What Makes a Good Preconditioner for Data Science?

Mitchell T. Scott

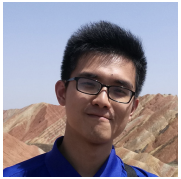
Department of Mathematics, Emory University

Preconditioning techniques for scientific and industrial applications
Precond 2026

May 28, 2026

Collaborators and Acknowledgments

Yousef Saad,
Minnesota



Tianshi Xu,
Emory

Yuanzhe Xi,
Emory

This work was partially supported by NSF DMS-2208412 and NSF DMS-2513118.

EMORY UNIVERSITY

Table of Contents

Stochastic gradient descent is the workhorse of ML training optimizers.

Example

I want to minimize some loss, but the dataset is so large, the gradient computation takes forever!
What do I do?

Stochastic gradient descent is the workhorse of ML training optimizers.

Example

I want to minimize some loss, but the dataset is so large, the gradient computation takes forever!
What do I do?

Definition (Preconditioned Stochastic Gradient Descent)

To minimize the stochastic optimization problem, we can write the resulting algorithm as

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \alpha_k \mathbf{M}^{-1} g(\mathbf{w}_k, \boldsymbol{\xi}_k), \tag{1}$$

where $g(\mathbf{w}_k, \boldsymbol{\xi}_k) = \nabla_w F_k(\mathbf{w})$ is the stochastic gradient, α_k is the learning rate, $\boldsymbol{\xi}_k$ is an i.i.d. sample drawn at iteration k, and $\mathbf{M}$ is the preconditioner.

Common preconditioners for SGD can consider isotropy and curvature.

Example (Adam [1])

Define $\mathbf{s}_k$ is an exponential moving average of squared gradients up until iteration k, then

$$\mathbf{M}_{\text{Adam}} = \text{diag}\left(\sqrt{\mathbf{s}_k} + \boldsymbol{\varepsilon}\right),$$

Common preconditioners for SGD can consider isotropy and curvature.

Example (Adam [1])

Define $\mathbf{s}_k$ is an exponential moving average of squared gradients up until iteration k, then

$$\mathbf{M}_{\text{Adam}} = \text{diag}\left(\sqrt{\mathbf{s}_k} + \boldsymbol{\varepsilon}\right),$$

Example (Full or mini-batch Hessian)

The Hessian, or mini-batch approximation to the Hessian, matrix of the loss function,

$$\mathbf{H}_{\mathcal{B}}(\boldsymbol{\theta}) = \nabla^2 \mathcal{L}_{\mathcal{B}}(\boldsymbol{\theta}).$$

Common preconditioners for SGD can consider isotropy and curvature.

Example (Adam [1])

Define $\mathbf{s}_k$ is an exponential moving average of squared gradients up until iteration k, then

$$\mathbf{M}_{\text{Adam}} = \text{diag}\left(\sqrt{\mathbf{s}_k} + \boldsymbol{\varepsilon}\right),$$

Example (Full or mini-batch Hessian)

The Hessian, or mini-batch approximation to the Hessian, matrix of the loss function,

$$\mathbf{H}_{\mathcal{B}}(\boldsymbol{\theta}) = \nabla^2 \mathcal{L}_{\mathcal{B}}(\boldsymbol{\theta}).$$

Example (Fisher Information Matrix)

Using the Fisher Information Matrix as a preconditioner corresponds to natural gradient descent [2]

$$\mathbf{F} = \mathbb{E}\left[\frac{d \log p(y|x,\theta)}{d\theta} \frac{d \log p(y|x,\theta)}{d\theta}^\top\right] := \mathbb{E}[\mathcal{D}\boldsymbol{\theta}\mathcal{D}\boldsymbol{\theta}^\top] = \mathbb{C}\text{ov}(\mathcal{D}\boldsymbol{\theta}, \mathcal{D}\boldsymbol{\theta})$$

Table of Contents

Classical Convergence Results for SGD

Theorem (Convergence of SGD [3])

Assume that a loss function F where $F_* := F(\mathbf{w}^*)$, has L-Lipschitz gradients and $\mathbb{E}\left[\|\nabla F(\mathbf{w})\|^2\right] \leq \sigma^2$, then SGD converges

$$\mathbb{E}\left[F(\mathbf{w}_k) - F_*\right] \leq \frac{F(\mathbf{w}_1) - F_*}{2\overline{\alpha}k}$$

Classical Convergence Results for SGD

Theorem (Convergence of SGD [3])

Assume that a loss function F where $F_* := F(\mathbf{w}^*)$, has L-Lipschitz gradients and $\mathbb{E}\left[\|\nabla F(\mathbf{w})\|^2\right] \leq \sigma^2$, then SGD converges

$$\mathbb{E}\left[F(\mathbf{w}_k) - F_*\right] \leq \frac{F(\mathbf{w}_1) - F_*}{2\overline{\alpha}k}$$

Theorem (Strongly Convex Objective, Fixed Stepsize (Thm 4.6, [4]))

Adding in stronger bounds on the gradient's first and second moments, and c-strong convexity. Running SGD with a small fixed step size $\overline{\alpha}$ satisfy

$$\mathbb{E}[F(\mathbf{w}_k) - F_*] \leq (1 - \overline{\alpha}c\mu)^{k-1}\left(F(\mathbf{w}_1) - F_* - \frac{\overline{\alpha}LK}{2c\mu}\right) + \frac{\overline{\alpha}LK}{2c\mu}, \tag{2}$$

where μ, K are constants associated with the stochastic gradients and variance, respectively.

Two important quantities for data science preconditioning.

Remark (Two stage convergence)

This highlights two late-stage drivers: a linear contraction factor $(1 - \alpha c\mu)$ *and a stochastic error floor*

$$\frac{\alpha L K}{2c\mu} = \frac{\alpha}{2\mu}\,\kappa\,K,$$

where $\kappa := \frac{L}{c}$ *is the (Euclidean) condition number associated with curvature.*

Two important quantities for data science preconditioning.

Remark (Two stage convergence)

This highlights two late-stage drivers: a linear contraction factor $(1 - \alpha c\mu)$ *and a stochastic error floor*

$$\frac{\alpha L K}{2c\mu} \;=\; \frac{\alpha}{2\mu}\,\kappa\, K,$$

where $\kappa := \frac{L}{c}$ *is the (Euclidean) condition number associated with curvature.*

Definition (Conditional Variance)

$$\mathbb{V}_{\boldsymbol{\xi}_k}\big[g(\mathbf{w}_k, \boldsymbol{\xi}_k), \|\cdot\|_{\mathbf{M}^{-1}}\big] := \mathbb{E}_{\boldsymbol{\xi}_k}\left[\|g(\mathbf{w}_k, \boldsymbol{\xi}_k)\|^2_{\mathbf{M}^{-1}}\right] - \left\|\mathbb{E}_{\boldsymbol{\xi}_k}[g(\mathbf{w}_k, \boldsymbol{\xi}_k)]\right\|^2_{\mathbf{M}^{-1}} = \operatorname{tr}\big(\mathbf{M}^{-1}\boldsymbol{\Sigma}(\mathbf{w})\big), \tag{3}$$

where $\boldsymbol{\Sigma}(\mathbf{w}) := \mathbb{C}\mathrm{ov}(g(\mathbf{w}, \boldsymbol{\xi}) \mid \mathbf{w})$. For late stage convergence

$$\mathbb{V}_{\boldsymbol{\xi}_k}\big[g(\mathbf{w}_k, \boldsymbol{\xi}_k), \|\cdot\|_{\mathbf{M}^{-1}}\big] \;\leq\; K + \mathcal{O}(\overline{\alpha} K),$$

Preconditioning helps convex problems converge faster with a lower loss.

Theorem (S., *et al.*, '26, [5])

Running (1) *with small* $\alpha_k = \overline{\alpha}$ *under preconditioned analogues of common assumptions, then, for all* $k \in \mathbb{N}$,

$$\mathbb{E}[F(\mathbf{w}_k) - F_*] \leq \frac{\overline{\alpha}\,\hat{L}\,K}{2\,\hat{c}\,\mu} + (1 - \overline{\alpha}\,\hat{c}\,\mu)^{k-1}\left(F(\mathbf{w}_1) - F_* - \frac{\overline{\alpha}\,\hat{L}\,K}{2\,\hat{c}\,\mu}\right) \xrightarrow{k\to\infty} \frac{\overline{\alpha}\,\hat{L}\,K}{2\,\hat{c}\,\mu}. \tag{4}$$

Preconditioning helps convex problems converge faster with a lower loss.

Theorem (S., *et al.*, '26, [5])

Running (1) *with small* $\alpha_k = \overline{\alpha}$ *under preconditioned analogues of common assumptions, then, for all* $k \in \mathbb{N}$,

$$\mathbb{E}[F(\mathbf{w}_k) - F_*] \;\leq\; \frac{\overline{\alpha}\,\hat{L}\,K}{2\,\hat{c}\,\mu} \;+\; (1 - \overline{\alpha}\,\hat{c}\,\mu)^{k-1}\left(F(\mathbf{w}_1) - F_* - \frac{\overline{\alpha}\,\hat{L}\,K}{2\,\hat{c}\,\mu}\right) \xrightarrow{k\to\infty} \frac{\overline{\alpha}\,\hat{L}\,K}{2\,\hat{c}\,\mu}. \tag{4}$$

Theorem

Now suppose (1) *has a decaying learning rate, then, for all* $k \in \mathbb{N}$,

$$\mathbb{E}[F(\mathbf{w}_k) - F_*] \;\leq\; \frac{\nu}{\gamma + k}, \qquad \nu := \max\left\{\frac{\beta^2 \hat{L} K}{2(\beta\hat{c}\mu - 1)}, (\gamma+1)\big(F(\mathbf{w}_1) - F_*\big)\right\}. \tag{5}$$

Polyak-Łojasiewicz functions are structured but nonconvex.

Assumption (Local PŁ functions on $\mathcal{N}$)

There exists $\hat{\mu}_{\mathrm{PL}} > 0$ *such that, for all* $\mathbf{w} \in \mathcal{N}$: $\quad 2\hat{\mu}_{\mathrm{PL}}\big(F(\mathbf{w}) - F_*\big) \;\leq\; \|\nabla F(\mathbf{w})\|^2$.

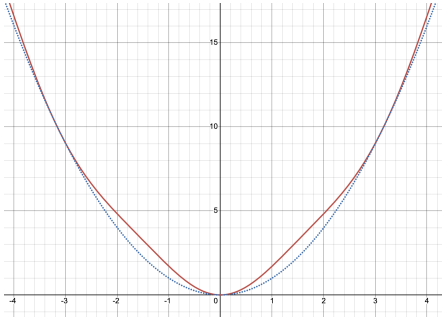


Figure 1: Comparing a convex function $F(x) = x^2$ with a $\mu-$PŁ function $F(x) = x^2 + \sin^2(x)$. [6]

Preconditioning helps in the nonconvex case, more realistic for SciML problems.

Theorem (S., *et al.*, '26 (Informal) [5])

Assume $\mathbf{w}_1$ *is in a quadratically growing local basin of radius* r, $\mathcal{N}_r$. *Let* $\tau := \inf\{k \geq 1 : \mathbf{w}_k \notin \mathcal{N}_r\}$.

Preconditioning helps in the nonconvex case, more realistic for SciML problems.

Theorem (S., *et al.*, '26 (Informal) [5])

Assume $\mathbf{w}_1$ *is in a quadratically growing local basin of radius* r, $\mathcal{N}_r$. *Let* $\tau := \inf\{k \geq 1 : \mathbf{w}_k \notin \mathcal{N}_r\}$.
For a fixed step size $\alpha_k = \overline{\alpha}$,

Preconditioning helps in the nonconvex case, more realistic for SciML problems.

Theorem (S., *et al.*, '26 (Informal) [5])

Assume $\mathbf{w}_1$ *is in a quadratically growing local basin of radius* r, $\mathcal{N}_r$. *Let* $\tau := \inf\{k \geq 1 : \mathbf{w}_k \notin \mathcal{N}_r\}$.
For a fixed step size $\alpha_k = \overline{\alpha}$,

- *we have linear convergence (depending on* $\uparrow \hat{\mu}_{\mathrm{PL}}$*)*

Preconditioning helps in the nonconvex case, more realistic for SciML problems.

Theorem (S., *et al.*, '26 (Informal) [5])

Assume $\mathbf{w}_1$ *is in a quadratically growing local basin of radius* r, $\mathcal{N}_r$. *Let* $\tau := \inf\{k \geq 1 : \mathbf{w}_k \notin \mathcal{N}_r\}$.

For a fixed step size $\alpha_k = \overline{\alpha}$,

- *we have linear convergence (depending on* $\uparrow \hat{\mu}_{\mathrm{PL}}$*)*
- *and convergence up until a noise floor, which depends on a tradeoff between*
  - $\downarrow \hat{L}/\hat{\mu}_{\mathrm{PL}}$ *and*

Preconditioning helps in the nonconvex case, more realistic for SciML problems.

Theorem (S., *et al.*, '26 (Informal) [5])

Assume $\mathbf{w}_1$ *is in a quadratically growing local basin of radius* r, $\mathcal{N}_r$. *Let* $\tau := \inf\{k \geq 1 : \mathbf{w}_k \notin \mathcal{N}_r\}$.

For a fixed step size $\alpha_k = \overline{\alpha}$,

- *we have linear convergence (depending on* $\uparrow \hat{\mu}_{\mathrm{PL}}$*)*
- *and convergence up until a noise floor, which depends on a tradeoff between*
  - $\downarrow \hat{L}/\hat{\mu}_{\mathrm{PL}}$ *and*
  - $\downarrow K$.

Preconditioning helps in the nonconvex case, more realistic for SciML problems.

Theorem (S., *et al.*, '26 (Informal) [5])

Assume $\mathbf{w}_1$ *is in a quadratically growing local basin of radius* r, $\mathcal{N}_r$. *Let* $\tau := \inf\{k \geq 1 : \mathbf{w}_k \notin \mathcal{N}_r\}$.

For a fixed step size $\alpha_k = \overline{\alpha}$,

- *we have linear convergence (depending on* $\uparrow \hat{\mu}_{\mathrm{PL}}$*)*
- *and convergence up until a noise floor, which depends on a tradeoff between*
  - $\downarrow \hat{L}/\hat{\mu}_{\mathrm{PL}}$ *and*
  - $\downarrow K$.

For a harmonic step size $\alpha_k \propto 1/k$

Preconditioning helps in the nonconvex case, more realistic for SciML problems.

Theorem (S., *et al.*, '26 (Informal) [5])

Assume $\mathbf{w}_1$ *is in a quadratically growing local basin of radius* r, $\mathcal{N}_r$. *Let* $\tau := \inf\{k \geq 1 : \mathbf{w}_k \notin \mathcal{N}_r\}$.

For a fixed step size $\alpha_k = \overline{\alpha}$,

- *we have linear convergence (depending on* $\uparrow \hat{\mu}_{\mathrm{PL}}$*)*
- *and convergence up until a noise floor, which depends on a tradeoff between*
  - $\downarrow \hat{L}/\hat{\mu}_{\mathrm{PL}}$ *and*
  - $\downarrow K$.

For a harmonic step size $\alpha_k \propto 1/k$

- *we have* $\mathcal{O}(\nu/k)$ *convergence, where* ν *depends on* $\downarrow \hat{L}/\hat{\mu}_{\mathrm{PL}}$.

EMORY UNIVERSITY

Preconditioning helps in the nonconvex case, more realistic for SciML problems.

Theorem (S., *et al.*, '26 (Informal) [5])

Assume $\mathbf{w}_1$ *is in a quadratically growing local basin of radius* r, $\mathcal{N}_r$. *Let* $\tau := \inf\{k \geq 1 : \mathbf{w}_k \notin \mathcal{N}_r\}$.

For a fixed step size $\alpha_k = \overline{\alpha}$,

- *we have linear convergence (depending on* $\uparrow \hat{\mu}_{\mathrm{PL}}$*)*
- *and convergence up until a noise floor, which depends on a tradeoff between*
  - $\downarrow \hat{L}/\hat{\mu}_{\mathrm{PL}}$ *and*
  - $\downarrow K$.

For a harmonic step size $\alpha_k \propto 1/k$

- *we have* $\mathcal{O}(\nu/k)$ *convergence, where* ν *depends on* $\downarrow \hat{L}/\hat{\mu}_{\mathrm{PL}}$.

We stay in the basin w.h.p. depending on $1 - \mathcal{O}\left(\hat{L}K\right) - \sum_k \delta_k$, *where* δ_k *is the "escape probability".*

Table of Contents

Diagnostic quadratic model allows for explicit control of preconditioned spectra.

Example (Quadratic model)

Let $\mathbf{w}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \mathbf{w}_* = \mathbf{0}, F_* = 0$ for the quadratic model:

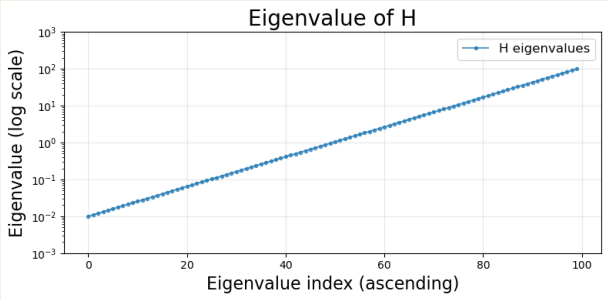
$$F(\mathbf{w}) = \frac{1}{2}(\mathbf{w} - \mathbf{w}_*)^\top \mathbf{H}(\mathbf{w} - \mathbf{w}_*),$$

where $\mathbf{H} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^\top$ has log-uniform spectrum from $[10^{-2}, 10^2]$, namely $\mathbf{H} \succ 0$.

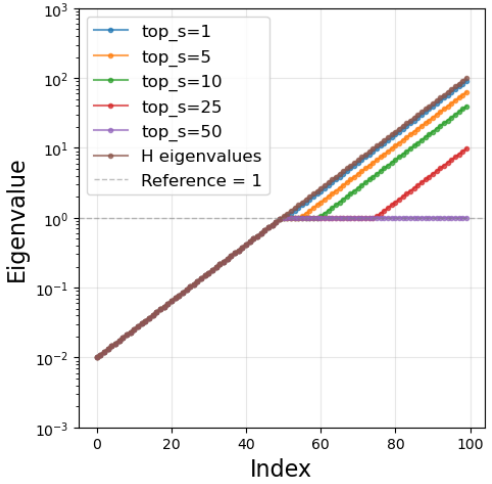
We add stochastic noise, so

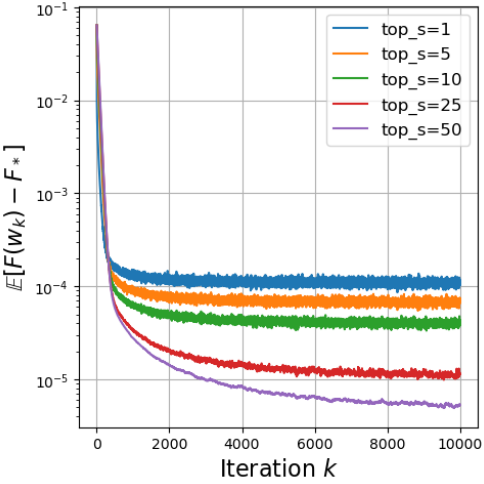
$$\nabla F(\mathbf{w}) = g(\mathbf{w}) + \boldsymbol{\epsilon}, \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$$

and run (preconditioned) SGD.

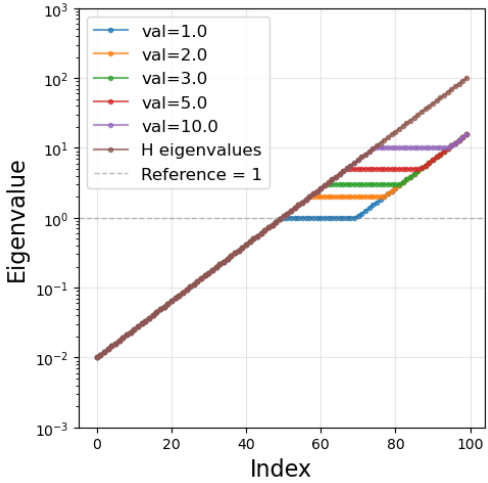


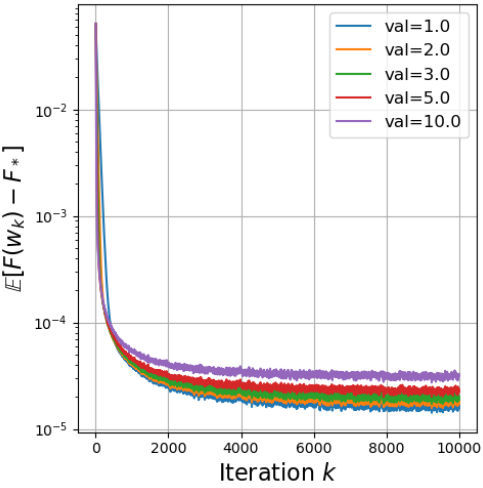
Deflation preconditioners lead to faster convergence and lower loss floor.





Generic deflation preconditioners demonstrate that a lower trace leads to a lower loss floor.

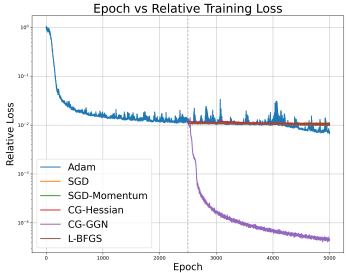


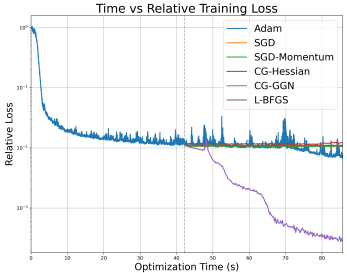


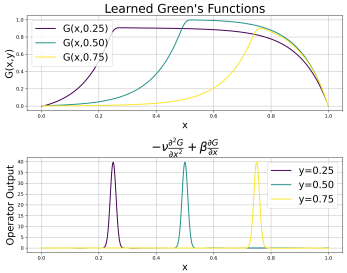
Convection dominated problems are hard, and Green's function learning is no different.

Example (Convection-dominated problem from Hao et al., [8])

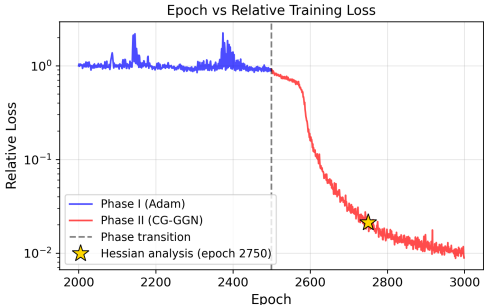
Find $G(x, y)$ such that $\mathcal{L}G = \delta(x - y)$ where $\mathcal{L}u := -0.1u'' + 1.0u', \quad u(0) = u(1) = 0$







Preconditioning empirically clusters the eigenvalues for SciML's Hessian.



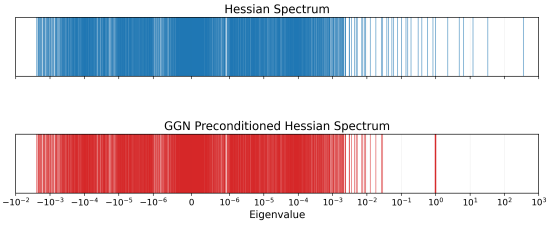


Table of Contents

Conclusion and Future Work

Conclusion:

Remark

A well-designed preconditioner $\mathbf{M}$ *does the following:*

Conclusion and Future Work

Conclusion:

Remark

A well-designed preconditioner $\mathbf{M}$ *does the following:*

1. *enhancing local conditioning—decreasing* $\hat{L}/\hat{c}$ *(*$\hat{L}/\hat{\mu}_{\mathrm{PL}}$ *for nonconvexity)* $\rightarrow$ *speeds up contraction,*

Conclusion and Future Work

Conclusion:

Remark

A well-designed preconditioner $\mathbf{M}$ *does the following:*

1. *enhancing local conditioning—decreasing* $\hat{L}/\hat{c}$ *(*$\hat{L}/\hat{\mu}_{\mathrm{PL}}$ *for nonconvexity)* $\rightarrow$ *speeds up contraction,*
2. *reducing the preconditioned noise level* K*, lowering the effective noise floor, and*

Conclusion and Future Work

Conclusion:

Remark

A well-designed preconditioner $\mathbf{M}$ *does the following:*

1. *enhancing local conditioning—decreasing* $\hat{L}/\hat{c}$ *(*$\hat{L}/\hat{\mu}_{\mathrm{PL}}$ *for nonconvexity)* $\rightarrow$ *speeds up contraction,*
2. *reducing the preconditioned noise level* K*, lowering the effective noise floor, and*
3. *increasing* α_{QG} *and permitting a larger basin radius* r*, which jointly improves basin stability.*

Conclusion and Future Work

Conclusion:

Remark

A well-designed preconditioner $\mathbf{M}$ *does the following:*

1. *enhancing local conditioning—decreasing* $\hat{L}/\hat{c}$ *(*$\hat{L}/\hat{\mu}_{\mathrm{PL}}$ *for nonconvexity)* $\rightarrow$ *speeds up contraction,*
2. *reducing the preconditioned noise level* K*, lowering the effective noise floor, and*
3. *increasing* α_{QG} *and permitting a larger basin radius* r*, which jointly improves basin stability.*

Future Work: Generalization of Preconditioners

- This work was based on late stage convergence for Scientific Machine Learning, where generalization is straightforward.

Conclusion and Future Work

Conclusion:

Remark

A well-designed preconditioner $\mathbf{M}$ *does the following:*

1. *enhancing local conditioning—decreasing* $\hat{L}/\hat{c}$ *(*$\hat{L}/\hat{\mu}_{\mathrm{PL}}$ *for nonconvexity)* $\rightarrow$ *speeds up contraction,*
2. *reducing the preconditioned noise level* K*, lowering the effective noise floor, and*
3. *increasing* α_{QG} *and permitting a larger basin radius* r*, which jointly improves basin stability.*

Future Work: Generalization of Preconditioners

- This work was based on late stage convergence for Scientific Machine Learning, where generalization is straightforward.
- There are other machine learning frameworks such as LLMs, which have different goals for generalization.

Conclusion and Future Work

Conclusion:

Remark

A well-designed preconditioner $\mathbf{M}$ *does the following:*

1. *enhancing local conditioning—decreasing* $\hat{L}/\hat{c}$ *(*$\hat{L}/\hat{\mu}_{\mathrm{PL}}$ *for nonconvexity)* $\rightarrow$ *speeds up contraction,*
2. *reducing the preconditioned noise level* K*, lowering the effective noise floor, and*
3. *increasing* α_{QG} *and permitting a larger basin radius* r*, which jointly improves basin stability.*

Future Work: Generalization of Preconditioners

- This work was based on late stage convergence for Scientific Machine Learning, where generalization is straightforward.
- There are other machine learning frameworks such as LLMs, which have different goals for generalization.
- We aim to have a parameter *a priori* that informs the generalization in terms of a regret bound.

Questions?

Thank You!

Read the Paper!

References

[1] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. Jan. 29, 2017. DOI: 10.48550/arXiv.1412.6980. arXiv: 1412.6980[cs]. URL: http://arxiv.org/abs/1412.6980 (visited on 09/13/2024).

[2] Shunichi Amari. "Natural Gradient Works Efficiently in Learning". In: *Neural Computation* 10.2 (Feb. 1998), pp. 251–276. ISSN: 0899-7667. DOI: 10.1162/089976698300017746. URL: https://doi.org/10.1162/089976698300017746 (visited on 03/23/2026).

[3] Herbert Robbins and Sutton Monro. "A Stochastic Approximation Method". In: *The Annals of Mathematical Statistics* 22.3 (1951). Publisher: Institute of Mathematical Statistics, pp. 400–407. ISSN: 0003-4851, 2168-8990. DOI: 10.1214/aoms/1177729586. URL: https://projecteuclid.org/journals/annals-of-mathematical-statistics/volume-22/issue-3/A-Stochastic-Approximation-Method/10.1214/aoms/1177729586.full (visited on 10/30/2024).

[4] Léon Bottou, Frank E. Curtis, and Jorge Nocedal. "Optimization Methods for Large-Scale Machine Learning". In: *SIAM Review* 60.2 (2018), pp. 223–311. DOI: 10.1137/16M1080173. eprint: https://doi.org/10.1137/16M1080173. URL: https://doi.org/10.1137/16M1080173.

EMORY UNIVERSITY

References (cont.)

[5] Mitchell T Scott et al. "Design Criteria for SGD Preconditioners: Local Conditioning, Noise Floors, and Basin Stability". In: *Transactions of Machine Learning Research* (2026).

[6] Guillaume Garrigos and Robert M. Gower. *Handbook of Convergence Theorems for (Stochastic) Gradient Methods*. Mar. 9, 2024. DOI: 10.48550/arXiv.2301.11235. arXiv: 2301.11235[math]. URL: http://arxiv.org/abs/2301.11235 (visited on 04/15/2025).

[7] Nicolas Boumal, Christopher Criscitiello, and Quentin Rebjock. *Smooth, globally Polyak-Łojasiewicz functions are nonlinear least-squares*. 2026. arXiv: 2604.07972 [math.OC]. URL: https://arxiv.org/abs/2604.07972.

[8] Wenrui Hao et al. *Multiscale Neural Networks for Approximating Green's Functions*. 2024. arXiv: 2410.18439 [math.NA]. URL: https://arxiv.org/abs/2410.18439.

[9] Richard Franke. *A Critical Comparison of Some Methods for Interpolation of Scattered Data*. Tech. rep. Graduate School of Operational and Information Sciences (GSOIS), 1979. URL: https://apps.dtic.mil/sti/citations/ADA081688 (visited on 05/10/2025).