)^{-1}\varepsilon$$
for some $\delta \in (0,1)$ and all $v \in \mathcal{M}$.
   - Otherwise, there exist spurious (global) empirical minima.
2. For numerical stability, we also want that
$$\|u - v\|^2 \leq \varepsilon \text{ implies } \|u - v\|_n^2 \leq (1+\delta)\varepsilon$$
for some $\delta \in (0,1)$ and all $v \in \mathcal{M}$.
   - Otherwise, numerically minimising $\|\bullet\|_n$ might not yield a minimum for $\|\bullet\|$.

With the set $S := \{u\} - \mathcal{M}$, both conditions can be combined into $\mathrm{RIP}_S(\delta)$

$$(1-\delta)\|v\|^2 \leq \|v\|_n^2 \leq (1+\delta)\|v\|^2, \quad v \in S.$$

The probability of $\mathrm{RIP}_S(\delta)$

Definition

For any set $S \subseteq \mathcal{H}$, define the *inverse Christoffel function* $\mathfrak{K}_S(y) := \sup_{v \in S} \frac{\|L_x v\|_2^2}{\|v\|^2}$.

Theorem (Eigel, Schneider, T – 2021)

Under suitable assumptions on the set $S \subseteq \mathcal{H}$, and for any $\delta \in (0,1)$, there exists C such that

$$\mathbb{P}[\neg\, \mathrm{RIP}_S(\delta)] \leq C \exp\left(-\frac{n}{2}\left(\frac{\delta}{\|w\mathfrak{K}_S\|_{L^\infty(\rho)}}\right)^2\right).$$

The constant C is independent of n and depends polynomially on δ and $\|w\mathfrak{K}_S\|_{L^\infty(\rho)}^{-1}$.

This probability is unbounded for neural networks.
It grows exponentially with the number of variables for tensor networks.

Practical bound

- We can restrict the model class to a neighbourhood $\mathcal{N} \subseteq \mathcal{M}$ of the solution.

Theorem

Let $r > 0$ and $\mathcal{N} \subseteq \mathcal{M} \cap B(u_{\mathcal{M}}, r)$ be a manifold with bounded curvature[1] $\kappa \leq \frac{1}{r}$. Then

$$\mathfrak{K}_{\mathbb{T}_{u_{\mathcal{M}}}\mathcal{N}} \leq \mathfrak{K}_S \leq \left(\sqrt{\mathfrak{K}_{\mathbb{T}_{u_{\mathcal{M}}}\mathcal{N}}} + \frac{\kappa r}{2}\sqrt{\mathfrak{K}_{\mathbb{T}^{\perp}_{u_{\mathcal{M}}}\mathcal{N}}}\right)^2.$$

This may explain successful applications,
but it is an unrealistic assumption.

[1] with bounded reach

Discussion

- The theory only applies to quadratic losses.
- But it shows even for those that we should not use i.i.d. samples.

Idea: Adapt the samples for each iteration of a SGD.

Optimal sampling for SGD

General framework

1. Compute the gradient
$$g_t := \nabla_v \mathcal{L}(v_t).$$
2. Define the "local linearisation" $\mathcal{T}_t$ and the empirical map $P_t^n : \mathcal{H} \to \mathcal{T}_t$.
3. Perform the linear update
$$\bar{v}_{t+1} := v_t - s_t P_t^n g_t.$$
4. Map $\bar{v}_{t+1}$ back to $\mathcal{M}$ via the recompression map
$$v_{t+1} := R_t(\bar{v}_{t+1}).$$

SGD and NGD correspond to different choices of P_t^n and R_t.

SGD

- Parameterise $v_t := V(\theta_t)$ with $\theta_t \in \mathbb{R}^D$ and recall that SGD defines the update direction

$$\nabla_\theta \mathcal{L}_n(V(\theta)).$$

- Let $\varphi_k := \partial_k V(\theta_t)$ for $k = 1, \dots, D$ and $\mathcal{T}_t := \operatorname{span}\{\varphi_k : k = 1, \dots, D\}$.
- Under suitable conditions on $\mathcal{L}$, and for $\mathcal{H} = L^2(\rho)$, it holds that

$$(\nabla_\theta \mathcal{L}_n(V(\theta_k)), e_k) = (\nabla_v \mathcal{L}(v), \varphi_k)_n.$$

- Hence, SGD corresponds to the choice

$$P_t^n g := \sum_{k=1}^{D} \hat{\zeta}_k \varphi_k, \qquad \hat{\zeta}_k := (g, \varphi_k)_n = \frac{1}{n} \sum_{i=1}^{n} w_t(x_i) g(x_i) \varphi_k(x_i).$$

- However, the SGD choice $R_t(V(\theta) - sP_t^n g) := V(\theta - s\hat{\zeta})$ does not satisfy our assumptions.

Convergence rates for "our" SGD

- The speed of convergence depends on the constants

$$\mathbb{E}\big[(g_t, P_t^n g_t) \,\big|\, \mathcal{F}_t\big] \geq c_{\mathrm{bias},1}\|P_t g_t\|^2 - c_{\mathrm{bias},2},$$
$$\mathbb{E}\big[\|P_t^n g_t\|^2 \,\big|\, \mathcal{F}_t\big] \leq c_{\mathrm{var},1}\|P_t g_t\|^2 + c_{\mathrm{var},2}\|(I - P_t)g_t\|^2.$$

- Namely, $\mathbb{E}[\mathcal{L}(v_{t+1})|v_t] \leq \mathcal{L}(v_t)$ requires a step size $s_t \lesssim \frac{c_{\mathrm{bias}}}{c_{\mathrm{var},1}}$.
- Define the Gramian matrix $G \in \mathbb{R}^{D\times D}$ by $G_{jk} := (\varphi_j, \varphi_k)$.
- Denote by λ_* the smallest positive eigenvalue and by λ^* the largest eigenvalue.
- Then SGD exhibits the constants

$$c_{\mathrm{bias}} = \lambda_*(G), \qquad c_{\mathrm{var},1} = \frac{\lambda^*(G)^2(n-1)+\lambda^*(G)\|w_t \mathfrak{K}_{\mathcal{T}_t}\|_{L^\infty(\rho)}}{n}, \qquad c_{\mathrm{var},2} = \frac{\lambda^*(G)\|w_t \mathfrak{K}_{\mathcal{T}_t}\|_{L^\infty(\rho)}}{n}.$$

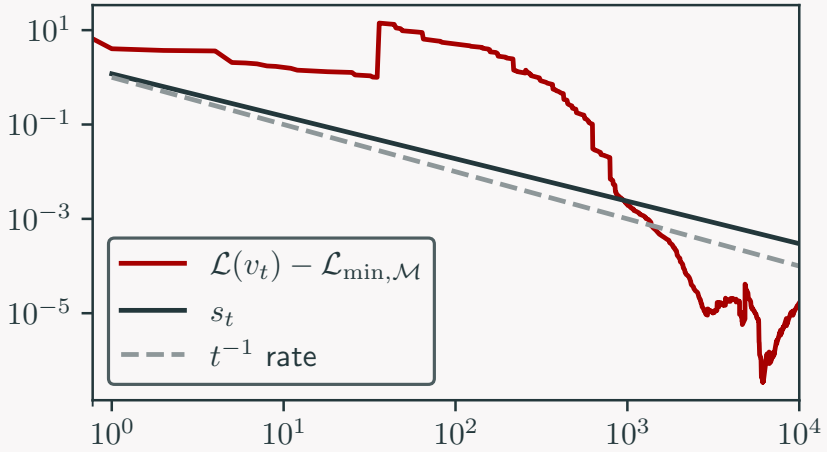
NGD and optimal sampling

- Improving the convergence rate requires two steps:
  1. orthogonalising the basis $\varphi_1, \dots, \varphi_D \rightsquigarrow \lambda_*(G) = \lambda^*(G) = 1$ is optimal.
  2. choosing an optimal weight function $w_t \propto \mathfrak{K}_{\mathcal{T}_t}^{-1} \rightsquigarrow \|w_t \mathfrak{K}_{\mathcal{T}_t}\|_{L^\infty(\rho)} = \dim(\mathcal{T}_t)$ remains bounded.
- The first step yields "our" version of NGD.
- But, notably, $\|\mathfrak{K}_{\mathcal{T}_t}\|_{L^\infty(\rho)}$ could still become unbearably large.
- Applying both simultaneously yields the uniform rates:

	GD	Best-cast	Worst-case	SGD
L-smoothness	$\mathcal{O}(t^{-1})$	$\mathcal{O}(t^{\varepsilon-1})$	$\mathcal{O}(t^{\varepsilon-1/2})$	$\mathcal{O}(t^{\varepsilon-1/2})$
λ-PŁ on $\mathcal{M}$	$\tilde{\mathcal{O}}(e^{-t})$	$\tilde{\mathcal{O}}(e^{-t})$	$\mathcal{O}(t^{\varepsilon-1})$	$\mathcal{O}(t^{\varepsilon-1})$

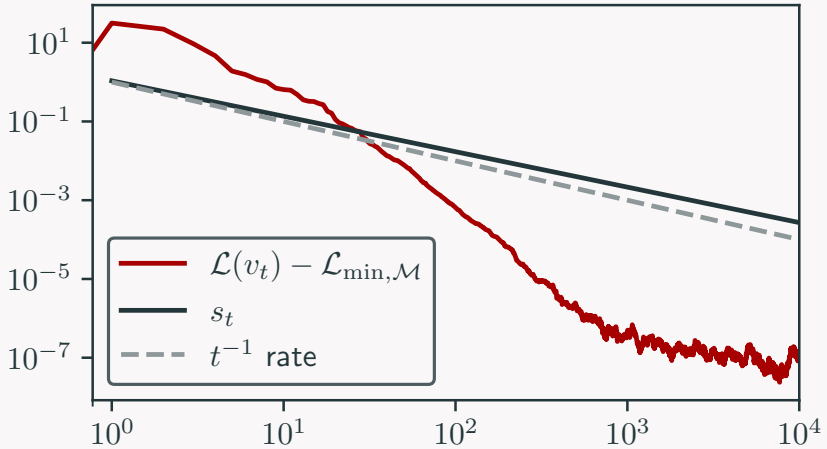
SGD for linear least squares

- approximates $u(x) := \exp(x)$ on $L^2(\rho)$ with $\rho = \mathcal{N}(0, 1)$
- uses 70 Gaussian samples per iteration



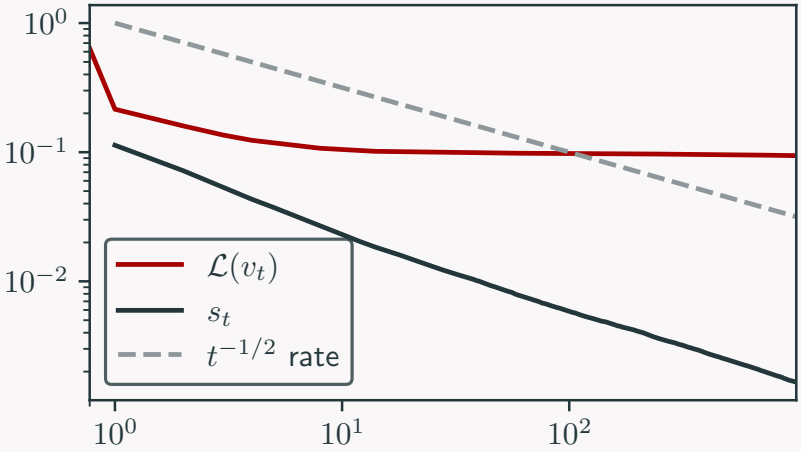
SGD for linear least squares

- approximates $u(x) := \exp(x)$ on $L^2(\rho)$ with $\rho = \mathcal{N}(0, 1)$
- uses 7 optimal samples per iteration



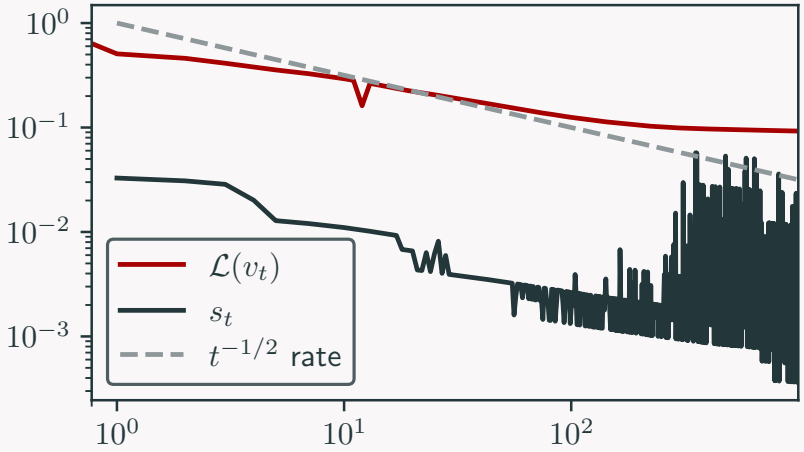
SGD for least squares with width-20 shallow neural networks

- approximates $u(x) := \sin(2\pi x)$ on $L^2(\rho)$ with $\rho = \mathcal{U}([0,1])$
- uses 200 uniform samples per iteration



SGD for least squares with width-20 shallow neural networks

- approximates $u(x) := \sin(2\pi x)$ on $L^2(\rho)$ with $\rho = \mathcal{U}([0,1])$
- uses 200 uniform samples per iteration and an adaptive step size



NGD for least squares with width-20 shallow neural networks

- approximates $u(x) := \sin(2\pi x)$ on $L^2(\rho)$ with $\rho = \mathcal{U}([0,1])$
- uses 200 optimal samples per iteration and an adaptive step size