Reassessing Muon for Matrix Factorization

Ali Parviz[†] ALPARVIZ@UCSD.EDU
Gal Mishne[†] GMISHNE@UCSD.EDU
Alex Cloninger[*†] ACLONINGER@UCSD.EDU

Abstract

`Muon` has recently emerged as a strong optimizer for large-scale deep learning, where it reshapes gradient updates through approximate orthogonalization and has been reported to outperform `Adam` and `AdamW` on large language model training. Its empirical success has motivated a growing theoretical literature that interprets `Muon` as steepest descent under the spectral norm. Yet it remains unclear which of `Muon`'s advantages stem from its update rule itself and which are artifacts of the scale, architecture, and data of modern deep networks. In this work we isolate the optimizer from these confounders by studying `Muon` on a simple, well-understood, and spectrally structured problem: low-rank matrix factorization. Through a controlled and systematically tuned comparison against adaptive baselines, we find that `Muon` does *not* consistently outperform `AdamW` in this setting, and that several previously reported advantages are sensitive to hyperparameter choices. Our results give a more nuanced picture of when spectrum-aware orthogonalization helps, and argue for evaluating modern optimizers on controlled problems in addition to end-to-end benchmarks.

1. Introduction

Recent advances in large-scale optimization have introduced a class of optimizers that explicitly exploit the matrix structure of gradient updates. Among these, `Muon` (MomentUm Orthogonalized by Newton–Schulz) has emerged as a promising alternative to standard first-order methods (Jordan, 2024). Rather than applying a momentum update directly, `Muon` transforms the update direction through an approximate orthogonalization step based on Newton–Schulz iterations, effectively reshaping the spectrum of the gradient.

Empirically, `Muon` has demonstrated strong performance on modern deep learning workloads, including GPT-style models, where it improves training efficiency and in some cases outperforms widely used optimizers such as `Adam` and `AdamW` (Jordan, 2024; Liu et al., 2025; Shah et al., 2025b). Notably, large-scale studies report up to a twofold speedup over `AdamW` in multi-billion-parameter language model training (Shah et al., 2025b). These successes have motivated a growing body of theoretical work that studies `Muon` through the lenses of spectral-norm steepest descent and constrained optimization. Existing analyses establish convergence guarantees under various simplifying assumptions, including exact orthogonalization, simplified momentum dynamics, and locally quadratic models (Pethick et al., 2025a; Li and Hong, 2025; Shen et al., 2025; Chen et al., 2025; Kovalev, 2025; Riabinin et al., 2025; Gruntkowska et al., 2025; Sato et al., 2025; Nagashima and Iiduka, 2026), while more recent

∗. Department of Mathematics, UC San Diego.
†. Halıcıoğlu Data Science Institute, UC San Diego.

work studies inexact Newton–Schulz iterations and how approximation errors propagate into convergence guarantees (Shulgin et al., 2025; Kim and Oh, 2026; Lau et al., 2025).

Despite this progress, it remains unclear which aspects of `Muon`'s advantage are intrinsic to the optimizer and which arise from the complexity of large-scale deep learning, where architectural, scale, and data-dependent effects are hard to disentangle. Existing analyses rarely show *when* and *why* `Muon` should outperform classical optimizers in concrete, well-specified problems, tending to concern idealized settings or purely local properties (Davis and Drusvyatskiy, 2025; Su, 2025). We take a complementary perspective and study `Muon` on low-rank matrix factorization: a simple yet fundamental problem that allows systematic, exhaustive evaluation and where `Muon`'s spectral nature might be expected to help, since optimization is governed by well-characterized curvature and singular-value structure. Our goal is to characterize both the strengths and limitations of `Muon` and to identify regimes where it does or does not outperform standard optimizers such as `Adam` and `AdamW`.

This focus exposes a gap between expectation and behavior, which we organize around three questions.

Q1: Does `Muon` offer any advantage in structured problems such as matrix factorization? Matrix factorization has explicit low-rank structure and is associated with a well-behaved optimization landscape. If `Muon` genuinely exploits spectral properties of the gradient, this should translate into faster convergence. If it does not, what limits its effectiveness relative to adaptive methods such as `AdamW`?

Q2: Are `Muon`'s benefits tied to specific problem structures? Which properties of an optimization problem make orthogonalized updates advantageous? Do their benefits stem from specific geometric or spectral features of the objective, or do they generalize across matrix factorization tasks with varying conditioning, over-parameterization, and constraints?

Q3: How does `Muon` compare to adaptive methods under ill-conditioning? Matrix factorization is sensitive to conditioning, especially when singular values decay rapidly. Optimizers such as `AdamW` implicitly adapt to coordinate-wise scaling. Does `Muon`'s spectral normalization compete with or conflict with such adaptivity, and under which regimes does one dominate?

Findings. Contrary to its strong performance in large-scale models, Muon's advantage in these canonical settings is problem-dependent. On low-rank factorization and matrix completion, its reported gains disappear under equal tuning, with AdamW and GD matching or exceeding its performance. On NMF, however, Muon retains a consistent advantage, likely because its orthogonalized updates discourage redundant factors and promote more diverse representations. This dependence on problem structure suggests that Muon's broader success may also arise from properties specific to deep learning and underscores the value of controlled optimization benchmarks.

2. Matrix factorization: symmetric and nonnegative variants

We begin by fixing notation used throughout the paper. For a matrix $\boldsymbol{M}$, let $\sigma_i(\boldsymbol{M})$ denote its i-th largest singular value and $\sigma_{\min}(\boldsymbol{M})$ its smallest. We write $\|\boldsymbol{M}\|_{\mathrm{op}}$ and $\|\boldsymbol{M}\|_{\mathrm{F}}$ for the spectral and Frobenius norms, respectively. For $k \le d$, let $\mathcal{O}_{d\times k}$ be the set of matrices in $\mathbb{R}^{d\times k}$ with orthonormal columns. Given scalars $a_1, \dots, a_d$, we write $\mathsf{diag}\{a_1, \dots, a_d\}$ for

the diagonal matrix with these entries. We discuss `Muon` (Jordan, 2024) in Section A and defer a detailed discussion of related work to Appendix C.

Symmetric matrix factorization We consider the symmetric matrix factorization problem

$$\min_{\boldsymbol{U}\in\mathbb{R}^{d\times k}} \quad f(\boldsymbol{U}) \;=\; \frac{1}{4}\left\|\boldsymbol{U}\boldsymbol{U}^\top - \boldsymbol{M}^\star\right\|_{\mathrm{F}}^2, \tag{1}$$

where $\boldsymbol{M}^\star \in \mathbb{R}^{d\times d}$ is a rank-r positive semidefinite matrix and $\boldsymbol{U} \in \mathbb{R}^{d\times k}$ is a factor with $k \geq r$ columns ($k = r$ is the exactly parameterized case and $k > r$ is over-parameterized). The goal is to recover $\boldsymbol{M}^\star$ through a low-rank factorization $\boldsymbol{U}\boldsymbol{U}^\top$ by solving equation 1. We assume $\boldsymbol{M}^\star$ admits the eigendecomposition $\boldsymbol{M}^\star = \boldsymbol{V}^\star\boldsymbol{\Lambda}^\star\boldsymbol{V}^{\star\top}$, where $\boldsymbol{\Lambda}^\star = \mathsf{diag}\{\lambda_1^\star, \ldots, \lambda_r^\star\}$ collects the nonzero eigenvalues with $\lambda_1^\star \geq \cdots \geq \lambda_r^\star > 0$ and $\boldsymbol{V}^\star \in \mathbb{R}^{d\times r}$ is orthonormal. The condition number of $\boldsymbol{M}^\star$ is

$$\kappa \;\coloneqq\; \lambda_1^\star/\lambda_r^\star. \tag{2}$$

The symmetric problem equation 1 is the positive semidefinite, factor-tied instance of a broader family of low-rank factorization problems; we defer the general bilinear factorization and the matrix completion variant to Appendix B.

Nonnegative matrix factorization. A closely related variant constrains both the target and the factors to be entrywise nonnegative. For a rank-r target $\boldsymbol{M}^\star \in \mathbb{R}^{d_1\times d_2}$, nonnegative matrix factorization (NMF) solves

$$\min_{\substack{\boldsymbol{U}\in\mathbb{R}_{\geq 0}^{d_1\times k}\\ \boldsymbol{V}\in\mathbb{R}_{\geq 0}^{d_2\times k}}} \quad \frac{1}{2}\left\|\boldsymbol{U}\boldsymbol{V}^\top - \boldsymbol{M}^\star\right\|_{\mathrm{F}}^2, \tag{3}$$

where $\mathbb{R}_{\geq 0}$ denotes the nonnegative reals and $k \geq r$. NMF is the nonnegativity-constrained special case of the general bilinear factorization equation 12 (Appendix B). Moreover, We analyze the exact softplus and its Taylor surrogate as nonlinear matrix factorizations and show the truncation imposes a rank cap that explains the high-rank gap in Fig. 11 (Appendix. J).

3. Experimental Design and Evaluation Protocol

We test the claim that optimizer comparisons in matrix factorization are highly sensitive to hyperparameters, so that conclusions drawn from a single (often default) configuration are not representative. Going beyond prior work (Ma et al., 2026), we evaluate five matrix-recovery problems under a controlled protocol that holds model capacity and parameterization fixed and varies only optimization-related hyperparameters, spanning well-conditioned to severely *ill-conditioned* targets. Two further analyses isolate *how* conditioning acts: the sensitivity of each optimizer to the *shape* of the eigenvalue distribution at a fixed, extreme condition number (Section 4.1), and the dynamics with which each recovers the target's spectral subspace (Section E).

3.1. Problem settings

All problems use ambient dimension $d = 100$ and a factorized parameterization $\hat{M} = UV^\top$ with $U, V \in \mathbb{R}^{d\times r}$, optimized from a small random initialization. We consider:

Table 1: Learning-rate-tuned final loss (geometric mean over 3 seeds; lower is better). The best optimizer per row is in **bold**. The tuned ranking varies across problems and, within each problem, differs from the ranking at a fixed default learning rate.

Problem	Setting	Muon	AdamW	GD	SignGD
Factorization	κ=1	2.3×10^{-8}	1.9×10^{-13}	$\mathbf{1.6\times10^{-13}}$	2.8×10^{-7}
	κ=5	3.0×10^{-7}	1.9×10^{-12}	$\mathbf{1.7\times10^{-12}}$	4.4×10^{-6}
	κ=25	3.0×10^{-6}	4.2×10^{-11}	$\mathbf{3.7\times10^{-11}}$	6.1×10^{-3}
	κ=125	4.1×10^{-7}	1.4×10^{-9}	$\mathbf{1.0\times10^{-9}}$	4.4×10^{-1}
	κ=625	9.4×10^{-5}	$\mathbf{2.6\times10^{-7}}$	1.3×10^{-1}	1.2×10^{1}
Completion (κ)	κ=1	$\mathbf{4.6\times10^{-16}}$	4.8×10^{-16}	4.5×10^{-11}	2.6×10^{-15}
	κ=5	5.7×10^{-15}	$\mathbf{5.5\times10^{-15}}$	1.6×10^{-9}	2.4×10^{-13}
	κ=25	$\mathbf{1.4\times10^{-13}}$	1.4×10^{-13}	4.7×10^{-7}	3.5×10^{-5}
	κ=125	3.3×10^{-7}	$\mathbf{1.7\times10^{-7}}$	1.8×10^{-2}	2.2×10^{-2}
	κ=625	$\mathbf{9.8\times10^{-3}}$	2.0×10^{-2}	3.4×10^{-2}	3.6×10^{-2}
Completion (rank)	r=4	5.7×10^{-15}	$\mathbf{5.5\times10^{-15}}$	1.6×10^{-9}	2.4×10^{-13}
	r=5	$\mathbf{4.5\times10^{-15}}$	1.8×10^{-12}	2.1×10^{-6}	4.9×10^{-8}
	r=100	2.1×10^{-15}	$\mathbf{1.5\times10^{-15}}$	1.8×10^{-14}	3.2×10^{-15}
NMF (uniform)	r=10	$\mathbf{1.5\times10^{-9}}$	6.1×10^{-3}	4.8×10^{-1}	8.7×10^{-1}
	r=50	4.5×10^{-9}	$\mathbf{4.1\times10^{-9}}$	4.1×10^{0}	1.3×10^{2}
	r=100	$\mathbf{8.9\times10^{-6}}$	7.8×10^{-3}	1.2×10^{2}	1.2×10^{3}
NMF (decay)	r=10	$\mathbf{5.1\times10^{-7}}$	2.3×10^{-3}	3.8×10^{-2}	1.2×10^{-1}
	r=50	$\mathbf{3.8\times10^{-9}}$	8.4×10^{-4}	1.0×10^{0}	1.5×10^{1}
	r=100	$\mathbf{1.2\times10^{-6}}$	2.1×10^{-3}	6.3×10^{0}	3.4×10^{2}

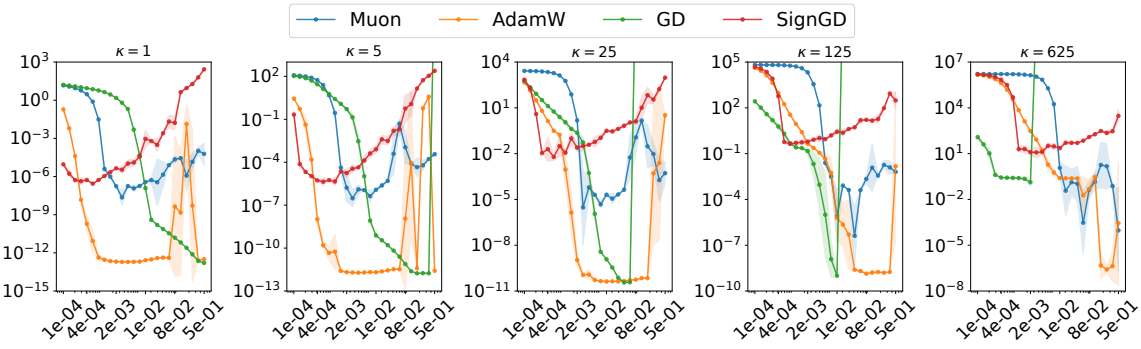

Figure 1: **Low-rank factorization, conditioning sweep** (target rank r=15, matched search rank k=15, dimension d=100). Tuned final loss vs. learning rate for each condition number κ, from well-conditioned (κ=1) to strongly ill-conditioned (κ=625) (geometric mean over 3 seeds, ±log-std band). The target's 15 singular values are spaced linearly from κ down to 1, so κ is exactly the condition number. When tuned, AdamW and GD reach near machine precision; the Muon configuration plateaus several orders higher.

- **Low-rank factorization (conditioning).** A fully observed symmetric PSD target $M^\star = U^\star \mathrm{diag}(s) U^{\star\top} \in \mathbb{R}^{d\times d}$ of rank $r = 15$, where $U^\star \in \mathbb{R}^{d\times r}$ has orthonormal

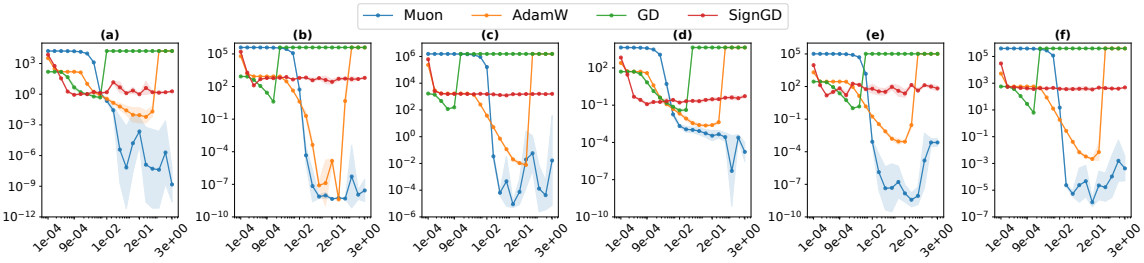

Figure 2: **Non-negative factorization.** Final loss (y-axis, log scale; geometric mean over 3 seeds, shaded ±one-log-standard-deviation band) versus learning rate (x-axis, log scale). Panels **(a–c)** use the uniform spectrum and **(d–f)** the decayed spectrum, at factor ranks $r = 10, 50, 100$ respectively. In both spectra, tuned Muon is the only method to fit the target across ranks; AdamW trails by several orders and GD/SignGD fail.

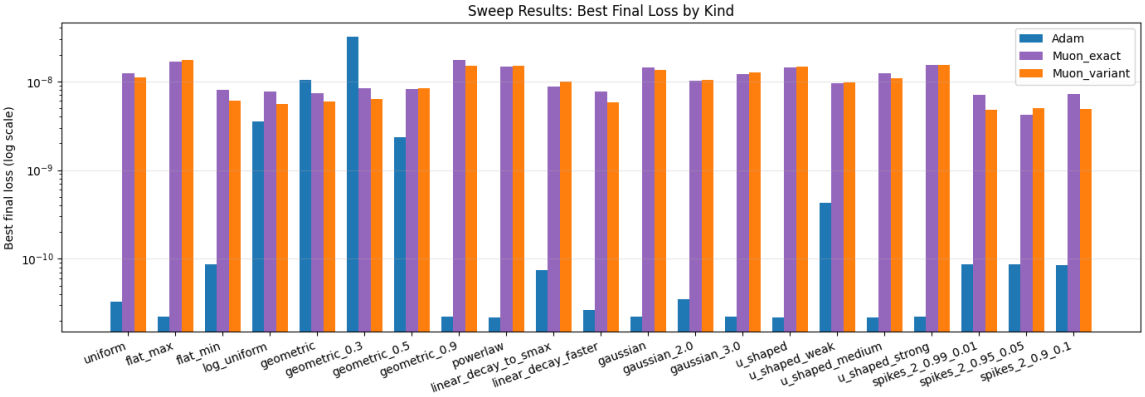

Figure 3: **AdamW outperforms Muon variants across spectrum shapes** (matrix factorization, target $m \times n = 100 \times 100$, rank r=5). Final loss after LR tuning for each of the spectral profiles at fixed $\kappa = 10^4$; for every (profile, optimizer) pair we report the best final loss over a logarithmic grid of learning rates, using a common target instance and a shared initialization. `Muon_exact` (exact polar factor via SVD, no momentum) and `Muon_variant` (Newton–Schulz orthogonalization with Nesterov momentum, $\mu = 0.95$) exhibit degraded relative performance on most spectral shapes, whereas AdamW remains effective despite the extreme ill-conditioning.

columns (the thin Q-factor of a matrix with i.i.d. $\mathcal{N}(0,1)$ entries) and $s = (s_1, \ldots, s_r)$ collects the nonzero eigenvalues of $M^\star$, spaced linearly between $s_1 = \kappa$ and $s_r = 1$:

$$s_i \;=\; \kappa - \tfrac{i-1}{r-1}\,(\kappa - 1), \qquad i = 1, \ldots, r.$$

Since $U^{\star\top} U^\star = I_r$, the condition number of $M^\star$ restricted to its range is exactly $s_1/s_r = \kappa$, and we sweep $\kappa \in \{1, 5, 25, 125, 625\}$ (at $\kappa = 1$ the spectrum is flat and $M^\star$ is a rank-r orthogonal projector). We solve for the symmetric factor $U \in \mathbb{R}^{d \times r}$ at matched search rank, minimizing $\frac{1}{4}\|UU^\top - M^\star\|_F^2$ (Figure 1).

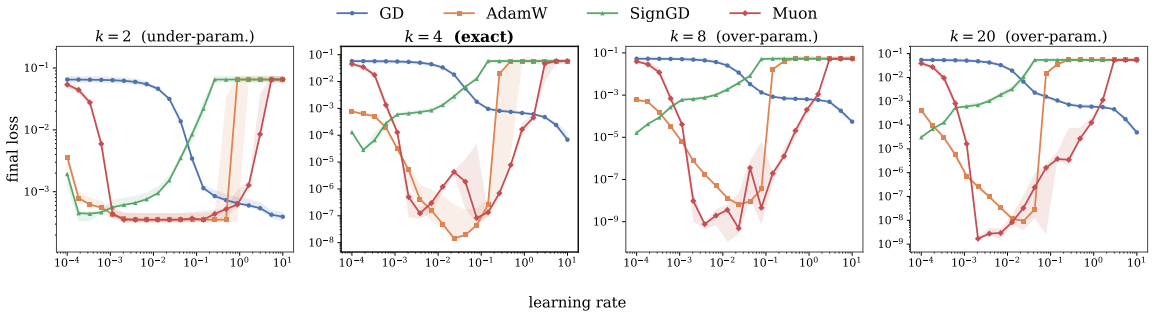

Figure 4: **Tensor-Train Noiseless regime** (true rank $r^\star = 4$). Final loss versus learning rate for each optimizer, across search ranks $r \in \{2, 4, 8, 20\}$ spanning under-, exactly- ($r = r^\star$, bold panel), and over-parameterized settings. Solid lines are the median over 3 seeds; shaded bands the min–max.

- **Matrix completion (conditioning).** A rank-4 symmetric PSD target with the same conditioning sweep, observed on a uniformly random 20% of entries; loss $\frac{1}{2}\|\mathcal{P}_\Omega(UV^\top - M^\star)\|_F^2$, the squared error on the observed support (Figure 7).
- **Matrix completion (over-parameterization).** The same completion task at fixed $\kappa = 5$ and true rank 4, sweeping the search rank $k \in \{4, 5, 100\}$ (matched, mildly, and heavily over-parameterized) (Figure 8).
- **Non-negative factorization.** A target $M^\star$ with non-negative entries, factorized at rank $k \in \{10, 50, 100\}$ subject to an *entrywise* non-negativity constraint on the factors, $U \in \mathbb{R}^{m\times k}_{\geq 0}$ (i.e. $U_{ij} \geq 0$ for all i, j; this is the componentwise order, not the positive-semidefinite one). The constraint is enforced by projected gradient descent: each optimizer step is followed by the Euclidean projection onto the non-negative orthant, $\Pi_{\geq 0}(X) = \max(X, 0)$, applied entrywise. We consider two target spectra: a *uniform* spectrum and a *decayed* (exponentially graded) spectrum, the latter removing the flat "DC-component" degeneracy of the uniform case, in which the leading non-negative component is nearly constant and all remaining directions carry equal weight (Figure 2).
- **Tensor-train factorization (over-parameterization, noise).** A symmetric positive semidefinite target $M^\star = U^\star U^{\star\top}$ recovered with untied cores U, V in a noiseless ($r^\star = 4$) and a noisy ($r^\star = 30$) regime, sweeping the search rank across under-, exactly-, and over-parameterized values. Objective $\frac{1}{2}\|UV^\top - M^\star\|_F^2$; see Section 4.3 and Figures 4 and 9.

We compare **Muon** (momentum 0.95, Nesterov, 5 Newton–Schulz orthogonalization steps) against **AdamW**, plain gradient descent (**GD**), and sign gradient descent (**SignGD**).

3.2. Evaluation protocol

We evaluate recovery with the *normalized mean-squared error* (NMSE), the squared reconstruction error relative to $\|\boldsymbol{M}^\star\|_{\mathrm{F}}^2$, which makes the metric scale-free and comparable across problems and conditioning regimes. For every (problem, condition, optimizer) triple we sweep the learning rate, the dominant axis and the usual source of misleading comparisons,

over a logarithmically spaced grid (25 points in $[10^{-4}, 5 \times 10^{-1}]$ for the factorization and completion tasks; 20 points in $[10^{-4}, 3.2]$ for NMF), holding all other hyperparameters (momentum, and for `Muon` the Newton–Schulz count J and the orthogonalization coefficients) at standard values. Each configuration runs for up to 3000 iterations under an identical patience-based learning-rate-decay schedule and is repeated over 3 seeds. Because the resulting NMSE values span roughly thirty orders of magnitude, we summarize each configuration by the *geometric mean* over seeds with a ±one-log-standard-deviation band, and compare optimizers by their *stable learning-rate range*, the band of step sizes over which each converges to the NMSE floor.

4. Experimental Results

A single learning rate is not a meaningful operating point. Within a fixed problem and optimizer, the final loss varies by a median of about 11 orders of magnitude across the learning-rate grid for the factorization and completion tasks, and about 6 orders for both NMF variants (Figures 1–2). Performance is therefore dominated by the learning rate, and any comparison made at one pre-selected value reflects that choice as much as the optimizer itself.

No shared learning rate is fair to all methods. The loss-minimizing learning rate differs *across optimizers* by a median of 1.5–3.1 decades (largest for NMF), so evaluating all methods at a common learning rate necessarily places at least one of them far from its own optimum. This appears as a horizontal shift between each method's basin in every panel.

Tuning reorders the methods, and the winner is problem-dependent. Table 1 reports each optimizer's best (learning-rate-tuned) final loss. In nearly all nineteen settings the ranking induced by a fixed "default" learning rate ($\approx 10^{-3}$) differs from the ranking obtained once each method is tuned individually, and methods that look worst at the default frequently become best after tuning. Under per-optimizer tuning no method dominates universally: AdamW and GD win on plain low-rank factorization, Muon and AdamW are comparable on matrix completion, and Muon wins clearly on both NMF variants. The detailed per-setting comparisons, including the specific default-vs-tuned reorderings, are deferred to Appendix D.

Ill-conditioning widens the gaps and shifts the winner. Conditioning is itself a hyperparameter of the problem, and it interacts strongly with the optimizer. As κ grows, every method's attainable loss degrades and its optimal learning rate migrates upward (Figures 1, 7). The effect is most dramatic for the non-adaptive methods on factorization, where GD goes from the best-tuned method in the well-conditioned regime to failing under extreme ill-conditioning while AdamW becomes best (see Appendix D for the precise values). Thus ill-conditioning does not merely raise the loss floor, it reorders the methods, so that no fixed configuration summarizes behavior across the conditioning range.

4.1. Sensitivity to spectrum shape under extreme ill-conditioning

Conditioning alone does not determine difficulty: at a *fixed* condition number, the convergence of first-order optimizers also depends on the *distribution* of eigenvalues within $[s_{\min}, s_{\max}]$. We therefore hold the conditioning fixed at an extreme value and vary only the interior shape of the spectrum, evaluating spectral profiles (defined in Appendix F). All

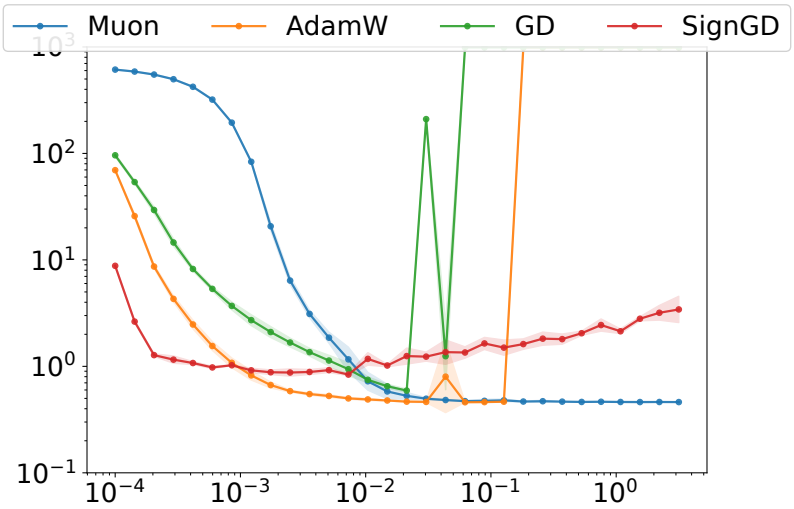

Figure 5: **Gaussian-kernel NMF at a principled bottleneck.** Non-negative factorization of a fixed Gaussian RBF kernel $K \in \mathbb{R}^{100\times100}$ at search rank r=24 (the smallest rank capturing 95% of K's spectral energy). On this smooth, well-behaved landscape Muon's orthogonalized updates give no advantage: AdamW reaches the same global floor as Muon.

families share the same endpoints $(s_{\min}, s_{\max}) = (10^{-3}, 10)$ and hence the same, deliberately extreme, condition number $\kappa = s_{\max}/s_{\min} = 10^4$; only the interior shape differs. Despite this identical (and severe) ill-conditioning, the distributions pose markedly different preconditioning challenges. As shown in Figure 3, AdamW adapts effectively to heavy-tailed and clustered spectra, whereas Muon underperforms across the majority of these shapes, evidence that its spectral updates are less robust to specific eigenvalue densities even when the overall condition number is held fixed. Ill-conditioning is therefore not a single axis of difficulty: the same κ can be easy or hard depending on how the eigenvalues are arranged, and the two optimizers respond to that arrangement differently.

4.2. Gaussian-kernel factorization: a well-behaved landscape

To complement the synthetic NMF problems, we study the Gaussian radial-basis-function kernel. We sample $N = 100$ points in $\mathbb{R}^5$ and form $K_{ij} = \exp(-\|x_i - x_j\|^2/2\sigma^2)$ with $\sigma = 2$, giving a symmetric positive-semidefinite matrix with strictly positive entries and a rapidly decaying spectrum. We factorize it as $\hat{K} = UV^\top$ under a non-negativity constraint ($U, V \geq 0$, enforced by projection after each step), minimizing $\frac{1}{2}\|UV^\top - K\|_F^2$.

Rather than fix the factor rank arbitrarily, we set it by a *principled bottleneck*: the smallest rank r whose leading singular values capture at least 95% of K's spectral energy, which yields $r = 24$ for this target. Each optimizer is tuned by a learning-rate sweep (30 points in $[10^{-4}, 10^{0.5}]$), and every configuration is repeated over three seeds that vary the initialization while holding the target fixed.

The result is shown in Figure 5. On this well-behaved landscape, Muon's advanced orthogonalization is unnecessary: standard AdamW performs just as effectively, converging seamlessly to the same global floor as Muon. This reinforces the study's central theme from

the opposite direction, on the adversarial NMF spectra Muon's structured updates help, but on a smooth, well-conditioned kernel target that advantage disappears and a well-tuned adaptive baseline is equally good.

4.3. Tensor-train factorization

We consider the tensor-train (TT) factorization of a matrix target. For an order-2 tensor $\boldsymbol{M}^\star \in \mathbb{R}^{d\times d}$, the TT decomposition contracts two cores $\mathcal{G}_1 \in \mathbb{R}^{d\times r}$ and $\mathcal{G}_2 \in \mathbb{R}^{r\times d}$ with boundary ranks $r_0 = r_2 = 1$ and a single internal rank $r_1 = r$,

$$\boldsymbol{M}^\star(i,j) \;=\; \sum_{\alpha=1}^{r} \mathcal{G}_1(i,\alpha)\,\mathcal{G}_2(\alpha,j), \qquad \Longleftrightarrow \qquad \boldsymbol{M}^\star \;=\; \mathcal{G}_1\mathcal{G}_2, \tag{4}$$

so the TT factorization coincides with the bilinear factorization of Appendix B under $\boldsymbol{U} = \mathcal{G}_1$, $\boldsymbol{V}^\top = \mathcal{G}_2$, with the lone TT-rank r playing the role of the factorization rank. In our setting the target $\boldsymbol{M}^\star = \boldsymbol{U}^\star\boldsymbol{U}^{\star\top}$ is symmetric positive semidefinite, but we do *not* tie the cores: recovery uses untied factors $\boldsymbol{U}, \boldsymbol{V}$, so the problem is the general bilinear factorization equation 12 applied to a symmetric PSD target, retaining the GL_r imbalance invariance of equation 12. Recovery minimizes

$$f(\boldsymbol{U},\boldsymbol{V}) \;=\; \frac{1}{2}\left\|\boldsymbol{U}\boldsymbol{V}^\top - \boldsymbol{M}^\star\right\|_{\mathrm{F}}^2, \qquad \boldsymbol{U}\in\mathbb{R}^{d\times r}, \boldsymbol{V}\in\mathbb{R}^{d\times r}, \tag{5}$$

the objective of equation 12 with search rank r as the TT-rank.

We study equation 5 in two regimes, a *noiseless* regime (true rank $r^\star = 4$, recovered exactly, loss limited only by convergence) and a *noisy* regime (true rank $r^\star = 30$ with additive observation noise, loss floored at the statistical noise level), and within each we sweep the search rank across under-, exactly-, and over-parameterized values; full specifications are given in Appendix G. The quantity of interest is the *stable learning-rate range* of each optimizer, the band of step sizes over which equation 5 converges to its regime-dependent floor, and how that band widens, narrows, or shifts as the parameterization regime and the noise level change (Figures 4 and 9). Beyond low-rank recovery, this factorized objective serves as a minimal model of the product parameterizations ubiquitous in deep learning, a connection we develop in Appendix H. We also study the effect of factorization depth in Appendix I.

5. Conclusion

In this work, we revisit optimizer comparisons for matrix factorization and show that conclusions are highly sensitive to hyperparameter selection and problem conditioning. Systematic learning-rate sweeps reveal that rankings reported under single configurations often do not hold: well-tuned AdamW and gradient descent frequently match or outperform Muon on standard low-rank factorization tasks, while Muon's advantages are mostly limited to NMF and some highly ill-conditioned completion problems. We further find that optimizer performance depends on both conditioning severity and spectral structure, altering method rankings across problem instances. These results underscore the need to evaluate optimizers across tuned hyperparameter ranges and conditioning regimes rather than relying on default settings or isolated benchmarks.

Acknowledgements

We thank Tianhao Wang for providing helpful references and for the discussions that assisted in developing the experimental setup. Ali Parviz was funded by NSF CIF-2403452.

References

Noah Amsel, David Persson, Christopher Musco, and Robert Gower. The polar express: Optimal matrix sign methods and their application to the Muon algorithm. *CoRR*, abs/2505.16932, 2025. doi: 10.48550/ARXIV.2505.16932. URL https://doi.org/10.48550/arXiv.2505.16932.

Kang An, Yuxing Liu, Rui Pan, Yi Ren, Shiqian Ma, Donald Goldfarb, and Tong Zhang. ASGO: Adaptive structured gradient optimization. *arXiv preprint arXiv:2503.20762*, 2025.

Sanjeev Arora, Nadav Cohen, and Elad Hazan. On the optimization of deep networks: Implicit acceleration by overparameterization. In *International conference on machine learning*, pages 244–253. PMLR, 2018.

Sanjeev Arora, Nadav Cohen, Wei Hu, and Yuping Luo. Implicit regularization in deep matrix factorization. *Advances in neural information processing systems*, 32, 2019.

Jeremy Bernstein and Laker Newhouse. Modular duality in deep learning. *arXiv preprint arXiv:2410.21265*, 2024a.

Jeremy Bernstein and Laker Newhouse. Old optimizer, new norm: An anthology. *arXiv preprint arXiv:2409.20325*, 2024b.

Nicolas Boumal and Antoine Gonon. Designing polynomials for Muon's polar factorization: Focus on quintics, 2025. URL www.racetothebottom.xyz/posts/polar-poly/. Accessed January 2026.

David Carlson, Volkan Cevher, and Lawrence Carin. Stochastic spectral descent for restricted Boltzmann machines. In *Artificial intelligence and statistics*, pages 111–119. PMLR, 2015a.

David Carlson, Ya-Ping Hsieh, Edo Collins, Lawrence Carin, and Volkan Cevher. Stochastic spectral descent for discrete graphical models. *IEEE Journal of Selected Topics in Signal Processing*, 10(2):296–311, 2015b.

David E Carlson, Edo Collins, Ya-Ping Hsieh, Lawrence Carin, and Volkan Cevher. Preconditioned spectral descent for deep learning. *Advances in neural information processing systems*, 28, 2015c.

Franz Louis Cesista, You Jiacheng, and Keller Jordan. Squeezing 1-2% efficiency gains out of Muon by optimizing the Newton-Schulz Coefficients, February 2025. URL https://leloykun.github.io/ponder/muon-opt-coeffs/. Accessed January 2026.

Lizhang Chen, Jonathan Li, and Qiang Liu. Muon optimizes under spectral norm constraints. *CoRR*, abs/2506.15054, 2025. doi: 10.48550/ARXIV.2506.15054. URL https://doi.org/10.48550/arXiv.2506.15054.

Nadav Cohen, Or Sharir, and Amnon Shashua. On the expressive power of deep learning: A tensor analysis. In *Conference on learning theory*, pages 698–728. PMLR, 2016.

Damek Davis and Dmitriy Drusvyatskiy. When do spectral gradient updates help in deep learning? *CoRR*, abs/2512.04299, 2025. doi: 10.48550/ARXIV.2512.04299. URL https://doi.org/10.48550/arXiv.2512.04299.

Chen Fan, Mark Schmidt, and Christos Thrampoulidis. Implicit bias of spectral descent and Muon on multiclass separable data. *arXiv preprint arXiv:2502.04664*, 2025.

Ekaterina Grishina, Matvey Smirnov, and Maxim V. Rakhuba. Accelerating Newton-Schulz iteration for orthogonalization via Chebyshev-type polynomials. *CoRR*, abs/2506.10935, 2025. doi: 10.48550/ARXIV.2506.10935. URL https://doi.org/10.48550/arXiv.2506.10935.

Kaja Gruntkowska, Alexander Gaponov, Zhirayr Tovmasyan, and Peter Richtárik. Error feedback for Muon and friends. *CoRR*, abs/2510.00643, 2025. doi: 10.48550/ARXIV.2510.00643. URL https://doi.org/10.48550/arXiv.2510.00643.

Suriya Gunasekar, Blake E Woodworth, Srinadh Bhojanapalli, Behnam Neyshabur, and Nati Srebro. Implicit regularization in matrix factorization. *Advances in neural information processing systems*, 30, 2017.

Vineet Gupta, Tomer Koren, and Yoram Singer. Shampoo: Preconditioned stochastic tensor optimization. In *International Conference on Machine Learning*, pages 1842–1850. PMLR, 2018.

Saurav Jha, Maryam Hashemzadeh, Ali Saheb Pasand, Ali Parviz, Min-Joong Lee, and Boris Knyazev. Ream: Merging improves pruning of experts in llms. *arXiv preprint arXiv:2604.04356*, 2026.

Keller Jordan. Muon: An optimizer for the hidden layers in neural networks. https://kellerjordan.github.io/posts/muon/, 2024. Accessed January 2026.

Valentin Khrulkov, Alexander Novikov, and Ivan Oseledets. Expressive power of recurrent neural networks. *arXiv preprint arXiv:1711.00811*, 2017.

Gyu Yeol Kim and Min-hwan Oh. Convergence of Muon with Newton–Schulz. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=lJSfxtLpLm.

Dmitry Kovalev. Understanding gradient orthogonalization for deep learning via non-euclidean trust-region optimization. *CoRR*, abs/2503.12645, 2025. doi: 10.48550/ARXIV.2503.12645. URL https://doi.org/10.48550/arXiv.2503.12645.

Tim Tsz-Kit Lau, Qi Long, and Weijie Su. Polargrad: A class of matrix-gradient optimizers from a unifying preconditioning perspective. *CoRR*, abs/2505.21799, 2025. doi: 10.48550/ARXIV.2505.21799. URL https://doi.org/10.48550/arXiv.2505.21799.

Jiaxiang Li and Mingyi Hong. A note on the convergence of Muon, 2025. URL https://arxiv.org/abs/2502.02900.

Jingyuan Liu, Jianlin Su, Xingcheng Yao, Zhejun Jiang, Guokun Lai, Yulun Du, Yidao Qin, Weixin Xu, Enzhe Lu, Junjie Yan, Yanru Chen, Huabin Zheng, Yibo Liu, Shaowei Liu, Bohong Yin, Weiran He, Han Zhu, Yuzhi Wang, Jianzhou Wang, Mengnan Dong, Zheng Zhang, Yongsheng Kang, Hao Zhang, Xinran Xu, Yutao Zhang, Yuxin Wu, Xinyu Zhou, and Zhilin Yang. Muon is scalable for LLM training. *CoRR*, abs/2502.16982, 2025. doi: 10.48550/ARXIV.2502.16982. URL https://doi.org/10.48550/arXiv.2502.16982.

Jianhao Ma, Yu Huang, Yuejie Chi, and Yuxin Chen. Preconditioning benefits of spectral orthogonalization in Muon, 2026. URL https://arxiv.org/abs/2601.13474.

Shuntaro Nagashima and Hideaki Iiduka. Improved convergence rates of Muon optimizer for nonconvex optimization, 2026. URL https://arxiv.org/abs/2601.19400.

Alexander Novikov, Dmitrii Podoprikhin, Anton Osokin, and Dmitry P Vetrov. Tensorizing neural networks. *Advances in neural information processing systems*, 28, 2015.

Ali Parviz and Yuichi Yoshida. Nonlinear laplacians improve signed-directed graph learning. In *New Perspectives in Graph Machine Learning*, 2025. URL https://openreview.net/forum?id=3CcP3VI6LW.

Thomas Pethick, Wanyun Xie, Kimon Antonakopoulos, Zhenyu Zhu, Antonio Silveti-Falls, and Volkan Cevher. Training deep learning models with norm-constrained lmos. In *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*. OpenReview.net, 2025a. URL https://openreview.net/forum?id=2Oqm2IzTy9.

Thomas Pethick, Wanyun Xie, Kimon Antonakopoulos, Zhenyu Zhu, Antonio Silveti-Falls, and Volkan Cevher. Training deep learning models with norm-constrained LMOs. *arXiv preprint arXiv:2502.07529*, 2025b.

Artem Riabinin, Egor Shulgin, Kaja Gruntkowska, and Peter Richtárik. Gluon: Making Muon & Scion great again! (bridging theory and practice of LMO-based optimizers for LLMs). *CoRR*, abs/2505.13416, 2025. doi: 10.48550/ARXIV.2505.13416. URL https://doi.org/10.48550/arXiv.2505.13416.

Naoki Sato, Hiroki Naganuma, and Hideaki Iiduka. Convergence bound and critical batch size of Muon optimizer, 2025. URL https://arxiv.org/abs/2507.01598.

Amin Sghaier, Ali Parviz, and Alexia Jolicoeur-Martineau. Probabilistic tiny recursive model. *arXiv preprint arXiv:2605.19943*, 2026.

Ishaan Shah, Anthony M Polloreno, Karl Stratos, Philip Monk, Adarsh Chaluvaraju, Andrew Hojel, Andrew Ma, Anil Thomas, Ashish Tanwer, and Darsh J Shah. Practical efficiency of Muon for pretraining. *arXiv preprint arXiv:2505.02222*, 2025a.

Ishaan Shah, Anthony M. Polloreno, Karl Stratos, Philip Monk, Adarsh Chaluvaraju, Andrew Hojel, Andrew Ma, Anil Thomas, Ashish Tanwer, Darsh J. Shah, Khoi Nguyen, Kurt Smith, Michael Callahan, Michael Pust, Mohit Parmar, Peter Rushton, Platon Mazarakis, Ritvik Kapila, Saurabh Srivastava, Somanshu Singla, Tim Romanski, Yash Vanjani, and Ashish Vaswani. Practical efficiency of Muon for pretraining. *CoRR*, abs/2505.02222, 2025b. doi: 10.48550/ARXIV.2505.02222. URL https://doi.org/10.48550/arXiv.2505.02222.

Wei Shen, Ruichuan Huang, Minhui Huang, Cong Shen, and Jiawei Zhang. On the convergence analysis of Muon. *CoRR*, abs/2505.23737, 2025. doi: 10.48550/ARXIV.2505.23737. URL https://doi.org/10.48550/arXiv.2505.23737.

Egor Shulgin, Sultan AlRashed, Francesco Orabona, and Peter Richtárik. Beyond the ideal: Analyzing the inexact Muon update. *CoRR*, abs/2510.19933, 2025. doi: 10.48550/ARXIV.2510.19933. URL https://doi.org/10.48550/arXiv.2510.19933.

Weijie Su. Isotropic curvature model for understanding deep learning optimization: Is gradient orthogonalization optimal? *arXiv preprint arXiv:2511.00674*, 2025.

Tian Tong, Cong Ma, and Yuejie Chi. Accelerating ill-conditioned low-rank matrix estimation via scaled gradient descent. *Journal of Machine Learning Research*, 22(150):1–63, 2021a.

Tian Tong, Cong Ma, and Yuejie Chi. Low-rank matrix recovery with scaled subgradient methods: Fast and robust convergence without the condition number. *IEEE Transactions on Signal Processing*, 69:2396–2409, 2021b.

Mark Tuddenham, Adam Prügel-Bennett, and Jonathan Hare. Orthogonalising gradients to speed up neural network optimisation. *arXiv preprint arXiv:2202.07052*, 2022.

Amund Tveit, Bjørn Remseth, and Arve Skogvold. Muon optimizer accelerates grokking. *arXiv preprint arXiv:2504.16041*, 2025.

Bhavya Vasudeva, Puneesh Deora, Yize Zhao, Vatsal Sharan, and Christos Thrampoulidis. How Muon's spectral design benefits generalization: A study on imbalanced data. *arXiv preprint arXiv:2510.22980*, 2025.

Shuche Wang, Fengzhuo Zhang, Jiaxiang Li, Cunxiao Du, Chao Du, Tianyu Pang, Zhuoran Yang, Mingyi Hong, and Vincent YF Tan. Muon outperforms Adam in tail-end associative memory learning. *arXiv preprint arXiv:2509.26030*, 2025.

Xingyu Xu, Yandi Shen, Yuejie Chi, and Cong Ma. The power of preconditioning in overparameterized low-rank matrix sensing. In *International Conference on Machine Learning*, pages 38611–38654. PMLR, 2023.

Gavin Zhang, Salar Fattahi, and Richard Y Zhang. Preconditioned gradient descent for overparameterized nonconvex Burer-Monteiro factorization with global optimality certification. *Journal of Machine Learning Research*, 24(163):1–55, 2023.

Jialun Zhang, Salar Fattahi, and Richard Y Zhang. Preconditioned gradient descent for overparameterized nonconvex matrix factorization. *Advances in Neural Information Processing Systems*, 34:5985–5996, 2021.

Thomas T Zhang, Behrad Moniri, Ansh Nagwekar, Faraz Rahman, Anton Xue, Hamed Hassani, and Nikolai Matni. On the concurrence of layer-wise preconditioning methods and provable feature learning. *arXiv preprint arXiv:2502.01763*, 2025.

Appendix A. Background: The Muon Optimizer

We briefly recall the `Muon` update and the spectral viewpoint that motivates it. Consider a weight matrix $\boldsymbol{W} \in \mathbb{R}^{m\times n}$ with loss gradient $\boldsymbol{G}_t = \nabla_{\boldsymbol{W}} L(\boldsymbol{W}_t)$ at iteration t. `Muon` maintains a momentum buffer $\boldsymbol{B}_t$ and applies an *orthogonalized* update:

$$\boldsymbol{B}_t = \mu\, \boldsymbol{B}_{t-1} + \boldsymbol{G}_t, \tag{6}$$

$$\boldsymbol{O}_t = \text{Newton-Schulz}(\boldsymbol{B}_t) \;\approx\; \text{polar}(\boldsymbol{B}_t), \tag{7}$$

$$\boldsymbol{W}_{t+1} = \boldsymbol{W}_t - \alpha\, \boldsymbol{O}_t. \tag{8}$$

Here $\alpha > 0$ is the step size and $\mu \in [0, 1)$ the momentum coefficient. The orthogonalization step equation 7 replaces the momentum direction by (an approximation of) its *polar factor*.

Polar factor and spectral steepest descent. Let $\boldsymbol{B}_t = \boldsymbol{P}\boldsymbol{\Sigma}\boldsymbol{Q}^\top$ be a singular value decomposition, with $\boldsymbol{P} \in \mathcal{O}_{m\times\rho}$, $\boldsymbol{Q} \in \mathcal{O}_{n\times\rho}$, and $\boldsymbol{\Sigma} = \mathsf{diag}\{\sigma_1, \ldots, \sigma_\rho\}$, where $\rho = \text{rank}(\boldsymbol{B}_t)$. The polar factor is

$$\text{polar}(\boldsymbol{B}_t) \;=\; \boldsymbol{P}\boldsymbol{Q}^\top, \tag{9}$$

which sets every nonzero singular value to one while preserving the singular vectors. Equivalently, $\boldsymbol{P}\boldsymbol{Q}^\top$ is the solution of the spectral-norm linear minimization oracle,

$$\boldsymbol{P}\boldsymbol{Q}^\top \;=\; \underset{\|\boldsymbol{X}\|_{\text{op}}\le 1}{\arg\max} \langle \boldsymbol{B}_t, \boldsymbol{X}\rangle\,, \tag{10}$$

so the `Muon` direction equation 7 is precisely the steepest-descent direction with respect to the spectral norm. This is the sense in which `Muon` is "spectrum aware": it discards the magnitude of the momentum's singular values and acts only along their directions.

Newton–Schulz orthogonalization. Computing the SVD at every step is expensive, so `Muon` approximates the polar factor with a fixed number of matrix-multiplication-only Newton–Schulz iterations. Starting from the normalized matrix $\boldsymbol{X}_0 = \boldsymbol{B}_t / \|\boldsymbol{B}_t\|_{\text{F}}$, one iterates a fixed odd polynomial

$$\boldsymbol{X}_{j+1} \;=\; a\,\boldsymbol{X}_j \;+\; b\,\boldsymbol{X}_j\boldsymbol{X}_j^\top\boldsymbol{X}_j \;+\; c\,\big(\boldsymbol{X}_j\boldsymbol{X}_j^\top\big)^2\boldsymbol{X}_j, \qquad j = 0, 1, \ldots, J-1, \tag{11}$$

with coefficients (a, b, c) chosen so that the induced scalar map $\sigma \mapsto a\sigma + b\sigma^3 + c\sigma^5$ pushes the singular values of $\boldsymbol{X}_j$ toward 1. The quintic coefficients $(a, b, c) = (3.4445, -4.7750, 2.0315)$ proposed by Jordan (2024) reach a usable approximation in roughly $J = 5$ iterations; subsequent work designs improved polynomials for this orthogonalization (Amsel et al., 2025; Grishina et al., 2025; Cesista et al., 2025; Boumal and Gonon, 2025). As $J \to \infty$ (with an exact map) the iteration converges to $\text{polar}(\boldsymbol{B}_t)$, recovering the idealized update; in practice J is small and $\boldsymbol{O}_t$ is only an approximate polar factor, which several recent analyses account for explicitly (Shulgin et al., 2025; Kim and Oh, 2026; Lau et al., 2025).

Appendix B. Matrix Factorization Variants

This appendix collects two generalizations of the symmetric matrix factorization problem equation 1: the general bilinear (asymmetric) factorization and the matrix completion setting in which only a subset of the entries of the target is observed.

General bilinear factorization. The symmetric problem equation 1 is a special case of the general *bilinear* (asymmetric) factorization, in which a possibly rectangular target $\boldsymbol{M}^\star \in \mathbb{R}^{d_1 \times d_2}$ of rank r is recovered from the product of two factors,

$$\min_{\substack{\boldsymbol{U}\in\mathbb{R}^{d_1\times k}\\ \boldsymbol{V}\in\mathbb{R}^{d_2\times k}}} \quad f(\boldsymbol{U},\boldsymbol{V}) \;=\; \frac{1}{2}\left\|\boldsymbol{U}\boldsymbol{V}^\top - \boldsymbol{M}^\star\right\|_{\mathrm{F}}^2, \tag{12}$$

with $k \geq r$. When $\boldsymbol{M}^\star$ is symmetric positive semidefinite and the factors are tied as $\boldsymbol{V} = \boldsymbol{U}$, equation 12 reduces to equation 1 up to a constant rescaling of the objective. Unlike the symmetric case, the asymmetric parameterization is invariant under the larger group of invertible transformations $(\boldsymbol{U},\boldsymbol{V}) \mapsto (\boldsymbol{U}\boldsymbol{R}, \boldsymbol{V}\boldsymbol{R}^{-\top})$ for any $\boldsymbol{R} \in \mathrm{GL}_k(\mathbb{R})$, since $\boldsymbol{U}\boldsymbol{V}^\top = (\boldsymbol{U}\boldsymbol{R})(\boldsymbol{V}\boldsymbol{R}^{-\top})^\top$. This invariance permits an arbitrary imbalance between the factor norms and is commonly controlled by a balancing regularizer $\frac{1}{2}\big(\|\boldsymbol{U}\|_{\mathrm{F}}^2 + \|\boldsymbol{V}\|_{\mathrm{F}}^2\big)$ or by enforcing $\boldsymbol{U}^\top\boldsymbol{U} = \boldsymbol{V}^\top\boldsymbol{V}$ along the trajectory.

Matrix completion. In many applications only a subset of the entries of $\boldsymbol{M}^\star$ is observed. Let $[n] \coloneqq \{1,\dots,n\}$ and let $\Omega \subseteq [d_1]\times[d_2]$ denote the set of observed indices. Define the sampling operator $\mathcal{P}_\Omega : \mathbb{R}^{d_1\times d_2} \to \mathbb{R}^{d_1\times d_2}$ by

$$\big[\mathcal{P}_\Omega(\boldsymbol{X})\big]_{ij} \;=\; \begin{cases} X_{ij}, & (i,j)\in\Omega,\\ 0, & \text{otherwise,}\end{cases} \tag{13}$$

which retains the observed entries and zeros out the rest. The factored matrix completion problem is

$$\min_{\substack{\boldsymbol{U}\in\mathbb{R}^{d_1\times k}\\ \boldsymbol{V}\in\mathbb{R}^{d_2\times k}}} \quad \frac{1}{2}\left\|\mathcal{P}_\Omega\big(\boldsymbol{U}\boldsymbol{V}^\top - \boldsymbol{M}^\star\big)\right\|_{\mathrm{F}}^2. \tag{14}$$

Taking $\Omega = [d_1]\times[d_2]$ recovers the fully observed problem equation 12, while the symmetric variant tied to equation 1 additionally imposes $\boldsymbol{V} = \boldsymbol{U}$. Exact recovery in this setting requires the ground-truth factors to be incoherent and the sample size $|\Omega|$ to be sufficiently large relative to the degrees of freedom $r(d_1 + d_2 - r)$. Under the Bernoulli model in which each entry is observed independently with probability p, the rescaled operator $p^{-1}\mathcal{P}_\Omega$ is an unbiased estimator of the identity, i.e. $\big[p^{-1}\mathcal{P}_\Omega(\boldsymbol{X})\big] = \boldsymbol{X}$.

Appendix C. Related Work

Spectrum-aware optimization and `Muon`. Much of the theoretical analysis of `Muon` builds on a perspective that interprets modern optimizers such as `Adam` and `Shampoo` as instances of steepest descent under non-Euclidean or norm-constrained geometries (Bernstein and Newhouse, 2024b). Within this view, `Muon` approximates a spectral-norm steepest-descent direction via polar factorization, connecting it to a broader class of methods that leverage matrix structure in gradients (Pethick et al., 2025a; Li and Hong, 2025; Shen et al., 2025; Chen et al., 2025). Spectral transformations and orthogonalization in optimization predate `Muon`, appearing in earlier work on stochastic and preconditioned methods (Carlson et al., 2015a,b,c; Tuddenham et al., 2022). A complementary line of work asks more directly when

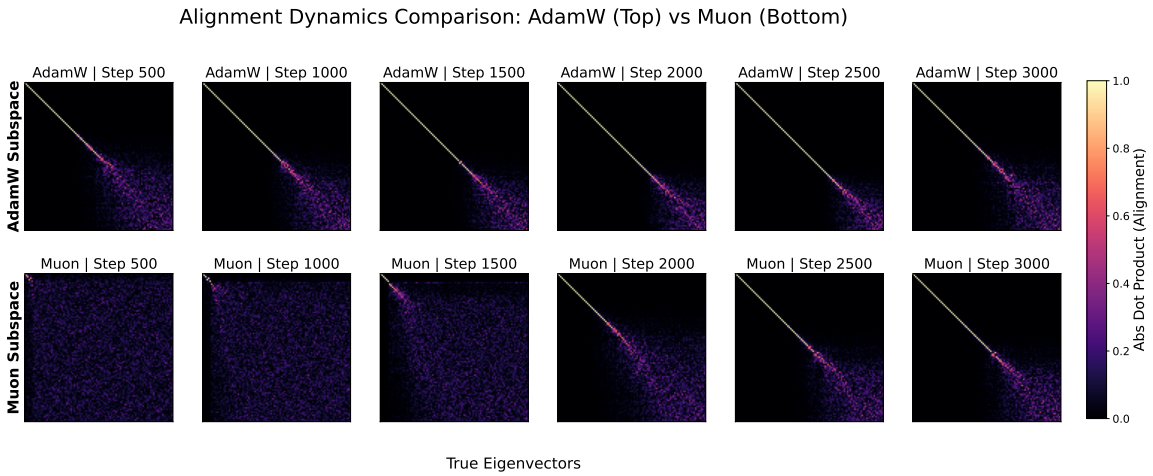

Figure 6: **Spectral subspace recovery: Muon vs. AdamW.** Absolute alignment $M_{ij} = |\langle u_i, e_j \rangle|$ (Eq. 15) between learned singular vectors u_i and true eigenvectors e_j of K, every 500 steps (AdamW top, Muon bottom). A bright, correctly-ordered diagonal indicates faithful recovery; off-diagonal mass reflects mixing. AdamW sharpens its diagonal steadily, while Muon stays near-random for ∼1500 steps, then snaps into a more diffuse alignment.

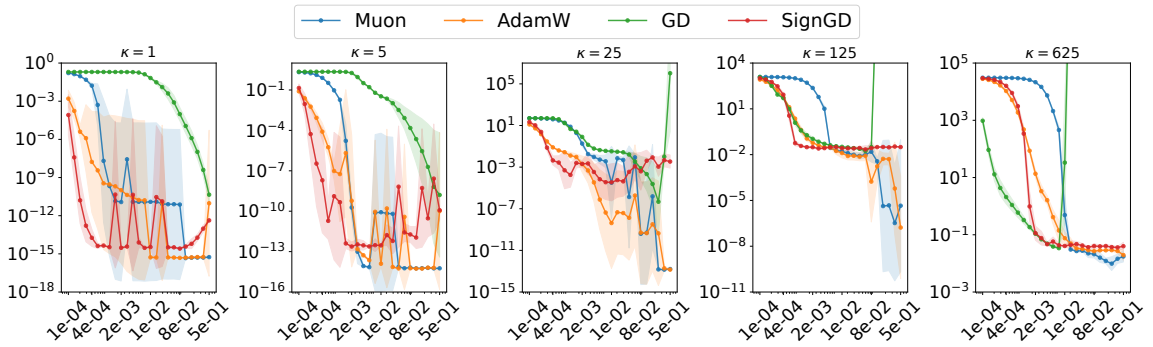

Figure 7: **Matrix completion, conditioning sweep.** Each panel plots the final reconstruction loss (y-axis, log scale; geometric mean over 3 seeds with a shaded ±one-log-standard-deviation band) against the learning rate (x-axis, log scale), for condition number κ. Muon and AdamW are indistinguishable in the well-conditioned regime; all methods degrade and converge toward one another as κ grows.

such spectral updates are beneficial in deep learning (Davis and Drusvyatskiy, 2025; Su, 2025).

Implicit bias and generalization. A second line of work studies the implicit bias induced by `Muon`. Fan et al. (2025) show that in linear classification, idealized `Muon` converges to a solution that maximizes margin with respect to the spectral norm, contrasting with the Euclidean and coordinate-wise biases of `SGD` and `Adam`. Complementary empirical studies suggest that spectrum-aware updates can improve generalization, particularly in imbalanced or long-tailed settings, by promoting more uniform learning across principal components

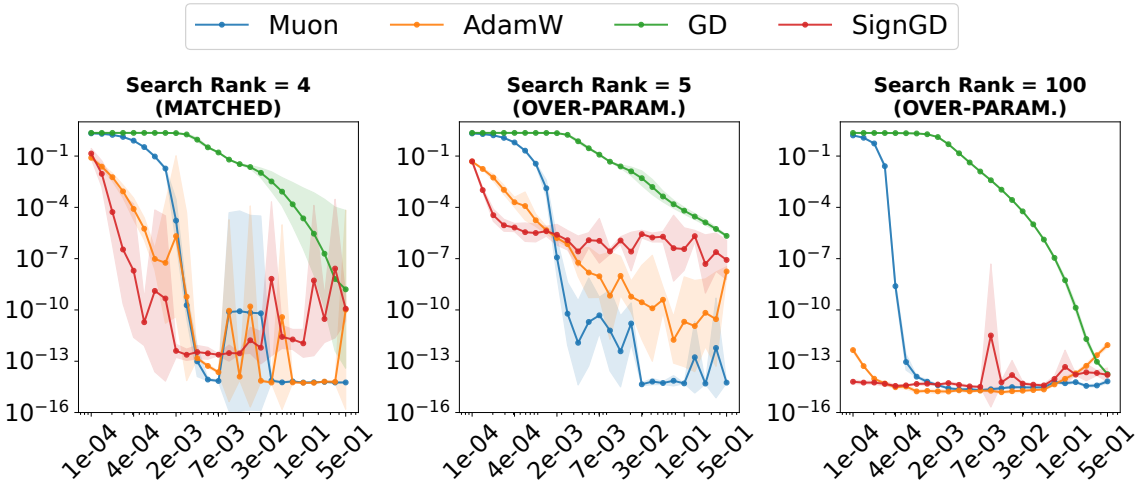

Figure 8: **Matrix completion, search-rank sweep** (true rank 4, κ=5). Over-parameterization (rank 100) widens the band of effective learning rates and lets every method except GD reach machine precision.

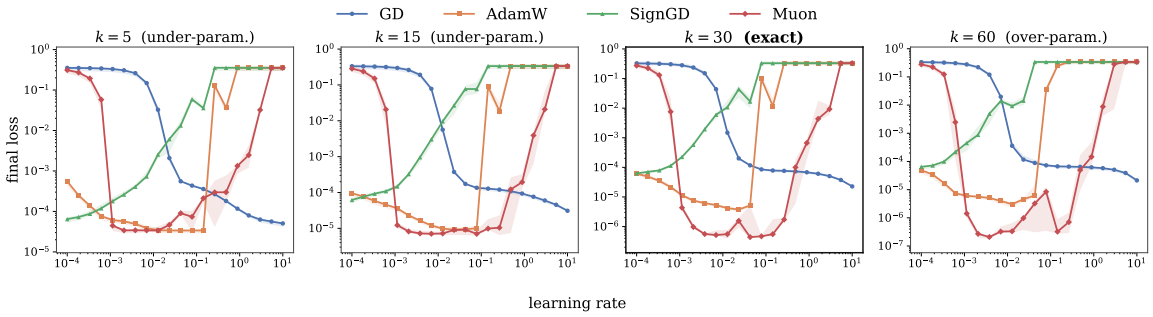

Figure 9: **Tensor-Train Noisy regime** (true rank $r^\star = 30$, additive observation noise). Final loss versus learning rate for each optimizer, across search ranks $r \in \{5, 15, 30, 60\}$ spanning under-, exactly- ($r = r^\star$, bold panel), and over-parameterized settings. Solid lines are the median over 3 seeds; shaded bands the min–max; the loss floors at the noise level rather than at zero.

rather than focusing on dominant directions (Vasudeva et al., 2025; Wang et al., 2025; Parviz and Yoshida, 2025). Related analyses indicate that `Muon` yields more isotropic singular-value spectra than `Adam` and may accelerate phenomena such as grokking in long-horizon training (Zhang et al., 2025; Wang et al., 2025; Vasudeva et al., 2025; Tveit et al., 2025; Jha et al., 2026). Connections have also been drawn between `Muon` and second-order or preconditioned methods, with several works interpreting its updates as approximations to `Shampoo` (Gupta et al., 2018; Jordan, 2024; Shah et al., 2025a). Alternative derivations and closely related formulations, some predating `Muon`, have appeared from different theoretical viewpoints (Pethick et al., 2025b; Carlson et al., 2015c; Lau et al., 2025; Bernstein and Newhouse, 2024a,b; An et al., 2025; Sghaier et al., 2026).

Preconditioning for matrix factorization. A separate body of work shows that preconditioning can dramatically accelerate optimization in matrix factorization. In particular, `ScaledGD` and its variants achieve linear convergence rates independent of the condition number under appropriate initialization, in both exactly parameterized and over-parameterized regimes (Tong et al., 2021a,b; Zhang et al., 2021, 2023), and these guarantees extend to small random initialization (Xu et al., 2023). Closest to our motivation, recent work analyzes the preconditioning effect of `Muon`'s spectral orthogonalization directly (Ma et al., 2026).

Summary. Taken together, these results show that spectrum-aware and preconditioned methods can offer strong theoretical and empirical benefits, but that their behavior depends heavily on problem structure and initialization. This motivates a closer examination of `Muon` in controlled settings such as matrix factorization, where powerful alternatives already enjoy strong guarantees and where the advantage of `Muon` over adaptive methods like `AdamW` is not immediately evident.

Appendix D. Detailed experimental results

This appendix expands the summary claims of Section 4 with the specific numerical comparisons. All values refer to Table 1.

Horizontal basin shifts. Evaluating all methods at a common learning rate necessarily places at least one of them far from its own optimum. This is visible as a horizontal shift between each method's basin in every panel: for instance, GD's basin lies near the high end of the grid while Muon's and AdamW's lie one to three decades lower.

Default-vs-tuned reordering. The ranking induced by a fixed default learning rate ($\approx 10^{-3}$) differs from the tuned ranking in nearly all nineteen settings, and methods that look worst at the default frequently become best after tuning. On low-rank factorization GD is among the worst at the default rate yet attains the lowest loss of all methods once tuned (Figure 1); symmetrically, on both NMF variants Muon is *last* at the default rate but *first* once tuned (Figures 2).

Effect of ill-conditioning. The effect of ill-conditioning is most dramatic for the non-adaptive methods on factorization: GD, the best-tuned method in the well-conditioned regime (1.6×10^{-13} at κ=1), collapses to 1.3×10^{-1} at κ=625, where AdamW becomes best (2.6×10^{-7}), Muon second (9.4×10^{-5}), and SignGD diverges. Matrix completion shows the same upward drift of the loss floor: at κ=625 all four methods are confined to $O(10^{-2})$ and become nearly indistinguishable.

Problem-dependence of the tuned winner. Under per-optimizer tuning there is no universally dominant method. On *plain low-rank factorization*, AdamW and GD drive the loss to 10^{-13}–10^{-9} across the whole conditioning range, while the Muon configuration plateaus several orders higher (10^{-8}–10^{-5}). On *matrix completion*, Muon and AdamW are both reaching 10^{-16}–10^{-13} in the well-conditioned regime, and over-parameterization (search rank 100) lets all methods except GD reach machine precision (Figure 8). On *NMF*, the picture reverses: Muon is the clear winner under both spectra, reaching 10^{-9}–10^{-6} while AdamW stalls near 10^{-3} (with the single exception of the uniform-spectrum rank-50 case,

where AdamW matches Muon at $\sim 4 \times 10^{-9}$) and GD/SignGD fail to fit. Whichever optimizer a single-configuration study would crown thus depends entirely on the problem and the learning rate chosen.

Appendix E. Dynamics of spectral subspace recovery

The aggregate losses above tell us *whether* a method fits the target but not *how* it does so. To probe the mechanism, we factorize a fixed Gaussian-kernel target $K \in \mathbb{R}^{100\times100}$, symmetric positive semidefinite, with eigendecomposition $K = Q\Lambda Q^\top$ and eigenvectors ordered by descending eigenvalue, as $\hat{K} = UV^\top$ at full search rank $k = 100$, training AdamW and Muon each at its own tuned learning rate. Every 500 iterations we take the left singular vectors $\{\hat{u}_i\}$ of the current reconstruction $UV^\top$ and form the *alignment matrix*

$$M_{ij} \;=\; \big|\langle \hat{u}_i,\, q_j \rangle\big|, \tag{15}$$

the absolute overlap between the i-th learned singular direction and the j-th true eigenvector. A faithful, correctly ordered recovery of the eigenbasis yields $M \to I$ (a sharp diagonal), whereas off-diagonal or diffuse mass indicates that the learned subspace mixes or reorders eigendirections. Figure 6 tracks M over training. AdamW sharpens toward a clean diagonal quickly and monotonically, recovering the eigenvectors faithfully and in order, while Muon's alignment remains comparatively diffuse, with persistent off-diagonal mass and slower diagonalization. This provides a mechanistic explanation for the loss-level gap of Sections 4–4.1: on these structured and ill-conditioned spectra, AdamW's per-coordinate adaptation locks onto the dominant eigendirections in the right order, whereas Muon's orthogonalized updates distribute capacity more evenly across directions and recover the eigenbasis less cleanly. The heatmaps shown are for a representative tuned run; the qualitative pattern is consistent across seeds.

Appendix F. Spectrum families.

Across all experiments we fix $s_{\min} = 10^{-3}$ and $s_{\max} = 10$, so that the condition number is $\kappa = 10^4$. The target is $M = U\,\mathrm{diag}(s)\,V^\top \in \mathbb{R}^{m\times n}$ with $m = n = 100$ and U, V drawn from the Q-factor of i.i.d. Gaussian matrices; the spectrum $s \in \mathbb{R}^r_{>0}$ therefore has length equal to the factorization rank $r = 5$, and its entries are singular values rather than eigenvalues.

Common post-processing. Every family below is generated, then sorted in descending order, and then has its endpoints overwritten exactly: $s_1 := s_{\max}$ and $s_r := s_{\min}$. Consequently all families realize $\kappa = 10^4$ by construction, even where the sampled bulk does not reach the endpoints; for `gaussian`, `u_shaped`, and `geometric_0.9` the two extreme values should be read as planted outliers relative to the bulk. Stochastic families are resampled per seed; `flat_max`, `flat_min`, `geometric`, `geometric_<q>`, and `powerlaw` are deterministic given r.

- `flat_max`: $s_r = s_{\min}$ and $s_1 = \cdots = s_{r-1} = s_{\max}$ (a single small singular value; mass at the top).
- `flat_min`: $s_1 = s_{\max}$ and $s_2 = \cdots = s_r = s_{\min}$ (a single large singular value; mass at the bottom).
- `uniform`: i.i.d. Unif$[s_{\min}, s_{\max}]$.

- `log_uniform`: $\tilde{s}_i = s_{\min}(s_{\max}/s_{\min})^{u_i}$ with $u_i \sim \mathrm{Unif}[0,1]$, i.e. log-uniform across the four decades.
- `geometric`: the deterministic geometric progression spanning both endpoints, $\tilde{s}_i = s_{\max}\,\rho^{\,i-1}$ with $\rho = (s_{\min}/s_{\max})^{1/(r-1)}$ ($\rho \approx 0.829$ at $r = 50$).
- `geometric_<q>`, $q \in \{0.3, 0.5, 0.9\}$: a deterministic geometric progression with *fixed* ratio, clipped below at $s_{\min}$: $\tilde{s}_i = \max\big(s_{\min},\, s_{\max} q^{\,i-1}\big)$. Note that the resulting shape depends strongly on q relative to r: at $r = 50$ the clip binds for 84% of the entries when $q = 0.3$ and 72% when $q = 0.5$ (yielding a large plateau at $s_{\min}$), whereas for $q = 0.9$ it never binds ($s_{\max} q^{49} \approx 5.7 \cdot 10^{-2}$) and the smallest value is supplied solely by the endpoint enforcement.
- `powerlaw` ($\alpha = 1$): the sequence $i^{-\alpha}$, $i = 1, \ldots, r$, is min–max normalized to $[0,1]$ and *reflected* before rescaling, $\tilde{s}_i = s_{\min} + (s_{\max} - s_{\min})\big(1 - \widetilde{i^{-\alpha}}\big)$. Despite the name this is not a heavy-tailed spectrum: at $\alpha = 1$, $r = 50$, 84% of the values lie above $0.9\, s_{\max}$, making it close to `flat_max` in shape.
- `linear_decay_to_smax`: $\tilde{s} = s_{\max} - (s_{\max} - s_{\min})\sqrt{u}$ with $u \sim \mathrm{Unif}[0,1]$, a density that decreases linearly toward $s_{\max}$ and hence concentrates near $s_{\min}$.
- `linear_decay_faster`: $\tilde{s} = s_{\min} + 2 - 2\sqrt{u}$ with $u \sim \mathrm{Unif}[0,1]$. The additive constant is absolute rather than proportional to the range, so at $s_{\max} = 10$ the sampled support is $[s_{\min},\, s_{\min} + 2]$ and the largest value comes from the endpoint enforcement alone.
- `gaussian_<k>`, $k \in \{2, 3\}$ (`gaussian` $\equiv k = 3$): a truncated normal on $[s_{\min}, s_{\max}]$. With $\mathrm{mid} := (s_{\min} + s_{\max})/2$ and $\sigma := (s_{\max} - s_{\min})/(2k)$, we draw $z_i \sim \mathcal{N}(0,1)$ conditioned on $|z_i| \le k$ (by rejection sampling, not clipping, so no atoms are placed at $\pm k$) and set $\tilde{s}_i = \mathrm{mid} + \sigma z_i$. The choice of σ makes the truncation limits coincide with $[s_{\min}, s_{\max}]$.
- `u_shaped_<a>`: symmetric $\mathrm{Beta}(a,a)$ on $[0,1]$ mapped affinely to $[s_{\min}, s_{\max}]$, $\tilde{s}_i = s_{\min} + (s_{\max} - s_{\min})z_i$ with $z_i \sim \mathrm{Beta}(a,a)$ and $0 < a < 1$. We use the presets $a = 0.9$ (`weak`), $a = 0.5$ (`medium`, the default `u_shaped`), and $a = 0.2$ (`strong`); smaller a concentrates more mass near both endpoints.
- `spikes_2_<pmin>_<pmax>`, with $(p_{\min}, p_{\max}) \in \{(0.99, 0.01), (0.95, 0.05), (0.9, 0.1)\}$: a two-point spectrum placing $\lceil p_{\min} r \rfloor$ values at $s_{\min}$ and $\lceil p_{\max} r \rfloor$ at $s_{\max}$, interpolating between `flat_min` and `flat_max`. Since $p_{\min} + p_{\max} = 1$ in all three cases, no residual mass remains to be allocated.

Appendix G. Tensor-train regimes and parameterization sweep

This appendix gives the full specification of the regimes and parameterization sweep summarized in Section 4.3.

Regimes. We study equation 5 in two regimes. In the *noiseless* regime the target has true rank $r^\star = 4$ and is recovered exactly, so the attainable loss is limited only by the optimizer's convergence; in the *noisy* regime the target has true rank $r^\star = 30$ and is corrupted by additive observation noise, so the loss is floored at the statistical noise level rather than driven to zero. The two regimes therefore probe complementary phenomena: optimization geometry in the well-specified case and robustness to a nonzero residual in the misspecified case.

Parameterization sweep. Within each regime we sweep the search rank, equivalently the TT-rank r of the factor $\boldsymbol{U}$, across under-, exactly-, and over-parameterized values: $r \in \{2, 4, 8, 20\}$ for the noiseless target (true rank $r^\star = 4$) and $r \in \{5, 15, 30, 60\}$ for the noisy target (true rank $r^\star = 30$), with $r = r^\star$ the exactly parameterized boundary case. Under-parameterization ($r < r^\star$) caps the attainable loss because $\boldsymbol{U}\boldsymbol{U}^\top$ cannot represent $\boldsymbol{M}^\star$, while over-parameterization ($r > r^\star$) enlarges the factor space and introduces additional flat directions in the landscape.

Appendix H. Relation to deep learning models

The factorized objective equation 5 is not only a tensor-train problem in its own right; it is the minimal instance of the product parameterizations that pervade deep learning, which is what makes the optimizer behavior we study here relevant beyond low-rank recovery.

Factorization as a shallow linear network. The map $\boldsymbol{U} \mapsto \boldsymbol{U}\boldsymbol{U}^\top$ (or $(\boldsymbol{U}, \boldsymbol{V}) \mapsto \boldsymbol{U}\boldsymbol{V}^\top$ in the asymmetric case of Appendix B) is exactly a two-layer *linear* network with no intervening nonlinearity: the factors are the layer weights, and the product is the end-to-end map. Consequently the loss is nonconvex in the factors despite being convex in the product, and the search rank, equivalently the TT-rank, plays the role of the network width. The over-parameterized regime $r > r^\star$ is precisely the width-overparameterized regime of modern networks, and the flat directions it introduces in the landscape are the source of the implicit bias of gradient methods studied for matrix and deep factorizations (Gunasekar et al., 2017; Arora et al., 2019). This is why over-parameterized factorization is a standard proxy for the optimization of deep models: it isolates the effect of redundant parameters on the trajectory while remaining analytically tractable.

Depth as a longer tensor-train chain. Increasing the order of the tensor-train chain, contracting L cores $\mathcal{G}_1, \ldots, \mathcal{G}_L$ rather than two, is the algebraic analogue of increasing the depth of a linear network, whose end-to-end map factorizes as $\boldsymbol{W} = \boldsymbol{W}_L \boldsymbol{W}_{L-1} \cdots \boldsymbol{W}_1$. The matrix case equation 4 ($L = 2$) is the shallowest such chain, and depth introduces exactly the ill-conditioning and balancing phenomena that motivate our conditioning sweep: the product of many factors amplifies spectral imbalance, and the overparameterized invariance group acts on each internal bond. Depth has been shown to act as an implicit preconditioner that accelerates gradient descent on these product objectives (Arora et al., 2018), so the sensitivity of each optimizer to conditioning at $L = 2$ is the base case of a phenomenon that sharpens with depth.

Tensor-train structure inside trained networks. Beyond serving as a proxy, tensor-train factorizations appear directly inside deep models. Reshaping a dense weight matrix into a high-order tensor and representing it in TT format compresses fully connected and embedding layers by orders of magnitude while preserving accuracy (Novikov et al., 2015), and specific architectures correspond to specific tensor decompositions: recurrent networks realize the tensor-train / matrix-product structure (Khrulkov et al., 2017), while convolutional arithmetic circuits realize the hierarchical Tucker decomposition (Cohen et al., 2016). In all of these the rank of the decomposition controls expressivity exactly as the TT-rank r controls the capacity of equation 5, so the interaction between search rank, conditioning,

and optimizer that we characterize on the factorized problem speaks directly to how these layers are trained.

Appendix I. Optimizer Comparison Across Tensor-Train Depth

We benchmark four optimizers—Muon, AdamW, gradient descent (GD), and SignGD—on non-negative tensor-train (TT) factorization. To systematically vary problem conditioning, we use the tensor-train depth (equivalently, the tensor order or number of TT cores) as the primary experimental factor. Each TT core is represented via a matrix unfolding, a necessary implementation detail because Muon's Newton–Schulz orthogonalization operates only on matrix-valued gradients; retaining a core in its native third-order form would effectively reduce Muon to momentum SGD on that factor. We evaluate two target regimes: a clean, low-rank, noise-free tensor and a noisy, higher-rank tensor. The study spans depths (2)–(6), an overparameterized factorization rank, a five-point learning-rate sweep, and three random seeds, yielding 600 total runs. Performance is reported as the best-tuned final loss for each optimizer, defined as the lowest mean squared error (MSE) achieved across the learning-rate grid and averaged over seeds.

The results differ markedly between the two regimes. For the clean target, optimizer rankings change substantially with depth. AdamW achieves the lowest losses at shallow depths, while Muon performs comparatively poorly. However, beginning at depth (4), the performance of AdamW, GD, and SignGD deteriorates as the factorization deepens, whereas Muon maintains consistently low error. By depth (6), Muon outperforms the competing methods by approximately one to two orders of magnitude (Figure 10, left). In contrast, performance on the noisy target remains relatively flat across depths, with all optimizers achieving similar losses (Figure 10, right). The added noise introduces an irreducible error floor that limits the optimization gains obtainable from improved conditioning, thereby diminishing Muon's advantage.

These findings suggest that Muon's benefits arise primarily from its ability to mitigate optimization difficulties associated with ill-conditioned deep factorizations, rather than from a general robustness advantage. At the same time, two limitations should be noted. First, increasing tensor-train depth changes the underlying optimization problem itself, so the observed trends reflect changes in relative optimizer performance across a family of increasingly difficult tasks rather than within a fixed loss landscape. Second, the reported results correspond only to the overparameterized setting considered here and may not fully characterize behavior at other scales or rank regimes.

Appendix J. Softplus, Its Taylor Surrogate, and Nonlinear Matrix Factorization

Let $X \in \mathbb{R}^{N\times d}$ collect the N inputs. We fit the Gaussian kernel $K \in \mathbb{R}^{N\times N}$, $K_{ij} = \exp(-\|x_i - x_j\|^2/2\sigma^2)$, with the model $\hat{K} = \phi(XW)\,V^\top$, where $W \in \mathbb{R}^{d\times R}$, $V \in \mathbb{R}^{N\times R}$, the search rank R is the hidden width, and ϕ acts entrywise. This is a *nonlinear* matrix factorization: for $\phi = \mathrm{id}$ it collapses to the bilinear low-rank model $\hat{K} = X(WV^\top)$.

Softplus and its expansion. The softplus $\phi(z) = \log(1 + e^z)$ has derivative $\phi' = \sigma$ (the logistic sigmoid). Since $\sigma - \frac{1}{2}$ is odd, the Maclaurin series of ϕ keeps only the constant,

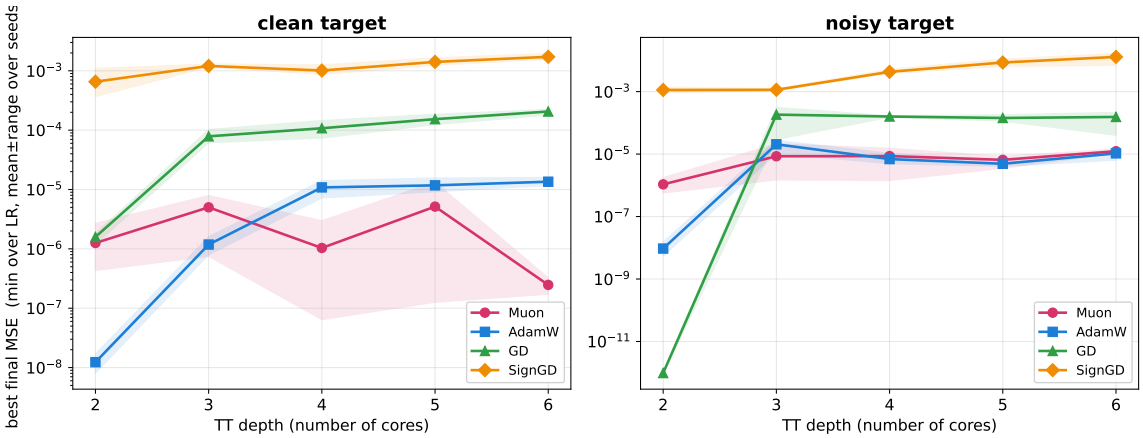

Figure 10: **Best-tuned reconstruction loss versus tensor-train depth.** Each curve shows the final MSE of one optimizer at its best learning rate (minimum over the grid) for a given depth, averaged over three seeds; shaded bands span the per-seed min–max. **Left:** clean, low-rank target. **Right:** noisy, higher-rank target. Search bond rank is fixed to the over-parameterized regime (R=12 clean, R=30 noisy). Lower is better; the y-axis is logarithmic.

linear, and even-order terms,

$$\phi(z) = \log 2 + \tfrac{1}{2}z + \tfrac{1}{8}z^2 - \tfrac{1}{192}z^4 + \mathcal{O}(z^6). \tag{16}$$

The surrogate used in the experiments, $\tilde{\phi}(z) = \log 2 + \frac{1}{2}z + \frac{1}{8}z^2$, is the truncation after the quadratic term (the cubic coefficient vanishes), and is accurate when the pre-activations XW are small, as enforced by the input scale.

The quadratic surrogate is an explicit feature factorization. With $A = \tilde{\phi}(XW)$, expanding term by term gives

$$\hat{K} = AV^\top = \underbrace{\log 2\, \mathbf{1}_{N\times R} V^\top}_{\text{constant } (\text{rank}\le 1)} + \underbrace{\tfrac{1}{2}(XW)V^\top}_{\text{rank}\le \min(R,d)} + \underbrace{\tfrac{1}{8}(XW)^{\odot 2}V^\top}_{\text{degree-2 lift}}, \tag{17}$$

where $\odot$ is the Hadamard product. The linear term is exactly the classical rank-R bilinear factorization, whose rank is further capped by $\text{rank}(X) \le d$. The Hadamard-square term injects degree-2 monomial features: each column of XW lies in the $\le d$-dimensional column space $\text{col}(X)$, so the columns of $(XW)^{\odot 2}$ lie in its symmetric square $\text{Sym}^2(\text{col}(X))$, of dimension $\le \binom{d+1}{2}$. Hence the model factorizes K through the fixed feature set $\{\mathbf{1}\} \cup \text{col}(X) \cup \text{Sym}^2(\text{col}(X))$, and

$$\text{rank}(\hat{K}) \;\le\; \min\Big(R,\; \tbinom{d+2}{2}\Big). \tag{18}$$

The quadratic model therefore *cannot exceed* rank $\binom{d+2}{2}$ ($= 21$ for $d = 5$) no matter how large the search rank R is.

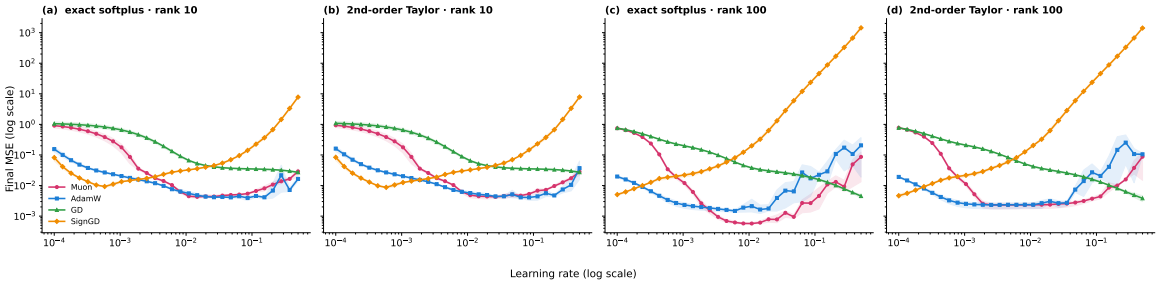

Figure 11: **Learning-rate stability across activation and capacity.** Final MSE (log) vs. learning rate (log) for four optimizers fitting the Gaussian kernel as $\hat{K} = \phi(XW)V^\top$; mean over three seeds, bands show min–max, divergent runs capped at 10^5. The activation ϕ is exact softplus or its second-order Taylor surrogate, and R is the search rank: **(a)** softplus, R=10; **(b)** Taylor, R=10; **(c)** softplus, R=100; **(d)** Taylor, R=100. At R=10 the rank bottleneck binds and the two activations match (a)≈(b); at R=100 the degree-2 surrogate is rank-capped (Eq. 18) while exact softplus is not, so (c) reaches a lower error than (d). Muon attains the lowest loss over the widest stable range; SignGD diverges at large learning rates.

Consequence for the comparison. Equation equation 18 predicts a sharp regime split, which Fig. 11 confirms. When $R \leq \binom{d+2}{2}$ (here $R = 10$), the search rank is the binding constraint and the truncation is invisible: exact softplus and its surrogate behave almost identically (panels (a) vs (b)). When $R > \binom{d+2}{2}$ (here $R = 100$), the degree-2 cap binds for the surrogate but not for exact softplus, so the exact activation attains a strictly lower error (panel (c) below (d)). In both regimes the activation is a mild perturbation of the same nonlinear-factorization family, while the search rank R governs how much of the kernel's polynomial spectrum the model can represent, and, empirically, is what separates the optimizers.

Appendix K. Analytical Solutions of the Matrix Factorization for AdamW and Muon

K.1. Problem Setup

We consider the Matrix Factorization objective function:

$$\mathcal{L}(U, V) = \frac{1}{2}\|UV^T - R\|_F^2, \tag{19}$$

where $U, V \in \mathbb{R}^{N\times r}$. At iteration t, we seek the optimal update step ΔU for the factor U (the derivation for V is symmetric). Let $G_t = \nabla_U \mathcal{L} = (U_t V_t^T - R)V_t$ be the gradient at step t.

Since the global optimization is intractable, iterative optimizers solve a **local proxy problem** at each step. We derive the analytical solution for these proxy problems. Throughout, v_t (lowercase) denotes the per-coordinate second-moment estimate, with entries $v_{t,ij}$; this is distinct from the factor V_t.

K.2. The AdamW Analytical Solution

Proposition 1 *Under the decoupled–weight-decay approximation of Loshchilov & Hutter (2017), the AdamW update is the analytical solution to minimizing the linearized loss subject to an adaptive Mahalanobis-distance trust region, with weight decay applied as a separate additive step.*

The Optimization Problem. We first consider the linearized loss penalized by the local curvature estimated by the diagonal second-moment estimate v_t:

$$\Delta U^\dagger = \underset{\Delta U \in \mathbb{R}^{N\times r}}{\arg\min} \left(\underbrace{\langle G_t, \Delta U \rangle}_{\text{Linear Descent}} + \underbrace{\frac{1}{2\eta} \sum_{i,j} \sqrt{v_{t,ij}}\, (\Delta U_{ij})^2}_{\text{Adaptive Trust Region}} \right). \tag{20}$$

Derivation of the trust-region step. The objective is separable across coordinates. Taking the derivative with respect to a single entry ΔU_{ij} and setting it to zero,

$$G_{t,ij} + \frac{1}{\eta} \sqrt{v_{t,ij}}\, \Delta U_{ij} = 0 \quad \Longrightarrow \quad \Delta U^\dagger_{ij} = -\,\eta\, \frac{G_{t,ij}}{\sqrt{v_{t,ij}}}. \tag{21}$$

Decoupled weight decay. AdamW then applies weight decay as a *separate* step that is not passed through the preconditioner:

$$\boxed{\Delta U^*_{ij} = -\,\eta \left(\frac{G_{t,ij}}{\sqrt{v_{t,ij}}} + \lambda\, U_{t,ij} \right).} \tag{22}$$

Remark (why decoupling matters). Had we instead folded the penalty $\frac{\lambda}{2}\|U_t + \Delta U\|_F^2$ directly into the proxy, the exact stationary point would be

$$\left(\frac{1}{\eta} \sqrt{v_{t,ij}} + \lambda \right) \Delta U_{ij} = -\big(G_{t,ij} + \lambda U_{t,ij}\big) \implies \Delta U_{ij} = -\,\eta\, \frac{G_{t,ij} + \lambda U_{t,ij}}{\sqrt{v_{t,ij}} + \eta\lambda}. \tag{23}$$

This *couples* the decay to the preconditioner (dividing the decay term by $\sqrt{v_{t,ij}}$) and is precisely Adam with L_2 regularization, *not* AdamW. AdamW is recovered by (i) neglecting $\eta\lambda$ in the denominator and (ii) decoupling the decay so that $\lambda U_{t,ij}$ is *not* rescaled by the curvature. Hence the boxed expression is an approximation, not the exact minimizer of the coupled proxy.

Conclusion. AdamW rescales every coordinate independently based on the diagonal curvature $\sqrt{v_{t,ij}}$, and adds an unpreconditioned weight-decay term.

K.3. The Muon Analytical Solution

Proposition 2 *The Muon update is the exact analytical solution to the Orthogonal Procrustes Problem. It finds the orthonormal-column update direction that aligns maximally with the* negative *gradient; this direction is the polar factor of $-G_t$.*

The Optimization Problem. We seek the matrix $O \in \mathbb{R}^{N\times r}$ closest to the negative gradient $-G_t$, constrained to have orthonormal columns:

$$O^* = \underset{O\,:\,O^TO=I_r}{\arg\min} \|(-G_t) - O\|_F^2. \tag{24}$$

The update is then $\Delta U = \eta\, O^*$.

K.3.1. Step-by-Step Derivation

Step 1: Expand the objective.

$$\| - G_t - O\|_F^2 = \mathrm{Tr}\big((G_t + O)^T(G_t + O)\big) \tag{25}$$

$$= \mathrm{Tr}(G_t^T G_t) + \mathrm{Tr}(O^T O) + 2\ \mathrm{Tr}(G_t^T O). \tag{26}$$

The term $\mathrm{Tr}(G_t^T G_t)$ is constant, and $\mathrm{Tr}(O^T O) = \mathrm{Tr}(I_r) = r$ is constant. The cross term carries a **plus** sign, so minimizing the norm is equivalent to **minimizing** $\mathrm{Tr}(G_t^T O)$, equivalently maximizing the alignment with the negative gradient, $\mathrm{Tr}\big((-G_t)^T O\big)$:

$$\min_{O\,:\,O^TO=I_r} \mathrm{Tr}(G_t^T O). \tag{27}$$

Step 2: Singular Value Decomposition. Let the (thin) SVD of the gradient be $G_t = P\Sigma Q^T$, where $P \in \mathbb{R}^{N\times r}$ has orthonormal columns ($P^TP = I_r$), $\Sigma \in \mathbb{R}^{r\times r}$ is diagonal with singular values $\sigma_i \geq 0$, and $Q \in \mathbb{R}^{r\times r}$ is orthogonal. Then

$$\mathrm{Tr}(G_t^T O) = \mathrm{Tr}\big(Q\Sigma P^T O\big) = \mathrm{Tr}\big(\Sigma\,(P^T O Q)\big), \tag{28}$$

using the cyclic property of the trace.

Step 3: Bound the matrix Z. Define $Z = P^TOQ \in \mathbb{R}^{r\times r}$. Each of P, O has orthonormal columns and Q is orthogonal, so each is a (semi-)isometry with spectral norm 1. Hence $\sigma_{\max}(Z) \leq \|P^T\|_2\, \|O\|_2\, \|Q\|_2 = 1$. (Note Z need *not* be orthogonal when $N > r$, since $OO^T \neq I_N$; we only require the norm bound.) In particular, every diagonal entry satisfies $|Z_{ii}| \leq \sigma_{\max}(Z) \leq 1$. The objective becomes

$$\mathrm{Tr}(\Sigma Z) = \sum_{i=1}^{r} \sigma_i Z_{ii}. \tag{29}$$

Step 4: Minimize. Since $\sigma_i \geq 0$ and $|Z_{ii}| \leq 1$, the sum $\sum_i \sigma_i Z_{ii}$ is **minimized** when $Z_{ii} = -1$ for all i, i.e. $Z = -I_r$ (attained by the admissible choice $O = -PQ^T$). Solving back,

$$P^TOQ = -I_r \tag{30}$$

$$O^* = -\,P\,Q^T. \tag{31}$$

$$\boxed{O^* = -\,PQ^T, \qquad \Delta U = \eta\, O^* = -\,\eta\, PQ^T.} \tag{32}$$

Equivalently, O^* is the polar factor of the *negative* gradient $-G_t$, and the update $\Delta U = -\eta\, PQ^T$ is a genuine descent step (it removes the singular-value magnitudes of G_t, keeping only its PQ^T "direction").

K.3.2. Connection to Newton–Schulz

The matrix PQ^T is the **polar factor** of G_t. Computing the SVD at every step is expensive, so Muon computes PQ^T via the Newton–Schulz iteration

$$X_{k+1} = \frac{1}{2} X_k \big(3I - X_k^T X_k\big), \qquad X_0 = \frac{G_t}{\|G_t\|_2}, \tag{33}$$

which converges quadratically to PQ^T. The descent update is then $\Delta U = -\eta\, X_\infty = -\eta\, PQ^T$, consistent with the boxed solution above.

K.4. Implications for Matrix Factorization

This derivation highlights the fundamental difference in update scaling. Both updates act on matrices in $\mathbb{R}^{N\times r}$.

1. AdamW scaling (dimension dependent). Assuming active gradients ($G_{ij}/\sqrt{v_{ij}} \approx \pm 1$) and negligible weight decay, the squared norm of the update scales with the number of entries $N \times r$:

$$\|\Delta W_{\text{Adam}}\|_F^2 \approx \sum_{i,j} \eta^2 = \eta^2 (N \cdot r). \tag{34}$$

2. Muon scaling (rank dependent). Since $O^* = -PQ^T$ has orthonormal columns, $(O^*)^T O^* = QP^T PQ^T = I_r$, so the sign is immaterial to the norm:

$$\|\Delta W_{\text{Muon}}\|_F^2 = \eta^2 \ \mathrm{Tr}\big((O^*)^T O^*\big) = \eta^2 \ \mathrm{Tr}(I_r) = \eta^2 r. \tag{35}$$

Conclusion.

$$\frac{\|\Delta W_{\text{Adam}}\|_F}{\|\Delta W_{\text{Muon}}\|_F} \approx \sqrt{N}. \tag{36}$$

In our experiments with $N = 100$, AdamW naturally takes steps ≈ 31.6 times larger than Muon for the same learning rate η. This explains the necessity of scaling $\eta_{\text{Muon}} \approx \sqrt{N} \cdot \eta_{\text{AdamW}}$ to achieve comparable convergence rates.