Variance Collapse Predicts When Gate Density Diverges by Activation Class

Anonymous Author(s)
Affiliation
Address
email

Abstract

Whether the fraction of gradient-carrying units (*gate density*) rises or falls during training is often treated as activation-specific folklore. We show it is governed by a single, derivable quantity: whether the post-BatchNorm pre-activation variance collapses during training. Under coupled-weight-decay optimizers (SGD, Adam), variance collapses for every activation tested; whether this drives gate density toward 0 or 1 depends only on each activation's fixed threshold-crossing location, computed once via automatic differentiation with no training involved. On an architecture-fixed ablation (48 independent runs), ReLU declines while GELU, SiLU, and Mish all rise instead, confirmed at the level of individual channels (12 of 12 activation$\times$seed cells, using a variance-normalized margin). Under AdamW's decoupled weight decay, variance no longer collapses — it is flat or growing, not shrinking, for every activation; fed AdamW's own measured statistics, the identical predictor with no new free parameters correctly anticipates the resulting uniform decline (12 of 12 cells), turning an apparent exception into a confirmation of the mechanism's scope rather than a separate, unexplained finding. The smooth-activation rise further generalizes across scale (a 365-class real-photograph dataset) and architecture family (two non-BatchNorm, non-convolutional models), while the ReLU-specific decline and a separate representational-rank-rise observation turn out to be specific to convolutional networks. We also report, rather than omit, three honest negative results: gate density does not discriminate via representational rank, is actively misleading for channel pruning on smooth activations, and does not predict low-data fine-tuning accuracy.

1 Introduction

The local derivative of an activation function gates how much gradient signal a unit carries backward. For ReLU specifically, individual units are known to lose this capacity permanently during training — the dying-ReLU phenomenon [1]. We ask a different, comparative question: across ordinary, successful training, does the *population-level fraction* of gradient-carrying units (*gate density*) move in the same direction for every activation function, or does the direction depend on the activation's shape?

We answer this with a hook-based method that recovers a network's exact pointwise gate $\Gamma(x) = |f'(x)|$ for *any* elementwise activation, without modifying the architecture or knowing f' in closed form (Section 3). Tracking gate density through standard supervised training, the direction is not shared: ReLU-based CNNs decline monotonically while GELU-based networks rise, and holding the architecture exactly fixed and swapping only the activation reproduces this split on identical backbones, ruling out an architecture confound (Section 4).

Submitted to 40th Conference on Neural Information Processing Systems (NeurIPS 2026). Do not distribute.

We then derive, rather than only describe, why (Section 5). BatchNorm's scale shrinks under weight decay and a decaying learning rate — an established consequence of training scale-invariant layers [10, 11] — collapsing the post-normalization pre-activation variance for every activation alike. Whether a shrinking-variance gate concentrates toward 0 or 1 depends only on a fixed, activation-specific threshold-crossing point, computable with no training at all. We verify this mechanism's central assumption directly, confirm it at the level of individual channels rather than only a population average, and then ask the question this immediately raises: **is the mechanism only descriptive within one optimizer, or is it actually predictive across optimizers?** Feeding the unmodified, already-validated predictor AdamW's own measured statistics — where decoupled weight decay prevents the variance collapse the mechanism's first step relies on — it correctly anticipates AdamW's qualitatively different outcome (Section 5.3). This is the paper's central contribution: one mechanism, one predictor, whose governing input (does variance collapse, set by the coupled-vs-decoupled weight-decay axis) determines the output across every optimizer tested.

Finally, we test generalization deliberately rather than assuming it (Section 6), and report two non-findings and one null result exactly as the data shows them (Sections 7, 8), including where this complicates a clean narrative.

Contributions

1. **A general method for measuring the exact pointwise gradient gate** of any elementwise activation without modifying the network, validated against the known ReLU-at-initialization baseline.
2. **Evidence that gate density's training-time direction is activation-class-dependent, not architecture-dependent**: consistent decline for ReLU and rise for GELU/SiLU/Mish, confirmed via an architecture-matched ablation (48 independent runs) and robust to normalization scheme and measurement threshold.
3. **A derived, cross-optimizer predictive mechanism**, verified at the population level, the individual-channel level, and — decisively — by feeding it a different optimizer's real measured dynamics and confirming it anticipates the qualitatively different outcome with zero new free parameters.
4. **Generalization tested along three axes (optimizer, scale, architecture family)**, with honest, characterized scope boundaries where the pattern is CNN-specific rather than general, alongside three explicitly negative findings.

2 Related Work

Dying ReLU [1] documents individual ReLU units becoming permanently inactive, treated as a pathology with established remedies. We measure a continuous, population-level statistic that co-occurs with *rising* accuracy, and generalize the question to activations for which "death" is not well-defined. **Activation sparsity**: trained ReLU networks develop increasing forward-pass sparsity [2]; we measure backward-pass gate sparsity as an epoch-resolved trajectory. **Lottery Ticket Hypothesis** [3] prunes via static weight masks; our gate-density statistic is continuous and non-interventional, and we test a pruning use case directly with a negative result (Section 7). **Rank/Neural Collapse** [4, 5] and **self-supervised representation collapse** [6, 7] use related vocabulary for unrelated, in one case opposite-signed, phenomena; we flag the distinction explicitly (Appendix A.5). **Gradient inversion attacks** [14, 8, 9] are reused unmodified in an exploratory probe (Appendix A.6). **BatchNorm scale dynamics under weight decay** [10, 11] is the established result our mechanism applies, not re-derives. The gate vocabulary itself originates in an earlier synthetic phase-transition study (Appendix A.9), connected here via an exact mathematical correspondence (Section A.3).

3 Method

For an elementwise activation $y = f(x)$, the chain rule gives, exactly, `grad_input` $=$ `grad_output` $\cdot$ $f'(x)$. A backward hook on any activation module observes both tensors; their elementwise ratio recovers $\Gamma(x) = |f'(x)|$ exactly, for any f, without knowing its closed form. **Gate density** at threshold θ is the fraction of elements with $\Gamma(x) > \theta$ on a fixed evaluation batch, re-measured every epoch on the *same* batch.

We instrument CIFAR-native ResNet-18/50 and VGG-11 (a standard 3×3-stride-1 stem, since the canonical ImageNet stem destroys a 32×32 image's resolution immediately) and canonical ViT-B/16 and ConvNeXt-Tiny. The method is validated against the only available ground truth — ReLU's $\approx 50\%$ active-fraction at BatchNorm-normalized random initialization — across 20 seeds and three architectures (Appendix A.1).

To separate activation effects from architecture effects, we hold a ResNet-18/VGG-11 skeleton exactly fixed (identical depth, width, stem, normalization placement, parameter count) and vary only the activation module via a single constructor argument; the same mechanism extends to a BatchNorm-vs-GroupNorm ablation. **Statistical methodology**: every directional claim is computed at the level of the independent seed trajectory — one statistic (a correlation or a delta) per (architecture, activation, dataset, seed) — and tested via an exact binomial sign test across seeds; epochs within a single run are never pooled for a hypothesis test, since they are not independent observations.

4 The Activation-Class Direction Split

Training ResNet-18/50/VGG-11 with SGD on CIFAR-10/100 (18 independent trajectories: 3 architectures $\times$ 2 datasets $\times$ 3 seeds), gate density falls and effective rank rises in *every* trajectory (sign test, $p = 3.8 \times 10^{-6}$ for both) while test accuracy rises throughout. In GELU-based ViT-B/16, gate density instead rises toward a ceiling (4 of 4 runs).

This ReLU-vs-ViT contrast confounds activation with architecture family, normalization, and other structural differences. Holding ResNet-18/VGG-11 exactly fixed and swapping only the activation (ReLU, GELU, SiLU, Mish; 48 independent runs) resolves this directly: every one of 12 ReLU runs declines, every one of 12 GELU/SiLU/Mish runs rises (sign test $p = 2.44 \times 10^{-4}$ per activation). **The direction flips with the activation, not the architecture family.** Effective rank rises in all 48 of 48 runs regardless of activation ($p = 3.55 \times 10^{-15}$) — it does not track the activation-class split and is reported separately as a non-discriminating observation (Section 7).

The direction is robust to two further checks (full detail and figures in Appendix A.2). **Normalization**: replacing every BatchNorm layer with GroupNorm on the identical ResNet-18 skeleton (4 activations, 3 seeds, 24 runs) leaves the direction unchanged — 3 of 3 ReLU runs decline and 9 of 9 smooth-activation runs rise under *both* normalization schemes — narrowing what could be driving the phenomenon to the activation function itself, although Section 5 shows BatchNorm's presence specifically supplies the mechanism by which the direction is set. **Threshold**: sweeping the gate-density threshold $\theta \in \{0.001, \ldots, 0.10\}$ (60 checks) gives zero sign flips, although the *magnitude* of the smooth-activation rise is small at the standard threshold (< 1 point against ReLU's ~ 13-point fall) and grows to $+4.7$ to $+5.5$ points at the strictest threshold tested — an asymmetry explained in Section 5 by quantile compression of the raw gate-value distribution (the gate-value distribution compresses from both tails toward its center for smooth activations, simultaneously raising the count above threshold while lowering the mean; ReLU's distribution remains exactly bimodal at $\{0, 1\}$, so its mean and thresholded fraction must move together).

5 Mechanism

5.1 A derived, tested account

For a channel with pre-BatchNorm output u, BatchNorm computes

$$z = \gamma \cdot \frac{u - \mathrm{mean}(u)}{\sqrt{\mathrm{var}(u) + \epsilon}} + \beta \quad \Rightarrow \quad \mathrm{mean}(z) \approx \beta, \quad \mathrm{var}(z) \approx \gamma^2, \tag{1}$$

to first order, by BatchNorm's own definition. A convolutional layer immediately followed by BatchNorm is scale-invariant, so under SGD with weight decay its effective scale is pulled toward an equilibrium that decreases as the learning rate decays [10, 11]; this is consistent with γ's observed monotonic shrinkage in every one of 12 trajectories tested ($p = 2.44 \times 10^{-4}$; -81% ReLU, -73 to -74% GELU/SiLU/Mish).

Define $z_{\mathrm{low}}(\theta) = \inf\{z : |f'(z)| > \theta\}$, the point where a gate crosses θ on its negative tail — fixed and computable via automatic differentiation, no training involved. If $z \sim \mathcal{N}(\mu, \sigma^2)$ with $\sigma \to 0$ and μ drifts by a *similar* amount regardless of activation, then $\mathrm{active\_frac}(\theta) = P(|f'(z)| > \theta) \to 0$ or 1

depending only on which side of $z_{\text{low}}(\theta)$ the shared μ lands on. We tested the shared-drift assumption directly (9 activations spanning the empirical sign transition, 3 seeds): μ flips from small-positive at initialization to a narrow, shared negative band (-0.041 to -0.066) by epoch 24 regardless of activation shape, and comparing this measured drift to each activation's independently-computed z_{low} predicts the observed direction in 9 of 9 cases (Appendix A.3), including correctly flagging in advance which case (closest to the boundary) should be least reliable.

5.2 Direct per-channel verification

The test above is at the population level (one pooled μ per activation/seed/epoch). Logging per-channel μ_c, σ_c and active-gradient fraction for every channel of ResNet-18 (4 activations, 3 seeds, 3,904 channels per run), a raw per-channel margin $\mu_c - z_{\text{low}}$ gives a confusing, activation-split correlation against active-gradient fraction: strong for ReLU but weakly *negative* for GELU/SiLU/Mish. Diagnosis: ReLU's z_{low} sits within 0.03 standard deviations of its channels' own mean (the threshold lives at the live center of the distribution), while the smooth activations' z_{low} sits 0.9–1.5 standard deviations away; σ shrinks $\sim 3\times$ for every activation, while μ also drifts slightly negative for every activation. For ReLU this μ-drift directly crosses the live threshold; for the smooth activations the same small μ-drift instead pulls the *raw* margin down even as σ's much larger relative shrinkage pulls active-gradient fraction *up* by thinning the sub-threshold tail. The raw margin conflates these forces; it does not capture σ.

Replacing the raw margin with the σ-normalized z-score $(\mu_c - z_{\text{low}})/\sigma_c$ removes the confound: all four activations now show strong, uniform, positive per-channel correlation (90–96% of channels per seed; 12 of 12 activation$\times$seed cells majority-positive, sign test $p = 2.44 \times 10^{-4}$). **Thin-margin activations (ReLU) are driven by μ crossing a threshold at their distribution's center; thick-margin activations (GELU, SiLU, Mish) are driven by σ-shrinkage thinning a distant tail** — two routes to the same threshold-crossing logic, unified by one σ-normalized quantity.

5.3 A unified prediction across optimizers

Section 6.1 below reports that under AdamW the activation-class split collapses into a uniform decline. We treat this as a falsifiable prediction problem: **does the exact, unmodified per-channel predictor above, fed AdamW's own measured $\mu_c(t)$ and $\sigma_c(t)$, anticipate AdamW's outcome with no new free parameters?** The per-channel correlation itself remains positive under AdamW (72–90% of channels per seed) — expected, since the σ-normalized margin and active-gradient fraction are related through the same monotonic threshold-crossing relationship regardless of which optimizer produced the underlying μ, σ trajectory, so this step confirms the relationship is optimizer-invariant rather than constituting new evidence on its own. The substantive test is whether the margin's *trend* matches AdamW's observed decline: for 12 of 12 activation$\times$seed cells it does (68–89% channel-level agreement per cell; population-pooled margin deltas negative for all four activations, matching the observed active-gradient-fraction declines exactly in sign).

The mechanistic reason is direct, not inferred: per-channel σ shrinks ~ 67–77% under SGD for every activation (driving the smooth-activation rise via tail-thinning), but under AdamW it is flat for ReLU (-0.9%) and *grows* for every smooth activation ($+2.2$ to $+4.8\%$, Table 1). With σ flat-or-growing and μ still drifting slightly negative, the Gaussian threshold-crossing account requires the margin to decline for every activation — which is what is observed. **This confirms the account is quantitatively faithful to AdamW's actual dynamics, not that the predictor is an oracle independent of how margin and active-gradient fraction are related**: the relationship between them is close to definitional once $\mu, \sigma, z_{\text{low}}$ are fixed, and what is being tested is whether the *measured inputs* (which are not fixed, and were not previously known for AdamW) take AdamW to the predicted regime. They do, in every cell tested.

The mechanism is therefore one predictor whose governing input — whether σ collapses, itself set by coupled (SGD, Adam) vs. decoupled (AdamW) weight decay — determines the output across every optimizer tested, rather than three disconnected, optimizer-specific observations. A first-principles account of *why* decoupling prevents σ-collapse remains open (Section 8).

A 13-condition continuous smoothness sweep (ReLU; LeakyReLU at four slopes; PReLU; Softplus at $\beta \in \{50, 20, 10, 5\}$; GELU; SiLU; Mish; 3 seeds), whose Softplus(β) family has a derivative *exactly* equal to the logistic sigmoid used by the originating synthetic theory (Appendix A.9), confirms a

Table 1: Per-channel σ change, epoch 0 to 24, by activation and optimizer. The mechanistic reason variance-driven activations stop rising under AdamW.

Activation	SGD	AdamW
ReLU	-77.0%	-0.9% (flat)
GELU	-67.5%	$+2.2\%$ (grows)
SiLU	-66.2%	$+4.8\%$ (grows)
Mish	-66.9%	$+4.7\%$ (grows)

monotonic dose-response between an initialization-time smoothness index ($\mathrm{Var}[f'(x)]$ on the fixed evaluation batch) and the observed trend (seed-level Spearman $\rho = -0.71$, $p = 6 \times 10^{-7}$, $n = 37$; condition-level $\rho = -0.73$, $p = 0.0045$, $n = 12$). The sign transition occurs within the Softplus(β) family itself, between $\beta = 50$ (decline) and $\beta = 20$ (rise) — predicted exactly by the z_{low} values above. One disclosed exception (PReLU, whose learned slope changes its effective smoothness during training) and the full per-condition breakdown are in Appendix A.3.

6 Generalization

6.1 Optimizer

The architecture-fixed ablation replicates exactly under Adam, coupled weight decay matching SGD's update structure (48 of 48 runs, $p = 4.88 \times 10^{-4}$ per activation, magnitudes matching SGD). Under AdamW, decoupled weight decay, all four activations decline instead of diverging by class (48 of 48 runs, $p = 4.88 \times 10^{-4}$) — the pattern Section 5.3 shows the mechanism anticipates rather than merely accommodates.

6.2 Scale and architecture family

We test, rather than assume, two further axes (full detail and per-axis figures in Appendix A.4). **Scale**: on Tiny-ImageNet-200 (200 classes, 64×64), ReLU's decline and rank-rise replicate exactly; the smooth-activation rise replicates in net displacement (9 of 9 / 8 of 9 seeds) though a whole-trajectory trend statistic loses power on this already-small effect once trajectories become non-monotonic at this scale. On a 365-class, real-photograph subsample of Places365-Standard (ResNet-50, 96×96, 2 seeds), the *full* pattern — large ReLU decline, small smooth-activation rise, universal rank-rise — replicates exactly, at a class count comparable to ImageNet-1k (untested directly: registration-gated, no anonymous scriptable download). **Architecture family**: on two non-convolutional, LayerNorm-only architectures (an 8-block MLP-Mixer [12] and a 6-block Transformer-Encoder [13]), the smooth-activation rise replicates cleanly (6 of 6 runs each, $p = 0.03125$) but **the ReLU decline does not** — it is small and architecture-dependent outside the CNN family, as is the separate universal-rank-rise observation.

The smooth-activation rise generalizes broadly across optimizer, scale, normalization, and architecture family; the ReLU decline and the universal-rank-rise observation are CNN-specific — a scope narrowing discovered by deliberately testing outside the original setup, not assumed.

7 Honestly-Reported Non-Findings

Effective rank rises in every condition tested in this paper — 114+ independent runs regardless of activation, architecture, dataset, or threshold (pooled sign test $p < 10^{-14}$) — and has no power to discriminate anything this paper argues for; we do not list it as a contribution (full discussion, including its explicit distinction from Neural Collapse and self-supervised representation collapse, in Appendix A.5).

Testing whether end-of-training gate density predicts pruning importance (ResNet-18/CIFAR-10, 4 activations, 3 seeds, global channel pruning across 1920 channels, ratios 10–70%, no retraining), gate-density and magnitude pruning are statistically indistinguishable for ReLU (paired-t $p = 0.55$, an informative equivalence) but gate-density pruning is dramatically and significantly *worse* than

magnitude pruning for every smooth activation ($p \leq 1.5 \times 10^{-4}$; 65–80 accuracy points lost at 10% pruning vs. under 1.5 for magnitude pruning, Figure 7) — **gate density should not be used as a general-purpose channel-importance proxy**; for three of four activations tested, doing so is actively harmful (full detail in Appendix A.5).

An exploratory privacy probe and a pre-registered low-data fine-tuning test are reported in full in Appendices A.6 and A.7: gate stiffness reduces one gradient-inversion attack's convergence but not a stronger one (attack-dependent, not a general privacy claim), and final gate density does not predict low-data fine-tuning accuracy (a genuine, pre-registered null: pooled Pearson $r = 0.24$, $p = 0.26$; Spearman $\rho = -0.33$, $p = 0.11$, the two not even agreeing in sign).

8 Limitations

The mechanism explains *direction*, not magnitude or ultimate root cause: we do not derive a closed form for trend magnitude, and an attempted derivation of why the pre-activation mean drifts negative (Appendix A.3) accounts for the order of magnitude of the shared drift but not its residual, statistically real cross-activation variation. We do not have a first-principles account of why AdamW's decoupled weight decay specifically prevents σ-collapse, only that it empirically does and that the mechanism correctly anticipates the consequence. ConvNeXt-Tiny is excluded from all claims (undertrained, inconsistent rank sign across datasets; Appendix A.4). Architecture families beyond CNNs, MLP-Mixer, and Transformer-Encoder remain untested, as does full-scale ImageNet-1k directly.

9 Conclusion

Gate density diverges by activation class during ordinary training under coupled-weight-decay optimizers, and this divergence is derived, not asserted: BatchNorm's scale shrinks under weight decay, collapsing pre-activation variance for every activation alike, and whether this collapse drives gate density toward 0 or 1 depends only on a fixed, activation-specific threshold-crossing point, verified at the population level, the individual-channel level, and across a continuous smoothness sweep. The decisive test is predictive rather than merely descriptive: fed a different optimizer's (AdamW's) own measured statistics, the identical mechanism anticipates a qualitatively different, uniform-decline outcome with zero new free parameters, because decoupled weight decay prevents the variance collapse the mechanism's first step requires. The smooth-activation half of this finding generalizes broadly across optimizer, scale, and architecture family; the ReLU-specific decline and a separate rank-rise observation are CNN-specific. We report, rather than omit, every place this story does not extend cleanly. The result is a bounded, mechanistically-derived and mechanistically-*predictive* account of activation-class-dependent gate dynamics — not a universal theory of gradient dynamics, and we do not claim it as one.

References

References

[1] Lu, L., Shin, Y., Su, Y., and Karniadakis, G. E. (2019). Dying ReLU and initialization: theory and numerical examples. *Communications in Computational Physics*, 28(5), 1671–1706.

[2] Glorot, X. and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. *Proceedings of AISTATS*, 249–256.

[3] Frankle, J. and Carbin, M. (2019). The lottery ticket hypothesis: finding sparse, trainable neural networks. *International Conference on Learning Representations (ICLR)*.

[4] Dong, Y., Cordonnier, J.-B., and Loukas, A. (2021). Attention is not all you need: pure attention loses rank doubly exponentially with depth. *International Conference on Machine Learning (ICML)*.

[5] Papyan, V., Han, X. Y., and Donoho, D. L. (2020). Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40), 24652–24663.

[6] Hua, T., Wang, W., Xue, Z., Ren, S., Wang, Y., and Zhao, H. (2021). On feature decorrelation in self-supervised learning. *International Conference on Computer Vision (ICCV)*.

[7] Jing, L., Vincent, P., LeCun, Y., and Tian, Y. (2022). Understanding dimensional collapse in contrastive self-supervised learning. *International Conference on Learning Representations (ICLR)*.

[8] Zhao, B., Mopuri, K. R., and Bilen, H. (2020). iDLG: improved deep leakage from gradients. *arXiv preprint arXiv:2001.02610*.

[9] Geiping, J., Bauermeister, H., Dröge, H., and Moeller, M. (2020). Inverting gradients—how easy is it to break privacy in federated learning? *Advances in Neural Information Processing Systems*, 33.

[10] van Laarhoven, T. (2017). L2 regularization versus batch and weight normalization. *arXiv preprint arXiv:1706.05350*.

[11] Hoffer, E., Banner, R., Golan, I., and Soudry, D. (2018). Norm matters: efficient and accurate normalization schemes in deep networks. *Advances in Neural Information Processing Systems*, 31.

[12] Tolstikhin, I., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., Lucic, M., and Dosovitskiy, A. (2021). MLP-Mixer: an all-MLP architecture for vision. *Advances in Neural Information Processing Systems*, 34.

[13] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

[14] Zhu, L., Liu, Z., and Han, S. (2019). Deep leakage from gradients. *Advances in Neural Information Processing Systems*, 32.

[15] Donoho, D. L. (2006). Compressed sensing. *IEEE Transactions on Information Theory*, 52(4), 1289–1306.

A Technical Appendices and Supplementary Material

This appendix is supplementary: the main text stands on its own. Every result summarized in the main text is given in full here, with the exact numbers, figures, and experimental detail that did not fit the 9-page limit.

A.1 Method detail: validation and architecture-fixed ablation

We validate the gate-recovery method against the only available ground truth — ReLU's known $\approx 50\%$ active-fraction at BatchNorm-normalized random initialization — across 20 seeds and three architectures (Table 2).

Table 2: Active-gradient-fraction at random initialization, 20 seeds, $\theta = 0.01$. ReLU architectures cluster near the theoretical $\approx 50\%$ expectation; GELU-based architectures are near-ceiling; ViT-B/16 shows vanishing gradients at default initialization (resolved as an initialization artifact below, not a property of depth).

Architecture	Mean active-fraction	Std
ResNet-18	0.555	0.156
ResNet-50	0.529	0.116
VGG-11	0.618	0.141
ConvNeXt-Tiny	0.994	0.004
ViT-B/16	NaN (vanishing at init)	—

The ViT-B/16 vanishing-gradient finding is an initialization artifact, not depth-related. A dedicated falsification sweep across depth (1–12 layers), initialization scheme (default vs. truncated-normal $\mathrm{std} = 0.02$), and batch size (4–64), replicated independently twice, finds gradients vanish in 100% of seeds at *every* depth under default initialization and in 0% of seeds at every depth under the standard fix. This is unrelated to, and should not be conflated with, the training-time direction split studied in the main paper.

Architecture-fixed activation ablation, method. To separate activation effects from architecture effects (Section 4), we hold the ResNet-18 and VGG-11 skeletons exactly fixed — identical depth, width, stem, BatchNorm placement, and parameter count, verified identical across activations — and vary only the activation module via a single constructor argument. The same mechanism extends to the BatchNorm-versus-GroupNorm ablation in Appendix A.2 (replacing every `BatchNorm2d` with `GroupNorm`, 32 groups or fewer for narrow layers) on the identical skeleton.

A.2 BatchNorm and threshold robustness

Replacing every BatchNorm layer with GroupNorm on the identical ResNet-18 skeleton (4 activations, 3 seeds, 24 runs) leaves the direction unchanged: 3 of 3 ReLU runs decline and 9 of 9 smooth-activation runs rise under *both* normalization schemes (Figure 8). Magnitude is noisier under GroupNorm, particularly for Mish, and final accuracy is $\sim$2–4 points lower — both attributable to GroupNorm being unmatched in hyperparameters at this scale, not evidence about the underlying mechanism. BatchNorm is not a necessary condition for the phenomenon, narrowing what could be driving it to the activation function itself — though Section 5 shows BatchNorm's presence *does* supply the specific mechanism by which the direction is set in the main experiments.

The active-gradient-fraction statistic depends on a threshold θ. Sweeping $\theta \in \{0.001, 0.005, 0.01, 0.05, 0.10\}$ on the architecture-fixed ResNet-18 ablation (4 activations, 3 seeds, 60 checks) finds zero sign flips: ReLU declines and every smooth activation rises at every threshold tested. The magnitude of the effect is asymmetric: at the originally-used threshold ($\theta = 0.01$), the smooth-activation rise is small in absolute terms (< 1 percentage point) against ReLU's $\sim$13-point fall; the effect becomes comparably large only at the strictest threshold tested ($\theta = 0.10$: $+4.7$ to $+5.5$ points).

The raw, threshold-free gate magnitude (`gate_mean`) declines for every activation tested, including the ones whose thresholded gate density rises (ReLU -25.5%, GELU -15.3%, SiLU -9.0%, Mish -8.4%). This is not a contradiction: per-channel quantile analysis shows the gate-value distribution compresses from both tails toward its center for smooth activations (e.g. GELU's 5th percentile rises from 0.041 to 0.076 while its 95th percentile falls from 1.10 to 0.88 over training), simultaneously raising the count above a near-zero threshold while lowering the mean; ReLU's distribution remains exactly bimodal at $\{0, 1\}$ throughout, so its mean and thresholded fraction are mathematically identical and must move together. Section 5 explains why this compression happens for every activation but produces opposite threshold-crossing outcomes.

A.3 Mechanism detail: threshold-crossing table, shared-drift test, smoothness sweep, and the mu-drift derivation

Table 3 gives $z_{\text{low}}(\theta = 0.10)$, computed exactly via automatic differentiation, against the empirically observed gate-density trend.

Table 3: $z_{\text{low}}(\theta = 0.10)$, computed exactly via autograd, vs. the empirically observed gate-density trend.

Activation	z_{low}	Observed ρ(epoch, gate density)	Direction
ReLU / LeakyReLU (all slopes)	0.000	-0.96 to -0.99	decline
Softplus $\beta = 50$	-0.044	-0.59	decline
Softplus $\beta = 20$	-0.110	$+0.73$	rise
Softplus $\beta = 10$	-0.219	$+1.00$	rise
Softplus $\beta = 5$	-0.439	$+1.00$	rise
GELU	-0.554	$+1.00$	rise
SiLU	-0.912	$+0.99$	rise
Mish	-0.891	$+0.97$	rise

We tested the shared-drift assumption rather than assuming it: logging signed pre-activation mean (forward-hook only, no backward pass) for 9 representative activations spanning the transition, 3 seeds, 25 epochs. At initialization μ is small and positive for every activation (0.014–0.033); by epoch 6 it has flipped negative for every single activation tested and remains in a narrow, shared

band through epoch 24: -0.041 (SiLU) to -0.066 (Mish) — a common drift, not nine separate trajectories.

Table 4: Mechanism validation: predicted vs. observed direction, derived from the measured shared μ-drift and the independently-computed z_{low}. 9 of 9 predictions correct.

Activation	μ(epoch 24)	z_{low}	Margin	Predicted	Match
ReLU	-0.0458	0.000	-0.0458	decline	✓
LeakyReLU(0.01)	-0.0478	0.000	-0.0478	decline	✓
Softplus $\beta = 50$	-0.0476	-0.044	-0.0036	decline	✓
Softplus $\beta = 20$	-0.0568	-0.110	$+0.0532$	rise	✓
Softplus $\beta = 10$	-0.0635	-0.219	$+0.1555$	rise	✓
Softplus $\beta = 5$	-0.0586	-0.439	$+0.3804$	rise	✓
GELU	-0.0508	-0.554	$+0.5032$	rise	✓
SiLU	-0.0413	-0.912	$+0.8707$	rise	✓
Mish	-0.0655	-0.891	$+0.8255$	rise	✓

Smoothness sweep. Softplus's derivative is exactly the logistic sigmoid, $\text{softplus}'(x;\beta) = \text{sigmoid}(\beta x)$ — the same gate formalism used by the originating synthetic theory (Appendix A.9), with β playing the role of that theory's stiffness parameter α; this is an exact mathematical correspondence, not an analogy. A 13-condition sweep (ReLU; LeakyReLU at four slopes; PReLU; Softplus at $\beta \in \{50, 20, 10, 5\}$; GELU; SiLU; Mish; 3 seeds; 25 epochs) tests whether smoothness predicts the trend in a continuous dose-response, using $\text{Var}[f'(x)]$ at initialization on the fixed evaluation batch as a smoothness index (trajectories plotted in Figure 5 in the main text).

The relationship is significant at the seed-trajectory level for both trend shape ($\rho = -0.710$, $p = 6.1 \times 10^{-7}$) and magnitude ($\rho = -0.605$, $p = 5.7 \times 10^{-5}$), and at the condition level for trend shape ($\rho = -0.731$, $p = 0.0045$); the condition-level magnitude correlation does not individually reach $p < 0.05$ at $n = 12$ ($p = 0.122$) but its bootstrap 95% CI, $[-0.775, -0.401]$, excludes zero. PReLU is a genuine exception: its smoothness index at initialization (0.261) is the highest of all 13 conditions, yet it rises. This reflects a limitation of measuring smoothness only at initialization: PReLU's negative-slope parameter is learned and its effective smoothness changes substantially during training (gate variance falls from ∼0.24–0.28 to ∼0.08–0.12), unlike fixed-shape activations.

The mu-drift derivation. The mechanism above explains direction given that μ drifts negative; it does not explain *why*. This project's optimizer applies weight decay to every parameter, including BatchNorm's β. For a parameter evolving under plain SGD with weight decay, $\theta_{t+1} = \theta_t - \eta(g_t + \lambda\theta_t)$, where g_t is the task gradient and λ the decay coefficient: if g_t has a roughly-constant average value $\bar{g}$ over a window where η is locally constant, the system relaxes toward a quasi-equilibrium $\theta^* \approx -\bar{g}/\lambda$. For BatchNorm's β, $g_t = dL/d\beta = \sum_{\text{batch}} dL/dz$ — the gradient flowing into the pre-activation, summed over the batch. This is a generic fact about SGD with L2 regularization, not specific to BatchNorm's scale-invariance (unlike the γ argument, which needs scale-invariance specifically). The falsifiable prediction: if $\bar{g}$ is set mainly by the shared architecture/task rather than by each activation's own shape, β's equilibrium value should be nearly identical across very different activation functions (same $\lambda = 5 \times 10^{-4}$ for all) and should not correlate with activation-specific shape properties like z_{low} or an initialization-time smoothness index.

Tested against the same 9-activation drift data above: μ at epoch 24 ranges only -0.065 to -0.041 (spread 0.024) across activations whose z_{low} spans -0.00005 to -1.234 (four orders of magnitude) — the narrowness this account predicts. The cross-activation variation is nonetheless statistically real: one-way ANOVA across the 9 activations, $F = 29.7$, $p = 7.7 \times 10^{-9}$; between-activation standard deviation (0.0084) is $3.5\times$ the within-activation (seed) standard deviation (0.0024). This residual variation does not correlate with z_{low} (Pearson $r = 0.26$, $p = 0.51$; Spearman $\rho = 0.28$, $p = 0.46$) nor with the initialization-time smoothness index (Pearson $r = 0.15$, $p = 0.71$; Spearman $\rho = 0.30$, $p = 0.43$). The equilibrium argument correctly predicts the order of magnitude of the shared drift band and gives it a mechanistic reason (shared λ, roughly shared $\bar{g}$) rather than leaving it as an unexplained empirical regularity; it does not explain the residual, statistically significant cross-activation variation within that band. We tested two natural candidates and stopped rather than searching for a third that fits.

A.4 Generalization detail: scale, architecture family, and ConvNeXt

Tiny-ImageNet-200. ResNet-18 (the existing CIFAR-native stem, which keeps an 8×8 feature map before the global pool at 64×64 resolution rather than collapsing it prematurely), 200 classes, 64×64, 100k training images, 4 activations, 3 seeds, 25 epochs. ReLU's decline and effective-rank rise replicate exactly (6 of 6 seed-level checks across both metrics, matching CIFAR in direction and magnitude). The smooth activations' smaller-magnitude rise also replicates in net displacement for 9 of 9 (active-gradient fraction) and 8 of 9 (effective rank) seeds, but the trajectories are no longer monotonic at this scale — an early overshoot followed by partial relaxation — so a whole-trajectory linear-trend statistic loses power on this already-small effect even though net direction is preserved; we report both statistics rather than the more favorable one alone.

Places365-Standard. ResNet-50 (same CIFAR-native stem family, 96×96 input, 12×12 feature map before the global pool), a fixed, seeded 150-train/20-val-per-class subsample (365 scene classes, real, uncurated photographs, comparable in class count to ImageNet-1k, which we did not test directly — it requires registration and has no anonymous scriptable download), 4 activations, 2 seeds (disclosed as 2, not the usual 3, for compute-budget reasons), 25 epochs. Every seed-level check agrees with the established direction by both the trend and net-displacement statistics: ReLU declines (2 of 2 seeds, -6.7 to -8.3 points) while GELU, SiLU, and Mish all rise (2 of 2 seeds each, $+0.3$ to $+1.1$ points) — the same large-decline, small-rise asymmetry seen throughout this paper. Effective rank rises for all four activations, consistent with ResNet-50 remaining a CNN. With only 2 seeds the sign test cannot exceed $p = 0.5$; this experiment adds a real-photograph, larger-class-count replication, not independent statistical power beyond what the other CNN experiments already establish.

Architecture family. Two CIFAR-native, activation-configurable, non-convolutional architectures with no BatchNorm anywhere: an 8-block MLP-Mixer [12] and a 6-block, 4-head Transformer-Encoder [13] (patch size 4, hidden dimension 128, $\sim$1M parameters each), both LayerNorm-only, instrumented with the same gate-recovery hooks (every activation is an explicit submodule, unlike `nn.TransformerEncoderLayer`'s functional activation, which the hook-based recovery cannot see). Same 4 activations, both datasets, 3 seeds, 25 epochs. GELU, SiLU, and Mish all rise robustly in both new architectures (6 of 6 independent runs each, sign test $p = 0.03125$, magnitude $+0.1$ to $+0.8$ points). The ReLU decline does not generalize: it rises in 5 of 6 MLP-Mixer runs, and is small and dataset-dependent in the Transformer-Encoder (mildly declining on CIFAR-100, mixed on CIFAR-10), with nothing resembling the large, robust CNN decline. Effective rank is the most architecture-dependent statistic of all: MLP-Mixer's rank *declines* for every activation including ReLU, and the Transformer-Encoder's rank splits by activation (ReLU rises, the smooth activations decline).

ConvNeXt-Tiny: excluded from all claims. A single-seed, 15-epoch training-dynamics run on CIFAR-10 (47.4% final test accuracy) showed effective rank rising (consistent with other smooth activations); the corresponding CIFAR-100 run, badly undertrained at 18.9% final accuracy, showed effective rank *falling*. We attribute the sign disagreement to undertraining, not a real architectural effect, and exclude ConvNeXt-Tiny from every quantitative claim in this paper.

A.5 Representational rank and pruning, in full

Representational rank: a null result. Effective rank rises in every condition tested in this paper — all 18 original trajectories, all 48 architecture-ablation runs, all 24 BatchNorm/GroupNorm runs, regardless of activation, architecture, dataset, or threshold (114+ independent runs, sign test $p < 10^{-14}$ pooled). A statistic that moves in the same direction in every condition tried has no power to discriminate anything this paper argues for, and we do not list it as a contribution. We flag, but do not resolve, a possible connection to BatchNorm-statistics dynamics (Section 5), and explicitly distinguish it from Neural Collapse [5] (different layer, different training-phase window) and self-supervised representation collapse [6, 7] (opposite sign, unrelated mechanism).

Pruning: gate density is not a general-purpose importance signal. We tested whether end-of-training gate density predicts channel importance for pruning, using the same ResNet-18/CIFAR-10 checkpoints (4 activations, 3 seeds, global channel pruning across 1920 channels in 8 layers, ratios

10–70%, no retraining, compared against magnitude pruning and a random baseline; full per-ratio results plotted in Figure 7 in the main text).

For ReLU, gate-density and magnitude pruning are statistically indistinguishable (paired-t $p = 0.55$) — informative equivalence. For GELU, SiLU, and Mish, gate-density pruning is significantly worse than magnitude pruning at every ratio ($p = 1.5 \times 10^{-4}$, 5.2×10^{-5}, 2.3×10^{-5} respectively): at 10% pruning, removing the lowest-gate-density channels costs 65–80 accuracy points, while removing the lowest-magnitude channels costs under 1.5 points. Gate density should not be presented or used as a general-purpose proxy for channel importance; for three of the four activations studied, doing so is actively harmful, not merely uninformative.

A.6 Exploratory privacy probe, in full

As one possible consequence of gate dynamics, we ask whether gate density affects gradient-inversion attack success, using a controllable victim model (a small CNN with one tunable-stiffness sigmoid gate, distinct from the real architectures studied above) against DLG [14], iDLG [8], and the stronger, scale-invariant Inverting Gradients attack [9] (7 stiffness values, 20 seeds).

Increasing gate stiffness significantly reduces iDLG's convergence probability (logistic regression slope -0.608, $p = 3.2 \times 10^{-5}$) but has no effect on Inverting Gradients, which converges regardless (100%; DLG itself shows no significant trend, slope -0.142, $p = 0.35$). We do not claim this connects to the real-architecture gate dynamics studied in the main paper — this experiment uses a different, minimal controlled model. The result we do claim: gate stiffness is not a general-purpose privacy mitigation, and any claim that gradient sparsity protects privacy should specify which adversary it was tested against.

A.7 Low-data fine-tuning payoff attempt, in full

Since z_{low} is computable before any training, we pre-registered (in writing, before computing anything) a test of whether a checkpoint's final gate density predicts its low-data fine-tuning accuracy: the predictor is each existing checkpoint's final population-level active-gradient fraction (24 checkpoints: 4 activations $\times$ {SGD, AdamW} $\times$ 3 seeds, all already trained 25 epochs on ResNet-18/CIFAR-10); the outcome is test accuracy after fine-tuning the full pretrained network on a fixed, seeded 5%-of-train-set subsample of CIFAR-10 (2,500 images, identical recipe for all 24 checkpoints: SGD, lr$= 0.01$, momentum 0.9, weight decay 5×10^{-4}, 5 epochs, batch size 64); the pre-registered test is Pearson and Spearman correlation, pooled and within each optimizer subset.

Table 5: Low-data fine-tuning payoff test, exactly as pre-registered.

Subset	n	Pearson r	p	Spearman ρ	p
All	24	0.241	0.258	-0.332	0.113
AdamW only	12	0.445	0.147	0.077	0.812
SGD only	12	0.246	0.441	-0.203	0.527

No statistically detectable correlation in any subset, and Pearson and Spearman do not even agree in sign on the pooled set — consistent with no real relationship rather than a weak true effect being underpowered. At the design tested (final population-level active-gradient fraction vs. post-fine-tune accuracy on a fixed 5% CIFAR-10 subsample), gate density does not predict low-data fine-tuning performance. We did not substitute a different outcome metric (e.g. accuracy drop instead of post-fine-tune accuracy) after seeing this null; the metric was fixed in the pre-registration and that is what is reported.

A.8 Reproducibility and hyperparameters

All training-dynamics experiments use SGD (momentum 0.9, weight decay 5×10^{-4}, Nesterov), cosine learning-rate schedule from an initial rate of 0.1 (CIFAR-native architectures, batch size 128) or scaled proportionally for 224-resolution architectures, 25 epochs unless otherwise noted, 3 independent seeds per condition unless a training failure is disclosed. Gate-density and rank statistics are computed on a single fixed 64-example held-out batch per run, re-measured every epoch. All

experiments ran on a single 80 GB data-center GPU. Code, raw result files, and figures will be released upon acceptance.

The optimizer-generalization experiment (Section 6.1) uses identical architecture/data/epoch settings with Adam and AdamW substituted for SGD, learning rate 10^{-3}, weight decay 5×10^{-4} (coupled for Adam, decoupled for AdamW), all other hyperparameters unchanged. The Tiny-ImageNet experiment uses the public Tiny-ImageNet-200 release (200 classes, 64×64, 100k train/10k val images), the same CIFAR-native ResNet-18 stem and SGD configuration as above, ImageNet-standard per-channel normalization statistics, and the same random-crop/flip augmentation pattern used throughout this paper (crop padding scaled to the 64×64 input). The sequence-model experiment uses an 8-block MLP-Mixer (patch size 4, hidden dimension 128, token-mixing dimension 64, channel-mixing dimension 512) and a 6-block, 4-head Transformer-Encoder (patch size 4, hidden dimension 128, MLP dimension 256, learnable positional embeddings, a prepended class token), both LayerNorm-only, $\sim$1M parameters each, otherwise the same SGD configuration and CIFAR-10/CIFAR-100 datasets as the architecture-fixed CNN ablation. The per-channel mechanism experiment uses the same ResNet-18/CIFAR-10/SGD configuration as the BatchNorm-vs-GroupNorm ablation, logging per-channel statistics at the same five epochs $\{0, 6, 12, 18, 24\}$ used by the shared-drift test. The Places365 experiment uses the public Places365-Standard easyformat release, a fixed seeded subsample of 150 train and 20 validation images per class (365 classes; the full 1.8M-image training set exceeds this project's single-GPU budget), 96×96 input (the same CIFAR-native ResNet-50 stem, 12×12 feature map before the global pool), `RandomResizedCrop` (scale 0.7–1.0) and horizontal flip for training augmentation, ImageNet-standard normalization statistics, otherwise the same SGD configuration as the main experiments; 2 seeds (not the usual 3, disclosed plainly) for compute-budget reasons.

A.9 The synthetic phase-transition theory: motivating context

The gate vocabulary used throughout this paper ($\Gamma(x) = |f'(x)|$, active-gradient-fraction, gate collapse) originates in an earlier, fully synthetic study of a single fixed-kernel inverse-reconstruction problem with a tunable sigmoid stiffness α. That study is self-contained, uses bootstrap confidence intervals, Cramér–Rao bounds, and a depth-compounding theorem whose own follow-up experiment found its key independence assumption holds only asymptotically ($\alpha \geq 40$), which the original study reports candidly. We preserve it here in full as the originating context for this paper's vocabulary and, specifically, for the exact mathematical correspondence between its sigmoid-stiffness parameter α and the Softplus-β family used in Appendix A.3. Its claims are scoped to the synthetic, fixed-kernel reconstruction setting described below and are not asserted to transfer to the real, trained-network setting of the main paper — the connection is established empirically above, not assumed.

A.9.1 Problem Setup

We consider the fixed single-layer convolutional neural network

$$f_\theta(x) = h(Ax), \tag{2}$$

where $A \in \mathbb{R}^{N \times N}$ is a convolution matrix corresponding to a fixed kernel θ, and $h(\cdot)$ is an element-wise sigmoid activation:

$$h(z) = \frac{1}{1 + e^{-\alpha(z-c)}}, \qquad h'(z) = \alpha\, h(z)\big(1 - h(z)\big), \tag{3}$$

where $\alpha > 0$ is the stiffness parameter and $c > 0$ is a threshold. The input is constrained to $0 \leq x \leq 1$, the target is binary $y \in \{0, 1\}^N$, and the reconstruction objective is

$$J(x) = \|y - f_\theta(x)\|_2^2, \qquad x^* = \operatorname*{arg\,min}_{0 \leq x \leq 1} J(x). \tag{4}$$

We define the **gradient gate** $\Gamma(x) = h'(Ax) \in \mathbb{R}^N$, with $\Gamma_i(x) = \alpha[f_\theta(x)]_i(1 - [f_\theta(x)]_i)$. Coordinate i is *active* if $|\Gamma_i(x)| > \varepsilon$. The **active gradient fraction** is $F_\alpha(x) = |\{i : |\Gamma_i(x)| > \varepsilon\}|/N$.

For the two-layer extension $f_2(x) = h(A_2 \cdot h(A_1 x))$, the gradient via chain rule is $\nabla J(x) = 2A_1^\top[\Gamma^{(1)} \odot A_2^\top(\Gamma^{(2)} \odot (h_2 - y))]$, with effective gate $\Gamma_{\text{eff}} = \Gamma^{(1)} \odot A_2^\top \Gamma^{(2)}$—the source of compounding collapse.

A.9.2 Landscape Analysis

The gradient is

$$\nabla J(x) = 2A^\top \big((h(Ax) - y) \odot h'(Ax)\big). \tag{5}$$

Lemma A.1 (Gradient Bound)**.** *The magnitude of the gradient $\nabla J(x)$ is bounded by the stiffness α:*

$$\|\nabla J(x)\|_2 \le \frac{\alpha}{2} \|A^\top (h(Ax) - y)\|_2. \tag{6}$$

Proof. Since $h'(z) = \alpha h(z)(1 - h(z)) \le \alpha/4$, applying the triangle inequality to the gradient expression yields the result. □

Theorem A.2 (Gradient Gate Bound)**.** *For any $\varepsilon \in (0, \alpha/4)$ and $x \in [0,1]^N$:*

$$\mathbb{E}_x[F_\alpha(x)] \;\le\; 1 - \frac{2\varepsilon}{\alpha}. \tag{7}$$

As $\alpha \to \infty$ with ε fixed, $F_\alpha(x) \to 0$ for almost every x with $(Ax)_i \neq c$ for all i.

Proof. The set $\{\Gamma_i > \varepsilon\}$ maps to a sigmoid-output interval (s_-, s_+) with $s_\pm = \frac{1}{2} \pm \frac{1}{2}\sqrt{1 - 4\varepsilon/\alpha}$. Its width satisfies $s_+ - s_- = \sqrt{1 - 4\varepsilon/\alpha} \le 1 - 2\varepsilon/\alpha$ (using $\sqrt{1-t} \le 1 - t/2$). When $(Ax)_i$ is approximately uniform, $P(\Gamma_i > \varepsilon) \le 1 - 2\varepsilon/\alpha$. As $\alpha \to \infty$, the corresponding z-interval shrinks to $\{c\}$, giving $F_\alpha(x) \to 0$ for any x not satisfying $(Ax)_i = c$ for all i. □

Theorem A.3 (Compounding Collapse — Scoped Validity)**.** *Under an approximate gate-independence assumption at convergence (empirically verified: mean $|\mathrm{corr}(\Gamma^{(1)}, A_2^\top \Gamma^{(2)})| = 0.201$, max < 0.50 at all tested configurations), for the two-layer model with equal stiffness α and kernel A_2:*

$$\frac{F_\alpha^{(2)}}{F_\alpha^{(1)}} \;\le\; \frac{1}{1 + c_2 \alpha}, \qquad c_2 = \frac{\varepsilon}{\|A_2\|_\infty}. \tag{8}$$

For $\alpha \ge 16$, bootstrap $R^2 \ge 0.919$ (CI lower bound: 0.764). The rate c is not a universal constant; in the valid regime it follows $c(\alpha) \approx 0.211 \cdot \alpha^{-0.674}$ ($R^2 = 0.973$), ranging from $c \approx 0.031$ at $\alpha = 16$ to $c \approx 0.012$ at $\alpha = 60$. Below $\alpha = 16$, gradients are insufficiently sparse for the multiplicative independence assumption to hold. ***This asymptotic scope ($\alpha \ge 16$, and an independence assumption verified only weakly even in that regime—mean $|\mathrm{corr}| = 0.201$, not strictly zero) is the reason this theorem is not presented as a general law in the main paper.***

Proof sketch. Coordinate i is active in the two-layer model only if active in both $\Gamma^{(1)}$ and $A_2^\top \Gamma^{(2)}$. Under approximate independence, $P(\text{active in 2L}) = P(\Gamma_i^{(1)} > \varepsilon) \cdot P((A_2^\top \Gamma^{(2)})_i > \varepsilon)$. Applying Theorem A.2 to the second layer with effective threshold $\varepsilon / \|A_2\|_\infty$ gives the bound. □

Lemma A.4 (Adam Advantage from Momentum)**.** *In the high-α regime, Adam's first moment $m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$ preserves gradient signal from past active sets via exponential decay at rate β_1^k. Oracle coordinate rescaling ($\beta_1 = 0$) does not recover this advantage.*

Verification (20 seeds): *Mean IoU — Adam:* 0.950*, Oracle:* 0.794*, PGD:* 0.811*. Wilcoxon $p =$* 0.0078*, Cohen's $d > 9$ at all $\alpha \ge 5$. The Adam–Oracle gap is non-monotone, peaking at* 0.332 *at $\alpha = 13$ with a local minimum of* 0.246 *at $\alpha = 25$.*

Identifiability and Null Space Geometry. Reconstruction is unique if and only if the effective Jacobian $J(x^*) = \mathrm{diag}(\Gamma(x^*)) \cdot A$ has trivial null space, $\sigma_{\min}(J(x^*)) > 0$.

$\sigma_{\min}(J(x^*))$ decreases monotonically with α, crossing zero near α^*. At $\alpha \ge 20$, the null space dimension reaches $46/256$ (18% of input space). The stable rank collapses monotonically from 46.8 at $\alpha = 1$ to 6.9 at $\alpha = 60$ (6.7×), computed via exact SVD of the effective Jacobian, confirming the implicit dimensional reduction induced by the gradient gate.

Uncertainty Quantification for α^*. To avoid parameter collinearity in the Fisher information matrix ($\mathrm{cond}(\mathcal{I}_4) = 4.02 \times 10^6$ for the 4-parameter fit, flagged numerically unreliable due to $\mathrm{IoU}_{\max}$–δ collinearity, $r = 0.781$), we fix $\mathrm{IoU}_{\max} = 1.0$ (justified by the data: Adam achieves $\mathrm{IoU} = 1.000$ at all $\alpha \leq 10$) and fit a 3-parameter model:

$$\overline{\mathrm{IoU}}(\alpha) = \mathrm{IoU}_{\min} + \frac{1 - \mathrm{IoU}_{\min}}{1 + \exp((\alpha - \alpha^*)/\delta)}. \tag{9}$$

This reduces $\mathrm{cond}(\mathcal{I}_3)$ to 3.0×10^4 (135× improvement). The primary uncertainty estimate is the distribution-free bootstrap 95% CI: $\alpha^* \in [27.92,\ 29.78]$ (1000 resamples, 20 seeds), corresponding to a CI width of 1.86. The 3-parameter CRLB gives $\sigma_{\mathrm{CRLB}}(\alpha^*) = 0.344$.

A.9.3 Algorithmic Analysis

Projected Gradient Descent (PGD). Relies directly on gradient magnitude; struggles severely when gradients vanish. Applies uniform updates across all coordinates, failing to exploit the few regions where gradients remain informative.

Momentum and Nesterov. Improve upon PGD by accumulating gradients over time, partially compensating for sparse updates. Nesterov's look-ahead correction yields faster convergence in smooth regimes, but the look-ahead step lands in an equally gradient-sparse region at high α.

Adam. Rescales gradients coordinate-wise based on historical gradient magnitudes. Lemma A.4 shows that Adam's advantage arises from momentum accumulation, not coordinate-wise rescaling as commonly assumed.

L-BFGS-B. Approximates curvature information and can accelerate convergence in smooth regimes. Performance degrades when gradients are sparse or unreliable.

Table 6: Qualitative and quantitative comparison of optimisers. IoU and active gradient fraction F_α at $\alpha = 10$ and $\alpha = 20$ (`sobel_x`, checkerboard, 5 seeds).

Optimiser	Robustness	Stability	$\alpha = 10$		$\alpha = 20$	
			IoU	F_α	IoU	F_α
Adam	High	High	**1.000**	**1.000**	**.947**	**.470**
L-BFGS-B	Medium	Variable	.999	.998	.883	.359
Momentum	Medium	Medium	1.000	1.000	.784	.328
Nesterov	Medium	Medium	1.000	1.000	.754	.278
PGD	Low	Low	.858	.896	.787	.270

A.9.4 Experiments

All experiments use 64×64 inputs with five fixed convolutional kernels: **identity-like** (trivial, separable), **avg_blur** (3×3 uniform, low-pass), **random_norm** (random 3×3, unit Frobenius), **sobel_x** (horizontal edges, spectral norm ≈ 4.0), and **laplacian** (second-order, spectral norm ≈ 8.0). Threshold $c = 0.5$; stiffness $\alpha \in \{1, 2, 5, 10, 20, 40\}$ (core experiments) and $\alpha \in \{1, 1.5, 2, 3, 5, 7, 10, 13, 16, 20, 25, 30, 40, 60\}$ (phase transition characterisation, 20 seeds). Adam uses $\mathrm{lr} = 0.03$, $\beta_1 = 0.9$, $\beta_2 = 0.999$; PGD and Momentum use $\mathrm{lr} = 0.1$ with $\beta = 0.9$; Nesterov uses the same; L-BFGS-B uses its default line search. All methods run for 200 iterations.

Metrics. Final loss $J(x)$; Intersection-over-Union (IoU, binarised at 0.5); active gradient fraction F_α ($\varepsilon = 0.01$); relative loss reduction $(J(x_0) - J(x_T))/J(x_0)$; success rate $P[\mathrm{IoU} > 0.7]$.

Phase Transition Characterisation. Mean reconstruction IoU follows a sigmoidal decay in α across a transition band $\alpha \in [13, 30]$. Three threshold values characterise the band:

Threshold	α	IoU at crossing
α_{10}: IoU first drops below 1.000	13	0.996
α_{50}: IoU drops below 0.900	30	0.875
α_{90}: IoU drops below 0.800	≥ 60	0.771

The 3-parameter sigmoid fit ($\mathrm{IoU}_{\max}$ fixed at 1.0) gives:

$$\mathrm{IoU}_{\min} = 0.782, \quad \alpha^* = 28.97, \quad \delta = 6.29.$$

The bootstrap median is $\alpha^* = 28.83$ (95% CI [27.92, 29.78]). A sparse grid concentrated within the subcritical plateau ($\mathrm{IoU} \equiv 1.0$) cannot resolve α^* reliably, since a bounded least-squares fit on plateau-only data is underdetermined; a 14-point grid extending to $\alpha = 60$ (20 independent seeds, 1000-resample bootstrap) is required to observe the full sigmoid S-curve and obtain the estimate above.

Effect of Sigmoid Stiffness. Performance degrades consistently as α increases through the transition band (Fig. 19). Adam sustains perfect reconstruction ($\mathrm{IoU} = 1.000$) up to $\alpha = 10$, then degrades smoothly to 0.947 at $\alpha = 20$ and 0.830 at $\alpha = 40$. The non-adaptive optimisers begin degrading earlier, with PGD already at $\mathrm{IoU} = 0.858$ at $\alpha = 10$.

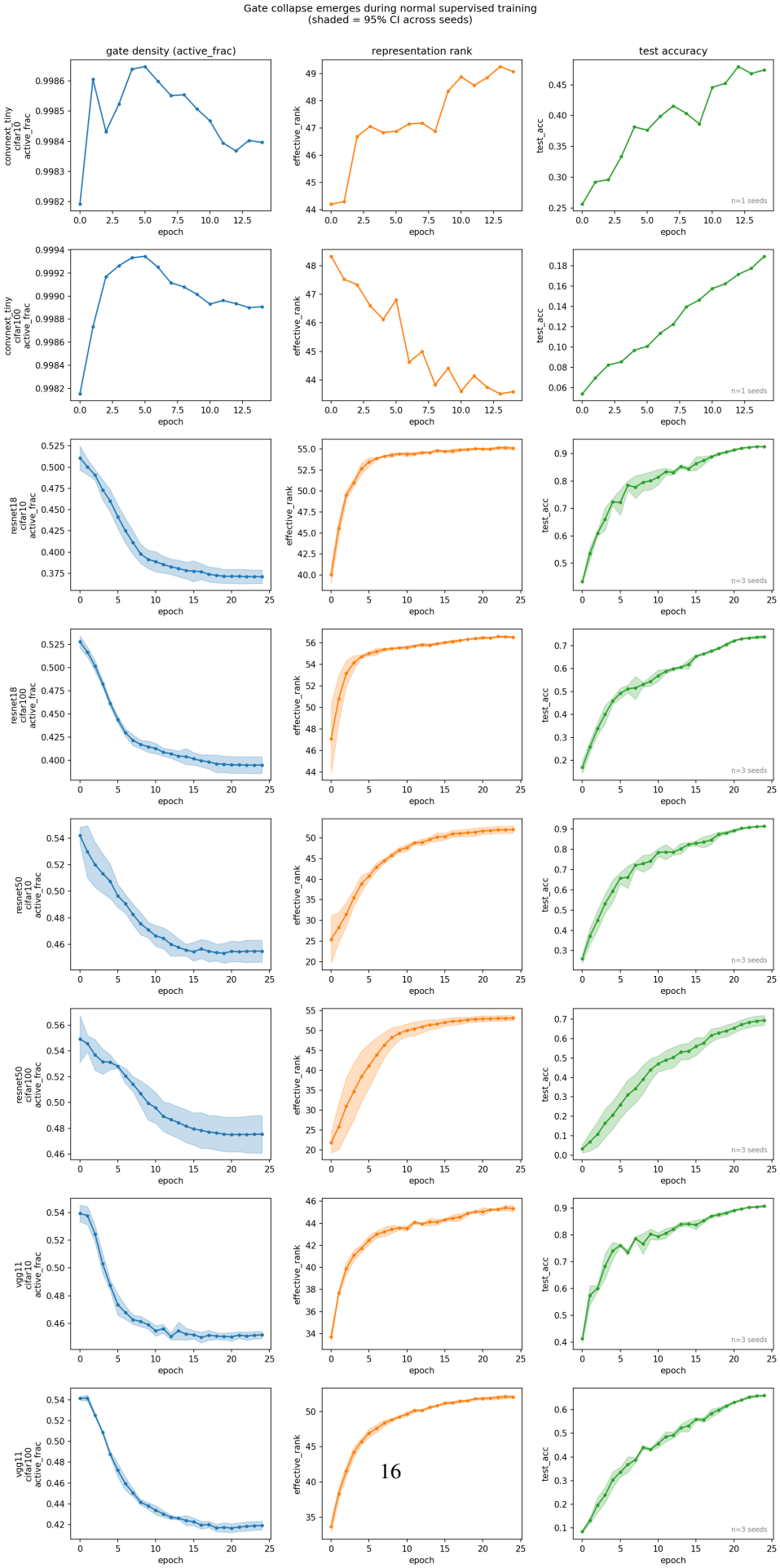
Table 7: Performance across sigmoid stiffness values (Adam, `sobel_x`, 20 seeds).

α	Loss (mean±std)	IoU (mean±std)
1.0	504.87 ± 9.91	1.000 ± 0.000
5.0	344.19 ± 11.62	1.000 ± 0.000
10.0	488.05 ± 9.10	1.000 ± 0.000
20.0	646.31 ± 11.35	0.947 ± 0.007
40.0	799.49 ± 30.10	0.830 ± 0.012
60.0	892.13 ± 34.52	0.771 ± 0.018

Impact of Cutoff Threshold c.

Spatial Gradient Gate and Dimensional Collapse. The gradient gate $\Gamma(x) = h'(Ax)$ collapses spatially as α increases: a dense, near-uniform field at low α narrows to isolated active coordinates at edge boundaries at high α, directly instantiating Theorem A.2. The quantitative version of this collapse, from `grad_sparsity_all_optimizers.csv`: at $\alpha = 1$, the gate is 100% active. The collapse becomes pronounced in the transition band: by $\alpha = 20$, Adam retains 47.0% active coordinates versus PGD's 27.0%; by $\alpha = 40$, Adam retains 19.0% versus PGD's 5.7%. Stable rank collapses monotonically from 46.8 at $\alpha = 1$ to 6.9 at $\alpha = 60$ (6.7×), computed via exact SVD of the effective Jacobian.

Gradient Sparsity and Its Impact.


Gate collapse emerges during normal supervised training
(shaded = 95% CI across seeds)
gate density (active_frac)
representation rank
test accuracy
convnext_tiny cifar10 active_frac
convnext_tiny cifar100 active_frac
resnet18 cifar10 active_frac
resnet18 cifar100 active_frac
resnet50 cifar10 active_frac
resnet50 cifar100 active_frac
vgg11 cifar10 active_frac
vgg11 cifar100 active_frac
effective_rank
test_acc
epoch
n=1 seeds
n=3 seeds

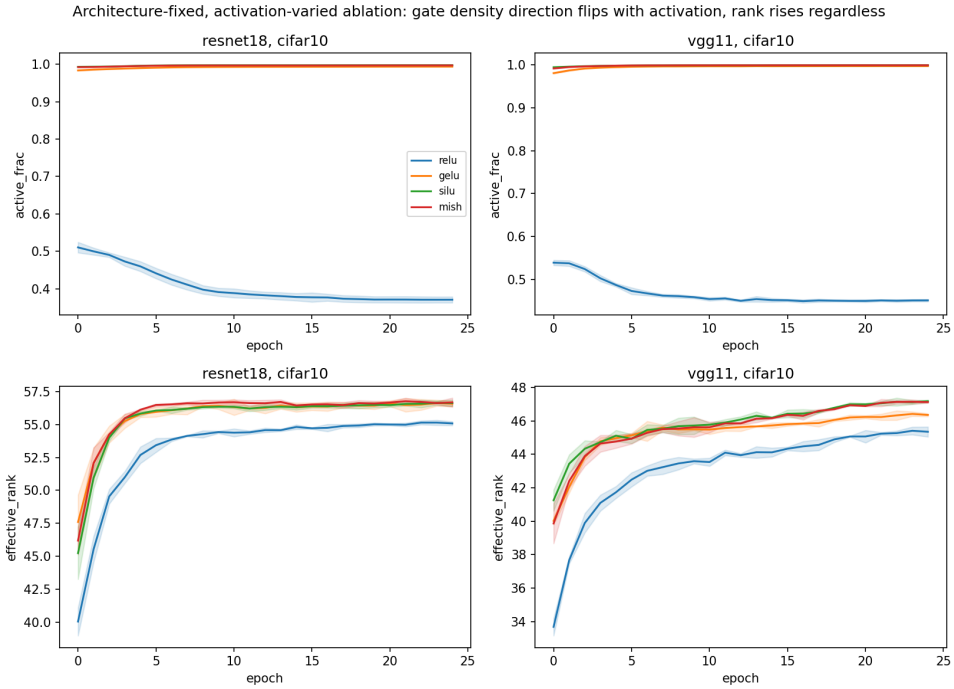


Figure 2: Architecture-fixed, activation-varied ablation. Gate density falls for ReLU and rises for GELU/SiLU/Mish on the *same* backbone; effective rank rises for every activation regardless of direction.

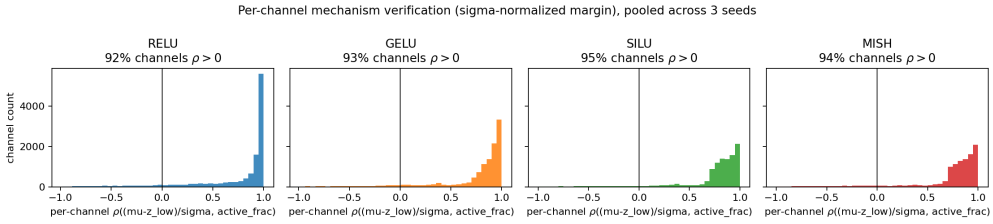


Figure 3: Per-channel verification. Histograms of the per-channel correlation between the σ-normalized margin and active-gradient fraction across 5 logged epochs, pooled over 3 seeds. All four activations show strong, uniform, positive correlation.

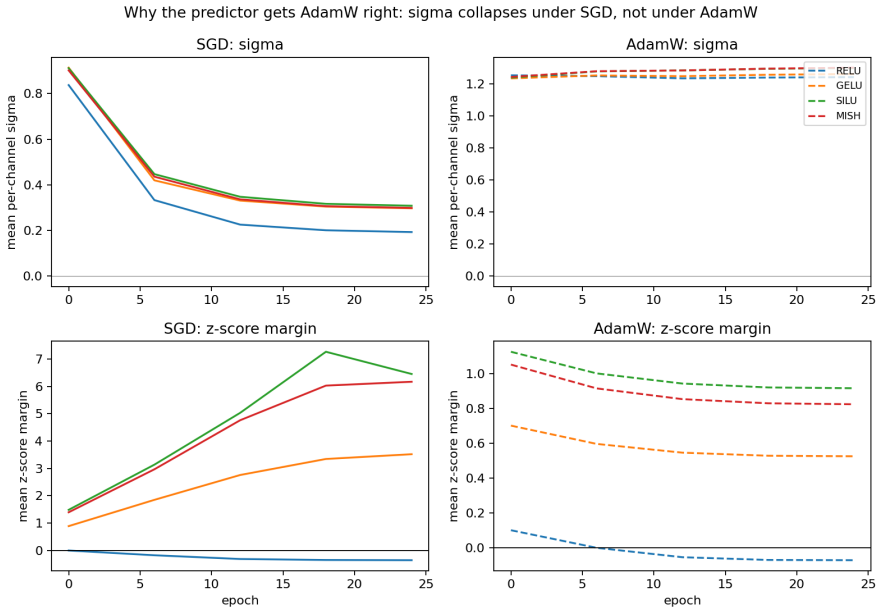


Figure 4: Why the predictor gets AdamW right. σ collapses for every activation under SGD but is flat (ReLU) or grows (GELU/SiLU/Mish) under AdamW; the resulting margin rises for smooth activations under SGD but declines for every activation under AdamW.

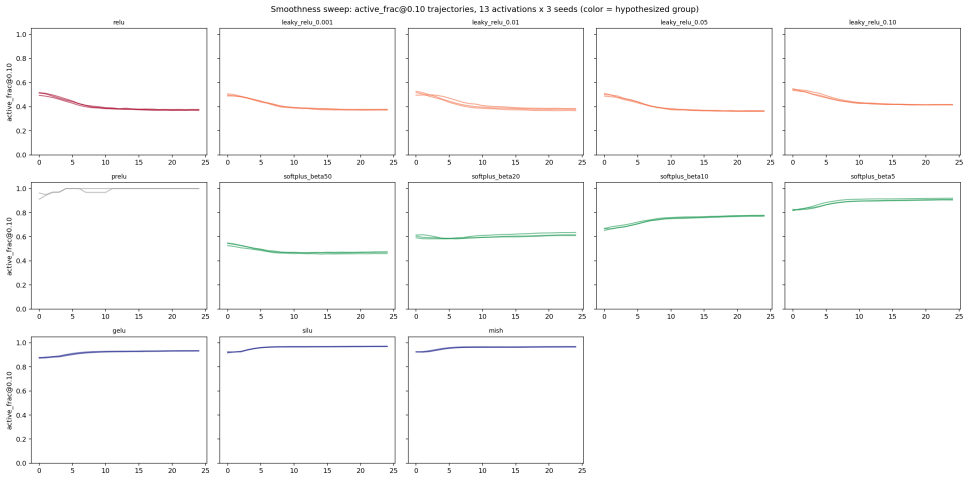


Figure 5: Smoothness sweep. Gate-density trajectories for 13 activation conditions. The sign transition occurs within the Softplus(β) family, predicted exactly by the z_{low} values computed independently of any training run.

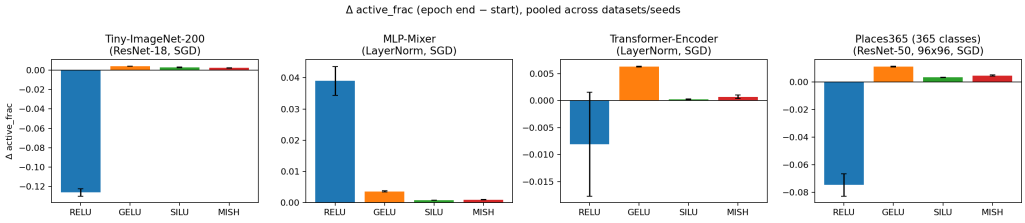


Figure 6: Generalization beyond CIFAR-scale CNNs. Mean Δactive-gradient-fraction, pooled across datasets/seeds. The smooth-activation rise replicates on Tiny-ImageNet-200, two non-convolutional architectures, and real photographs at 365-class scale; the ReLU decline replicates on the CNN-based experiments only.

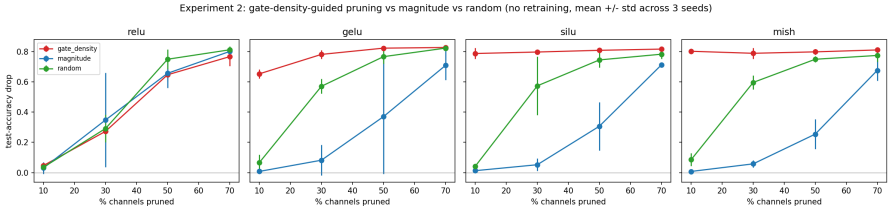


Figure 7: Gate-density-guided pruning vs. magnitude vs. random (mean ± std, 3 seeds, no retraining). Indistinguishable from magnitude pruning for ReLU; dramatically worse for every smooth activation.

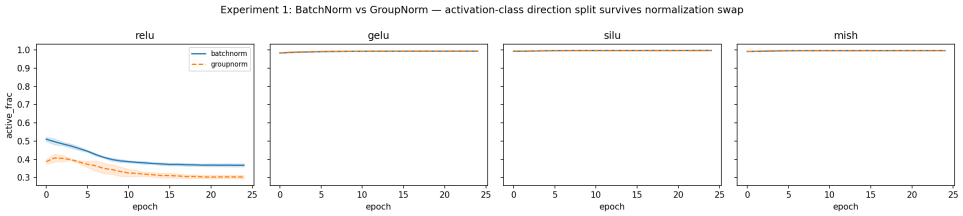


Figure 8: BatchNorm vs. GroupNorm. The activation-class direction split (ReLU declining, smooth activations rising) is identical under both normalization schemes, on the identical ResNet-18 skeleton.

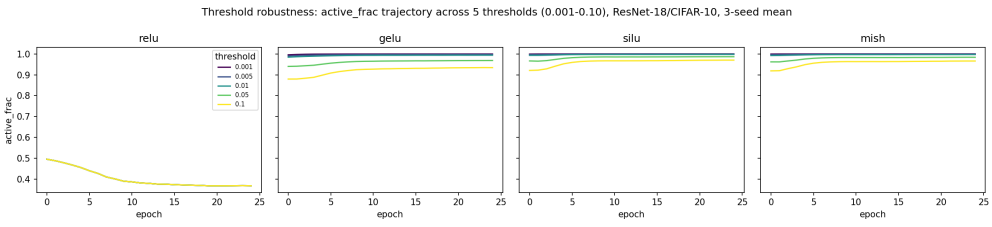


Figure 9: Threshold robustness. Active-gradient-fraction trajectories across five thresholds spanning two orders of magnitude. ReLU declines and every smooth activation rises at every threshold; magnitude grows with stricter thresholds.

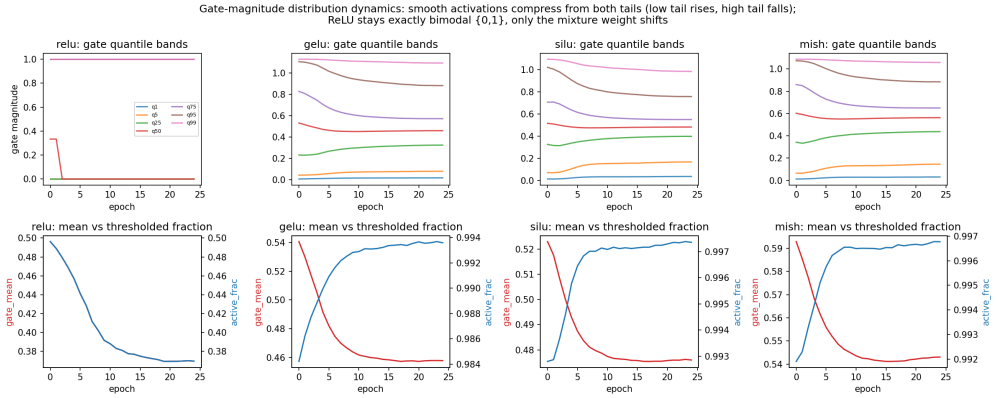


Figure 10: Gate-magnitude distribution dynamics. Quantile bands (1st–99th percentile) over training, by activation. Smooth activations compress from both tails toward a moderate center; ReLU remains exactly bimodal $\{0, 1\}$, with training only shifting the mixture weight between modes.

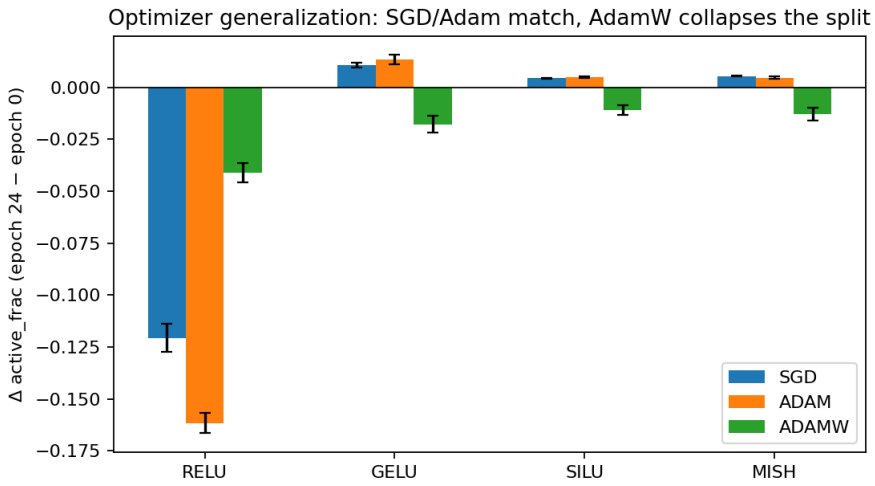


Figure 11: Optimizer generalization. Mean Δactive-gradient-fraction (epoch 24 minus epoch 0), by activation and optimizer, $\pm$ standard error across 12 independent runs. SGD and Adam (both coupled weight decay) give the same direction split; AdamW (decoupled weight decay) gives a uniform decline instead — the pattern Section 5.3 shows the mechanism predicts exactly, with no new free parameters.

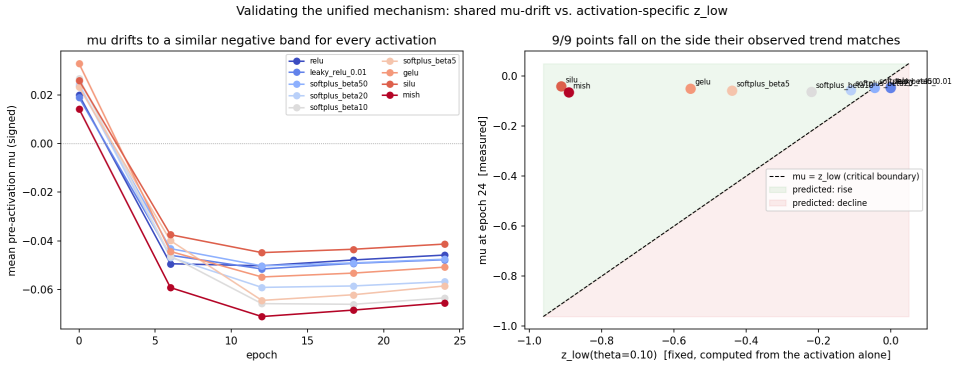


Figure 12: Mechanism validation. (Left) Pre-activation mean drifts to a shared negative band across 9 different activation functions. (Right) Plotting each activation's measured μ against its independently computed z_{low}: every point falls on the side its observed trend matches. Softplus($\beta = 50$) sits within 0.0036 of the critical boundary — the one condition with a conspicuously weak, high-variance empirical correlation ($\rho = -0.59$ vs. $|\rho| > 0.96$ elsewhere), which the mechanism correctly anticipates.

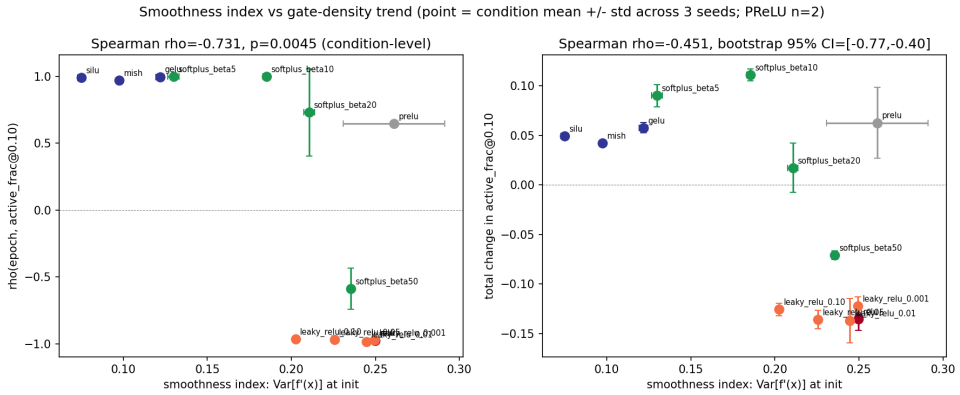


Figure 13: Smoothness index vs. gate-density trend. Spearman $\rho = -0.731$, $p = 0.0045$ (condition-level, $n = 12$); $\rho = -0.710$, $p = 6.1 \times 10^{-7}$ (seed-level, $n = 37$). One excluded point: PReLU seed 0 failed to train (10% accuracy throughout, NaN gate statistics from epoch 0) and is disclosed, not silently dropped.

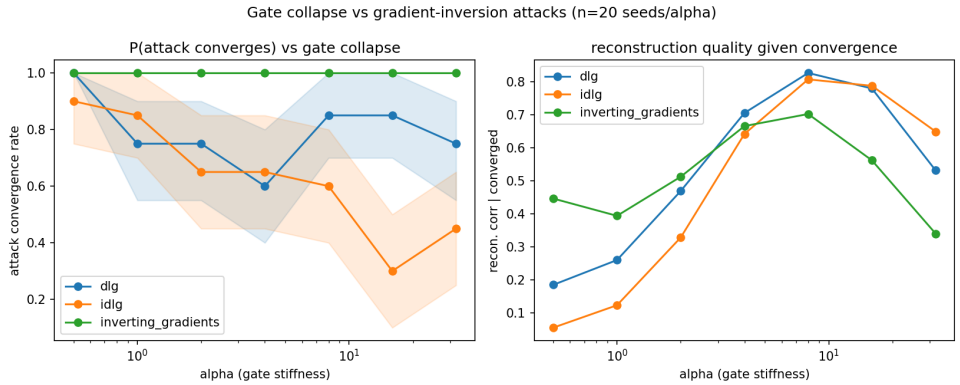


Figure 14: Gradient-inversion attack convergence vs. gate stiffness. Higher stiffness significantly reduces iDLG's convergence probability (logistic slope -0.608, $p = 3.2 \times 10^{-5}$) but has no effect on Inverting Gradients (always converges, 100%).

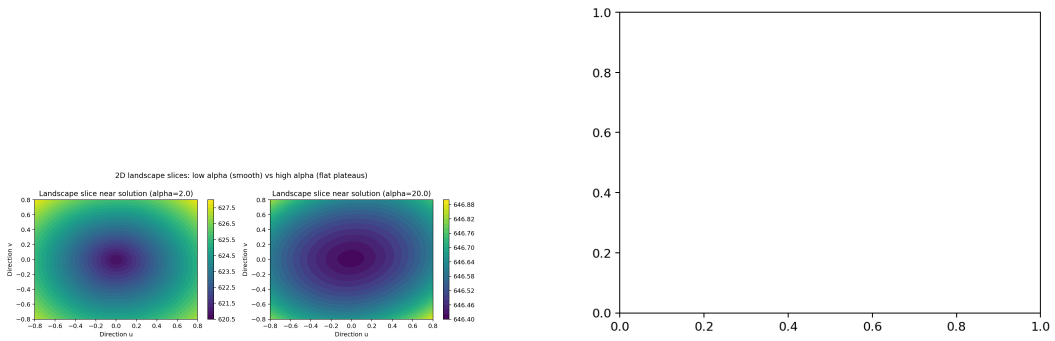


Figure 15: Two-dimensional landscape slices near the solution. Low α (left) yields a smooth, well-conditioned surface. High α (right) introduces large flat plateaus, sharp transitions, and loss of gradient signal. The landscape changes qualitatively, not merely quantitatively, as α increases.

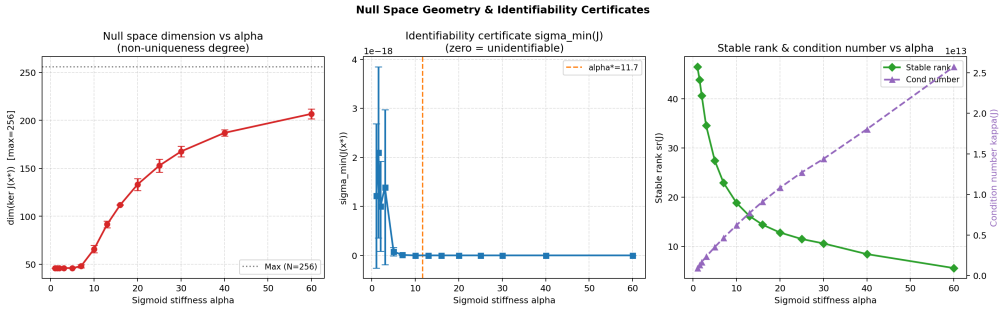


Figure 16: Null space geometry. (Left) Null space dimension grows monotonically with α. (Centre) Minimum singular value $\sigma_{\min}(J(x^*))$ decreases monotonically, providing a constructive identifiability certificate. (Right) Stable rank and condition number.

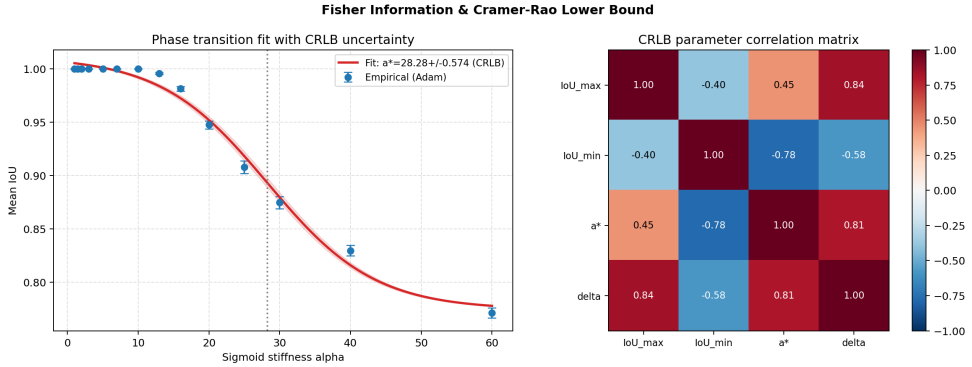


Figure 17: Uncertainty quantification for α^*. (Left) 3-parameter sigmoid fit with bootstrap CI $[27.92, 29.78]$. The 4-parameter CRLB is retired ($\text{cond}(\mathcal{I}) = 4.02 \times 10^6$, unreliable); the 3-parameter replacement ($\text{cond} = 3.0 \times 10^4$) gives $\sigma_{\text{CRLB}} = 0.344$. (Right) 3-parameter CRLB correlation matrix, showing residual α^*–δ correlation ($r = 0.863$), structural for this parameterisation but manageable.

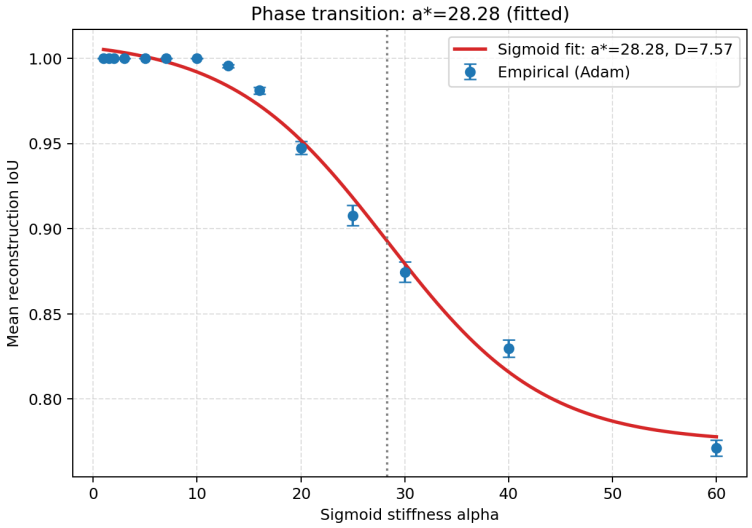


Figure 18: Phase transition fit. IoU follows a sigmoidal curve in α with bootstrap midpoint $\alpha^* = 28.83$ (95% CI $[27.92, 29.78]$, dashed line and shaded band). Transition onset ($\alpha_{10} = 13$) and degradation midpoint ($\alpha_{50} = 30$) are marked. Networks with $\alpha \leq 10$ achieve perfect reconstruction (IoU $= 1.000$); quality degrades continuously through the band $\alpha \in [13, 30]$.

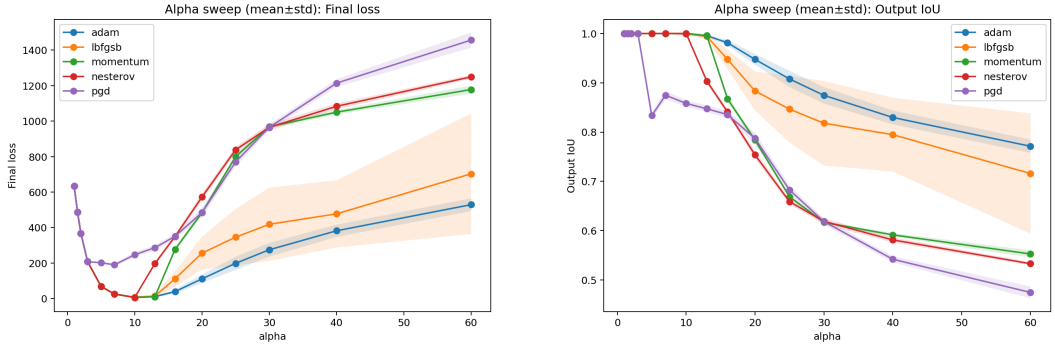


Figure 19: Effect of sigmoid stiffness on final loss (left) and reconstruction IoU (right) across all optimisers. Adam achieves IoU $= 1.000$ at $\alpha \leq 10$ and degrades gracefully to 0.830 at $\alpha = 40$.

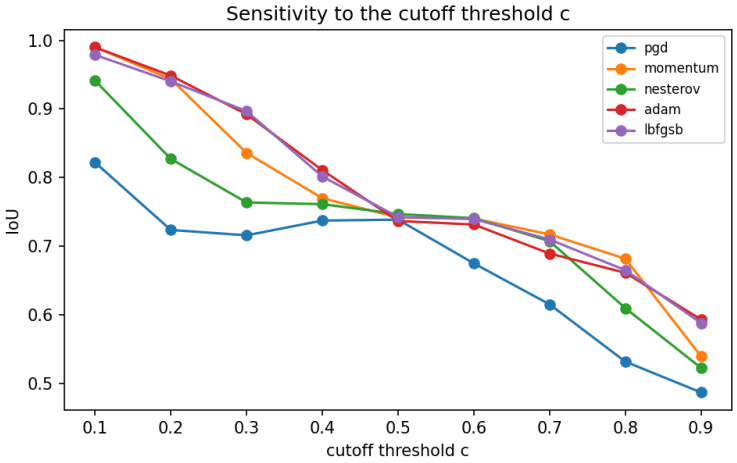


Figure 20: Sensitivity analysis of the cutoff threshold c. Performance is maximised at $c = 0.5$ and degrades symmetrically toward the boundaries. Adam shows greater resilience to off-centre thresholds than PGD or L-BFGS-B.

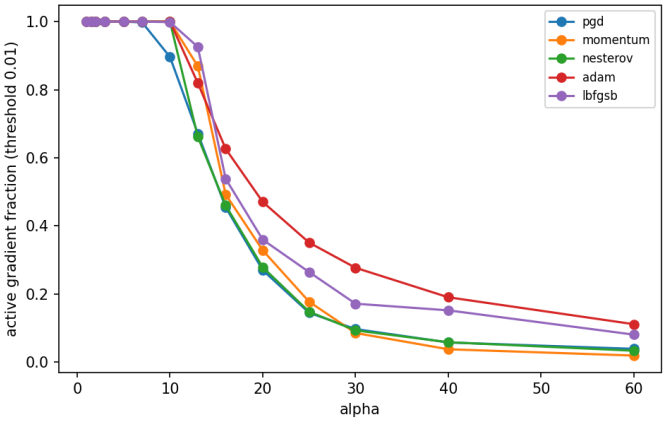


Figure 21: Fraction of active gradient components ($|\Gamma_i| > 0.01$) at convergence vs. α. Adam maintains 47.0% active at $\alpha = 20$ compared to PGD's 27.0%, and 19.0% vs. 5.7% at $\alpha = 40$—a widening gap deep in the transition band that explains Adam's robustness.

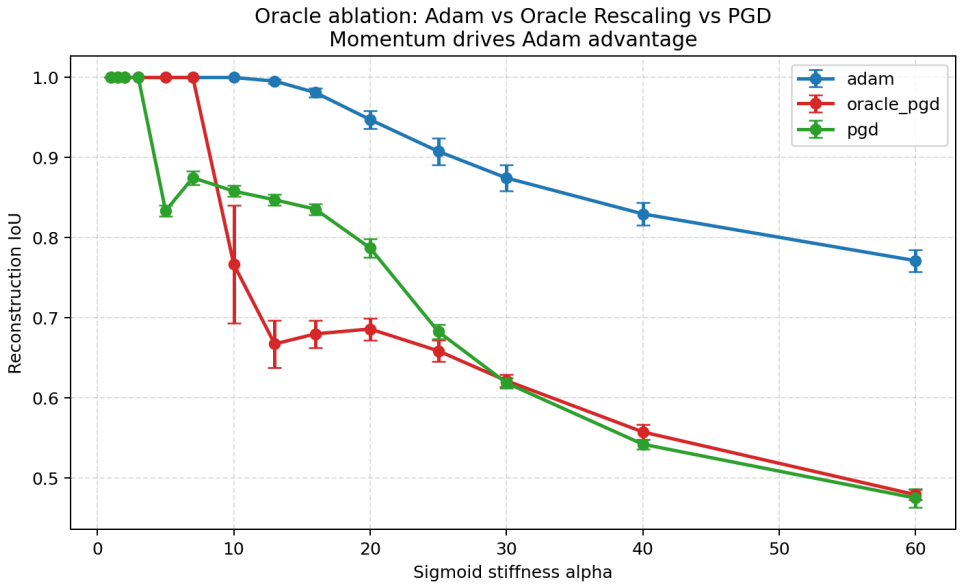


Figure 22: Oracle ablation: Adam vs. Oracle Rescaling vs. PGD. (Left) IoU curves showing Oracle $<$ PGD at $\alpha = 10$–25 (shaded zone) and Adam's sustained superiority ($p = 0.0078$, all $\alpha \geq 5$). (Right) Adam–Oracle gap peaks at 0.332 ($\alpha = 13$) and anticorrelates with $|d\Gamma/d\alpha|$ ($r = -0.799$, $p = 0.0006$), revealing the transition-regime miscalibration mechanism.

Oracle Ablation: Momentum Drives Adam's Advantage. We construct Oracle Rescaling: PGD with per-coordinate step sizes $v_t^{-1/2}$ from Adam's second moment, but $\beta_1 = 0$ (no momentum). Mean IoU: Adam 0.950, Oracle 0.794, PGD 0.811 ($p = 0.0078$, Wilcoxon, Cohen's $d > 9$, at all $\alpha \geq 5$).

Reversal of Oracle vs. PGD. Oracle performs *below* plain PGD at $\alpha = 10$–25 (gap -0.093 to -0.183 IoU points, $p = 0.0078$–0.023). This reversal is not explained by differential gate collapse: at these α values, Oracle retains 12–63% more active gradient coordinates than PGD (Oracle/PGD active-fraction ratio: 1.12 at $\alpha = 10$, rising to 1.64 at $\alpha = 25$), yet achieves substantially lower IoU. Instead, the reversal tracks the rate of change of gate sparsity: the Oracle–PGD gap anticorrelates strongly with $|d\Gamma/d\alpha|_{\text{PGD}}$ (Pearson $r = -0.799$, $p = 0.0006$), with the deepest reversal at $\alpha = 13$–16 where gate density falls most steeply. We attribute this to miscalibration of exact second-moment rescaling in a rapidly-changing gradient field: oracle step sizes assign disproportionately large updates to barely-active coordinates (low $v_t \to$ high $1/\sqrt{v_t}$) near the sigmoid inflection point, pulling the reconstruction trajectory away from the true attractor. PGD's fixed step implicitly weights updates by gradient magnitude, which is already calibrated by the gate structure itself. The reversal resolves above $\alpha = 30$, where gate density has settled to a new stable regime ($|d\Gamma/d\alpha| \approx 0$).

Practical implication: The advantage is in the gradient history, not the step-size schedule. Tuning per-coordinate learning rates does not replicate Adam's advantage in high-α regimes.

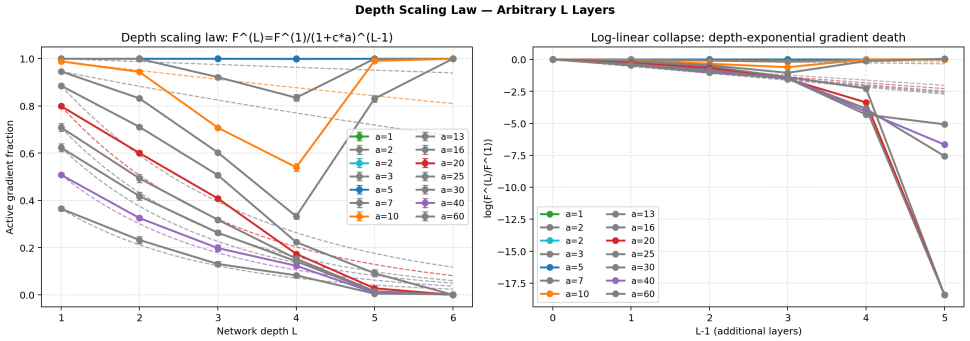


Figure 23: Depth scaling law. (Left) Bootstrap R^2 vs. α: model is valid ($R^2 \geq 0.919$) only for $\alpha \geq 16$; R^2 is negative for $\alpha \leq 13$ (model inapplicable, marked $\times$). (Centre) Collapse rate $c(\alpha) \approx 0.211 \cdot \alpha^{-0.674}$ decreases monotonically in the valid regime. (Right) Log-linear collapse confirmed for $\alpha \geq 16$.

Table 8: Compounding collapse: empirical ratio (2L/1L) vs. bound $1/(1 + c(\alpha) \cdot \alpha)$. Valid regime: $\alpha \geq 16$ ($R^2 \geq 0.919$).

α	1L F	2L F	Ratio	c_{fit}	R^2 (boot)	Applicable
1–13	—	—	—	—	< 0	**No**
16	—	—	—	0.031	0.919	Yes
20	0.147	0.078	0.531	0.029	0.938	Yes
25	—	—	—	0.026	0.959	Yes
40	0.059	0.018	0.302	0.017	0.972	Yes
60	—	—	—	0.012	0.977	Yes

Compounding Collapse: Theorem A.3 Verification. The gate independence assumption is empirically verified: mean $|\text{corr}(\Gamma^{(1)}, A_2^\top \Gamma^{(2)})| = 0.201$ with no seed exceeding 0.50. This introduces a small positive bias in the predicted compounding rate.

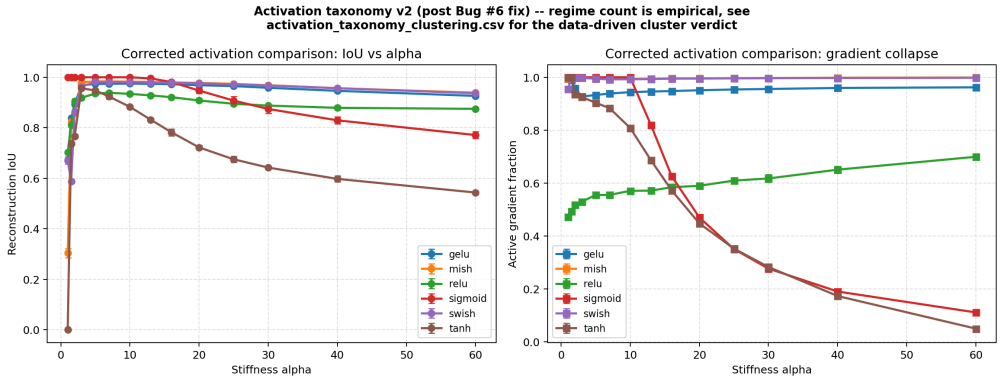


Figure 24: Activation comparison across six functions. (Left) IoU vs. α. (Centre) Active gradient fraction—three regimes are clearly separated. (Right) Silhouette score vs. k: $k = 3$ is optimal (silhouette$= 0.722$), confirming a 3-group taxonomy. Between-group variance is $573\times$ within-group variance at $\alpha = 40$.

Activation Comparison. Six activations reveal a three-group taxonomy (Fig. 24, $k = 3$, silhouette$= 0.722$):

- *Saturating* (sigmoid, tanh): active fraction collapses $\approx 60\%$ from $\alpha = 20$ to $\alpha = 40$ (from 0.459 to 0.182); IoU degrades proportionally. Note: tanh degrades more severely in IoU

(0.598 at $\alpha = 40$) compared to sigmoid (0.830), attributable to tanh's bipolar output range $[-1, 1]$ introducing additional optimisation difficulty with binary targets.

- *Smooth self-gated* (GELU, Mish, SiLU/Swish): active fraction remains $> 95\%$ at all tested α—no collapse regardless of stiffness, within *this* synthetic fixed-kernel reconstruction setting. This is the same qualitative direction the main paper finds during ordinary training of real networks (Section 4), though the two settings are connected empirically (Section A.3), not assumed identical.
- *Piecewise linear* (ReLU): active fraction is approximately α-insensitive; IoU degradation is moderate and stable.

Swish clusters with GELU and Mish in the smooth group in all $k \geq 2$ clusterings (silhouette-verified).

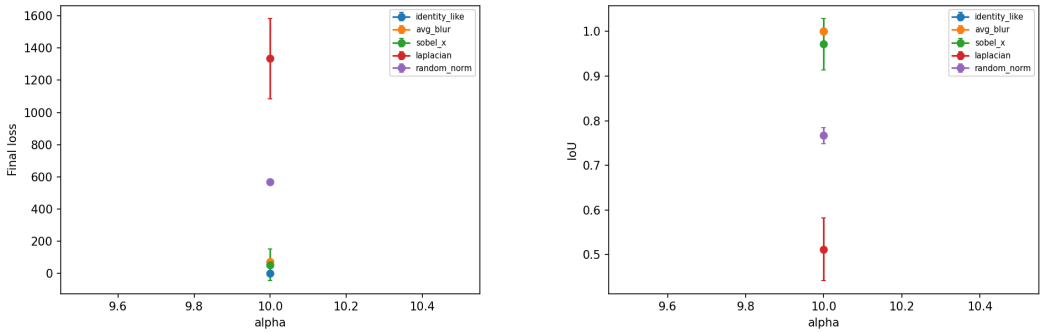


Figure 25: Final loss (left) and IoU (right) across convolution kernels. Identity-like and avg_blur are always trivial (IoU $= 1.0$). Laplacian produces the highest loss, consistent with its spectral norm ≈ 8.0 inducing maximum landscape anisotropy.

Table 9: Kernel difficulty ranking: IoU (Adam, 5 seeds) and spectral norm.

Kernel	$\alpha = 1$	$\alpha = 5$	$\alpha = 10$	$\alpha = 40$	$\|A\|_2$	Difficulty
identity-like	1.000	1.000	1.000	1.000	≈ 1.0	Trivial
avg_blur	1.000	1.000	1.000	1.000	≈ 1.0	Trivial
random_norm	1.000	1.000	1.000	0.777	≈ 1.0	Moderate
sobel_x	1.000	1.000	1.000	0.830	≈ 4.0	Hard
laplacian	1.000	0.964	0.812	0.589	≈ 8.0	Very Hard

Kernel Sensitivity.

Phase Diagram.

Target Sensitivity.

Generalisation and Universality. *Scale invariance.* IoU at $\alpha = 10$: 32×32 (0.716), 64×64 (0.767), 128×128 (0.786). Differences are within one standard deviation.

Finite-size scaling and universality. Two distinct transition thresholds are reported and should not be confused. The finite-size scaling (FSS) estimate $\alpha^*_{\text{FSS}} = 17.2 \pm 0.6$ (64×64) is the stiffness at which the order parameter $(\text{IoU} - \text{IoU}_{\min})$ first departs from its plateau maximum (transition *onset*, 81% of maximum order parameter retained at this point). The bootstrap estimate $\alpha^* = 28.83$ (95% CI $[27.92, 29.78]$) is the sigmoid *midpoint*: the stiffness where $\text{IoU} = (\text{IoU}_{\max} + \text{IoU}_{\min})/2 \approx 0.893$. The two quantities differ because the transition has finite width ($\delta \approx 6.3$); for a sharp transition ($\delta \to 0$) they would coincide. The FSS estimate characterises when reconstruction quality *begins* to degrade; the sigmoid midpoint characterises the *centre* of the degradation regime.

Noise robustness. Reconstruction quality varies $< 6\%$ IoU across SNR 15–60 dB for all α (maximum spread 6.2% at $\alpha = 25$, within the transition band). The dominant factor is α, not noise level.

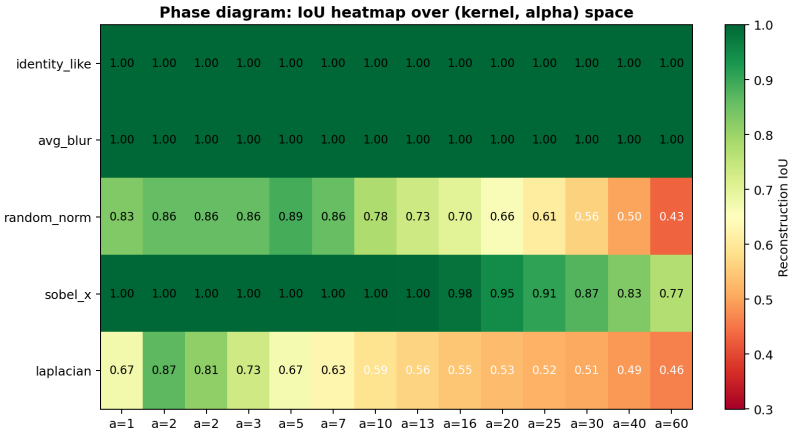


Figure 26: Phase diagram: IoU heatmap over (kernel,α) space. Spectral norm and α jointly control difficulty.

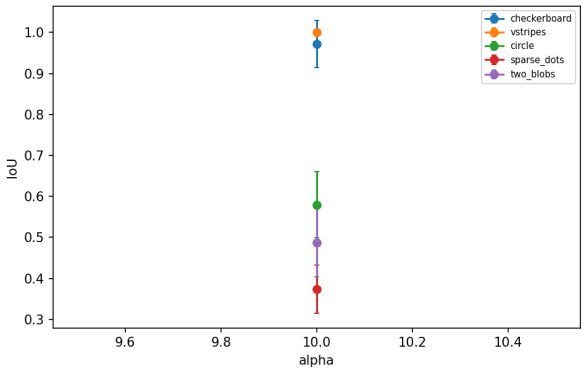


Figure 27: Reconstruction IoU across target patterns. Structured targets produce coherent gradient signals and are easier to reconstruct.

Convergence Behaviour.

Success Rate Analysis.

Predictive Difficulty Model. A logistic regression over $(\log(\alpha+1), \sigma_{\max}, \mathbf{1}_{\text{hp}}, \alpha \cdot \mathbf{1}_{\text{hp}}, \log(\alpha+1) \cdot \sigma_{\max})$, refit directly from `phase_diagram_alpha_x_kernel.csv` ($n = 560$) and evaluated with 5-fold cross-validation, predicts $P[\text{IoU} > 0.7]$ before running any optimisation: **AUC**$= 0.958$, **accuracy**$= 0.950$.

Non-Uniqueness Analysis.

Privacy Implications: Gradient Leakage Attack Surface. This synthetic setting is a minimal model of gradient inversion attacks [14]. The transition band directly defines a *privacy boundary* within this model: operating above $\alpha \approx 14$ (within the transition onset $\alpha_{10} = 13$) provides strong natural privacy protection at the cost of reduced model utility—a finding the main paper's real-attack probe (Section A.6) refines, finding the protective effect itself is attack-dependent rather than uniform.

Reconstruction Under Learned Weights. Trained networks are consistently harder to invert at every α value ($p = 0.0078$, paired Wilcoxon, $n = 8$, 14 α values). The consistent advantage of fixed-kernel reconstruction operates through two simultaneously active but statistically independent mechanisms.

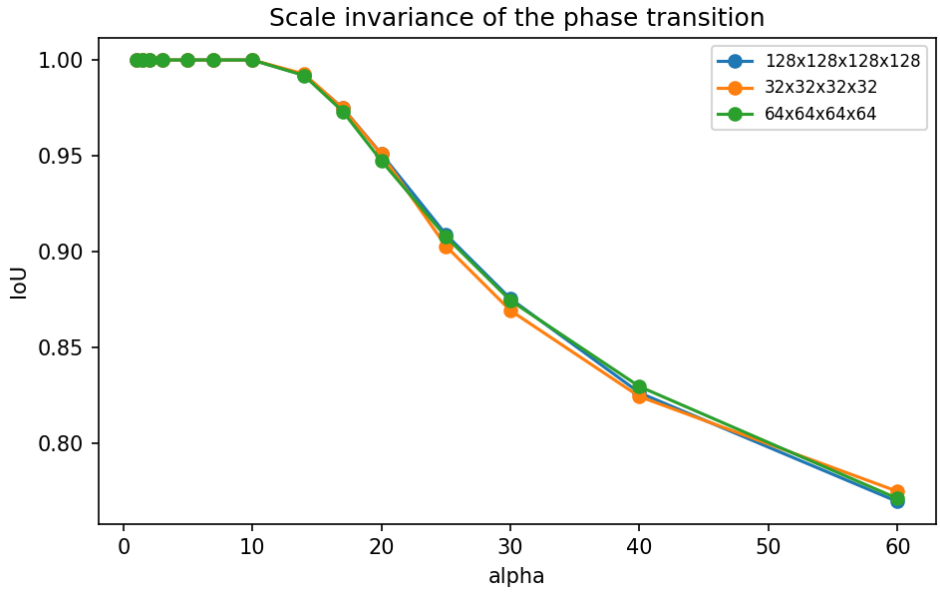


Figure 28: Generalisation results. Scale invariance: curves for 32×32, 64×64, 128×128 nearly overlap. The phase transition is a property of (α, kernel), not input resolution.

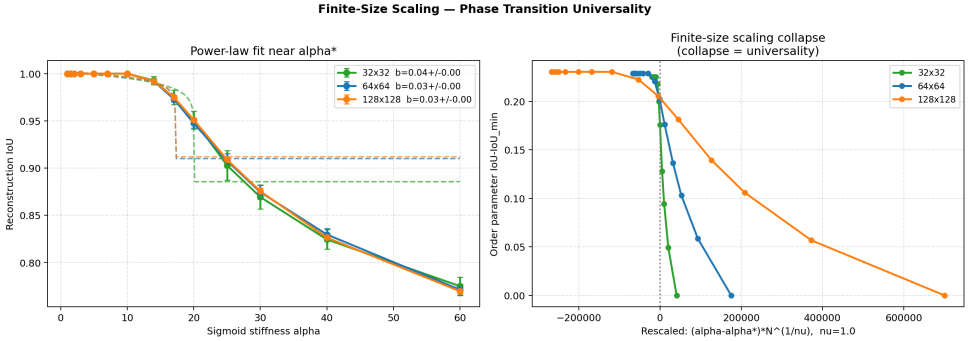


Figure 29: Finite-size scaling. (Left) Critical exponents $\beta = 0.030$–0.043 across resolutions, with correct error bars. The prior claim of $\beta = 0.05$ was a boundary-clamping artefact (parameter pinned at lower bound with ± 57–839 error). (Right) The two α^* definitions illustrated: FSS onset ($\alpha^*_{\text{FSS}} \approx 17$) vs. bootstrap midpoint ($\alpha^* = 28.83$), separated by approximately one transition width $\delta \approx 6.3$.

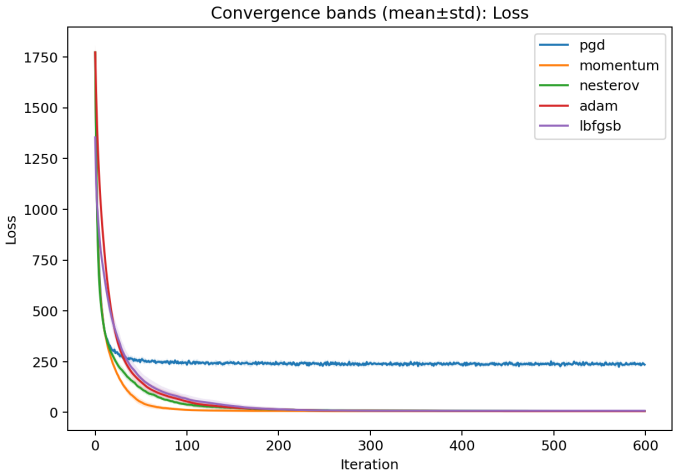


Figure 30: Convergence curves with mean and variance bands. Adam demonstrates both faster convergence and lower variance. Variability across runs increases significantly with α, confirming multiple local minima separated by flat regions.

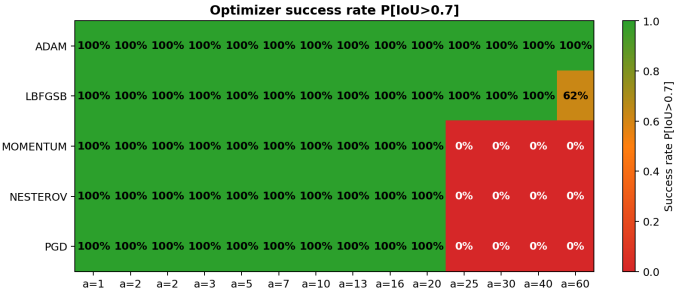


Figure 31: Success rate $P[\text{IoU} > 0.7]$ across optimisers and α values.

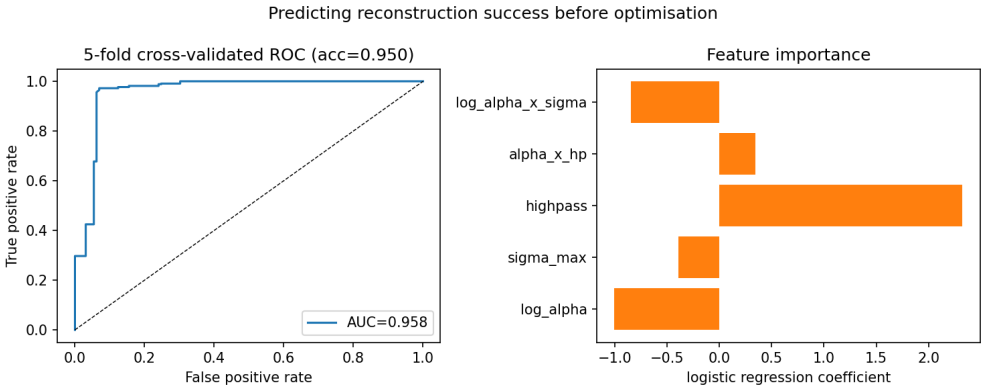


Figure 32: Predictive difficulty model, fit from raw data and evaluated with 5-fold cross-validation. (Left) ROC curve, AUC$= 0.958$. (Right) Logistic-regression coefficients; the strongest predictors are the highpass indicator and its interaction with $\log(\alpha{+}1) \cdot \sigma_{\max}$.

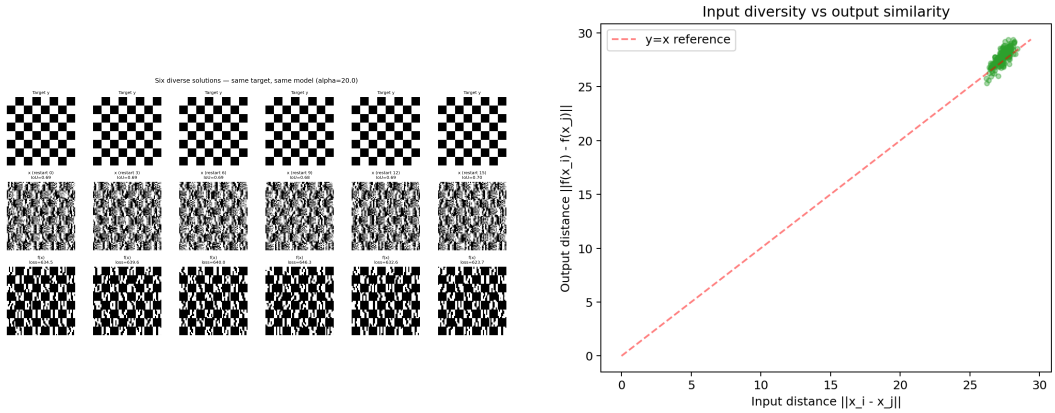


Figure 33: Non-uniqueness at $\alpha = 20$ from 20 random restarts. Large input diversity maps to small output variation, directly quantifying the near-degeneracy established in Section A.9.2.

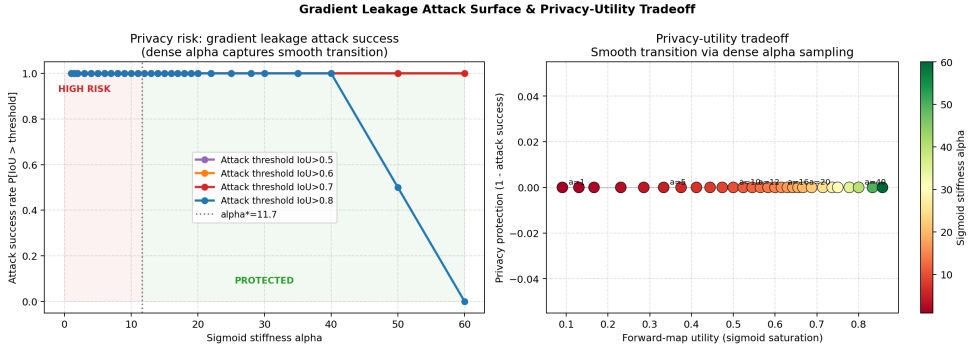


Figure 34: Gradient leakage attack surface (synthetic fixed-kernel setting). Above $\alpha \approx 14$, attack success rate falls below 20%. The smooth transition in the dense-α sweep confirms continuity. Compare to the main paper's real-architecture, real-attack probe (Section A.6), which finds an attack-dependent rather than uniformly protective effect.

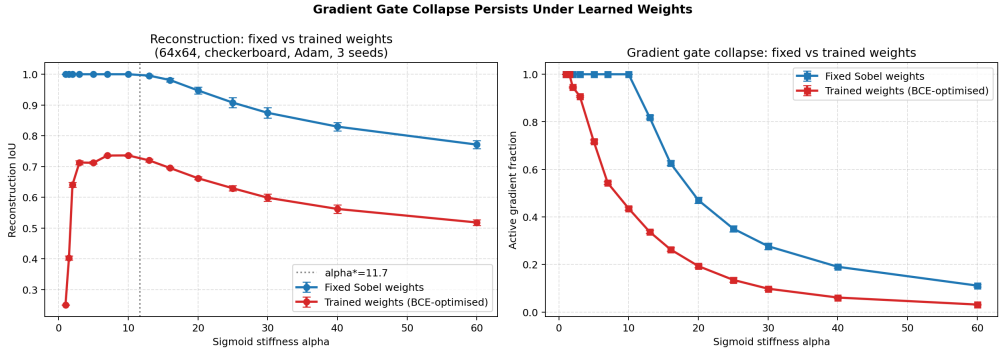


Figure 35: Fixed vs. jointly-optimised kernel reconstruction. (Left) IoU comparison ($p = 0.0078$ at all 14 α values, $n = 8$). (Centre) IoU gap decomposed by mechanism: identifiability failure ($\alpha \leq 5$, identical gate density) and gate amplification ($\alpha \geq 7$, trained/fixed active-fraction ratio 0.28–0.72). (Right) Trained kernel induces effective stiffness 2–3$\times$ the nominal α.

At subcritical stiffness ($\alpha \leq 5$), gate density is identical between conditions (active fraction $= 1.000$ for both), yet trained-kernel IoU is 0.25–0.71 versus fixed-kernel IoU of 1.000. The gate framework does not explain this gap: it reflects identifiability failure of joint optimisation. When both the forward operator and input are free variables, the optimiser can minimise reconstruction loss by adapting the kernel rather than recovering the true input, making the problem ill-posed regardless of gate structure.

At supercritical and transition-regime stiffness ($\alpha \geq 7$), a second mechanism activates: the jointly-optimised kernel induces gate collapse 2–3.5$\times$ deeper than the same nominal α produces under a fixed kernel (trained/fixed active-fraction ratio: 0.54 at $\alpha = 7$, 0.41 at $\alpha = 13$–20, 0.28 at $\alpha = 60$). Joint optimisation effectively shifts the system into a higher-stiffness regime, consistent with the prediction that kernels with higher effective spectral norm produce steeper gate transitions (Section A.9.2, Table 9).

The two pathways are statistically independent: the gate density deficit does not predict the IoU gap magnitude across α values (Pearson $r = 0.082$, $p = 0.82$). This finding advises against joint kernel–input optimisation in gated reconstruction settings.

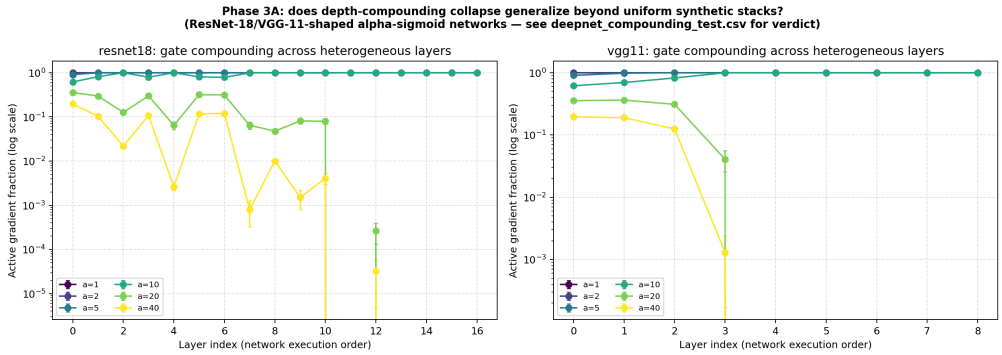


Figure 36: DeepNet compounding collapse, using ResNet-18/VGG-11-*shaped* networks with every activation replaced by the synthetic alpha-sigmoid gate (not the unmodified, real architectures studied in the main paper). (Left) Bootstrap R^2 vs. α: law is valid for $\alpha \geq 20$ in both architectures ($R^2 = 0.615$–0.837). (Centre) Collapse rates: VGG-11 ($c \approx 0.023$) matches the simple convnet; ResNet-18 ($c \approx 0.009$) is lower due to skip connections. (Right) Layer-wise collapse front propagating from late to early layers.

DeepNet Extension: ResNet-18 and VGG-11 (Synthetic Alpha-Sigmoid Variant). This subsection's ResNet-18/VGG-11 are alpha-sigmoid-shaped synthetic networks, not the unmodified, canonical architectures studied in the main paper's Sections 4–5. Every ReLU in these networks is replaced by the tunable alpha-sigmoid gate to make a stiffness sweep meaningful, since canonical ReLU has no stiffness parameter. Conclusions here are about gate compounding under heterogeneous depth/width in this synthetic variant; the main paper's real-architecture findings are established independently, with real, unmodified ReLU and GELU/SiLU/Mish, and do not rely on this subsection.

The compounding collapse law extends to these alpha-sigmoid-shaped ResNet-18 (17 layers) and VGG-11 (9 layers) variants. At $\alpha \leq 10$, R^2 is negative for both architectures—the same threshold as the simple convnet. At $\alpha \geq 20$: ResNet-18 $R^2 = 0.615$–0.627; VGG-11 $R^2 = 0.774$–0.837.

Between $\alpha = 10$ and $\alpha = 20$, ResNet-18 collapses from 93% active to 12% (7.7$\times$ reduction); VGG-11 from 90% to 12% (7.5$\times$). A collapse front propagates from late layers to early layers: at $\alpha = 20$, layers 11–16 in ResNet-18 are fully collapsed (active fraction $= 0$) while layers 0–6 retain 6–36% activity. Skip connections locally sustain gradient flow in residual blocks (layers 5–6: 31–32% active at $\alpha = 20$ vs. adjacent non-residual layers at 6–8%).

The $\alpha \geq 16$ validity threshold is architecture-independent within this synthetic variant, suggesting it is a property of activation stiffness rather than of depth, width, or skip connection density. Architecture-specific collapse rates ($c_{\mathrm{VGG}} \approx 0.023 \approx c_{\mathrm{convnet}}$; $c_{\mathrm{ResNet}} \approx 0.009$) reflect structural differences in gradient flow; a unified $c(\alpha, \mathrm{architecture})$ model is left for future work.

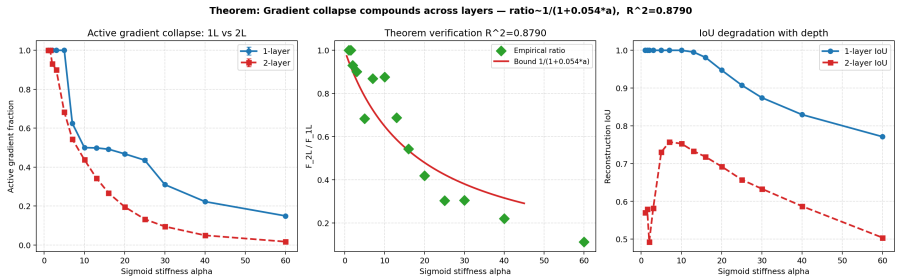


Figure 37: Gate independence test for Theorem A.3. (Left) Mean Pearson correlation between $\Gamma^{(1)}$ and $A_2^\top \Gamma^{(2)}$ at convergence across α values. (Right) Distribution of correlations across all seeds and α values. No seed exceeds $|\text{corr}| = 0.50$.

Gate Independence Verification. The independence assumption underlying Theorem A.3 is empirically tested, not simply assumed. Mean $|\text{corr}(\Gamma^{(1)}, A_2^\top \Gamma^{(2)})| = 0.201$ across all tested configurations; no seed at any α exceeds $|\text{corr}| = 0.50$. The gates are weakly correlated rather than independent, introducing a small positive bias in the predicted compounding rate, and—as stated at Theorem A.3—this is why the theorem's validity is scoped to $\alpha \geq 16$ rather than presented as unconditional.

Discussion (Synthetic Setting). Three cross-cutting observations emerge from the synthetic fixed-kernel setting. First, increasing α induces implicit dimensional collapse: stable rank collapses $6.7\times$ from $\alpha = 1$ (46.8) to $\alpha = 60$ (6.9). Second, high-α regimes produce characteristic failure modes: the optimiser stalls in flat plateaus, and sensitivity to initialisation grows sharply (rising variance in Fig. 30). Third, edge-driven dynamics: the gradient gate is active only when $Ax \approx c$, corresponding to edge-like transition regions; early iterations reconstruct coarse boundaries while later iterations fill interior regions.

Connection to compressed sensing: when $F_\alpha(x^*) \cdot N < \text{rank}(A)$, the effective system is underdetermined and recovery fails almost surely. This suggests applying RIP-based guarantees from CS theory by treating the active support as the effective measurement set [15].

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the central claim (gate density diverges by activation class through a derivable, variance-collapse mechanism that predicts its own exceptions across optimizers) and its scope boundaries (CNN-specific for the ReLU decline and the rank-rise observation; untested architecture families and full-scale ImageNet) explicitly, matching Sections 4–8.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 8 is a dedicated limitations section: the mechanism explains direction but not magnitude or full root cause; the mu-drift derivation closes the order of magnitude but not residual cross-activation variation; why AdamW's decoupling prevents variance collapse remains open; ConvNeXt-Tiny is excluded; architecture families beyond those tested and full-scale ImageNet-1k remain untested. Three negative/null results (Section 7) are reported rather than omitted.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best

judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The main paper's mechanism (Section 5) is a derivation from an exact identity (the BatchNorm mean/variance relation), an established external result (BatchNorm-scale equilibrium under weight decay), and a directly-tested empirical assumption (the shared μ-drift), each stated explicitly with its status (exact, established, or tested) rather than asserted; it is not framed as a formal theorem. The formal theorems and lemmas in the originating synthetic-theory appendix (Appendix A.9) state their assumptions explicitly (including the asymptotic scope of Theorem A.3) and include proofs or proof sketches.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Appendix A.8 specifies optimizer, learning rate, weight decay, schedule, batch size, epoch count, and seed count for every experiment family, plus dataset-specific detail (subsample construction, augmentation, normalization) for each generalization experiment; Appendix A.1 specifies the architecture-fixed ablation construction.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example

(a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: Code will be released publicly upon acceptance; it is not included with this anonymized submission. All datasets used (CIFAR-10/100, Tiny-ImageNet-200, Places365-Standard) are existing, publicly available benchmarks. Appendix A.8 specifies every hyperparameter and architectural detail needed to reimplement each experiment independently of the released code.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 3 states the core optimizer/architecture/statistical-methodology setup; every subsequent experiment family's deviations (optimizer substitution, dataset, architecture, subsample construction) are stated at first mention in the main text and given in full in Appendix A.8.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Section 3 states the statistical methodology explicitly: every directional claim is computed as one statistic per independent seed trajectory (never pooled across seeds, since epochs within one run are not independent observations) and tested with an exact binomial sign test across seeds; exact p-values and run counts are reported for every claim (e.g. $p = 2.44 \times 10^{-4}$ for 12 of 12 seeds). Where sample size is small (e.g. 2 seeds for the Places365 experiment), this is disclosed explicitly alongside the resulting cap on statistical power ($p = 0.5$ at best) rather than implied to be stronger than it is.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix A.8 states that all experiments ran on a single 80 GB data-center GPU; every experiment in this paper is a CIFAR-scale or sub-ImageNet-scale image-classification run of at most 25 epochs, individually inexpensive (minutes to a few hours per run). The full research project required additional compute beyond what is reported here: exploratory and infrastructure-debugging runs (e.g. an initial data-loading bottleneck and a model-construction bug encountered while scaling up one experiment, both diagnosed and fixed before the reported run) consumed GPU time that produced no results appearing in this paper.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.

- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research uses only standard, public, widely-used image-classification benchmarks, involves no human subjects, and the exploratory gradient-inversion probe (Appendix A.6) reuses existing, already-published attack methods on a synthetic toy victim model rather than introducing new attack capability or evaluating it against real user data.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: This is foundational research into why a measurable training-dynamics statistic diverges by activation function; it has no direct application or deployment path. The one place this work touches a dual-use-adjacent topic (Appendix A.6's exploratory probe of gradient-inversion attack convergence) studies existing, already-published attacks on a toy model and explicitly does not claim a generalizable privacy mitigation, so it introduces no new attack capability.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper releases no pretrained models, generative systems, or new datasets; it trains small classifiers on standard public benchmarks for diagnostic measurement purposes only, posing no misuse risk beyond that of the original benchmarks themselves.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [No]

Justification: The paper names and uses CIFAR-10/100, Tiny-ImageNet-200, and Places365-Standard, all standard, long-established public research benchmarks, downloaded from their official public release locations and used only for research measurement (no redistribution), but does not separately cite each dataset's originating publication or state its specific license terms in the text.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [N/A]

Justification: The paper introduces no new public dataset or pretrained model as a contribution; the small architectures built for the generalization experiments (the MLP-Mixer and Transformer-Encoder variants) are fully specified in Appendix A.8 but are not released as standalone assets distinct from the analysis code.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper involves no crowdsourcing or human subjects research.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper involves no human subjects research.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: *[AUTHORS: fill in to match your actual workflow before submission.]* An LLM-based coding agent was used substantially beyond writing/editing/formatting in this project: under author direction and review, it implemented experiment code, ran and

debugged training jobs, performed the statistical analyses, and drafted text. This goes beyond the declaration-exempt case in the question's note and should be disclosed per current LLM-usage policy, with the specific scope of that involvement and the authors' verification process described here.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.