Muon Does Not Converge on Convex Lipschitz Functions

Tetiana Parshakova

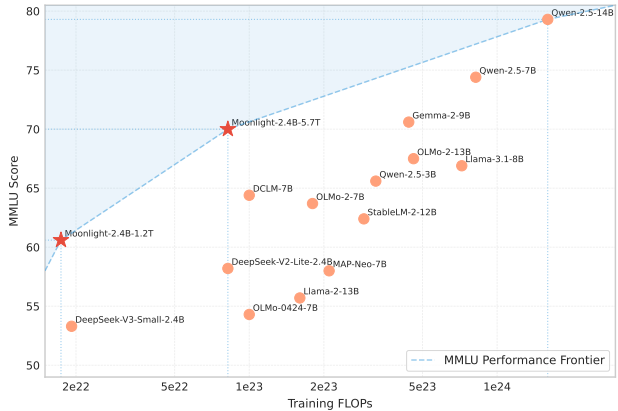
joint work with Ahmed Khaled, Guillaume Garrigos, Michael Crawshaw, Robert Gower

ISL, Stanford

04/30/2026

Muon a new optimizer for deep learning

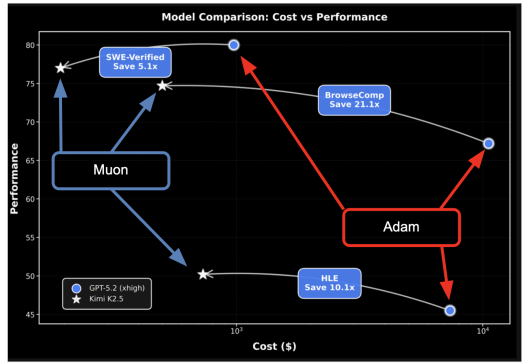
- MMLU score
  - 57 multiple choice exams
  - math, history, law, CS, etc
- Muon advances Pareto frontier of performance vs training FLOPs
  - Moonlight trained with Muon
- 1T parameters Kimi model [Kimi team, 2025]
- 1.6T parameters Deepseek-V4-Pro [Deepseek-AI, 2026]



[Liu et al. (2025)]

Muon a new optimizer for deep learning

- three agentic benchmarks
  - HLE (humanity last exam): 3000 expert-level questions in advanced subjects
  - BrowseComp: 1266 reasoning questions using browser
  - SWE-Verified: 500 github issues with unit tests



[Kimi K2.5, 2026]

Motivation

- Muon [Jordan et al., 2024]
  - consistently outperforms Adam on deep learning tasks
  - existing convergence theory requires **smoothness**
- convex and Lipschitz function class
  - led to development of AdaGrad, RMSprop, Adam and Shampoo
  - predicts deep learning dynamics surprisingly well

- **what about Muon on convex Lipschitz functions?**

Outline

Outline

Non-Euclidean subgradient methods

optimization problem

$$\underset{W \in \mathbb{R}^{m \times n}}{\text{minimize}} \quad f(W)$$

- regularized subgradient descent in norm $\|\cdot\|$

$$\begin{aligned} G_t &\in \partial f(W_t) \\ W_{t+1} &\in \underset{W}{\operatorname{argmin}} \left(f(W_t) + \langle G_t, W - W_t \rangle + \frac{1}{2\lambda_t} \|W - W_t\|^2 \right) \end{aligned}$$

- convex subproblem since $X \mapsto \|X\|^p$ is convex for $p \geq 1$ and any norm $\|\cdot\|$

Example: Euclidean subgradient descent

- **which norm over matrices should we use?**

- Frobenius norm yields Euclidean subgradient descent

$$\begin{aligned} G_t &\in \partial f(W_t) \\ W_{t+1} &= \operatorname*{argmin}_W \left(f(W_t) + \langle G_t, W - W_t \rangle + \frac{1}{2\lambda_t} \|W - W_t\|_F^2 \right) \\ &= W_t - \lambda_t G_t \end{aligned}$$

Matrix parameters are linear maps

- **which norm over matrices should we use?**

- x is current token embedding and W is linear layer weights
- $x' = Wx$ output token embedding
- stable training requires controlling scale of activations $\|x'\|_2 = O(1)$

$$\|x'\|_2 = \|Wx\|_2 \leq \|W\|_{\text{op}} \|x\|_2, \qquad \|W\|_{\text{op}} = \sup_{\|x\|_2=1} \|Wx\|_2$$

- control spectral norm $\|W\|_{\text{op}}$ and $\|\Delta W\|_{\text{op}}$ for feature learning [Yang et al., 2024]

Spectral descent

- $G_t \in \partial f(W_t)$
- regularized spectral descent:

$$W_{t+1} \in \operatorname*{argmin}_W \left(f(W_t) + \langle G_t, W - W_t \rangle + \frac{1}{2\tilde{\lambda}_t} \|W - W_t\|_{\mathrm{op}}^2 \right)$$

- constrained spectral descent:

$$W_{t+1} \in \operatorname*{argmin}_{W \in \mathbb{R}^{m \times n}} \left(f(W_t) + \langle G_t, W - W_t \rangle \right) \quad \text{subject to} \quad \|W - W_t\|_{\mathrm{op}} \leq \tilde{\lambda}_t$$

- $W_{t+1} = W_t - \lambda_t \operatorname{polar}(G_t)$
- $\operatorname{polar}(G_t) = U_t V_t^\top$, where $G_t = U_t \Sigma_t V_t^\top$ is reduced SVD

Muon is spectral descent with momentum

Muon = spectral descent + momentum

$$\begin{aligned} G_t &\in \partial f(W_t) \\ M_t &= \beta M_{t-1} + (1-\beta) G_t \\ W_{t+1} &= W_t - \lambda_t \operatorname{polar}(M_t) \end{aligned}$$

Muon on diagonal matrices = signed momentum (signum)

Theorem (Informal). If function $f : \mathbb{R}^{m\times n} \to \mathbb{R}$ depends only on the (main) diagonal entries of its argument, then Muon reduces to signed momentum.

- polar factor of a (rectangular) diagonal matrix $A = \mathrm{diag}(a) \in \mathbb{R}^{m\times n}$ for some $a \in \mathbb{R}^{\min\{m,n\}}$ is entrywise sign_0

$$\mathrm{polar}(A) = \mathrm{sign}_0(A) = \mathrm{diag}(\mathrm{sign}_0(a))$$

- single-valued sign operator: $\mathrm{sign}_0 = \begin{cases} +1 & x > 0 \\ 0 & x = 0 \\ -1 & x < 0 \end{cases}$

Muon on diagonal matrices = signed momentum (signum)

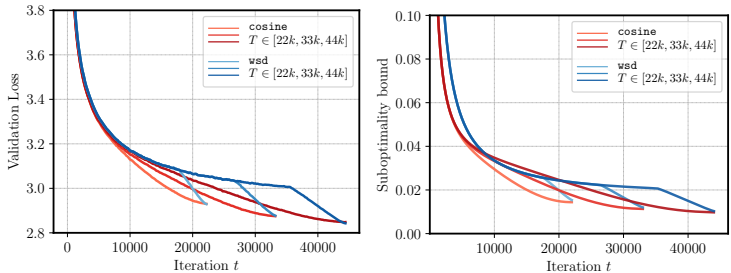
- run Muon on $f : \mathbb{R}^{m\times n} \to \mathbb{R}$ that depends on diagonal entries
  - subgradients $G_t \in \partial f(W_t)$ are diagonal matrices
  - diagonal entries evolve as signed momentum on $\mathbb{R}^{\min\{m,n\}}$
  - off-diagonal entries are frozen

$$\begin{array}{lcl} G_t & \in & \partial f(W_t) \\ M_t & = & \beta M_{t-1} + (1-\beta) G_t \\ W_{t+1} & = & W_t - \lambda_t \operatorname{polar}(M_t) \end{array} \qquad \Longrightarrow \qquad \begin{array}{lcl} g_t & = & \operatorname{diag}(G_t) \\ m_t & = & \beta m_{t-1} + (1-\beta) g_t \\ w_{t+1} & = & w_t - \lambda_t \operatorname{sign}_0(m_t) \end{array}$$

Outline

Convex Lipschitz in deep learning optimization

- fundamental function class for deep learning optimization
  - development of AdaGrad [Duchi et al., 2011], RMSprop [Tieleman and Hinton, 2012], and Adam [Kingma and Ba, 2014] and Shampoo [Gupta et al., 2018]
- predicts deep learning dynamics surprisingly well [Schaipp et al., 2024]
  - 210M Llama model trained with AdamW
  - normalized convex problem scale—not calibrated for loss scale

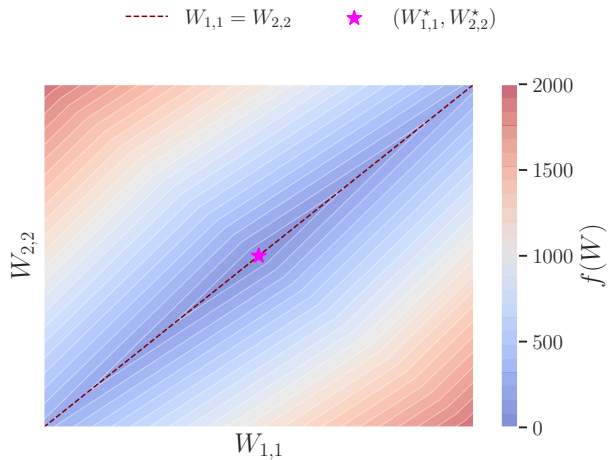


The counterexample function

- for $c \in (0, 1)$, let

$$f(W) = c|W_{11}+W_{22}|+|W_{11}-W_{22}|$$

- convex and Lipschitz
- minimizer: $W_{11}^\star = W_{22}^\star = 0$
- only depends on two diagonal entries
- adaptated from [Karimireddy et al., 2019]



Counterexample: offline decreasing stepsizes

Theorem. Let $\beta \in [0, 1)$ and (offline) strictly decreasing $\{\lambda_t\}_{t=0}^{\infty}$ with $\lambda_t \to 0$. Let f be the counterexample function with $c = \frac{1-\beta}{2}$. Starting from $M_{-1} = 0$, there exists W_0 such that

$$f(W_t) - \inf f \;\geq\; 1 - \beta, \qquad t = 0, 1, \ldots.$$

Further for $W^\star = 0$,

$$\|W_t - W^\star\|_F \geq \frac{1}{\sqrt{2(1+c^2)}}(f(W_t) - \inf f) \geq \frac{1-\beta}{\sqrt{2(1+c^2)}}.$$

- rules out global convergence

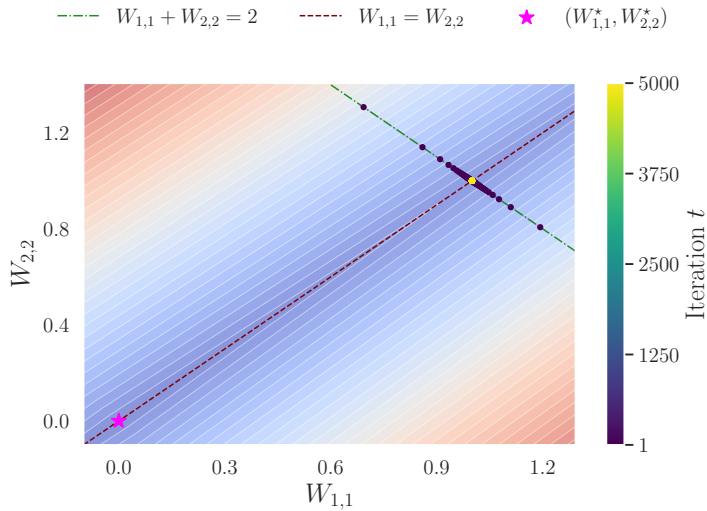
Counterexample: adaptive stepsizes

Theorem. Let f be the counterexample function with $c \in (0, 1)$, and $\beta \in [0, \frac{1-c}{2})$. Stepsize λ_t may depend on $G_0, \ldots, G_{t-1}$. Then almost surely over initialization W_0, the Muon iterates cannot converge to a minimizer of f.

- covers all $\beta \in [0, \frac{1}{2})$
- rules out local convergence of constrained Muon and regularized Muon

Muon cycling on counterexample

- $\beta = 0.9$
- $\lambda_t = \frac{1}{t+1}$



Outline

Signed gradient descent with error feedback

- signSGD does not converge on convex Lipschitz [Karimireddy et al., 2019]
- trick is to view sign as a compression operator
- incorporate error-feedback (EF) [Seide et al., 2014] in signSGD
  - EF accumulates information lost during compression
  - adds if back into next step—every gradient update is eventually used
- **can we lift EF to fix any non-Euclidean subgradient descent with momentum?**

Polar factor step is a compressor

Definition. (δ-compressor). Let $\delta \in (0, 1]$, $\mathcal{C} : \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n}$ is a δ-compressor if

$$\|\mathcal{C}(W) - W\|_F^2 \le (1-\delta)\|W\|_F^2, \quad W \in \mathbb{R}^{m\times n}.$$

- polar factor for $W \in \mathbb{R}^{m\times n}$

$$\mathcal{C}(W) = \frac{\|W\|_{\text{nuc}}}{\min\{m,n\}} \operatorname{polar}(W), \qquad \delta = \frac{1}{\min\{m,n\}}$$

- sign for $w \in \mathbb{R}^d$

$$\mathcal{C}(w) = \frac{\|w\|_1}{d} \operatorname{sign}(w), \qquad \delta = \frac{1}{d}$$

Error feedback with momentum (EF-M)

Muon

For $t = 0, 1, \ldots, T$:

1. sample stochastic subgradient G_t
2. $M_t = \beta M_{t-1} + (1-\beta) G_t$ [momentum]
3. $\Delta_t = \lambda_t \mathcal{C}(M_t)$ [compress]
4. $W_{t+1} = W_t - \Delta_t$ [update]

EF-M

For $t = 0, 1, \ldots, T$:

1. sample stochastic subgradient G_t
2. $M_t = \beta M_{t-1} + (1-\beta) G_t$ [momentum]
3. $P_t = E_t + \lambda_t M_t$ [add error]
4. $\Delta_t = \mathcal{C}(P_t)$ [compress]
5. $W_{t+1} = W_t - \Delta_t$ [update]
6. $E_{t+1} = P_t - \Delta_t$ [save error]

Convergence of EF-M

Theorem. Let f be convex with minimizer $W^\star$. Suppose $\mathbf{E}_t[G_t] \in \partial f(W_t)$ and $\mathbf{E}_t\left[\|G_t\|_F^2\right] \le \sigma^2$. Let $\mathcal{C}$ be a δ-compressor. Then

$$\mathbf{E}\left[f(\bar{W}_T)\right] - f(W^\star) \;\le\; \frac{\|W_0 - W^\star\|_F^2}{2\lambda(T+1)} + \lambda\sigma^2\left(\frac{1}{2} + \frac{2\sqrt{1-\delta}}{\delta} + \frac{\beta}{1-\beta}\right).$$

Choosing $\lambda = \Theta(1/\sqrt{T+1})$ gives $\mathbf{E}\left[f(\bar{W}_T)\right] - f(W^\star) = O(1/\sqrt{T+1})$.

- rate matches standard stochastic subgradient method
- holds for any compressor

Decaying stepsize variant

Decaying stepsizes $\lambda_t = 1/\sqrt{t+1}$:

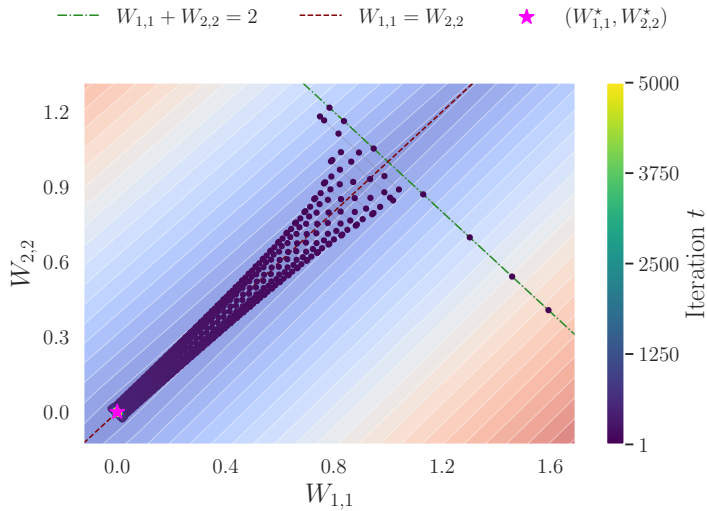
$$\mathbf{E}\big[f(\bar{W}_T)\big] - f(W^\star) \;\le\; \frac{\|W_0 - W^\star\|_F^2}{2\sqrt{T+1}} + \sigma^2\left(\frac{1}{2} + \frac{2\sqrt{1-\delta}}{\delta} + \frac{\beta}{1-\beta}\right)\frac{1+\ln(T+1)}{\sqrt{T+1}}$$

- $O(\log T/\sqrt{T})$ — near-optimal for the nonsmooth convex class
- anytime convergence

Outline

EF-Muon on counterexample

- $\beta = 0.9$
- $\lambda_t = \frac{1}{\sqrt{t+1}}$



Product space of matrices over different layers

- weights $W = (W^1, \dots, W^L, \theta)$, where $W_\ell \in \mathbb{R}^{m_\ell \times n_\ell}$, $\theta \in \mathbb{R}^d$
- norm that recovers separable compression operator across the layers

$$\|W\| := \left(\sum_{\ell=1}^{L} d_\ell \|W^\ell\|_{\text{op}}^2 + d\|\theta\|_\infty^2 \right)^{1/2}, \qquad d_\ell = \min\{m_\ell, n_\ell\}$$

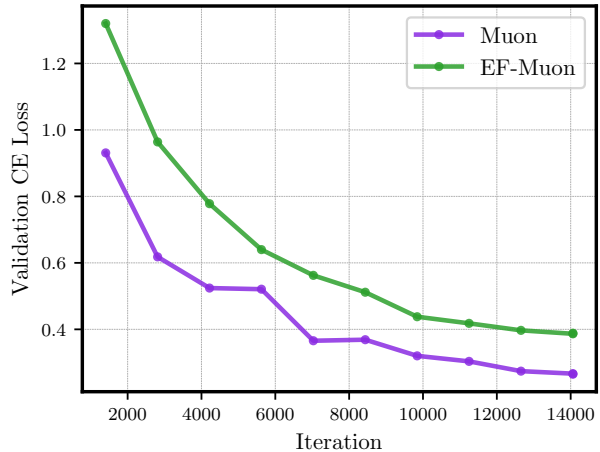
- EF-Muon update: $P_t = (P_t^1, \dots, P_t^L, p_t^\theta)$

$$\mathcal{C}(P_t) = \left(\frac{\|P_t^1\|_{\text{nuc}}}{d_1} \operatorname{polar}(P_t^1), \dots, \frac{\|P_t^L\|_{\text{nuc}}}{d_L} \operatorname{polar}(P_t^L), \frac{\|p_t^\theta\|_1}{d} \operatorname{sign}_0(\theta) \right)$$

$$\operatorname{polar}(P_t) = \frac{1}{\|P_t\|_{\text{nuc}}} \left(\frac{\|P_t^1\|_{\text{nuc}}}{d_1} \operatorname{polar}(P_t^1), \dots, \frac{\|P_t^L\|_{\text{nuc}}}{d_L} \operatorname{polar}(P_t^L), \frac{\|p_t^\theta\|_1}{k} \operatorname{sign}_0(p_t^\theta) \right)$$

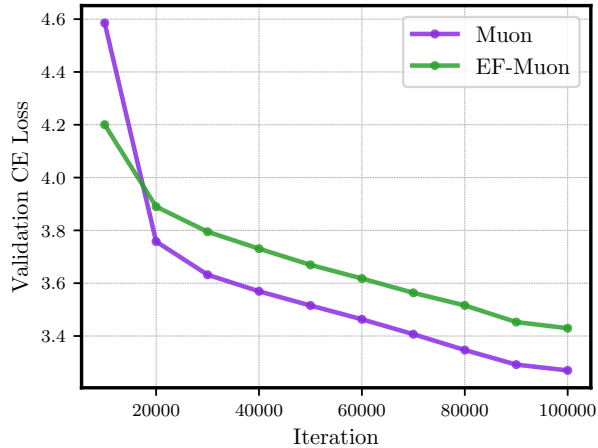
CIFAR-10 image classification

- WideResNet-28-10
  - depth 28, width multiplier 10, 4 blocks
  - 36M parameters
- CIFAR-10
- tuned over 60 trials
  - Signum pair optimizer
  - Muon lr $100\times$ larger
  - EF-Muon lr $30\times$ larger
  - cosine schedule without warmup



FineWeb language modeling

- GPT-2 small decoder-only Transformer
  - 12 layers
  - 206M parameters
- FineWeb-EDU 10B
  - sequence length 1024
  - batch size 64
- tuned over 40 trials
  - NAdamW pair optimizer
  - Muon lr $30\times$ larger
  - cosine schedule with 10% warmup



Questions

- should we expect this method to work well in practice?
  - EF makes the EF-Muon behave more like Euclidean subgradient descent with momentum
  - but Muon outperforms it in practice
  - spectral update is not just an error to be corrected; it is the useful bias of the method
- what does this story tell us?
  - convex Lipschitz is wrong model class for understanding Muon
  - why this departure from the Euclidean subgradient descent with momentum and adaptive methods?

Outline

Summary

- Muon = spectral subgradient + momentum
- Muon on diagonal matrices = signed descent + momentum

- can fail to converge on convex Lipschitz functions
- EF recovers convergence of Muon but degrades the practical performance
- convex Lipschitz is wrong model class for understanding Muon
  - while AdaGrad, Shampoo and ScheduleFree are meaningfully informed by convex Lipschitz theory

- why this departure from the Euclidean and adaptive methods?
- what is so special about smoothness that Muon requires smoothness?

Thanks!

Muon and Adam

- Adam [Kingma and Ba, 2015; Loshchilov and Hutter, 2019]

$$
\begin{aligned}
g_t &\in \partial f(w_t) && \text{(subgradient)} \\
m_t &= \beta_1 m_{t-1} + (1-\beta_1) g_t && \text{(first moment)} \\
v_t &= \beta_2 v_{t-1} + (1-\beta_2) g_t^2 && \text{(second moment)} \\
w_{t+1} &= w_t - \lambda_t m_t / (\sqrt{v_t} + \epsilon) && \text{(update)}
\end{aligned}
$$

- Muon [Jordan et al., 2024]

$$
\begin{aligned}
G_t &\in \partial f(W_t) && \text{(subgradient)} \\
M_t &= \beta M_{t-1} + (1-\beta)\, G_t && \text{(momentum)} \\
W_{t+1} &= W_t - \lambda_t \operatorname{polar}(M_t) && \text{(update)}
\end{aligned}
$$

Muon and Adam

- Adam without EMA is signed momentum

$$\begin{array}{lcl} g_t & \in & \partial f(w_t) \\ m_t & = & \beta_1 m_{t-1} + (1-\beta_1) g_t \\ v_t & = & \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \\ w_{t+1} & = & w_t - \lambda_t m_t / (\sqrt{v_t} + \epsilon) \end{array} \quad \Longrightarrow \quad \begin{array}{lcl} g_t & \in & \partial f(w_t) \\ w_{t+1} & = & w_t - \lambda_t g_t / \sqrt{g_t^2} \end{array}$$

- Muon is signed momentum on diagonal matrices

$$\begin{array}{lcl} G_t & \in & \partial f(W_t) \\ M_t & = & \beta M_{t-1} + (1-\beta) G_t \\ W_{t+1} & = & W_t - \lambda_t \operatorname{polar}(M_t) \end{array} \quad \Longrightarrow \quad \begin{array}{lcl} g_t & = & \operatorname{diag}(G_t) \\ m_t & = & \beta m_{t-1} + (1-\beta) g_t \\ w_{t+1} & = & w_t - \lambda_t \operatorname{sign}_0(m_t) \end{array}$$

Muon and one-sided Shampoo / ASGO

- one-sided Shampoo [Xie et al., 2025] and ASGO [An et al., 2025]

$$\begin{aligned} G_t &\in \partial f(W_t) \\ V_t &= \epsilon I_m + (\textstyle\sum_{s=0}^{t} G_s G_s^\top)^{\frac{1}{2}} \\ W_{t+1} &= W_t - \lambda_t V_t^{-1} G_t \end{aligned}$$

- Muon using $\mathrm{polar}(M_t) = (M_t M_t^\top)^{-1/2} M_t$

$$\begin{aligned} G_t &\in \partial f(W_t) \\ M_t &= \beta M_{t-1} + (1-\beta) G_t \\ W_{t+1} &= W_t - \tfrac{\lambda_t}{1-\beta} \left(\textstyle\sum_{s_1, s_2 = 1}^{t} \beta^{2t - s_1 - s_2} G_{s_1} G_{s_2}^\top \right)^{-\frac{1}{2}} M_t \end{aligned}$$

AdaGrad [Duchi et al., 2011]

- parameter $W_t \in \mathbb{R}^{m \times n}$, gradient G_t
- vectorized versions $w_t = \mathrm{vec}(W_t) \in \mathbb{R}^{mn}$ and $g_t = \mathrm{vec}(G_t) \in \mathbb{R}^{mn}$

$$H_t = \epsilon I_{mn} + \left(\sum_{s=0}^{t} g_s g_s^\top \right)^{\frac{1}{2}}, \qquad w_{t+1} = w_t - \lambda H_t^{-1} g_t$$

- online learning second-order algorithm that maintains a $mn \times mn$ preconditioner
- regret analysis via online convex optimization
- diagonal AdaGrad: replace dense $mn \times mn$ preconditioner by diagonal matrix

Shampoo [Gupta et al., 2018]

- parameter $W_t \in \mathbb{R}^{m \times n}$, gradient G_t

$$L_t = \epsilon I_m + \sum_{s=0}^{t} G_s G_s^\top, \qquad R_t = \epsilon I_n + \sum_{s=0}^{t} G_s^\top G_s$$

$$W_{t+1} = W_t - \lambda L_t^{-\frac{1}{4}} G_t R_t^{-\frac{1}{4}}$$

- approximates full-matrix AdaGrad [Duchi et al., 2011]
- convergence analysis using online convex optimization
- bounds $mn \times mn$ preconditioner by Kronecker product of $m \times m$ and $n \times n$ matrices
- Newton iteration for computing 4th inverse root

SOAP [Vyas et al., 2025]

- SOAP = Adam in Shampoo's eigenbasis
- build Shampoo factors and rotate the gradient

$$L_t = Q_t \Lambda_t Q_t^\top, \qquad R_t = P_t \Sigma_t P_t^\top, \qquad \widetilde{G}_t = Q_t^\top G_t P_t$$

- run Adam in the rotated coordinates

$$M_t = \beta_1 M_{t-1} + (1-\beta_1)\widetilde{G}_t, \qquad S_t = \beta_2 S_{t-1} + (1-\beta_2)\widetilde{G}_t^{\odot 2}$$

$$W_{t+1} = W_t - \eta\, Q_t \left(\frac{M_t}{\sqrt{S_t} + \epsilon} \right) P_t^\top$$

- update Q_t, P_t infrequently via power iteration + QR
- intuition: Shampoo finds the right coordinates; Adam chooses the step sizes there

Non-Euclidean subgradient methods

$$\underset{W \in \mathbb{R}^{m \times n}}{\text{minimize}} \quad f(W)$$

regularized subgradient descent in norm $\|\cdot\|$

$$\begin{aligned} G_t &\in \partial f(W_t) \\ W_{t+1} &\in \underset{W}{\operatorname{argmin}} \left(f(W_t) + \langle G_t, W - W_t \rangle + \frac{1}{2\lambda_t} \|W - W_t\|^2 \right) \end{aligned}$$

- $W_{t+1} = W_t - \lambda_t \|G_t\|_* \mathsf{LMO}_{\|\cdot\|}(G_t)$
- $\mathsf{LMO}_{\|\cdot\|}(G) \in \partial \|G\|_*$ is a linear maximization oracle, least norm element
- $\|G\|_* = \langle \mathsf{LMO}_{\|\cdot\|}(G), G \rangle$ is the dual norm

Non-Euclidean subgradient methods

$$\underset{W \in \mathbb{R}^{m \times n}}{\text{minimize}} \quad f(W)$$

constrained subgradient descent in norm $\|\cdot\|$

$$\begin{aligned} G_t &\in \partial f(W_t) \\ W_{t+1} &\in \underset{W \in \mathbb{R}^{m \times n}}{\operatorname{argmin}} \left(f(W_t) + \langle G_t, W - W_t \rangle\right) \quad \text{subject to} \quad \|W - W_t\| \leq \lambda_t \end{aligned}$$

- $W_{t+1} = W_t - \lambda_t \mathsf{LMO}_{\|\cdot\|}(G_t)$
- $\mathsf{LMO}_{\|\cdot\|}(G) \in \partial \|G\|_*$ is a linear maximization oracle, least norm element
- $\|G\|_* = \langle \mathsf{LMO}_{\|\cdot\|}(G), G \rangle$ is the dual norm

Polar factor on GPU

- $\mathrm{polar}(G) = UV^\top$ where $G = U\Sigma V^\top$ is reduced SVD

how to compute polar factor efficiently on GPU?

- NNs are trained in low precision `bfloat16`
- use (odd) matrix polynomials with $p(x) \approx 1$ for all $x \geq 1$

$$\mathrm{polar}(G) = UV^\top \approx p(G) = a_0 G + a_1 G(G^\top G) + \cdots + a_q G(G^\top G)^q = Up(\Sigma)V^\top$$

  - $a_0, \ldots, a_q$ are unknown coefficients
- PolarExpress [Amsel et al., 2025] optimal (min-max) composition of polynomials

EF-Muon proof sketch

$$\widetilde{W}_t := W_t - E_t, \qquad Y_t := \widetilde{W}_t - \frac{\beta}{1-\beta}\lambda M_t \quad \Longrightarrow \quad Y_{t+1} = Y_t - \lambda G_t$$

- let $S_t := \mathbb{E}_t[G_t] \in \partial f(W_t)$, then

$$\mathbb{E}_t\|Y_{t+1} - W^\star\|_F^2 \le \|Y_t - W^\star\|_F^2 - 2\lambda\langle S_t, W_t - W^\star\rangle + \text{perturbation terms}$$

- control perturbations by $O(\lambda^2\sigma^2)$

$$\mathbb{E}\|M_t\|_F^2 \le \sigma^2, \qquad \mathbb{E}\|E_t\|_F^2 \le \frac{4(1-\delta)}{\delta^2}\lambda^2\sigma^2$$

- using convexity, and suming over t

$$\mathbb{E}[f(\bar{W}_T)] - f(W^\star) \le \frac{\|W_0 - W^\star\|_F^2}{2\lambda(T+1)} + \lambda\sigma^2\left(\frac{1}{2} + 2\frac{\sqrt{1-\delta}}{\delta} + \frac{\beta}{1-\beta}\right)$$