ARE PARAMETER-EFFICIENT FINE-TUNING METHODS REALLY DIFFERENT?

Anonymous authors
Paper under double-blind review

ABSTRACT

Parameter-efficient fine-tuning (PEFT) offers many parameterizations, yet their methodological and functional differences remain unclear. We compare six methods in language and diffusion models to examine how their parameterizations relate to task performance, forgetting, and changes in pretrained weight geometry. Motivated by the spectrum-preserving design of orthogonal fine-tuning (OFT), we first ask whether spectral preservation is itself important for adaptation and retention. We find that the selected LoRA-family methods also approximately preserve pretrained geometry, and that restoring their slightly drifted singular-value spectra largely preserves task performance, questioning the necessity of explicit geometric preservation. Beyond this, we observe that some methods exhibit distinct adaptation–retention trade-offs that vary across settings: LoRA most consistently limits forgetting at competitive performance, DoRA achieves higher mean task scores than LoRA in most comparisons, while PiSSA often incurs greater retention costs. Further intervention experiments suggest that while performance gains from different PEFT methods can be attributed to modifications in different groups of spectral components, we consistently find that restoring dominant rather than intermediate or trailing components produces the largest mean reduction in general-text NLL or base-image drift. Together, these results motivate evaluating geometric constraints through their functional consequences rather than preservation alone.

1 INTRODUCTION

Parameter-efficient fine-tuning (PEFT) (Houlsby et al., 2019; Li & Liang, 2021; Hu et al., 2022) has become a standard approach for adapting large pretrained models. Over the past several years, a large number of PEFT algorithms have been proposed, often by modifying a small number of recurring parameterizations.

One major paradigm is the LoRA family, which adapts pretrained weights through *additive* low-rank updates (Hu et al., 2022). Subsequent variants modify the optimization, scaling, initialization, or magnitude–direction decomposition of these updates (Hayou et al., 2024; Kalajdzievski, 2023; Wang et al., 2024; Meng et al., 2024; Liu et al., 2024). A second paradigm is the OFT family, which adapts weights *multiplicatively* through structured orthogonal transformations and their variants (Qiu et al., 2023; 2025b;a; Yuan et al., 2024). These methods are motivated by seemingly different principles, and many are presented as principled improvements over vanilla LoRA. Yet it remains unclear which of these differences survive careful tuning and, more importantly, which differences matter functionally. This raises a basic question:

When do typical PEFT methods really differ, and which differences matter?

Recent work has begun to question whether apparent methodological improvements reflect genuinely different operating behavior. For the LoRA family, Schulman & Thinking Machines Lab (2025) identify rescaling invariances linking initialization scale, update scaling, and factor learning rates, and show that vanilla LoRA can be highly competitive when properly configured. Lee et al. (2026) likewise find similar peak performance between LoRA and its variants after method-specific learning-rate tuning, yet 46 of 64 surveyed studies report a fixed learning rate. Hugging Face PEFT's mathematical-reasoning and image-generation benchmarks (Mangrulkar et al., 2022, v0.21.0) also

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

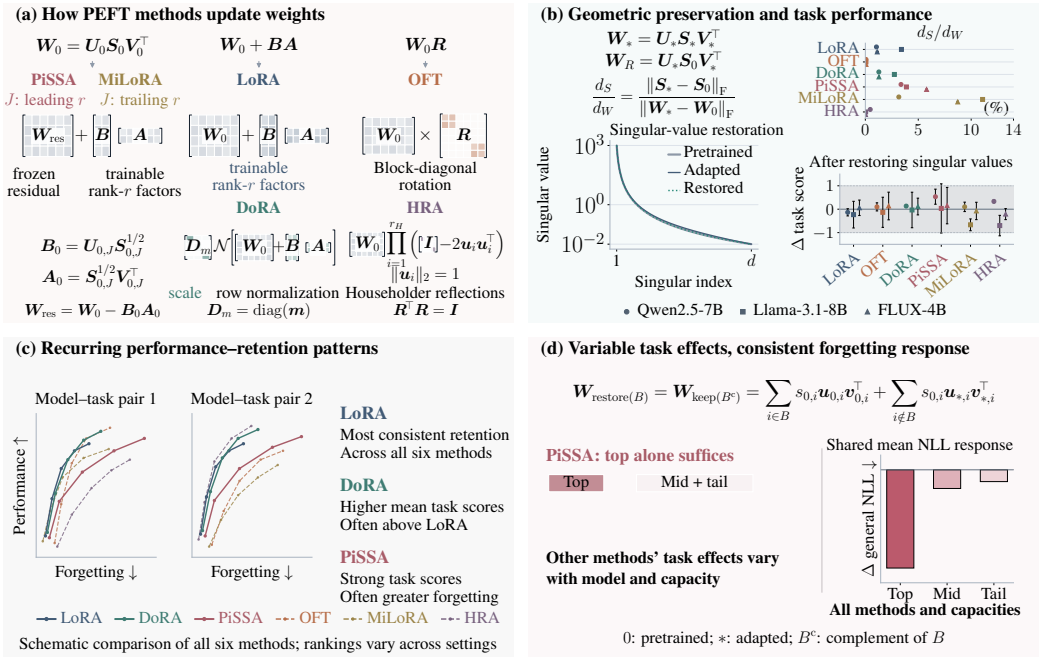

Figure 1: Overview of the questions studied in this work. (a) Six representative PEFT parameterizations. (b) Changes in pretrained singular spectra and the singular-value restoration intervention. (c) Performance–retention comparisons across PEFT methods. (d) Spectral interventions that selectively retain or restore dominant, intermediate, and trailing components. Singular-value decay curves and bar heights are schematic.

report a single training seed and, for most methods, only one learning-rate setting. Comparison settings also differ across studies. On GSM8K and MATH, PiSSA's main comparison adapts all linear layers across three 7B backbones for one epoch (Meng et al., 2024), whereas HRA (Yuan et al., 2024) adapts only LLaMA2-7B's query and value projections for two epochs. These observations motivate common experimental settings, method-specific tuning and multiple training seeds to separate configuration effects from methodological gains.

Performance alone, however, may not reveal the full distinction between methods. Two adapters can achieve similar downstream scores while differing substantially in how much pretrained behavior they disrupt. This issue is particularly relevant to orthogonal adaptation. OFT explicitly preserves hyperspherical geometry (Qiu et al., 2023), while HRA connects orthogonal transformations and low-rank adaptation through Householder reflections (Yuan et al., 2024). Such geometric constraints are motivated in part by improved generalization and preservation of pretrained knowledge (Qiu et al., 2023; Yuan et al., 2024; Huang et al., 2026). But two questions remain unresolved: do unconstrained LoRA-family methods actually depart substantially from these geometries in practice, and does tighter geometric preservation translate into better adaptation or retention?

1.1 Contributions and Findings

We address these questions by combining controlled empirical comparisons with direct interventions on pretrained weight geometry. We study six representative methods, including LoRA, DoRA, PiSSA, MiLoRA, OFT, and HRA, across language-model adaptation and diffusion personalization. Rather than comparing parameterizations only through final task scores, we examine three levels of behavior: their operating performance after method-specific tuning, their retention of pretrained capabilities, and the functional consequences of the geometric changes they induce. As summarized in Figure 1, this analysis reveals a recurring mismatch: *parameterization labels can overstate geometric differences, whereas task scores can understate functional differences.* Our study leads to three main findings.

Explicit geometric preservation is not what primarily separates PEFT methods. We first characterize adaptation relative to the two-sided orthogonal orbit of the pretrained weights. For small isotropically oriented additive updates, the fraction of weight displacement appearing as singular-value displacement scales as $1/\sqrt{\max(m,n)}$, showing that limited spectral drift can arise even without an explicit orthogonality constraint. Consistent with this prediction, the LoRA-family methods we study exhibit only modest spectral drift and very small changes in hyperspherical energy. More importantly, restoring pretrained singular values while retaining the adapted singular bases largely preserves task performance. Thus, explicit geometric preservation is neither unique to successful PEFT nor, in our tested settings, necessary for retaining its task gains.

Similar task performance can conceal more persistent differences in retention. After method-specific learning-rate tuning, the six methods often achieve competitive downstream performance, but their retention behavior differs more systematically. Across the tested settings, LoRA most consistently limits forgetting while maintaining competitive performance. DoRA achieves higher mean task scores than LoRA in most comparisons, typically with some additional forgetting, whereas PiSSA often incurs larger retention costs. Crucially, tighter preservation of pretrained geometry does not reliably predict better retention. The more persistent distinctions between methods therefore appear in their functional consequences, rather than simply in the geometric invariants imposed by their parameterizations.

Adaptation is spectrally heterogeneous, whereas forgetting is spectrally structured. We next intervene on dominant, intermediate, and trailing spectral components, ordered by pretrained singular value. For task adaptation, there is no universal spectral location of useful change: PiSSA consistently concentrates task-relevant changes in the dominant components, whereas for the other methods the sufficient components vary with the model, capacity, and task. For forgetting, however, a markedly more consistent pattern emerges: restoring the dominant components produces the largest mean reduction in general-text NLL or base-image drift across all six methods and tested settings. This ordering persists even in cases where the dominant-component edit is smaller in norm than alternative edits. These results suggest that downstream adaptation can be realized through multiple spectral routes, while forgetting is more consistently coupled to changes in dominant pretrained components.

Taken together, our results suggest that PEFT methods are better distinguished by their operating regimes and the functional consequences of learned weight geometry than by parameterization labels or explicit geometric preservation alone.

1.2 Related work

Adaptation, retention and parameterization. Within the LoRA family, several methods use singular directions of pretrained weights or gradients to initialize adaptation (Meng et al., 2024; Wang et al., 2024; Zhang et al., 2025; Ponkshe et al., 2024). Optimization and scaling studies explain how parameterization affects adaptation (Hayou et al., 2024; Kalajdzievski, 2023; Wang et al., 2024); Lee et al. (2026) link preferred learning rates to loss curvature. Biderman et al. (2024) compare LoRA's adaptation and forgetting with full fine-tuning. PEFT-Arena compares PEFT methods using a shared SFT learning rate and one run per configuration, leaving method-specific tuning and seed uncertainty unresolved (Huang et al., 2026). ETHER questions the practical necessity of hyperspherical-energy preservation (Bini et al., 2024). We examine preservation alongside tuned retention comparisons and spectral interventions.

Spectral structure and functional interventions. Shuttleworth et al. (2025) identify intruder singular directions after LoRA adaptation. He & Cao (2025) study spectrum and vector replacement after full fine-tuning. Jin et al. (2026) connect forgetting and recovery during full-parameter SFT/RL to singular-vector changes. CASA studies spectral preservation in diffusion-LoRA transfer (Wang et al., 2026), while SVFT learns combinations of pretrained singular-vector pairs (Lingam et al., 2024). Our contribution is to connect tuning-controlled comparisons with complementary keep/reset interventions, distinguishing recurring retention responses from setting-dependent contributions to task performance.

Table 1: Selected PEFT methods and their parameterizations.

Method	Adapted matrix $\boldsymbol{W}_*$	Initialization or structure
LoRA	$\boldsymbol{W}_0 + s\boldsymbol{B}\boldsymbol{A}$	Zero initial update
DoRA	$\mathrm{diag}(\boldsymbol{m})\mathcal{N}(\boldsymbol{W}_0 + s\boldsymbol{B}\boldsymbol{A})$	Learned magnitudes and directions
PiSSA	$\boldsymbol{W}_{\mathrm{res}} + s\boldsymbol{B}\boldsymbol{A}$	Leading singular components
MiLoRA	$\boldsymbol{W}_{\mathrm{res}} + s\boldsymbol{B}\boldsymbol{A}$	Trailing singular components
OFT	$\boldsymbol{W}_0\boldsymbol{R}$	$\boldsymbol{R} = \mathrm{blkdiag}(\boldsymbol{R}_j)$
HRA	$\boldsymbol{W}_0\boldsymbol{R}$	$\boldsymbol{R} = \prod_{i=1}^{r_H}(\boldsymbol{I} - 2\boldsymbol{u}_i\boldsymbol{u}_i^\top)$

2 Methods and Study Design

We consider six representative PEFT algorithms: LoRA (Hu et al., 2022), DoRA (Liu et al., 2024), PiSSA (Meng et al., 2024), MiLoRA (Wang et al., 2025), OFT (Qiu et al., 2023), and HRA (Yuan et al., 2024).

Parameterizations. The six methods span low-rank addition, magnitude–direction decomposition, spectral initialization and orthogonal transformations. For pretrained $\boldsymbol{W}_0 \in \mathbb{R}^{m\times n}$, Table 1 summarizes the six methods. The low-rank factors are $\boldsymbol{B} \in \mathbb{R}^{m\times r}$ and $\boldsymbol{A} \in \mathbb{R}^{r\times n}$, with scale $s = \alpha/r$. PiSSA and MiLoRA freeze $\boldsymbol{W}_{\mathrm{res}} = \boldsymbol{W}_0 - s\boldsymbol{B}_0\boldsymbol{A}_0$, preserving the starting model. For DoRA, $\mathcal{N}$ normalizes each row and $\boldsymbol{m}$ contains learned row magnitudes. OFT uses Cayley blocks $\boldsymbol{R}_j = (\boldsymbol{I} + \boldsymbol{Q}_j)(\boldsymbol{I} - \boldsymbol{Q}_j)^{-1}$ with skew-symmetric $\boldsymbol{Q}_j$. HRA uses $\boldsymbol{u}_i$ with $\|\boldsymbol{u}_i\|_2 = 1$.

Experiment setup. We evaluate mathematics, coding and diffusion personalization, reporting means and sample SDs over three training seeds unless otherwise specified. Qwen2.5-7B (Qwen Team, 2024) and Llama-3.1-8B (Grattafiori et al., 2024) each train on 20,000 examples per task from MetaMath (Yu et al., 2024) for mathematics and CodeFeedback (Zheng et al., 2024) for coding. Methods share data exposure and adapted modules, with approximately matched parameter budgets. We select LRs by mean GSM8K (Cobbe et al., 2021) development accuracy for mathematics or held-out response NLL for coding, then report held-out GSM8K accuracy and greedy HumanEval+/MBPP+ pass@1 (Liu et al., 2023a). FineWiki (Penedo, 2025) $\Delta\mathrm{NLL} = \mathrm{NLL}_* - \mathrm{NLL}_0$ measures forgetting, with lower values indicating better retention. For FLUX.2-klein-base-4B personalization (Black Forest Labs, 2026), we compare all six methods on the benchmark cat at four approximately matched budgets and extend the LoRA/OFT comparison to 28 objects (Ruiz et al., 2023). Validation DINO (Oquab et al., 2024) selects LRs, while held-out test DINO measures fidelity. Base-image drift, CLIP alignment (Radford et al., 2021) and COCO flow loss (Lin et al., 2014) assess complementary retention outcomes. Spectral interventions use the same development/validation selection criteria across all six methods on mathematics and diffusion and jointly edit all adapted matrices. Each Pareto point represents one capacity/LR configuration.

3 Geometric Preservation and Functional Consequences

We examine how PEFT methods change pretrained weight geometry and whether these changes are necessary for adaptation. We first explain singular-value and singular-basis changes theoretically, then measure spectral and hyperspherical energy preservation in checkpoints. Finally, we restore pretrained singular values to test whether adapted singular bases alone retain task gains.

Theoretical perspective. For $\boldsymbol{W}_0, \boldsymbol{W}_* \in \mathbb{R}^{m\times n}$, let $\boldsymbol{W}_j = \boldsymbol{U}_j\,\mathrm{diag}(\boldsymbol{s}_j)\boldsymbol{V}_j^\top$ denote thin SVDs with decreasing singular values, and let $O(k)$ denote the orthogonal group. Relative weight and spectral displacement are defined by $d_W = \|\boldsymbol{W}_* - \boldsymbol{W}_0\|_{\mathrm{F}}/\|\boldsymbol{W}_0\|_{\mathrm{F}}$ and $d_S = \|\boldsymbol{s}_* - \boldsymbol{s}_0\|_2/\|\boldsymbol{s}_0\|_2$.

Proposition 3.1 (Distance to the orthogonal orbit)**.** *The restored matrix* $\boldsymbol{W}_R = \boldsymbol{U}_*\,\mathrm{diag}(\boldsymbol{s}_0)\boldsymbol{V}_*^\top$ *satisfies*

$$\min_{\boldsymbol{Q}_L\in O(m),\,\boldsymbol{Q}_R\in O(n)} \|\boldsymbol{W}_* - \boldsymbol{Q}_L\boldsymbol{W}_0\boldsymbol{Q}_R^\top\|_{\mathrm{F}} = \|\boldsymbol{W}_* - \boldsymbol{W}_R\|_{\mathrm{F}} = \|\boldsymbol{s}_* - \boldsymbol{s}_0\|_2, \tag{1}$$

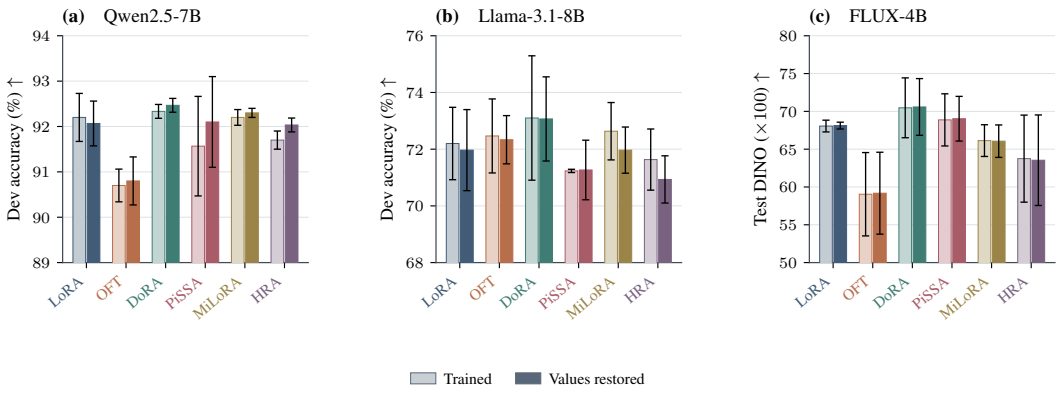

Figure 2: Task performance largely survives singular-value restoration. At the largest capacities and development/validation-selected LRs, pale and saturated bars show trained and restored scores with sample SD error bars. LLM panels report development accuracy, and FLUX reports same-run test DINO for all six methods.

and attains this minimum (Mirsky, 1975). For nonzero base and update norms, d_S/d_W is the distance to this orbit relative to the update norm.

A small ratio means that changing the singular bases while keeping pretrained singular values closely approximates the adapted weights, with error measured relative to the update norm.

Proposition 3.2 (First-order spectral and basis motion)**.** *Let $q = \min(m, n)$ and suppose $\boldsymbol{W}_0$ has q distinct positive singular values. For $\boldsymbol{W}_*(\epsilon) = \boldsymbol{W}_0 + \epsilon \boldsymbol{E}$, set $\dot{\boldsymbol{s}} = \mathrm{d}\boldsymbol{s}(\epsilon)/\mathrm{d}\epsilon|_{\epsilon=0}$. Then*

$$\dot{s}_i = \boldsymbol{u}_{0,i}^\top \boldsymbol{E} \boldsymbol{v}_{0,i}, \qquad \boldsymbol{E}_N = \boldsymbol{U}_0 \operatorname{diag}(\dot{\boldsymbol{s}}) \boldsymbol{V}_0^\top, \qquad \boldsymbol{E}_T = \boldsymbol{E} - \boldsymbol{E}_N \tag{2}$$

are the spectral derivative (Stewart, 1990), the normal component and the tangent component to the two-sided orthogonal orbit, respectively. In particular, $\boldsymbol{E}_T$ changes only singular bases to first order, and $\|\boldsymbol{E}\|_{\mathrm{F}}^2 = \|\dot{\boldsymbol{s}}\|_2^2 + \|\boldsymbol{E}_T\|_{\mathrm{F}}^2$. If $\|\boldsymbol{E}\|_{\mathrm{F}} = 1$ and $\mathbb{E}[\operatorname{vec}(\boldsymbol{E}) \operatorname{vec}(\boldsymbol{E})^\top] = \boldsymbol{I}_{mn}/(mn)$, then

$$\mathbb{E}\|\dot{\boldsymbol{s}}\|_2^2 = \frac{1}{\max(m,n)}, \qquad \lim_{\epsilon \to 0} \sqrt{\mathbb{E}\left[\left(\frac{d_S}{d_W}\right)^2\right]} = \frac{1}{\sqrt{\max(m,n)}}. \tag{3}$$

Small isotropic updates therefore have RMS $d_S/d_W \approx 1/\sqrt{\max(m,n)}$: spectral displacement can be a small share of weight displacement without an orthogonality constraint. This conditional prediction also holds for uniformly oriented low-rank updates (proofs and pooling in Appendix E).

Empirical evaluation. At the largest capacities, spectral displacement is a modest fraction of weight displacement even for unconstrained methods (Figure 1b): about 11% for Llama MiLoRA versus 3–4% for LoRA, DoRA, and PiSSA, and 5.8% and 8.7% for FLUX PiSSA and MiLoRA, respectively. We also measure hyperspherical energy (HE), whose preservation motivates OFT's approach to adaptation (Qiu et al., 2023). For nonzero rows with distinct normalized directions, $H(\boldsymbol{W}) = \sum_{i \neq j} \|\widehat{\boldsymbol{w}}_i - \widehat{\boldsymbol{w}}_j\|_2^{-1}$, where $\widehat{\boldsymbol{w}}_i = \boldsymbol{w}_i/\|\boldsymbol{w}_i\|_2$. Right-orthogonal transformations preserve hyperspherical energy exactly, whereas preserving singular values alone does not. Nevertheless, all six methods approximately preserve HE: every checkpoint's mean absolute relative change across adapted matrices is below 0.01% at the largest capacities (Appendix C.3). LoRA-family methods therefore approximately preserve both quantities without enforcing orthogonality. Their small spectral displacement further supports a two-sided orthogonal approximation of adapted weights (Proposition 3.1). To assess the functional importance of singular-basis changes relative to singular-value scaling, we restore pretrained singular values while retaining adapted bases (He & Cao, 2025; Jin et al., 2026). Across LLM and FLUX checkpoints, all points at the largest capacities remain within one percentage point of reconstruction (Figure 1b). See Appendix C for additional results at other capacities.

Table 2: Development-selected LLM test scores (%) and FineWiki forgetting ($10^3\Delta$NLL). Budgets are approximately matched: S/M/L for GSM8K; one for CodeFeedback. HE+ denotes HumanEval+. Base is one unadapted evaluation. Shading marks each column's best adapted mean, including ties.

(a) Mathematics: GSM8K test accuracy (%, ↑)

	Qwen2.5-7B			Llama-3.1-8B		
Method	S	M	L	S	M	L
LoRA	$83.07_{\pm 0.54}$	$83.22_{\pm 0.16}$	$83.65_{\pm 0.42}$	$70.84_{\pm 1.09}$	$\mathbf{71.65}_{\pm 0.35}$	$71.90_{\pm 0.12}$
OFT	$83.35_{\pm 0.36}$	$82.92_{\pm 0.46}$	$83.50_{\pm 0.27}$	$\mathbf{71.42}_{\pm 1.45}$	$71.34_{\pm 0.13}$	$71.75_{\pm 1.18}$
DoRA	$83.27_{\pm 0.04}$	$82.64_{\pm 0.59}$	$83.12_{\pm 0.53}$	$69.95_{\pm 0.12}$	$70.91_{\pm 0.44}$	$\mathbf{71.97}_{\pm 1.16}$
PiSSA	$\mathbf{83.85}_{\pm 0.47}$	$\mathbf{83.67}_{\pm 0.23}$	$83.17_{\pm 0.20}$	$70.33_{\pm 1.49}$	$69.98_{\pm 0.27}$	$70.26_{\pm 0.98}$
MiLoRA	$83.35_{\pm 0.24}$	$\mathbf{83.67}_{\pm 0.46}$	$\mathbf{83.73}_{\pm 0.12}$	$70.43_{\pm 0.42}$	$71.06_{\pm 0.61}$	$70.68_{\pm 0.23}$
HRA	$83.60_{\pm 0.12}$	$83.65_{\pm 0.49}$	$83.22_{\pm 0.16}$	$67.63_{\pm 0.42}$	$70.31_{\pm 1.37}$	$71.04_{\pm 0.20}$

(b) Coding: test pass@1 and general-text forgetting

	Qwen2.5-7B			Llama-3.1-8B		
Method	HE+ ↑	MBPP+ ↑	$10^3\Delta$NLL ↓	HE+ ↑	MBPP+ ↑	$10^3\Delta$NLL ↓
Base	51.22	64.02	0	31.71	50.79	0
LoRA	$52.24_{\pm 0.35}$	$64.29_{\pm 0.53}$	$\mathbf{1.60}_{\pm 0.49}$	$36.99_{\pm 0.93}$	$54.76_{\pm 0.46}$	$\mathbf{10.73}_{\pm 0.56}$
OFT	$51.83_{\pm 0.61}$	$\mathbf{64.55}_{\pm 0.53}$	$2.23_{\pm 0.47}$	$39.02_{\pm 1.06}$	$54.41_{\pm 1.88}$	$25.01_{\pm 2.49}$
DoRA	$\mathbf{52.64}_{\pm 0.70}$	$64.37_{\pm 0.61}$	$1.89_{\pm 0.47}$	$37.80_{\pm 1.06}$	$\mathbf{56.08}_{\pm 0.00}$	$13.51_{\pm 0.97}$
PiSSA	$52.44_{\pm 1.61}$	$63.93_{\pm 0.61}$	$4.25_{\pm 0.31}$	$\mathbf{40.45}_{\pm 0.93}$	$53.88_{\pm 0.40}$	$43.00_{\pm 0.73}$
MiLoRA	$51.02_{\pm 0.93}$	$64.02_{\pm 0.53}$	$3.11_{\pm 0.35}$	$37.80_{\pm 0.00}$	$54.06_{\pm 0.67}$	$22.61_{\pm 1.53}$
HRA	$51.83_{\pm 1.83}$	$64.46_{\pm 0.85}$	$2.92_{\pm 1.08}$	$35.57_{\pm 0.70}$	$55.64_{\pm 1.19}$	$15.09_{\pm 1.11}$

Summary. In our settings, LoRA-family methods approximately preserve the spectra and hyperspherical energy that OFT is designed to preserve exactly. Singular-value restoration largely retains task performance. Tighter preservation must be judged by its task and retention outcomes.

4 Competitive Performance, Recurring Retention Differences

In this section, we study task performance and forgetting across PEFT methods and ask whether closer geometric preservation leads to better adaptation–retention trade-offs. We also examine method-specific patterns across settings.

4.1 Language Models: Competitive Performance, Different Retention

We first examine whether comparable task performance in mathematics and coding is accompanied by comparable retention of general language modeling. We consider both development-selected test results and the mathematics performance–retention frontiers across learning rates and capacities.

LoRA most consistently limits forgetting among all six methods. Its mean NLL increase is lowest in 4 of 6 mathematics settings and both coding backbones (Table 2), while task performance remains competitive across settings. See Appendix B.6 for robustness checks. OFT forgets least for Llama at the largest capacity, with LoRA second. For Qwen at the smallest capacity, LoRA's high-LR selection forgets most and MiLoRA least.

DoRA usually exceeds LoRA's mean task scores. It leads in 10/14 comparisons: all four coding comparisons, all four cat budgets, and two mathematics settings (small Qwen and large Llama; Table 2, Appendix B.6). DoRA's forgetting ranks among the lowest three methods in every selected LLM setting.

PiSSA frequently incurs greater retention costs. It has the largest mean NLL increase on both coding backbones and half the mathematics settings, yet leads Llama HumanEval+. At matched

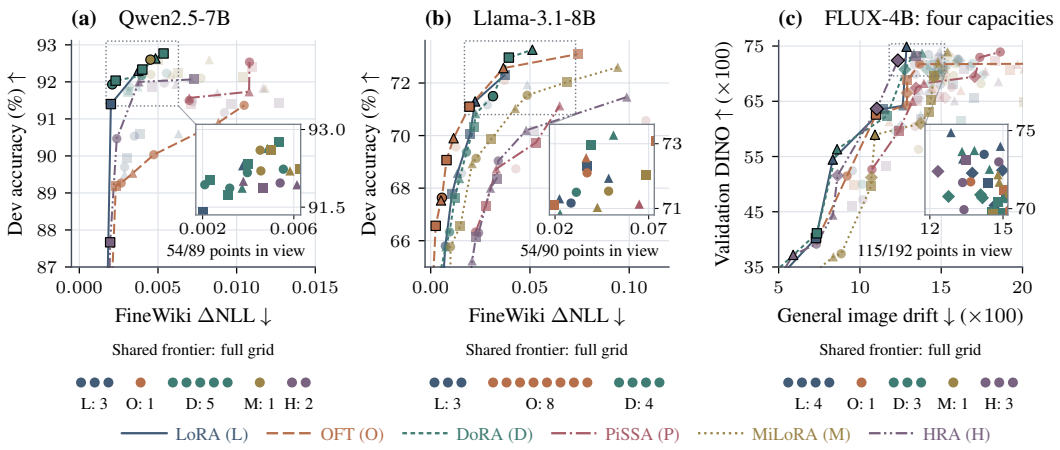

Figure 3: Development performance–retention envelopes across LRs and capacities. Shapes denote increasing capacity: circle, square, triangle, then diamond (FLUX only). Faint points are dominated within their method. Line styles identify method frontiers. Black borders mark the shared frontier across methods. Each icon counts one configuration in the full grid (89 Qwen, 90 Llama, 192 FLUX). Insets magnify boxed regions with uniform point styling. (a,b) Dev accuracy versus FineWiki ΔNLL. (c) Cat validation DINO versus general-image drift.

LRs and capacities, PiSSA forgets most in every complete LLM comparison and constituent seed (further comparisons in Appendix B.6).

PiSSA also illustrates why closer geometric preservation need not improve retention. At the largest Llama capacity, it has smaller HE and pairwise-cosine changes than LoRA (Appendix C.3) but a greater NLL increase in every seed (means 0.0646 versus 0.0487).

On the development frontiers, LoRA and DoRA also appear repeatedly, with DoRA reaching the highest accuracy on both backbones (Figure 3a,b). OFT contributes many Llama frontier points, whereas PiSSA contributes none.

4.2 Diffusion: recurring drift patterns, distinct retention metrics

We next examine whether the retention patterns observed in language extend to diffusion personalization. We compare all six methods on cat personalization across parameter budgets, then extend the LoRA–OFT comparison across objects using complementary measures of retention.

The retention pattern recurs across cat budgets. LoRA has the lowest mean general drift at 3 of 4 validation-selected budgets and the second lowest at the other (Table 3a). DoRA has higher mean test DINO than LoRA at every budget, with additional drift, and leads fidelity at the smallest and largest budgets. PiSSA exceeds LoRA's drift in all 12 paired seeds; its medium-budget fidelity lead again accompanies a retention cost.

HRA has the highest mean test fidelity and lowest drift at the small budget and extends the pooled frontier, but its advantage does not persist at larger capacities. LoRA and DoRA also contribute; PiSSA contributes no frontier point (Figure 3c). MiLoRA has lower mean fidelity and higher drift than LoRA at every selected budget.

Across 28 objects, LoRA and OFT have close test fidelity. LoRA has lower image drift and higher CLIP alignment at all four selected capacity pairs, while OFT has lower COCO flow loss at three (Table 3b). These metrics distinguish preservation of generated behavior from preservation of denoising predictions at fixed noisy states.

Summary. Although task scores are often close, LoRA most consistently limits forgetting among all six methods at competitive performance. DoRA usually exceeds LoRA's mean task scores, while PiSSA often forgets more. Other methods' performance and forgetting depend on the setting. The

Table 3: FLUX-4B test/retention scores. DINO and general drift/CLIP are $\times 100$; COCO loss is unscaled. Panel (b) averages objects per seed. Shaded bold means mark the best method per budget/metric in (a), or capacity pair in (b). Budgets list rank/block sizes; HRA uses 16/32/62/124 vectors.

(a) Cat: six methods at four approximately matched budgets

Budget	Metric	LoRA	OFT	DoRA	PiSSA	MiLoRA	HRA
r4/b32	DINO ↑	$68.90_{\pm 0.87}$	$67.43_{\pm 4.54}$	$\mathbf{69.92}_{\pm 2.05}$	$68.29_{\pm 3.41}$	$66.51_{\pm 1.07}$	$68.40_{\pm 1.09}$
	Drift ↓	$\mathbf{14.23}_{\pm 1.01}$	$27.66_{\pm 4.85}$	$16.93_{\pm 3.89}$	$18.63_{\pm 4.99}$	$15.89_{\pm 3.13}$	$14.83_{\pm 4.11}$
	CLIP ↑	$27.00_{\pm 1.24}$	$24.88_{\pm 1.90}$	$26.81_{\pm 0.62}$	$\mathbf{27.80}_{\pm 1.04}$	$26.94_{\pm 1.39}$	$26.95_{\pm 2.05}$
r8/b64	DINO ↑	$68.48_{\pm 1.29}$	$68.90_{\pm 1.03}$	$68.77_{\pm 2.44}$	$69.41_{\pm 1.43}$	$65.87_{\pm 2.53}$	$\mathbf{70.10}_{\pm 2.95}$
	Drift ↓	$14.38_{\pm 0.68}$	$15.03_{\pm 3.55}$	$14.77_{\pm 0.57}$	$18.32_{\pm 2.22}$	$14.58_{\pm 4.24}$	$\mathbf{13.47}_{\pm 4.26}$
	CLIP ↑	$28.09_{\pm 1.48}$	$27.88_{\pm 2.43}$	$27.40_{\pm 2.09}$	$28.04_{\pm 0.55}$	$\mathbf{28.25}_{\pm 0.21}$	$26.63_{\pm 1.47}$
r16/b128	DINO ↑	$65.78_{\pm 1.41}$	$63.11_{\pm 1.24}$	$66.38_{\pm 2.31}$	$\mathbf{69.98}_{\pm 5.48}$	$65.14_{\pm 1.60}$	$50.89_{\pm 23.55}$
	Drift ↓	$\mathbf{12.87}_{\pm 2.45}$	$30.83_{\pm 4.67}$	$13.83_{\pm 2.73}$	$19.69_{\pm 2.20}$	$15.38_{\pm 2.56}$	$32.82_{\pm 12.12}$
	CLIP ↑	$27.99_{\pm 1.60}$	$23.23_{\pm 1.11}$	$\mathbf{28.27}_{\pm 0.66}$	$26.83_{\pm 0.78}$	$27.07_{\pm 0.13}$	$21.10_{\pm 5.13}$
r32/b256	DINO ↑	$67.63_{\pm 0.88}$	$59.54_{\pm 5.56}$	$\mathbf{70.97}_{\pm 3.86}$	$68.92_{\pm 3.47}$	$65.96_{\pm 1.88}$	$63.63_{\pm 5.99}$
	Drift ↓	$\mathbf{14.98}_{\pm 2.53}$	$43.69_{\pm 6.55}$	$15.98_{\pm 2.36}$	$26.14_{\pm 2.62}$	$15.44_{\pm 2.85}$	$32.82_{\pm 2.95}$
	CLIP ↑	$\mathbf{28.78}_{\pm 0.85}$	$16.51_{\pm 5.30}$	$27.34_{\pm 0.60}$	$26.79_{\pm 1.44}$	$27.69_{\pm 1.70}$	$23.27_{\pm 0.99}$

(b) 28 objects: four capacities, three LRs

Adapter	LR	Test DINO ↑	COCO flow ↓	General drift ↓	General CLIP ↑
LoRA $r = 4$	1×10^{-4}	$72.95_{\pm 0.42}$	$0.8877_{\pm 0.0035}$	$\mathbf{12.11}_{\pm 0.86}$	$\mathbf{25.63}_{\pm 0.52}$
LoRA $r = 8$	5×10^{-5}	$\mathbf{73.94}_{\pm 0.85}$	$0.8739_{\pm 0.0015}$	$\mathbf{11.38}_{\pm 0.60}$	$\mathbf{25.74}_{\pm 0.24}$
LoRA $r = 16$	3×10^{-5}	$71.88_{\pm 1.42}$	$\mathbf{0.8687}_{\pm 0.0017}$	$\mathbf{10.67}_{\pm 0.62}$	$\mathbf{25.98}_{\pm 0.43}$
LoRA $r = 32$	1×10^{-4}	$72.84_{\pm 1.18}$	$1.0164_{\pm 0.0058}$	$\mathbf{16.10}_{\pm 1.16}$	$\mathbf{24.62}_{\pm 0.93}$
OFT $b = 32$	1×10^{-4}	$\mathbf{73.44}_{\pm 2.27}$	$\mathbf{0.8588}_{\pm 0.0009}$	$17.29_{\pm 0.81}$	$23.99_{\pm 0.05}$
OFT $b = 64$	5×10^{-5}	$73.60_{\pm 1.95}$	$\mathbf{0.8596}_{\pm 0.0005}$	$19.21_{\pm 0.52}$	$23.41_{\pm 0.72}$
OFT $b = 128$	1×10^{-4}	$\mathbf{73.39}_{\pm 1.76}$	$0.9692_{\pm 0.0030}$	$43.59_{\pm 0.88}$	$12.24_{\pm 0.49}$
OFT $b = 256$	5×10^{-5}	$\mathbf{74.13}_{\pm 1.36}$	$\mathbf{0.9886}_{\pm 0.0038}$	$44.91_{\pm 0.10}$	$11.74_{\pm 0.41}$

orthogonal methods OFT and HRA offer no consistent performance–retention advantage over the four LoRA-family methods across the tested settings.

5 Spectrally Heterogeneous Adaptation, Structured Forgetting

We examine how dominant, intermediate, and trailing singular components contribute to adaptation and forgetting. Using pretrained singular values throughout, we selectively restore or retain adapted singular vectors to measure each group's contribution.

5.1 Restoring groups of spectral components

For a component group B, we restore its pretrained singular vectors and retain adapted vectors elsewhere:

$$\boldsymbol{W}_{\text{restore}(B)} = \boldsymbol{W}_{\text{keep}(B^c)} = \sum_{i \in B} s_{0,i} \boldsymbol{u}_{0,i} \boldsymbol{v}_{0,i}^\top + \sum_{i \notin B} s_{0,i} \boldsymbol{u}_{*,i} \boldsymbol{v}_{*,i}^\top. \tag{4}$$

Restoring dominant components consistently reduces forgetting. At the largest capacities, restoring dominant components gives the largest mean reduction in general-text NLL or general-image drift for every method (Figure 4). This ordering holds across language capacities (Appendix C). MiLoRA shares this pattern despite greater spectral drift concentration in trailing components.

	Qwen2.5-7B			Llama-3.1-8B			FLUX-4B		
	Δ accuracy (pp) ↑			Δ accuracy (pp) ↑			100 Δ DINO ↑		
LoRA	**−1.0**±0.1	0.2±0.2	−0.1±0.2	**−27.4**±1.1	−0.6±1.3	−0.6±1.1	**−12.4**±3.3	0.1±3.5	−0.3±2.9
OFT	1.0±0.5	**1.1**±0.9	0.6±0.7	**−19.0**±0.6	−0.8±1.1	−0.1±0.6	**3.1**±6.1	0.4±1.9	0.8±1.1
DoRA	−0.2±0.5	**0.7**±0.5	0.0±0.1	**−27.0**±1.6	−1.8±1.2	−1.2±1.4	**−22.9**±8.7	−0.6±1.3	−0.6±2.2
PiSSA	**−87.3**±1.1	0.4±0.5	0.2±0.7	**−71.2**±0.1	0.1±0.5	−0.1±0.2	**−33.1**±3.2	0.3±1.0	−0.1±0.3
MiLoRA	0.0±0.5	0.2±0.1	**−1.2**±0.4	**−5.8**±0.5	−1.0±1.1	−3.2±0.9	−2.5±2.2	**2.8**±0.2	−0.3±3.1
HRA	**−77.6**±0.6	−2.9±0.8	0.0±0.1	**−11.1**±0.8	−1.1±2.1	0.2±1.0	**−3.0**±5.1	0.4±1.5	−0.2±0.7
	Top	Middle	Tail	Top	Middle	Tail	Top	Middle	Tail
	100 Δ general NLL ↓			100 Δ general NLL ↓			100 Δ general drift ↓		
LoRA	**−0.3**±0.0	−0.1±0.0	−0.1±0.0	**−2.5**±0.3	1.6±0.4	0.1±0.1	**−6.7**±3.4	−1.8±0.5	−1.3±1.4
OFT	**−0.7**±0.0	−0.2±0.1	−0.2±0.0	**−3.4**±0.1	−1.5±0.0	−0.5±0.1	**−30.9**±1.7	−7.9±3.5	−0.4±0.5
DoRA	**−0.3**±0.1	−0.1±0.1	−0.1±0.0	**−2.3**±0.3	2.0±0.1	0.2±0.2	**−8.5**±1.8	−0.7±0.8	−1.2±0.3
PiSSA	**−1.1**±0.1	0.1±0.1	0.0±0.0	**−6.3**±0.1	0.8±0.1	0.1±0.0	**−24.9**±3.3	0.1±0.1	0.0±0.1
MiLoRA	**−0.4**±0.0	−0.3±0.0	−0.4±0.1	**−4.9**±0.3	−1.1±0.2	−2.7±0.3	**−4.5**±1.1	0.4±1.5	−3.1±3.5
HRA	**−0.3**±0.0	−0.1±0.0	0.0±0.0	**−8.9**±0.5	−3.1±0.1	−1.1±0.1	**−24.4**±3.0	−5.3±1.3	−2.4±1.3
	Top	Middle	Tail	Top	Middle	Tail	Top	Middle	Tail

Figure 4: Dominant-component restoration consistently reduces forgetting, with varying effects on task performance. Cells show mean ± SD changes from trained models at the largest capacities and selected LRs. Higher task scores and lower NLL/drift are better. FLUX DINO/drift are scaled by 100. Bold marks the largest absolute mean change across component groups per method.

The same restoration has different effects on task performance. At the largest language capacities, restoring dominant components largely preserves Qwen accuracy for LoRA, OFT, DoRA and MiLoRA, whereas PiSSA and HRA suffer large losses. The same edit reduces LoRA and DoRA accuracy by about 27 percentage points on Llama. PiSSA nearly loses all recorded accuracy on both models. In FLUX, PiSSA, LoRA and DoRA all incur substantial losses. OFT improves on average, with substantial seed variation. HRA and MiLoRA incur smaller mean losses. Thus, restoring dominant components consistently reduces forgetting, but its effect on task performance shows no consistent pattern across methods and models.

5.2 Which components retain task performance on their own?

Retaining only adapted dominant components preserves PiSSA's language accuracy within 1 percentage point of the full value-restored model $\boldsymbol{W}_R$ across every tested setting, whereas retaining only the remaining components does not. Other methods show no consistent sufficiency pattern across settings (Appendix C, Figure 12 and Table 19).

Summary. Task dependence varies across settings: PiSSA's adapted dominant components suffice across the tested settings, whereas other methods may require intermediate and trailing components. Restoring the dominant components consistently yields the largest mean reduction in NLL and drift, suggesting that forgetting is systematically associated with changes in dominant components.

6 Discussion

Across the tested settings, our comparison of selected PEFT methods finds no consistent performance–retention advantage from explicit geometric preservation. With method-specific learning-rate tuning, LoRA most consistently limits forgetting while maintaining competitive task performance. Restoring pretrained singular values largely preserves task performance, while restoring dominant components consistently yields the largest mean reduction in forgetting, although its effects on task performance vary. These findings highlight the importance of evaluating adaptation and retention jointly. In addition, shared learning rates and single-seed results can bias comparisons, underscoring the need for method-specific tuning and evaluation across multiple seeds. We hope this study supports more reliable evaluations, informs method selection, and guides PEFT designs that better balance adaptation with the retention of pretrained capabilities.

REPRODUCIBILITY STATEMENT

We provide our code in the supplementary material and detail the experimental protocols and intervention controls in the appendix.

STATEMENT OF AI USE

Generative AI assistants helped develop conceptual frameworks, formulate mathematical claims, provide proof arguments and assist in writing proofs, refine research questions and hypotheses, design checkpoint analyses and experiments, implement methods and analysis code, clean and reformat exported result data, and interpret results. They also helped review related work and citations, generate figures, and draft and revise the manuscript. Synthetic dataset generation, translation, and qualitative or thematic data analysis are not applicable to this work. The authors reviewed the AI-assisted work and checked the results. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. Program synthesis with large language models, 2021. URL https://arxiv.org/abs/2108.07732.

Dan Biderman, Jacob Portes, Jose Javier Gonzalez Ortiz, Mansheej Paul, Philip Greengard, Connor Jennings, Daniel King, Sam Havens, Vitaliy Chiley, Jonathan Frankle, Cody Blakeney, and John Patrick Cunningham. LoRA learns less and forgets less. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=aloEru2qCG. Featured Certification.

Massimo Bini, Karsten Roth, Zeynep Akata, and Anna Khoreva. ETHER: Efficient finetuning of large-scale models with hyperplane reflections. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 4007–4026. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/bini24a.html.

Black Forest Labs. FLUX.2-klein-base-4B model card. Official model release, 2026. URL https://huggingface.co/black-forest-labs/FLUX.2-klein-base-4B.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code, 2021. URL https://arxiv.org/abs/2107.03374.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The Llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Soufiane Hayou, Nikhil Ghosh, and Bin Yu. LoRA+: Efficient low rank adaptation of large models. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 17783–17806. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/hayou24a.html.

Xinyu He and Xianghui Cao. Understanding post-training structural changes in large language models, 2025. URL https://arxiv.org/abs/2509.17866.

Nicholas J. Higham. Computing the polar decomposition—with applications. *SIAM Journal on Scientific and Statistical Computing*, 7(4):1160–1174, October 1986. ISSN 2168-3417. doi: 10.1137/0907079. URL http://dx.doi.org/10.1137/0907079.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2790–2799. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/houlsby19a.html.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Yangyi Huang, Ruotian Peng, Zeju Qiu, Jiale Kang, Yandong Wen, Bernhard Schölkopf, and Weiyang Liu. PEFT-Arena: Understanding parameter-efficient finetuning from a stability-plasticity perspective. *arXiv preprint arXiv:2605.28819*, 2026.

Hangzhan Jin, Sitao Luan, Tianwei Ni, Sicheng Lyu, Guillaume Rabusseau, Reihaneh Rabbany, Doina Precup, and Mohammad Hamdaqa. RL fine-tuning heals OOD forgetting in SFT, 2026. URL https://arxiv.org/abs/2509.12235.

Damjan Kalajdzievski. A rank stabilization scaling factor for fine-tuning with LoRA, 2023. URL https://arxiv.org/abs/2312.03732.

Yu-Ang Lee, Ching-Yun Ko, Pin-Yu Chen, and Mi-Yen Yeh. Learning rate matters: Vanilla LoRA may suffice for LLM fine-tuning, 2026. URL https://arxiv.org/abs/2602.04998.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 4582–4597, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.353. URL https://aclanthology.org/2021.acl-long.353/.

Valerii Likhosherstov, Jared Davis, Krzysztof Choromanski, and Adrian Weller. CWY parametrization: a solution for parallelized optimization of orthogonal and Stiefel matrices. In Arindam Banerjee and Kenji Fukumizu (eds.), *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 55–63. PMLR, 13–15 Apr 2021. URL https://proceedings.mlr.press/v130/likhosherstov21a.html.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (eds.), *Computer Vision – ECCV 2014*, pp. 740–755, Cham, 2014. Springer International Publishing. ISBN 978-3-319-10602-1.

Vijay Lingam, Atula Tejaswi, Aditya Vavre, Aneesh Shetty, Gautham Krishna Gudur, Joydeep Ghosh, Alex Dimakis, Eunsol Choi, Aleksandar Bojchevski, and Sujay Sanghavi. SVFT: Parameter-efficient fine-tuning with singular vectors. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 41425–41446. Curran Associates, Inc., 2024. doi: 10.52202/079017-1311. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/48c368f105e8145b945227b73255635a-Paper-Conference.pdf.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and LINGMING ZHANG. Is your code generated by ChatGPT really correct? Rigorous evaluation of large language models for code generation. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 21558–21572. Curran Associates, Inc., 2023a. doi: 10.52202/

075280-0943. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/43e9d647ccd3e4b7b5baab53f0368686-Paper-Conference.pdf.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. DoRA: Weight-decomposed low-rank adaptation. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 32100–32121. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/liu24bn.html.

Weiyang Liu, Rongmei Lin, Zhen Liu, Lixin Liu, Zhiding Yu, Bo Dai, and Le Song. Learning towards minimum hyperspherical energy. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/177540c7bcb8db31697b601642eac8d4-Paper.pdf.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023b. URL https://openreview.net/forum?id=XVjTT1nw5z.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, Benjamin Bossan, and Marian Tietz. PEFT: State-of-the-art parameter-efficient fine-tuning methods. https://github.com/huggingface/peft, 2022.

Fanxu Meng, Zhaohui Wang, and Muhan Zhang. PiSSA: Principal singular values and singular vectors adaptation of large language models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 121038–121072. Curran Associates, Inc., 2024. doi: 10.52202/079017-3846. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/db36f4d603cc9e3a2a5e10b93e6428f2-Paper-Conference.pdf.

L. Mirsky. A trace inequality of John von Neumann. *Monatshefte für Mathematik*, 79(4):303–306, December 1975. ISSN 1436-5081. doi: 10.1007/bf01647331. URL http://dx.doi.org/10.1007/BF01647331.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=a68SUt6zFt. Featured Certification.

Guilherme Penedo. FineWiki, 2025. URL https://huggingface.co/datasets/HuggingFaceFW/finewiki. Source: Wikimedia Enterprise Snapshot API (https://api.enterprise.wikimedia.com/v2/snapshots). Text licensed under CC BY-SA 4.0 with attribution to Wikipedia contributors.

Kaustubh Ponkshe, Raghav Singhal, Eduard Gorbunov, Alexey Tumanov, Samuel Horvath, and Praneeth Vepakomma. Initialization using update approximation is a silver bullet for extremely efficient low-rank fine-tuning. *arXiv preprint arXiv:2411.19557*, 2024.

Zeju Qiu, Weiyang Liu, Haiwen Feng, Yuxuan Xue, Yao Feng, Zhen Liu, Dan Zhang, Adrian Weller, and Bernhard Schölkopf. Controlling text-to-image diffusion by orthogonal finetuning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 79320–79362. Curran Associates, Inc., 2023. doi: 10.52202/

075280-3472. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/faacb7a4827b4d51e201666b93ab5fa7-Paper-Conference.pdf.

Zeju Qiu, Simon Buchholz, Tim Xiao, Maximilian Dax, Bernhard Schölkopf, and Weiyang Liu. Reparameterized LLM training via orthogonal equivalence transformation. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 140775–140821. Curran Associates, Inc., 2025a. doi: 10.52202/085713-4704. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/cde43c5d698076ae17f3838a80f5a1bc-Paper-Conference.pdf.

Zeju Qiu, Weiyang Liu, Adrian Weller, and Bernhard Schölkopf. Orthogonal finetuning made scalable. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 31946–31963, Suzhou, China, November 2025b. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.1627. URL https://aclanthology.org/2025.emnlp-main.1627/.

Qwen Team. Qwen2.5: A party of foundation models, September 2024. URL https://qwenlm.github.io/blog/qwen2.5/.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. DreamBooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22500–22510, June 2023.

Robert Schreiber and Charles Van Loan. A storage-efficient WY representation for products of Householder transformations. *SIAM Journal on Scientific and Statistical Computing*, 10(1): 53–57, January 1989. ISSN 2168-3417. doi: 10.1137/0910005. URL http://dx.doi.org/10.1137/0910005.

John Schulman and Thinking Machines Lab. LoRA without regret. *Thinking Machines Lab: Connectionism*, September 2025. doi: 10.64434/tml.20250929. URL https://thinkingmachines.ai/blog/lora/.

Reece Shuttleworth, Jacob Andreas, Antonio Torralba, and Pratyusha Sharma. LoRA vs full fine-tuning: An illusion of equivalence. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 174627–174662. Curran Associates, Inc., 2025. doi: 10.52202/085713-5808. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/ff541950d1e885af90f523571564a401-Paper-Conference.pdf.

G. W. Stewart. Perturbation theory for the singular value decomposition. Technical Report UMIACS-TR-90-124, CS-TR-2539, University of Maryland, Institute for Advanced Computer Studies, 1990. URL https://api.drum.lib.umd.edu/server/api/core/bitstreams/a38cc6db-9bb7-4f3b-ae43-55b5827bb9d4/content.

Hanqing Wang, Yixia Li, Shuo Wang, Guanhua Chen, and Yun Chen. MiLoRA: Harnessing minor singular components for parameter-efficient LLM finetuning. In Luis Chiruzzo, Alan Ritter, and Lu Wang (eds.), *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 4823–4836, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-189-6. doi: 10.18653/v1/2025.naacl-long.248. URL https://aclanthology.org/2025.naacl-long.248/.

Shaowen Wang, Linxi Yu, and Jian Li. LoRA-GA: Low-rank adaptation with gradient approximation. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 54905–54931. Curran Associates, Inc., 2024. doi: 10.52202/079017-1741. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/62c4718cc334f6a0a62fb81c4a2095a1-Paper-Conference.pdf.

Yuchen Wang, Wenliang Zhong, Lichen Bai, Zikai Zhou, Shitong Shao, Bojun Cheng, Shuo Chen, Shuo Yang, and Zeke Xie. Exploring data-free LoRA transferability for video diffusion models. In *International Conference on Machine Learning*, 2026. URL https://icml.cc/virtual/2026/poster/64894.

Longhui Yu, Weisen JIANG, Han Shi, Jincheng YU, Zhengying Liu, Yu Zhang, James Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. MetaMath: Bootstrap your own mathematical questions for large language models. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 45040–45061, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/c400474e8a36d0812fdee52739288b12-Paper-Conference.pdf.

Shen Yuan, Haotian Liu, and Hongteng Xu. Bridging the gap between low-rank and orthogonal adaptation via Householder reflection adaptation. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 113484–113518. Curran Associates, Inc., 2024. doi: 10.52202/079017-3606. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/cdd0640218a27e9e2c0e52e324e25db0-Paper-Conference.pdf.

Yuanhe Zhang, Fanghui Liu, and Yudong Chen. LoRA-One: One-step full gradient could suffice for fine-tuning large language models, provably and efficiently. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 75513–75574. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/zhang25ax.html.

Tianyu Zheng, Ge Zhang, Tianhao Shen, Xueling Liu, Bill Yuchen Lin, Jie Fu, Wenhu Chen, and Xiang Yue. OpenCodeInterpreter: Integrating code generation with execution and refinement. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 12834–12859, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.762. URL https://aclanthology.org/2024.findings-acl.762/.

CONTENTS

A Experimental settings and metrics **16**

A.1 Language adaptation 16

A.2 Diffusion personalization 16

A.3 Metrics and uncertainty 17

B Complete selected comparisons and retention checks **18**

B.1 Six-method mathematics comparison and held-out evaluation 18

B.2 Separate HRA learning-rate extension 18

B.3 CodeFeedback selection and evaluation 20

B.4 Six-method cat comparison across learning rates and capacities 21

B.5 28-object comparison 23

B.6 Retention across all six methods 23

C Geometric measurements and language spectral interventions **25**

C.1 Checkpoints, controls and reference models 25

C.2 Singular-value displacement and restoration 28

C.3 Hyperspherical energy and pairwise geometry 30

C.4 Component effects, interactions and task performance 31

C.5 Consistency of the dominant-component response 33

C.6 Native OFT orthogonality and polar projection 34

D FLUX spectral interventions and numerical controls **34**

D.1 Complete six-method FLUX interventions 34

D.2 Task performance and forgetting across spectral components 36

E Geometry of additive updates **37**

E.1 Distance to a two-sided orthogonal transformation 37

E.2 First-order spectral and basis motion 38

A EXPERIMENTAL SETTINGS AND METRICS

Training and functional evaluations used a cluster of NVIDIA A100 GPUs. Each task used a single GPU, with at most 30 GPUs in use simultaneously. The post-hoc hyperspherical-energy measurements used NVIDIA L40 GPUs.

Table 4: Study overview. Each comparison uses a shared pretrained backbone. Divergent runs are reported with the corresponding results.

Backbone	Adaptation	Primary retention view
Qwen2.5-7B	MetaMath / six methods	General-text NLL
Llama-3.1-8B	MetaMath / six methods	General-text NLL
Qwen2.5-7B	CodeFeedback / six methods	General-text NLL
Llama-3.1-8B	CodeFeedback / six methods	General-text NLL
FLUX-4B	Cat / six methods	Image drift / CLIP
FLUX-4B	28 objects	Flow loss / drift / CLIP

A.1 LANGUAGE ADAPTATION

Qwen2.5-7B and Llama-3.1-8B use 20,000 MetaMath examples (Yu et al., 2024), microbatch size two, gradient accumulation over 16 steps, sequence length 768, 625 optimizer updates, cosine decay, 3% warmup, gradient clipping at one and zero weight decay. LoRA uses $\alpha = 2r$ and zero dropout. OFT uses unshared coordinate blocks and OFTv2's Cayley–Neumann parameterization with five terms (Qiu et al., 2025b). The seven projection types are query, key, value, output, up, gate and down, giving 196 adapted matrices in Qwen and 224 in Llama. LRs are $\{1, 3, 10, 30, 100\} \times 10^{-5}$ and seeds are 13, 37 and 73.

All six methods use AdamW (Loshchilov & Hutter, 2019). LoRA-family ranks are 7/14/28 for Qwen and 7/15/30 for Llama. OFT block sizes are 32/64/128, and HRA uses 16/32/64 Householder vectors. For each model/method/capacity, mean accuracy on the fixed 1,000-item GSM8K development panel selects one LR from fully completed configurations, with ties favoring the lower LR. The resulting 108 checkpoints are evaluated on the official 1,319-item test set. The diverged Qwen PiSSA configuration is excluded from LR selection and complete-configuration frontiers. Appendix B.1 reports all selected outcomes and the later HRA LR extension separately.

Coding uses the same two backbones, six methods and five-rate grid at one capacity per method. Its held-out response-NLL selection, completion-based testing and full settings are specified in Appendix B.3.

Standard and intervention retention use FineWiki-200 (Penedo, 2025), with contexts truncated to 768 tokens. The intervention panel evaluates GSM8K development accuracy on 1,000 items. General NLL and full-vocabulary KL use all 200 FineWiki documents, with token-pooled aggregation. General NLL scores valid next-token positions. All intervention comparisons use the same evaluation inputs and checkpoint-local references.

A.2 DIFFUSION PERSONALIZATION

FLUX.2-klein-base-4B uses AdamW, batch size two, weight decay 10^{-4}, gradient clipping at one, constant LR and 750 updates. The benchmark-cat set has 14 training, two validation and four test images. Four approximately matched parameter pairs are r4/b32, r8/b64, r16/b128 and r32/b256. LoRA uses $\alpha = r$. OFT uses unshared coordinate blocks with OFTv2's Cayley–Neumann parameterization (Qiu et al., 2025b). The LR grid is $\{0.5, 1, 3, 5, 10, 30, 50, 100\} \times 10^{-5}$, with seeds 0, 1 and 2. Generation uses resolution 512, 20 inference steps and guidance 3.5. For each method/capacity, we select a shared LR by the mean validation DINO at step 700, the last validation evaluation, then report test and retention scores at final step 750. The two validation and four test images are disjoint.

The complete six-method cat comparison uses LoRA/DoRA/PiSSA/MiLoRA ranks 4/8/16/32, OFT blocks 32/64/128/256, and HRA 16/32/62/124 Householder vectors. All 576 runs across six methods, four capacities and eight LRs are complete.

Target DINOv2 similarity averages the paired generated/reference test-image cosines over four examples and three generation repeats. General and class drift compare paired base/adapted generations using $(1 - \text{cosine})/2$, over five prompts in each bank. General CLIP is normalized text/image cosine averaged over five general prompts. Training and generation randomness contribute to the reported seed variation.

The 28-object study uses three training images, one validation image and one or two held-out test images per object. It crosses the same four capacity pairs with LRs $\{3, 5, 10\} \times 10^{-5}$ and seeds 0, 1 and 2, for 2,016 runs. For each method/capacity, we select one LR by mean step-700 validation DINO across all 28 objects and seeds, breaking ties toward the smaller LR. That LR is shared across objects and seeds. Test DINO, averaged over three generation repeats, and retention are measured at step 750. The retention bank uses 200 COCO image/caption pairs and four scheduler-index fractions, $\{0.125, 0.375, 0.625, 0.875\}$. Image drift and CLIP use the same four general prompts for every object, depicting an angry Mona Lisa, the Eiffel Tower at night, an astronaut riding a horse, and an avocado-shaped chair.

A.3 Metrics and uncertainty

Language likelihood. For document j and its valid next-token positions T_j, general-text NLL is

$$\mathrm{NLL}(\boldsymbol{\Theta}) = -\frac{1}{\sum_j |T_j|} \sum_j \sum_{t \in T_j} \log p_{\boldsymbol{\Theta}}(x_{j,t} | x_{j,<t}) \,.$$

All models use identical teacher-forced contexts and masks, excluding padding. The score weights tokens equally and measures fit to the general-text bank.

Diffusion outputs and predictive change. Write $\cos(\boldsymbol{a}, \boldsymbol{b}) = \frac{\boldsymbol{a}^\top \boldsymbol{b}}{\|\boldsymbol{a}\|_2 \|\boldsymbol{b}\|_2}$, with normalized DINOv2 features $\mathbf{f}(I)$ and CLIP image/text features $\mathbf{g}(I), \mathbf{h}(c)$. For generated image $I_j^{\boldsymbol{\Theta}}$, paired base image I_j^0, target reference image I_j^{ref} and prompt c_j, the reported averages are

$$\mathrm{DINO} = \frac{1}{J} \sum_j \cos\big(\mathbf{f}(I_j^{\boldsymbol{\Theta}}), \mathbf{f}(I_j^{\mathrm{ref}})\big) \,, \qquad D_{\mathrm{image}} = \frac{1}{J} \sum_j \frac{1 - \cos\big(\mathbf{f}(I_j^{\boldsymbol{\Theta}}), \mathbf{f}(I_j^0)\big)}{2},$$

$$\mathrm{CLIP} = \frac{1}{J} \sum_j \cos\big(\mathbf{g}(I_j^{\boldsymbol{\Theta}}), \mathbf{h}(c_j)\big) \,.$$

Each metric uses its own stated bank of J pairs. General and class drift use different prompt banks. Paired images share generation seeds. Repeats are averaged before training-seed summaries. The 28-object tables report absolute target DINO, COCO flow loss and CLIP alignment, together with base-relative image drift.

For the 28-object COCO flow loss, $\boldsymbol{y}_{j,k}$ contains the fixed noisy latent, scheduler time and conditioning for image/caption pair j in time bucket k. With d_j velocity coordinates, K buckets, flow target $\boldsymbol{u}_{j,k}$ and loss weight $w_{j,k}$,

$$\mathcal{L}_{\mathrm{flow}}(\boldsymbol{\Theta}) = \frac{1}{NK} \sum_{j,k} \frac{w_{j,k}}{d_j} \left\| \boldsymbol{v}_{\boldsymbol{\Theta}}(\boldsymbol{y}_{j,k}) - \boldsymbol{u}_{j,k} \right\|_2^2 .$$

The rectified-flow target is noise minus clean latent (Liu et al., 2023b). This fixed-state prediction loss is distinct from generated-image drift.

Uncertainty. For S training seeds with scores z_s, we report $\bar{z} = S^{-1} \sum_s z_s$ and sample SD $= \big(\sum_s (z_s - \bar{z})^2 / (S-1)\big)^{1/2}$. Perturbation and generation repetitions are averaged within a checkpoint first. For each 28-object method/capacity/LR setting, $z_s = 28^{-1} \sum_{o=1}^{28} z_{s,o}$. The SD describes variation across training seeds.

B COMPLETE SELECTED COMPARISONS AND RETENTION CHECKS

B.1 SIX-METHOD MATHEMATICS COMPARISON AND HELD-OUT EVALUATION

We extend the same Qwen2.5-7B/Llama-3.1-8B mathematics setup with four methods that vary adaptation structure. DoRA separates weight-vector magnitudes and directions, learning magnitudes alongside a low-rank directional update (Liu et al., 2024). PiSSA and MiLoRA initialize the trainable factors from leading and trailing singular components, respectively (Meng et al., 2024; Wang et al., 2025). In both cases, subtracting the initial low-rank component from the frozen base preserves the starting model. HRA learns a product of Householder reflections (Yuan et al., 2024). In an input-side convention,

$$\boldsymbol{W} = \boldsymbol{W}_0 \prod_{j=1}^{r_H} \boldsymbol{H}_j, \qquad \boldsymbol{H}_j = \boldsymbol{I} - 2\frac{\boldsymbol{v}_j \boldsymbol{v}_j^\top}{\boldsymbol{v}_j^\top \boldsymbol{v}_j}.$$

The implementation uses the compact WY representation (Schreiber & Van Loan, 1989; Likhosherstov et al., 2021), paired identity initialization and no Gram–Schmidt orthogonalization. Here r_H counts Householder vectors. DoRA/PiSSA/MiLoRA use training ranks 7/14/28 for Qwen and 7/15/30 for Llama, with $\alpha = 2r$. HRA uses 16/32/64 vectors. The capacity tiers approximately match trainable parameter counts. PiSSA/MiLoRA serialization expresses their final differences against the original base at rank $2r$. The tables report the actual training rank and trainable parameter count.

All methods use the same seven attention/MLP projection types, AdamW, batch 32, sequence length 768, 625 updates, cosine decay, warmup fraction 0.03, zero weight decay and gradient clipping at one. The original grid uses LRs $\{1, 3, 10, 30, 100\} \times 10^{-5}$ and seeds 13/37/73. It contains 540 training outcomes, comprising 180 LoRA/OFT and 360 added-method runs. Qwen PiSSA r7 at LR 10^{-3}, seed 73, diverged at step 55, leaving 539 final development evaluations. The group with two successful seeds is excluded from LR selection and complete-configuration frontiers.

Selection and test separation. One LR per model/method/capacity maximizes the exact total correct count across seeds on the fixed 1,000-problem, group-held-out GSM8K development panel. Ties choose the smaller LR. The resulting 36 configurations and 108 checkpoints are fixed before test evaluation, with one LR shared across seeds. Each checkpoint is evaluated on all 1,319 official GSM8K test problems. Prior test-set use for evaluation-harness development was separate from this selection. Generation uses the same native BF16 merged-adapter path, prompt and answer parser, greedy decoding, a 300-new-token cap, an 800-total-token cap and a 1,024-token engine context. The six-method intervention figures use development scores.

Retention reuses each run's recorded standard step-625 FineWiki-200 measurement, at length 768 and batch two. We report $\Delta\mathrm{NLL} = \mathrm{NLL}_{625} - \mathrm{NLL}_0$, using the corresponding pretrained reference. Smaller values indicate better general-text retention. We report sample SD across training seeds.

Performance and retention across settings. Qwen's small test-accuracy differences coexist with different retention costs. Its high-LR, small-rank LoRA selection is an outlier with greater forgetting than the other methods. LoRA's larger-rank selections have less forgetting. HRA's Qwen scores are competitive, and its original-grid Llama scores improve with capacity alongside increasing retention cost. These results support the main finding that competitive task scores can accompany different forgetting costs, with outcomes depending on the selected LR and capacity. Appendix B.6 summarizes the trends across tasks.

B.2 SEPARATE HRA LEARNING-RATE EXTENSION

All three original Llama HRA winners occur at the largest LR, 10^{-3}. A follow-up therefore adds 3×10^{-3} and 10^{-2} at each capacity and seed. The extension was proposed after the original test report and is reported separately. Of 18 training runs, 15 complete both development and official-test evaluations. All three r64 runs at 10^{-2} diverge. Selection over the seven-LR development grid chooses 3×10^{-3} at r16/r32 and retains 10^{-3} at r64. The main comparison retains the original five-LR selections.

Table 5: Qwen2.5-7B results for all development-selected configurations from the original five-LR grid. Parameters are in millions. Ranks are training ranks (HRA counts Householder vectors). Accuracy is in percent, and ΔNLL is the recorded FineWiki-200 NLL minus its matching pretrained value. LRs were selected using development accuracy only.

Adapter	Params (M)	LR	Dev ↑	Test ↑	ΔNLL ↓
LoRA r7	17.66	1×10^{-3}	$90.73_{\pm 0.35}$	$83.07_{\pm 0.54}$	$0.0237_{\pm 0.0012}$
LoRA r14	35.32	1×10^{-5}	$91.40_{\pm 0.44}$	$83.22_{\pm 0.16}$	$0.0020_{\pm 0.0001}$
LoRA r28	70.65	3×10^{-5}	$92.30_{\pm 0.46}$	$83.65_{\pm 0.42}$	$0.0038_{\pm 0.0002}$
OFT b32	17.55	3×10^{-4}	$91.37_{\pm 0.25}$	$83.35_{\pm 0.36}$	$0.0105_{\pm 0.0009}$
OFT b64	35.68	3×10^{-4}	$89.97_{\pm 0.38}$	$82.92_{\pm 0.46}$	$0.0182_{\pm 0.0008}$
OFT b128	71.92	1×10^{-4}	$90.90_{\pm 0.69}$	$83.50_{\pm 0.27}$	$0.0088_{\pm 0.0007}$
DoRA r7	19.05	3×10^{-4}	$92.23_{\pm 0.40}$	$83.27_{\pm 0.04}$	$0.0055_{\pm 0.0007}$
DoRA r14	36.71	3×10^{-4}	$92.77_{\pm 0.58}$	$82.64_{\pm 0.59}$	$0.0054_{\pm 0.0005}$
DoRA r28	72.04	1×10^{-4}	$92.63_{\pm 0.06}$	$83.12_{\pm 0.53}$	$0.0048_{\pm 0.0006}$
PiSSA r7	17.66	3×10^{-5}	$92.53_{\pm 0.32}$	$83.85_{\pm 0.47}$	$0.0108_{\pm 0.0009}$
PiSSA r14	35.32	3×10^{-5}	$92.40_{\pm 0.20}$	$83.67_{\pm 0.23}$	$0.0110_{\pm 0.0012}$
PiSSA r28	70.65	3×10^{-5}	$91.73_{\pm 0.83}$	$83.17_{\pm 0.20}$	$0.0108_{\pm 0.0010}$
MiLoRA r7	17.66	1×10^{-5}	$92.60_{\pm 0.17}$	$83.35_{\pm 0.24}$	$0.0045_{\pm 0.0002}$
MiLoRA r14	35.32	1×10^{-5}	$92.60_{\pm 0.10}$	$83.67_{\pm 0.46}$	$0.0050_{\pm 0.0003}$
MiLoRA r28	70.65	3×10^{-5}	$92.60_{\pm 0.30}$	$83.73_{\pm 0.12}$	$0.0083_{\pm 0.0010}$
HRA r16	18.12	1×10^{-4}	$91.97_{\pm 0.31}$	$83.60_{\pm 0.12}$	$0.0054_{\pm 0.0007}$
HRA r32	36.24	1×10^{-4}	$92.07_{\pm 0.32}$	$83.65_{\pm 0.49}$	$0.0073_{\pm 0.0001}$
HRA r64	72.48	1×10^{-5}	$92.00_{\pm 0.26}$	$83.22_{\pm 0.16}$	$0.0038_{\pm 0.0002}$

Table 6: Llama-3.1-8B results for all development-selected configurations from the original five-LR grid. Parameters are in millions. Ranks are training ranks (HRA counts Householder vectors). Accuracy is in percent, and ΔNLL is the recorded FineWiki-200 NLL minus its matching pretrained value. LRs were selected using development accuracy only.

Adapter	Params (M)	LR	Dev ↑	Test ↑	ΔNLL ↓
LoRA r7	18.35	3×10^{-4}	$71.17_{\pm 0.29}$	$70.84_{\pm 1.09}$	$0.0285_{\pm 0.0018}$
LoRA r15	39.32	3×10^{-4}	$72.30_{\pm 1.04}$	$71.65_{\pm 0.35}$	$0.0372_{\pm 0.0023}$
LoRA r30	78.64	3×10^{-4}	$71.93_{\pm 1.12}$	$71.90_{\pm 0.12}$	$0.0486_{\pm 0.0024}$
OFT b32	19.30	3×10^{-4}	$72.10_{\pm 1.31}$	$71.42_{\pm 1.45}$	$0.0371_{\pm 0.0015}$
OFT b64	39.22	3×10^{-4}	$73.10_{\pm 0.26}$	$71.34_{\pm 0.13}$	$0.0740_{\pm 0.0017}$
OFT b128	79.07	1×10^{-4}	$72.57_{\pm 0.71}$	$71.75_{\pm 1.18}$	$0.0366_{\pm 0.0011}$
DoRA r7	19.73	3×10^{-4}	$71.50_{\pm 0.78}$	$69.95_{\pm 0.12}$	$0.0314_{\pm 0.0013}$
DoRA r15	40.70	3×10^{-4}	$72.97_{\pm 1.01}$	$70.91_{\pm 0.44}$	$0.0394_{\pm 0.0014}$
DoRA r30	80.02	3×10^{-4}	$73.27_{\pm 2.18}$	$71.97_{\pm 1.16}$	$0.0512_{\pm 0.0033}$
PiSSA r7	18.35	1×10^{-4}	$70.57_{\pm 0.21}$	$70.33_{\pm 1.49}$	$0.1084_{\pm 0.0025}$
PiSSA r15	39.32	1×10^{-4}	$70.87_{\pm 0.61}$	$69.98_{\pm 0.27}$	$0.1521_{\pm 0.0030}$
PiSSA r30	78.64	3×10^{-5}	$71.13_{\pm 0.68}$	$70.26_{\pm 0.98}$	$0.0650_{\pm 0.0014}$
MiLoRA r7	18.35	3×10^{-4}	$71.53_{\pm 0.25}$	$70.43_{\pm 0.42}$	$0.0486_{\pm 0.0017}$
MiLoRA r15	39.32	3×10^{-4}	$72.03_{\pm 0.86}$	$71.06_{\pm 0.61}$	$0.0685_{\pm 0.0021}$
MiLoRA r30	78.64	3×10^{-4}	$72.60_{\pm 0.52}$	$70.68_{\pm 0.23}$	$0.0942_{\pm 0.0024}$
HRA r16	19.92	1×10^{-3}	$69.03_{\pm 0.61}$	$67.63_{\pm 0.42}$	$0.0340_{\pm 0.0017}$
HRA r32	39.85	1×10^{-3}	$70.17_{\pm 1.10}$	$70.31_{\pm 1.37}$	$0.0521_{\pm 0.0022}$
HRA r64	79.69	1×10^{-3}	$71.47_{\pm 1.01}$	$71.04_{\pm 0.20}$	$0.0988_{\pm 0.0031}$

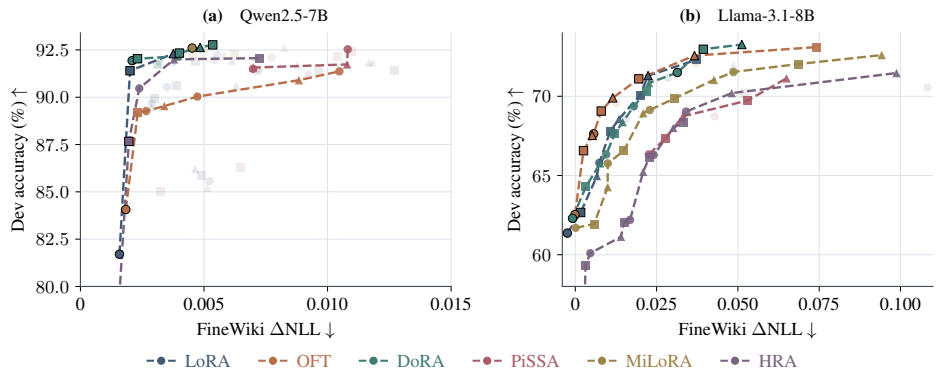

Figure 5: Six-method mathematics trade-offs, focused on the frontier regions. Each point is a complete capacity/LR configuration, with shapes denoting increasing capacity. Dashed lines and solid points mark within-method frontiers, and black borders mark the pooled frontier. Dominated points are faint. Frontiers use all 89 Qwen and 90 Llama configurations before axis clipping. The later HRA extension and diverged PiSSA configuration are excluded.

Table 7: Llama-3.1-8B HRA development-selected outcomes after the separate seven-LR follow-up. The r64 selection is unchanged from the original grid. n is the successful seed count. LR selection uses development scores.

Adapter	LR	n	Dev (%) ↑	Test (%) ↑	ΔNLL ↓
HRA r16	3×10^{-3}	3	$70.47_{\pm 0.42}$	$70.15_{\pm 0.27}$	$0.0553_{\pm 0.0028}$
HRA r32	3×10^{-3}	3	$70.87_{\pm 0.90}$	$71.37_{\pm 1.20}$	$0.1132_{\pm 0.0060}$
HRA r64	1×10^{-3}	3	$71.47_{\pm 1.01}$	$71.04_{\pm 0.20}$	$0.0988_{\pm 0.0031}$

The development-selected 3×10^{-3} extension raises mean test accuracy at both r16 and r32, while increasing their retention costs. At r64, 3×10^{-3} reduces accuracy and increases NLL, so development selection retains the original LR. HRA therefore illustrates the setting-dependent trade-offs in Section 4, with the useful LR range changing across capacities.

B.3 CodeFeedback selection and evaluation

We use the Python subset of CodeFeedback (Zheng et al., 2024), distributed for PiSSA (Meng et al., 2024). After exact prompt deduplication, a fixed split contains 20,000 training and 2,000 development examples. Exact prompt matching against the test sets removes no additional examples. All methods use one epoch, effective batch size 32, maximum training length 1,024, a cosine schedule with 3% warmup, and the same attention/MLP projections. LoRA, DoRA, PiSSA and MiLoRA train at rank 7. OFT uses block size 32 and HRA uses 16 Householder vectors. PiSSA/MiLoRA's converted rank-14 difference adapters retain the same training capacity.

For each method/backbone, the mean development response NLL across seeds 13, 37 and 73 selects one of $\{10^{-5}, 3 \times 10^{-5}, 10^{-4}, 3 \times 10^{-4}, 10^{-3}\}$, with ties favoring the lower LR. NLL pools valid response-token losses, with a 4,096-token evaluation limit. Thus the 180 development runs yield 36 selected test checkpoints, each using the best LR within the tested grid.

Test evaluation uses EvalPlus base completion prompts, greedy decoding, and one completion per problem, capped at 1,024 new tokens. We evaluate 164 HumanEval (Chen et al., 2021) and 378 MBPP (Austin et al., 2021) problems. Original and strengthened tests grade the same completion. Table 8 reports both versions. The base model has one fixed evaluation, so its row has no training-seed SD.

General retention uses the fixed FineWiki-200 corpus, a 768-token limit, and valid-token-pooled NLL with float32 logits. Each method's forgetting is its NLL minus the corresponding unadapted model's NLL.

Table 8: CodeFeedback results for all original and strengthened-test pass@1 scores (%). A single greedy completion is graded under both test suites. LR minimizes mean held-out response NLL. Only selected checkpoints have test evaluations. The Base row has no training-seed SD.

Model	Method	Selected LR	HumanEval	HE+	MBPP	MBPP+
Qwen7B	Base	—	58.54	51.22	76.72	64.02
	LoRA	1×10^{-4}	$59.96_{\pm 0.70}$	$52.24_{\pm 0.35}$	$76.90_{\pm 0.40}$	$64.29_{\pm 0.53}$
	OFT	1×10^{-4}	$58.94_{\pm 0.70}$	$51.83_{\pm 0.61}$	$77.43_{\pm 0.55}$	$64.55_{\pm 0.53}$
	DoRA	1×10^{-4}	$59.96_{\pm 0.93}$	$52.64_{\pm 0.70}$	$76.63_{\pm 1.00}$	$64.37_{\pm 0.61}$
	PiSSA	1×10^{-5}	$59.15_{\pm 1.06}$	$52.44_{\pm 1.61}$	$76.63_{\pm 0.40}$	$63.93_{\pm 0.61}$
	MiLoRA	1×10^{-4}	$58.94_{\pm 0.35}$	$51.02_{\pm 0.93}$	$76.63_{\pm 0.55}$	$64.02_{\pm 0.53}$
	HRA	1×10^{-3}	$58.94_{\pm 2.14}$	$51.83_{\pm 1.83}$	$77.25_{\pm 0.70}$	$64.46_{\pm 0.85}$
Llama8B	Base	—	38.41	31.71	61.38	50.79
	LoRA	3×10^{-4}	$41.46_{\pm 1.06}$	$36.99_{\pm 0.93}$	$65.87_{\pm 0.46}$	$54.76_{\pm 0.46}$
	OFT	3×10^{-4}	$43.09_{\pm 1.53}$	$39.02_{\pm 1.06}$	$65.61_{\pm 1.61}$	$54.41_{\pm 1.88}$
	DoRA	3×10^{-4}	$42.07_{\pm 0.61}$	$37.80_{\pm 1.06}$	$67.55_{\pm 1.10}$	$56.08_{\pm 0.00}$
	PiSSA	1×10^{-4}	$45.53_{\pm 0.35}$	$40.45_{\pm 0.93}$	$66.14_{\pm 0.26}$	$53.88_{\pm 0.40}$
	MiLoRA	3×10^{-4}	$42.28_{\pm 0.93}$	$37.80_{\pm 0.00}$	$65.61_{\pm 0.26}$	$54.06_{\pm 0.67}$
	HRA	1×10^{-3}	$39.84_{\pm 1.27}$	$35.57_{\pm 0.70}$	$66.75_{\pm 0.76}$	$55.64_{\pm 1.19}$

Task gains and retention costs. All six methods improve Llama's HumanEval+ and MBPP+ means over the base model, with rankings that differ across benchmarks and retention. PiSSA leads on HumanEval+, alongside higher retention costs. Together, these results extend the distinction between task performance and forgetting to coding, with the size of the task gain depending on the backbone.

B.4 Six-method cat comparison across learning rates and capacities

The complete cat study crosses six methods, four approximately matched parameter budgets, eight LRs and seeds 0, 1 and 2, giving 576 successful runs and 192 configuration means. All methods share the cat split (14 training, two validation and four test images), adapted modules, training exposure and evaluation steps. LoRA, DoRA, PiSSA and MiLoRA use ranks 4/8/16/32, OFT uses block sizes 32/64/128/256, and HRA uses 16/32/62/124 Householder vectors. PiSSA/MiLoRA ranks refer to training, with doubled ranks used to serialize their differences from the base model.

The LR grid is $\{5 \times 10^{-6}, 10^{-5}, 3 \times 10^{-5}, 5 \times 10^{-5}, 10^{-4}, 3 \times 10^{-4}, 5 \times 10^{-4}, 10^{-3}\}$. Within each method/capacity, mean validation DINO at step 700 selects one LR shared across seeds, with ties favoring smaller LR. Test and retention use step 750. Table 9 gives all 24 selections and parameter counts, with sample seed SD.

Table 9: Complete validation-selected FLUX cat results. Budgets are approximately matched, with HRA using 16/32/62/124 vectors and DoRA adding magnitude parameters. Parameters are in millions. Validation DINO at step 700 selects LR separately for each method/capacity. Step 750 supplies test DINO, general drift and general CLIP. Scores are $\times 100$.

Method	Capacity	Params	LR	Validation	Test DINO	Gen. drift	Gen. CLIP
LoRA	r4	4.79	3×10^{-4}	$73.48_{\pm 2.90}$	$68.90_{\pm 0.87}$	$14.23_{\pm 1.01}$	$27.00_{\pm 1.24}$
OFT	b32	4.76	3×10^{-4}	$74.58_{\pm 2.56}$	$67.43_{\pm 4.54}$	$27.66_{\pm 4.85}$	$24.88_{\pm 1.90}$
DoRA	r4	5.68	5×10^{-4}	$71.87_{\pm 2.31}$	$69.92_{\pm 2.05}$	$16.93_{\pm 3.89}$	$26.81_{\pm 0.62}$
PiSSA	r4	4.79	1×10^{-4}	$73.93_{\pm 1.62}$	$68.29_{\pm 3.41}$	$18.63_{\pm 4.99}$	$27.80_{\pm 1.04}$
MiLoRA	r4	4.79	1×10^{-4}	$71.79_{\pm 5.90}$	$66.51_{\pm 1.07}$	$15.89_{\pm 3.13}$	$26.94_{\pm 1.39}$
HRA	h16	4.92	5×10^{-4}	$72.99_{\pm 1.85}$	$68.40_{\pm 1.09}$	$14.83_{\pm 4.11}$	$26.95_{\pm 2.05}$
LoRA	r8	9.58	1×10^{-4}	$71.65_{\pm 1.88}$	$68.48_{\pm 1.29}$	$14.38_{\pm 0.68}$	$28.09_{\pm 1.48}$
OFT	b64	9.68	5×10^{-5}	$71.17_{\pm 0.91}$	$68.90_{\pm 1.03}$	$15.03_{\pm 3.55}$	$27.88_{\pm 2.43}$
DoRA	r8	10.48	3×10^{-4}	$70.42_{\pm 2.84}$	$68.77_{\pm 2.44}$	$14.77_{\pm 0.57}$	$27.40_{\pm 2.09}$
PiSSA	r8	9.58	5×10^{-5}	$72.23_{\pm 3.77}$	$69.41_{\pm 1.43}$	$18.32_{\pm 2.22}$	$28.04_{\pm 0.55}$
MiLoRA	r8	9.58	5×10^{-5}	$69.65_{\pm 2.02}$	$65.87_{\pm 2.53}$	$14.58_{\pm 4.24}$	$28.25_{\pm 0.21}$
HRA	h32	9.83	3×10^{-4}	$73.11_{\pm 0.95}$	$70.10_{\pm 2.95}$	$13.47_{\pm 4.26}$	$26.63_{\pm 1.47}$
LoRA	r16	19.17	1×10^{-4}	$74.88_{\pm 1.44}$	$65.78_{\pm 1.41}$	$12.87_{\pm 2.45}$	$27.99_{\pm 1.60}$
OFT	b128	19.51	1×10^{-4}	$70.87_{\pm 7.10}$	$63.11_{\pm 1.24}$	$30.83_{\pm 4.67}$	$23.23_{\pm 1.11}$
DoRA	r16	20.06	5×10^{-5}	$73.73_{\pm 1.29}$	$66.38_{\pm 2.31}$	$13.83_{\pm 2.73}$	$28.27_{\pm 0.66}$
PiSSA	r16	19.17	5×10^{-5}	$71.70_{\pm 3.16}$	$69.98_{\pm 5.48}$	$19.69_{\pm 2.20}$	$26.83_{\pm 0.78}$
MiLoRA	r16	19.17	1×10^{-4}	$73.96_{\pm 2.00}$	$65.14_{\pm 1.60}$	$15.38_{\pm 2.56}$	$27.07_{\pm 0.13}$
HRA	h62	19.05	5×10^{-4}	$70.95_{\pm 3.50}$	$50.89_{\pm 23.55}$	$32.82_{\pm 12.12}$	$21.10_{\pm 5.13}$
LoRA	r32	38.34	5×10^{-5}	$72.46_{\pm 5.94}$	$67.63_{\pm 0.88}$	$14.98_{\pm 2.53}$	$28.78_{\pm 0.85}$
OFT	b256	39.17	1×10^{-4}	$74.71_{\pm 4.21}$	$59.54_{\pm 5.56}$	$43.69_{\pm 6.55}$	$16.51_{\pm 5.30}$
DoRA	r32	39.23	1×10^{-4}	$72.33_{\pm 2.88}$	$70.97_{\pm 3.86}$	$15.98_{\pm 2.36}$	$27.34_{\pm 0.60}$
PiSSA	r32	38.34	5×10^{-5}	$71.87_{\pm 2.13}$	$68.92_{\pm 3.47}$	$26.14_{\pm 2.62}$	$26.79_{\pm 1.44}$
MiLoRA	r32	38.34	1×10^{-4}	$73.32_{\pm 2.90}$	$65.96_{\pm 1.88}$	$15.44_{\pm 2.85}$	$27.69_{\pm 1.70}$
HRA	h124	38.09	3×10^{-4}	$72.51_{\pm 4.16}$	$63.63_{\pm 5.99}$	$32.82_{\pm 2.95}$	$23.27_{\pm 0.99}$

Low selected drift and capacity-dependent fidelity. LoRA has the lowest selected mean general drift at 3 of 4 budgets and ranks among the lowest two at all 4. PiSSA has greater drift than LoRA in all 12 paired seeds. PiSSA has the highest selected drift at the small budget, OFT at the smallest and largest budgets, and HRA at the medium budget. DoRA exceeds LoRA's mean test DINO at every selected budget and leads at the smallest and largest. MiLoRA has lower mean test DINO than LoRA at all four budgets, with a mean CLIP advantage only at the small budget. These results support LoRA's recurring retention advantage and DoRA's higher mean task scores, alongside setting-dependent outcomes for the other methods.

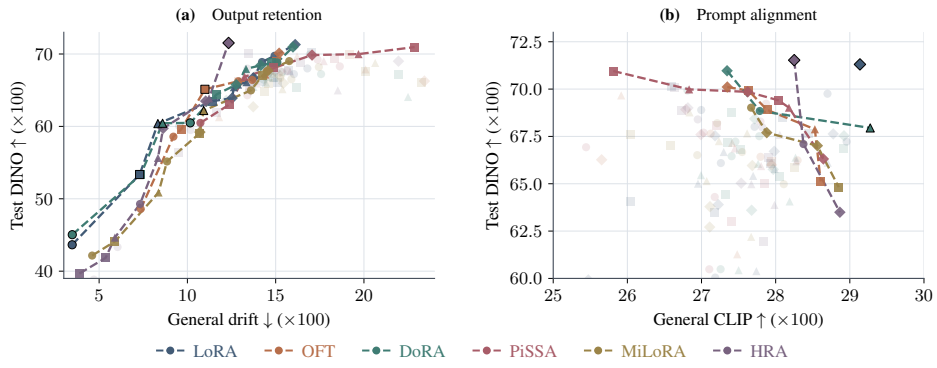

Figure 6: Six-method FLUX cat test trade-offs, focused on the frontier regions. Shapes denote increasing capacity. Dashed lines and solid points mark within-method frontiers, and black borders mark the pooled frontier. Dominated points are faint. Frontiers use all 192 configurations before axis clipping, with separate y ranges for each panel. Test DINO describes final-step outcomes, while validation DINO selects LR.

Retention metrics. General drift and CLIP use five unrelated prompts, while class drift uses five cat prompts. All six methods have these measurements. Drift measures preservation of generated outputs, and CLIP measures prompt alignment. Appendix D.1 examines the role of spectral components through six-method interventions at the largest budget.

B.5 28-OBJECT COMPARISON

Table 3b reports every validation-selected capacity pair. Each score first averages the 28 objects within a seed, then reports the mean and sample SD across seeds. The shared-LR selection and evaluation banks are specified in Appendix A.

Retention across objects and metrics. At the validation-selected settings, OFT has lower COCO flow loss at three of four capacities. LoRA has lower general-image drift and higher prompt alignment at every capacity. The same distinction appears at a shared LR of 5×10^{-5}, where OFT b64 has lower COCO loss than LoRA r8, but higher image drift and lower CLIP. Flow loss evaluates velocity predictions on fixed noisy latents, whereas drift and CLIP evaluate the final generated images. The object comparison supports LoRA's retention advantage in generated outputs and shows why the metrics capture different aspects of preservation.

B.6 RETENTION ACROSS ALL SIX METHODS

We compare LoRA's forgetting with *each of the other five methods* across twelve settings, comprising six mathematics model/capacity pairs, two coding backbones and four cat budgets. Each method's LR is chosen by task performance on the development set. We compare forgetting within each setting, so language NLL and image drift remain on their own scales.

LoRA most consistently limits forgetting. LoRA has the lowest mean forgetting in 9/12 settings. OFT, MiLoRA and HRA each have the lowest forgetting in one setting, and DoRA and PiSSA in none (Table 10). LoRA also has the lowest average rank, 1.58, followed by DoRA at 2.75. The three exceptions are small-capacity Qwen, where LoRA's outlying high-LR selection forgets most, large-capacity Llama, where OFT has the lowest forgetting, and cat rank 8, where HRA has the lowest forgetting.

Consistency across selection and weighting choices. First, we choose LR using two seeds and measure forgetting on the remaining seed, repeating for every seed. The original grids, eligibility rules and smaller-LR tie rule remain unchanged. LoRA still has the lowest mean forgetting in 8/12 settings. Each other method does so in at most one. Its average rank remains lowest (2.08, versus 2.50 for DoRA). Second, we give each model/task group equal importance, preventing groups with

Table 10: Forgetting across all six methods. The lowest-forgetting columns count settings in which a method has the lowest mean forgetting. A rank of 1 means least forgetting, and lower average rank is better. The right columns choose LR using the other two seeds, evaluate the remaining seed, then average those evaluations. LoRA has the lowest forgetting in the most settings and the lowest average rank under both selections.

	Original selection		LR chosen on other seeds	
Method	Lowest forgetting	Average rank	Lowest forgetting	Average rank
LoRA	**9/12**	**1.58**	**8/12**	**2.08**
OFT	1/12	4.58	1/12	4.33
DoRA	0/12	2.75	1/12	2.50
PiSSA	0/12	5.25	0/12	5.42
MiLoRA	1/12	3.42	1/12	3.25
HRA	1/12	3.42	1/12	3.42

more capacities from dominating the summary. Third, we remove each backbone or individual setting in turn. Finally, we compare results using one language seed and one image seed at a time, covering all nine combinations rather than matching unrelated seed numbers.

Across all these checks, LoRA has the lowest forgetting in more settings than any other method (Table 11). This supports the claim that LoRA most consistently limits forgetting across the tested methods. After LR selection on other seeds, DoRA has a better average rank in three seed combinations and when FLUX is removed.

Table 11: Robustness of the retention comparison. Frequency columns count versions in which LoRA has the lowest forgetting in more settings than any other method. Average-rank columns count versions in which LoRA has the lowest average rank. Equal weighting averages within the five model/task groups before averaging groups. The nine seed combinations cross all language seeds with all image seeds.

	Original selection		LR chosen on other seeds	
Check	Frequency	Average rank	Frequency	Average rank
Equal model/task weights	1/1	1/1	1/1	1/1
Remove one backbone	3/3	3/3	3/3	2/3
Remove one setting	12/12	12/12	12/12	12/12
Use individual seeds	9/9	9/9	9/9	6/9

Task scores and PiSSA's retention costs. DoRA has higher mean task scores than LoRA in 10/14 comparisons, covering two of six mathematics settings, all four coding comparisons and all four cat budgets. This supports the reported trend in mean task scores. Its coding and cat gains accompany additional forgetting.

With backbone, capacity and LR fixed across all six methods, PiSSA forgets most in all 29/29 complete mathematics comparisons and 10/10 coding comparisons, including all 87 and 30 seed comparisons. In cat, it ranks highest in 21/32 comparisons and among the highest two in all 32. HRA ranks lowest in 25/32. These matched-LR comparisons reinforce the finding that PiSSA often incurs greater retention costs and show that HRA can retain well in specific settings.

LoRA versus OFT across objects. After averaging seeds within each object, LoRA has lower drift than OFT for 23, 27, 28 and 28 of the 28 objects across increasing budgets. This extends LoRA's image-drift advantage beyond cat personalization.

C GEOMETRIC MEASUREMENTS AND LANGUAGE SPECTRAL INTERVENTIONS

C.1 CHECKPOINTS, CONTROLS AND REFERENCE MODELS

The intervention study covers 108 checkpoints selected by mean development accuracy within each model/method/capacity, with 22 interventions or controls per checkpoint. Llama HRA uses the extended-grid selections in Appendix B.2. All edits use the same GSM8K development and general-text banks. The tables below extend the largest-capacity main figures to all capacities and selected LRs.

Task and NLL changes use the trained model as reference. Value-restoration controls compare against reconstruction, while component sufficiency and factorial analyses use full value restoration. Forgetting recovery is the fraction of the trained model's NLL increase recovered by an edit, computed within each checkpoint before seed averaging.

Edits jointly cover all adapted matrices. SVD computations use FP32, followed by BF16 storage and evaluation. The dominant, intermediate and trailing component groups divide the pretrained spectrum into approximate thirds, with boundaries adjusted to nearby spectral gaps. Plots label these groups top, middle and tail.

For the collection $\mathcal{M}$ of adapted matrices, the normalized displacements are

$$d_W^2 = \frac{\sum_{M\in\mathcal{M}} \|\boldsymbol{W}_{*,M} - \boldsymbol{W}_{0,M}\|_{\mathrm{F}}^2}{\sum_{M\in\mathcal{M}} \|\boldsymbol{W}_{0,M}\|_{\mathrm{F}}^2}, \qquad d_S^2 = \frac{\sum_{M\in\mathcal{M}} \|\boldsymbol{s}_{*,M} - \boldsymbol{s}_{0,M}\|_2^2}{\sum_{M\in\mathcal{M}} \|\boldsymbol{s}_{0,M}\|_2^2}.$$

Each checkpoint pools all 196 Qwen or 224 Llama adapted matrices (80 for the FLUX results in Appendix D.1), weighting their contributions through the squared norms.

Table 12: Qwen2.5-7B spectral displacement at every selected capacity and LR.

Method	Capacity	LR	$100d_W$	$100d_S$	$100d_S/d_W$
LoRA	7	0.001	$8.154_{\pm 0.010}$	$1.67437_{\pm 0.01900}$	$20.53_{\pm 0.25}$
LoRA	14	$1e-05$	$0.151_{\pm 0.000}$	$0.00161_{\pm 0.00003}$	$1.07_{\pm 0.02}$
LoRA	28	$3e-05$	$0.307_{\pm 0.001}$	$0.00321_{\pm 0.00014}$	$1.05_{\pm 0.04}$
OFT	32	0.0003	$4.213_{\pm 0.006}$	$0.00250_{\pm 0.00062}$	$0.06_{\pm 0.01}$
OFT	64	0.0003	$6.368_{\pm 0.022}$	$0.00340_{\pm 0.00085}$	$0.05_{\pm 0.01}$
OFT	128	0.0001	$3.090_{\pm 0.009}$	$0.00183_{\pm 0.00011}$	$0.06_{\pm 0.00}$
DoRA	7	0.0003	$1.373_{\pm 0.010}$	$0.02003_{\pm 0.00212}$	$1.46_{\pm 0.15}$
DoRA	14	0.0003	$1.863_{\pm 0.016}$	$0.02374_{\pm 0.00088}$	$1.27_{\pm 0.04}$
DoRA	28	0.0001	$0.813_{\pm 0.004}$	$0.01036_{\pm 0.00023}$	$1.27_{\pm 0.03}$
PiSSA	7	$3e-05$	$0.522_{\pm 0.000}$	$0.02039_{\pm 0.00132}$	$3.91_{\pm 0.25}$
PiSSA	14	$3e-05$	$0.643_{\pm 0.001}$	$0.02227_{\pm 0.00119}$	$3.46_{\pm 0.19}$
PiSSA	28	$3e-05$	$0.817_{\pm 0.001}$	$0.02752_{\pm 0.00050}$	$3.37_{\pm 0.06}$
MiLoRA	7	$1e-05$	$0.139_{\pm 0.000}$	$0.00676_{\pm 0.00002}$	$4.87_{\pm 0.01}$
MiLoRA	14	$1e-05$	$0.144_{\pm 0.000}$	$0.00461_{\pm 0.00001}$	$3.20_{\pm 0.01}$
MiLoRA	28	$3e-05$	$0.346_{\pm 0.001}$	$0.01094_{\pm 0.00006}$	$3.16_{\pm 0.01}$
HRA	16	0.0001	$0.564_{\pm 0.004}$	$0.00153_{\pm 0.00000}$	$0.27_{\pm 0.00}$
HRA	32	0.0001	$0.636_{\pm 0.003}$	$0.00155_{\pm 0.00001}$	$0.24_{\pm 0.00}$
HRA	64	$1e-05$	$0.286_{\pm 0.003}$	$0.00131_{\pm 0.00001}$	$0.46_{\pm 0.01}$

Table 13: Qwen2.5-7B singular-value restoration and reconstruction controls, with accuracy in percent.

Method	Capacity	LR	Trained acc.	Rebuilt acc.	Restored acc.	ΔNLL vs. rebuilt
LoRA	7	0.001	$90.87_{\pm 0.90}$	$90.60_{\pm 0.85}$	$90.73_{\pm 0.55}$	$0.00756_{\pm 0.00546}$
LoRA	14	$1e-05$	$91.73_{\pm 0.29}$	$91.53_{\pm 0.29}$	$91.73_{\pm 0.15}$	$-0.00002_{\pm 0.00007}$
LoRA	28	$3e-05$	$92.20_{\pm 0.53}$	$92.23_{\pm 0.47}$	$92.07_{\pm 0.49}$	$0.00005_{\pm 0.00005}$
OFT	32	0.0003	$91.77_{\pm 0.71}$	$91.67_{\pm 0.75}$	$91.73_{\pm 0.68}$	$0.00001_{\pm 0.00009}$
OFT	64	0.0003	$89.87_{\pm 0.42}$	$90.17_{\pm 0.21}$	$89.63_{\pm 0.42}$	$-0.00014_{\pm 0.00023}$
OFT	128	0.0001	$90.70_{\pm 0.36}$	$91.07_{\pm 0.61}$	$90.80_{\pm 0.53}$	$0.00001_{\pm 0.00030}$
DoRA	7	0.0003	$92.27_{\pm 0.31}$	$92.23_{\pm 0.38}$	$92.30_{\pm 0.10}$	$0.00005_{\pm 0.00008}$
DoRA	14	0.0003	$92.83_{\pm 0.50}$	$92.53_{\pm 0.47}$	$92.80_{\pm 0.35}$	$-0.00007_{\pm 0.00008}$
DoRA	28	0.0001	$92.33_{\pm 0.15}$	$92.50_{\pm 0.36}$	$92.47_{\pm 0.15}$	$0.00004_{\pm 0.00012}$
PiSSA	7	$3e-05$	$92.40_{\pm 0.10}$	$92.30_{\pm 0.17}$	$92.33_{\pm 0.25}$	$-0.00034_{\pm 0.00008}$
PiSSA	14	$3e-05$	$92.10_{\pm 0.52}$	$92.07_{\pm 0.59}$	$92.07_{\pm 0.40}$	$0.00014_{\pm 0.00041}$
PiSSA	28	$3e-05$	$91.57_{\pm 1.10}$	$92.20_{\pm 1.15}$	$92.10_{\pm 1.00}$	$-0.00009_{\pm 0.00060}$
MiLoRA	7	$1e-05$	$92.53_{\pm 0.15}$	$92.57_{\pm 0.47}$	$92.77_{\pm 0.32}$	$0.00000_{\pm 0.00012}$
MiLoRA	14	$1e-05$	$92.60_{\pm 0.30}$	$92.70_{\pm 0.35}$	$92.57_{\pm 0.29}$	$0.00006_{\pm 0.00021}$
MiLoRA	28	$3e-05$	$92.20_{\pm 0.17}$	$92.50_{\pm 0.26}$	$92.30_{\pm 0.10}$	$-0.00010_{\pm 0.00015}$
HRA	16	0.0001	$91.93_{\pm 0.06}$	$91.83_{\pm 0.21}$	$91.97_{\pm 0.29}$	$-0.00004_{\pm 0.00003}$
HRA	32	0.0001	$92.10_{\pm 0.17}$	$92.00_{\pm 0.10}$	$91.90_{\pm 0.20}$	$0.00002_{\pm 0.00001}$
HRA	64	$1e-05$	$91.70_{\pm 0.20}$	$91.80_{\pm 0.17}$	$92.03_{\pm 0.15}$	$0.00006_{\pm 0.00008}$

Table 14: Qwen2.5-7B component-restoration effects relative to the trained model, with accuracy changes in percentage points.

Method	Capacity	LR	Top Δacc.	Middle Δacc.	Tail Δacc.	Top ΔNLL
LoRA	7	0.001	$-82.00_{\pm 2.40}$	$-0.70_{\pm 0.44}$	$0.07_{\pm 0.93}$	$-0.01099_{\pm 0.00343}$
LoRA	14	$1e-05$	$-49.67_{\pm 1.91}$	$-1.17_{\pm 0.12}$	$-0.33_{\pm 0.47}$	$-0.00139_{\pm 0.00026}$
LoRA	28	$3e-05$	$-1.03_{\pm 0.06}$	$0.20_{\pm 0.17}$	$-0.10_{\pm 0.20}$	$-0.00259_{\pm 0.00023}$
OFT	32	0.0003	$0.13_{\pm 0.75}$	$0.23_{\pm 0.84}$	$-0.17_{\pm 0.06}$	$-0.00772_{\pm 0.00056}$
OFT	64	0.0003	$2.27_{\pm 0.81}$	$1.37_{\pm 0.55}$	$0.30_{\pm 0.36}$	$-0.01298_{\pm 0.00104}$
OFT	128	0.0001	$1.03_{\pm 0.45}$	$1.07_{\pm 0.91}$	$0.60_{\pm 0.70}$	$-0.00669_{\pm 0.00015}$
DoRA	7	0.0003	$0.07_{\pm 0.29}$	$0.17_{\pm 0.60}$	$-0.33_{\pm 0.67}$	$-0.00218_{\pm 0.00102}$
DoRA	14	0.0003	$-1.17_{\pm 0.51}$	$-0.70_{\pm 0.62}$	$-0.93_{\pm 0.12}$	$-0.00147_{\pm 0.00068}$
DoRA	28	0.0001	$-0.20_{\pm 0.46}$	$0.67_{\pm 0.47}$	$0.03_{\pm 0.06}$	$-0.00272_{\pm 0.00066}$
PiSSA	7	$3e-05$	$-86.50_{\pm 0.52}$	$-0.07_{\pm 0.31}$	$-0.17_{\pm 0.15}$	$-0.01057_{\pm 0.00106}$
PiSSA	14	$3e-05$	$-86.40_{\pm 1.51}$	$-0.17_{\pm 0.15}$	$-0.00_{\pm 0.36}$	$-0.01119_{\pm 0.00137}$
PiSSA	28	$3e-05$	$-87.27_{\pm 1.07}$	$0.43_{\pm 0.47}$	$0.20_{\pm 0.66}$	$-0.01063_{\pm 0.00117}$
MiLoRA	7	$1e-05$	$-1.77_{\pm 0.21}$	$-0.93_{\pm 0.47}$	$-60.07_{\pm 1.01}$	$-0.00245_{\pm 0.00008}$
MiLoRA	14	$1e-05$	$-0.63_{\pm 0.40}$	$-0.53_{\pm 0.35}$	$-54.83_{\pm 1.30}$	$-0.00290_{\pm 0.00014}$
MiLoRA	28	$3e-05$	$-0.03_{\pm 0.45}$	$0.20_{\pm 0.10}$	$-1.17_{\pm 0.40}$	$-0.00405_{\pm 0.00029}$
HRA	16	0.0001	$-72.03_{\pm 1.29}$	$-0.03_{\pm 0.06}$	$-0.10_{\pm 0.26}$	$-0.00436_{\pm 0.00052}$
HRA	32	0.0001	$-50.87_{\pm 3.04}$	$-0.43_{\pm 0.12}$	$-0.10_{\pm 0.46}$	$-0.00576_{\pm 0.00019}$
HRA	64	$1e-05$	$-77.60_{\pm 0.56}$	$-2.87_{\pm 0.76}$	$-0.03_{\pm 0.12}$	$-0.00312_{\pm 0.00010}$

Table 15: Llama-3.1-8B spectral displacement at every selected capacity and LR.

Method	Capacity	LR	$100d_W$	$100d_S$	$100d_S/d_W$
LoRA	7	0.0003	$2.167_{\pm 0.020}$	$0.04985_{\pm 0.00207}$	$2.30_{\pm 0.08}$
LoRA	15	0.0003	$2.845_{\pm 0.026}$	$0.08243_{\pm 0.00717}$	$2.90_{\pm 0.23}$
LoRA	30	0.0003	$3.707_{\pm 0.024}$	$0.12641_{\pm 0.00536}$	$3.41_{\pm 0.13}$
OFT	32	0.0003	$5.023_{\pm 0.011}$	$0.00190_{\pm 0.00000}$	$0.04_{\pm 0.00}$
OFT	64	0.0003	$6.626_{\pm 0.015}$	$0.00193_{\pm 0.00000}$	$0.03_{\pm 0.00}$
OFT	128	0.0001	$3.488_{\pm 0.007}$	$0.00188_{\pm 0.00000}$	$0.05_{\pm 0.00}$
DoRA	7	0.0003	$2.200_{\pm 0.028}$	$0.04447_{\pm 0.00197}$	$2.02_{\pm 0.07}$
DoRA	15	0.0003	$2.871_{\pm 0.023}$	$0.06579_{\pm 0.00819}$	$2.29_{\pm 0.27}$
DoRA	30	0.0003	$3.712_{\pm 0.015}$	$0.10183_{\pm 0.00382}$	$2.74_{\pm 0.09}$
PiSSA	7	0.0001	$1.796_{\pm 0.006}$	$0.15402_{\pm 0.00041}$	$8.58_{\pm 0.01}$
PiSSA	15	0.0001	$2.309_{\pm 0.005}$	$0.19040_{\pm 0.00148}$	$8.25_{\pm 0.05}$
PiSSA	30	$3e-05$	$1.060_{\pm 0.001}$	$0.04113_{\pm 0.00028}$	$3.88_{\pm 0.02}$
MiLoRA	7	0.0003	$1.851_{\pm 0.015}$	$0.16542_{\pm 0.00088}$	$8.94_{\pm 0.02}$
MiLoRA	15	0.0003	$2.457_{\pm 0.013}$	$0.25238_{\pm 0.00124}$	$10.27_{\pm 0.04}$
MiLoRA	30	0.0003	$3.282_{\pm 0.019}$	$0.36372_{\pm 0.00318}$	$11.08_{\pm 0.05}$
HRA	16	0.003	$5.547_{\pm 0.023}$	$0.00188_{\pm 0.00000}$	$0.03_{\pm 0.00}$
HRA	32	0.003	$9.077_{\pm 0.052}$	$0.00189_{\pm 0.00000}$	$0.02_{\pm 0.00}$
HRA	64	0.001	$6.864_{\pm 0.027}$	$0.00189_{\pm 0.00000}$	$0.03_{\pm 0.00}$

Table 16: Llama-3.1-8B singular-value restoration and reconstruction controls, with accuracy in percent.

Method	Capacity	LR	Trained acc.	Rebuilt acc.	Restored acc.	ΔNLL vs. rebuilt
LoRA	7	0.0003	$71.00_{\pm 0.35}$	$71.30_{\pm 0.46}$	$71.07_{\pm 0.51}$	$0.00006_{\pm 0.00016}$
LoRA	15	0.0003	$72.33_{\pm 0.93}$	$72.50_{\pm 0.26}$	$72.23_{\pm 0.90}$	$-0.00023_{\pm 0.00025}$
LoRA	30	0.0003	$72.20_{\pm 1.28}$	$72.40_{\pm 0.89}$	$71.97_{\pm 1.43}$	$-0.00254_{\pm 0.00068}$
OFT	32	0.0003	$71.80_{\pm 1.37}$	$71.77_{\pm 0.81}$	$71.73_{\pm 1.29}$	$-0.00007_{\pm 0.00012}$
OFT	64	0.0003	$73.43_{\pm 0.50}$	$73.03_{\pm 0.29}$	$73.57_{\pm 0.49}$	$0.00003_{\pm 0.00010}$
OFT	128	0.0001	$72.47_{\pm 1.31}$	$72.40_{\pm 0.87}$	$72.33_{\pm 0.85}$	$0.00004_{\pm 0.00014}$
DoRA	7	0.0003	$71.10_{\pm 0.61}$	$71.47_{\pm 1.03}$	$71.17_{\pm 0.57}$	$-0.00029_{\pm 0.00029}$
DoRA	15	0.0003	$72.77_{\pm 0.87}$	$72.40_{\pm 1.01}$	$72.70_{\pm 0.17}$	$-0.00065_{\pm 0.00034}$
DoRA	30	0.0003	$73.10_{\pm 2.19}$	$73.33_{\pm 1.90}$	$73.07_{\pm 1.48}$	$-0.00208_{\pm 0.00085}$
PiSSA	7	0.0001	$70.77_{\pm 0.40}$	$70.67_{\pm 0.55}$	$70.47_{\pm 0.25}$	$-0.00509_{\pm 0.00037}$
PiSSA	15	0.0001	$71.00_{\pm 0.26}$	$70.83_{\pm 0.45}$	$71.30_{\pm 0.36}$	$-0.00390_{\pm 0.00071}$
PiSSA	30	$3e-05$	$71.23_{\pm 0.06}$	$71.67_{\pm 0.85}$	$71.27_{\pm 1.05}$	$-0.00063_{\pm 0.00010}$
MiLoRA	7	0.0003	$70.97_{\pm 0.80}$	$70.90_{\pm 0.44}$	$71.50_{\pm 0.30}$	$-0.00046_{\pm 0.00061}$
MiLoRA	15	0.0003	$71.90_{\pm 0.46}$	$71.97_{\pm 0.86}$	$71.70_{\pm 0.46}$	$-0.00042_{\pm 0.00058}$
MiLoRA	30	0.0003	$72.63_{\pm 1.01}$	$72.53_{\pm 0.92}$	$71.97_{\pm 0.81}$	$0.00208_{\pm 0.00065}$
HRA	16	0.003	$70.57_{\pm 0.95}$	$70.83_{\pm 0.87}$	$70.50_{\pm 0.56}$	$0.00010_{\pm 0.00006}$
HRA	32	0.003	$71.00_{\pm 1.67}$	$71.03_{\pm 1.29}$	$70.40_{\pm 1.30}$	$0.00012_{\pm 0.00015}$
HRA	64	0.001	$71.63_{\pm 1.08}$	$71.07_{\pm 1.18}$	$70.93_{\pm 0.83}$	$-0.00005_{\pm 0.00023}$

Table 17: Llama-3.1-8B component-restoration effects relative to the trained model, with accuracy changes in percentage points.

Method	Capacity	LR	Top Δacc.	Middle Δacc.	Tail Δacc.	Top ΔNLL
LoRA	7	0.0003	$-48.23_{\pm 11.06}$	$-0.67_{\pm 0.59}$	$-0.03_{\pm 0.71}$	$-0.02150_{\pm 0.00224}$
LoRA	15	0.0003	$-34.73_{\pm 6.28}$	$-1.57_{\pm 1.36}$	$-0.17_{\pm 0.61}$	$-0.02192_{\pm 0.00220}$
LoRA	30	0.0003	$-27.43_{\pm 1.07}$	$-0.60_{\pm 1.31}$	$-0.60_{\pm 1.14}$	$-0.02506_{\pm 0.00329}$
OFT	32	0.0003	$-23.30_{\pm 3.12}$	$-0.40_{\pm 1.56}$	$-0.03_{\pm 0.38}$	$-0.03143_{\pm 0.00099}$
OFT	64	0.0003	$-17.63_{\pm 3.27}$	$-1.43_{\pm 1.22}$	$-0.63_{\pm 1.42}$	$-0.06295_{\pm 0.00097}$
OFT	128	0.0001	$-18.97_{\pm 0.61}$	$-0.83_{\pm 1.12}$	$-0.07_{\pm 0.58}$	$-0.03398_{\pm 0.00105}$
DoRA	7	0.0003	$-45.40_{\pm 10.83}$	$-0.30_{\pm 0.52}$	$-0.27_{\pm 0.50}$	$-0.02029_{\pm 0.00029}$
DoRA	15	0.0003	$-42.10_{\pm 8.64}$	$-1.07_{\pm 0.42}$	$-1.00_{\pm 0.44}$	$-0.02078_{\pm 0.00252}$
DoRA	30	0.0003	$-27.03_{\pm 1.55}$	$-1.77_{\pm 1.17}$	$-1.20_{\pm 1.44}$	$-0.02313_{\pm 0.00277}$
PiSSA	7	0.0001	$-70.77_{\pm 0.40}$	$-0.10_{\pm 0.78}$	$-0.23_{\pm 0.38}$	$-0.10749_{\pm 0.00253}$
PiSSA	15	0.0001	$-71.00_{\pm 0.26}$	$-0.37_{\pm 0.50}$	$-0.07_{\pm 0.21}$	$-0.15029_{\pm 0.00406}$
PiSSA	30	$3e-05$	$-71.23_{\pm 0.06}$	$0.10_{\pm 0.53}$	$-0.13_{\pm 0.21}$	$-0.06317_{\pm 0.00101}$
MiLoRA	7	0.0003	$-11.93_{\pm 1.76}$	$0.67_{\pm 0.85}$	$-1.43_{\pm 0.49}$	$-0.03790_{\pm 0.00074}$
MiLoRA	15	0.0003	$-7.57_{\pm 0.25}$	$-1.07_{\pm 0.15}$	$-1.77_{\pm 0.85}$	$-0.04841_{\pm 0.00096}$
MiLoRA	30	0.0003	$-5.77_{\pm 0.49}$	$-1.00_{\pm 1.08}$	$-3.17_{\pm 0.91}$	$-0.04858_{\pm 0.00341}$
HRA	16	0.003	$-19.33_{\pm 3.96}$	$-0.77_{\pm 1.07}$	$-0.27_{\pm 1.43}$	$-0.04873_{\pm 0.00166}$
HRA	32	0.003	$-14.07_{\pm 5.22}$	$-0.73_{\pm 0.50}$	$0.40_{\pm 1.25}$	$-0.09701_{\pm 0.00496}$
HRA	64	0.001	$-11.07_{\pm 0.81}$	$-1.13_{\pm 2.12}$	$0.17_{\pm 1.04}$	$-0.08909_{\pm 0.00459}$

C.2 Singular-value displacement and restoration

At largest capacity, PiSSA concentrates over 99% of squared singular-value displacement in dominant components on all three models. MiLoRA concentrates about 99% on Qwen/FLUX and 95% on Llama in trailing components. These profiles follow their initialization choices, while restoration measures how much the spectral changes contribute to task performance.

Across all 108 checkpoints, singular-value restoration changes general NLL by -0.00535 to $+0.01365$ relative to reconstruction, alongside a maximum absolute task-accuracy change of 1.3 percentage points. These small task effects extend the restoration finding across language capacities.

A high-LR LoRA outlier. Qwen LoRA r7 selects the highest tested LR, 10^{-3}. Its spectral displacement is 1.67% of the base norm and 20.53% of the weight update, exceeding the ratios in the low-drift settings. Restoration still largely preserves task performance, with a mean accuracy change of $+0.13$ pp relative to reconstruction (Table 13). The restoration finding therefore also holds in this larger-ratio setting.

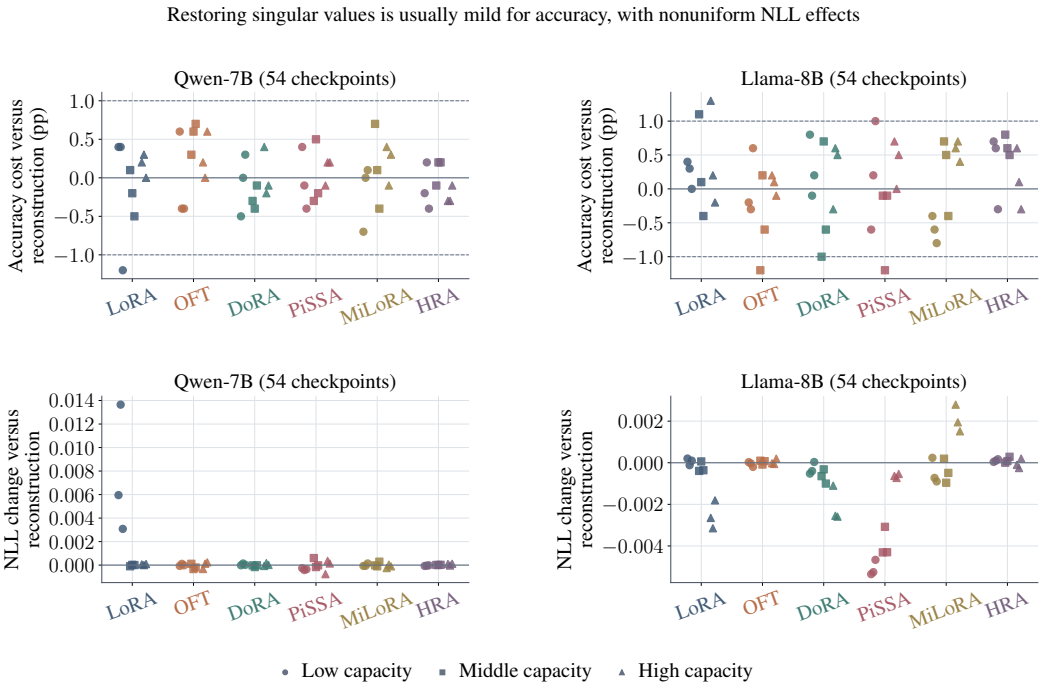

Figure 7: Singular-value restoration across capacities and seeds. Trained, reconstructed and restored models are shown separately, extending the main accuracy comparison with reconstruction controls and retention measurements.

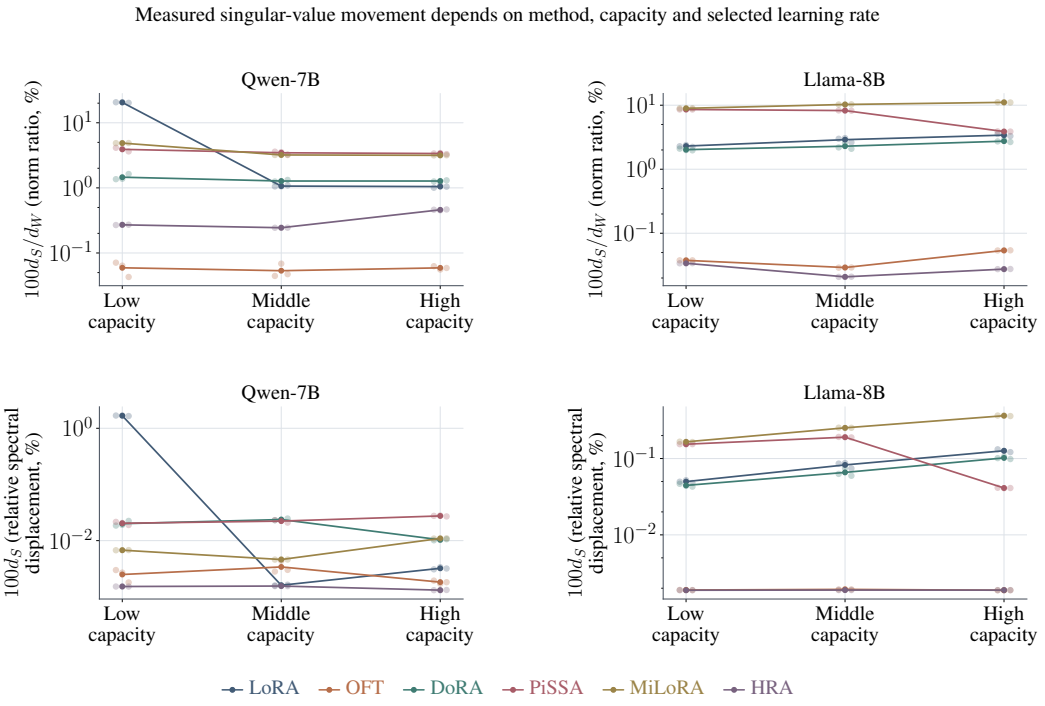

Figure 8: Measured weight and singular-value displacement across all six methods and both models at each selected capacity and LR. Ratios express spectral displacement relative to weight displacement.

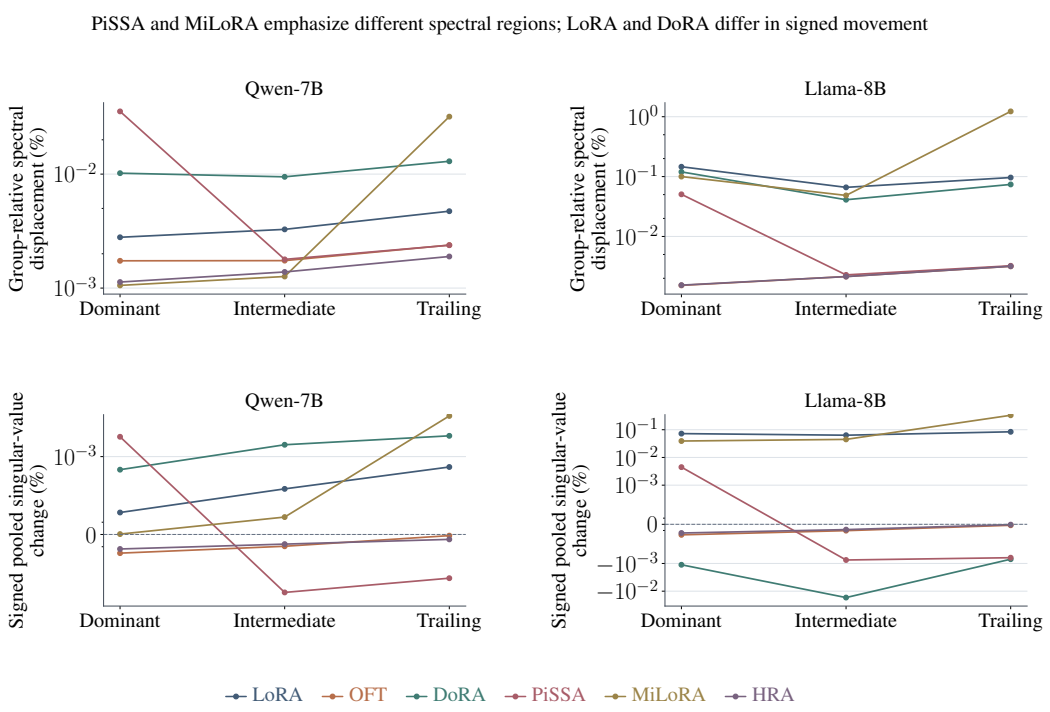

Figure 9: Singular-value changes across spectral components at largest capacity. PiSSA's displacement concentrates near the leading end, whereas MiLoRA's concentrates near the tail, consistent with their initialization choices. The pretrained reference and method colors are retained.

C.3 Hyperspherical energy and pairwise geometry

We measure all adapted matrices in the largest-capacity intervention checkpoints, with 196 per Qwen model, 224 per Llama model and 80 per FLUX cat model. The 54 checkpoints and 9,000 matrix comparisons use the same selected LRs as the spectral and functional analyses. This tests whether the approximate geometric preservation observed for LoRA-family methods also extends to HE.

Definitions and aggregation. For stored weights $\boldsymbol{W} \in \mathbb{R}^{n\times d}$, rows represent output neurons. Let $\widehat{\boldsymbol{w}}_i = \boldsymbol{w}_i/\|\boldsymbol{w}_i\|_2$. Following Liu et al. (2018) and Qiu et al. (2023), we compute

$$H(\boldsymbol{W}) = \sum_{i\neq j} \frac{1}{\|\widehat{\boldsymbol{w}}_i - \widehat{\boldsymbol{w}}_j\|_2}, \qquad \delta_H = \frac{|H(\boldsymbol{W}_*) - H(\boldsymbol{W}_0)|}{H(\boldsymbol{W}_0)}.$$

The sum covers every ordered distinct row pair. To measure pairwise changes without signed cancellation, we also compute

$$\delta_C = \left[\frac{1}{n(n-1)} \sum_{i\neq j} \left(\widehat{\boldsymbol{w}}_{*,i}^\top \widehat{\boldsymbol{w}}_{*,j} - \widehat{\boldsymbol{w}}_{0,i}^\top \widehat{\boldsymbol{w}}_{0,j}\right)^2\right]^{1/2}.$$

We average each metric equally over matrices within a checkpoint, then summarize training seeds. The absolute HE change is taken before matrix averaging. A common right-orthogonal transformation preserves row inner products and HE.

Results. All checkpoint means of δ_H are below 0.01%, with maximum 0.00980% (Table 18). This supports approximate HE preservation across the selected LoRA-family methods as well as the orthogonal methods. The pairwise measure shows how closely individual neuron relationships are preserved alongside aggregate HE.

Table 18: Geometric preservation at largest selected capacities. Each checkpoint averages matrices equally, followed by the seed mean and sample SD. HE changes are in units of $10^{-4}\%$, and cosine changes are in units of 10^{-4}.

Method	Qwen2.5-7B	Llama-3.1-8B	FLUX-4B cat
(a) Absolute relative HE change ($\times 10^{-4}\%$)			
LoRA	$14.8_{\pm 0.96}$	$1.72_{\pm 0.17}$	$0.281_{\pm 0.025}$
OFT	$1.5_{\pm 0.2}$	$0.0215_{\pm 0.0014}$	$0.533_{\pm 0.065}$
DoRA	$80.3_{\pm 15}$	$1.7_{\pm 0.069}$	$0.714_{\pm 0.072}$
PiSSA	$91.9_{\pm 4.6}$	$1.53_{\pm 0.099}$	$3.02_{\pm 0.24}$
MiLoRA	$1.35_{\pm 0.37}$	$4.36_{\pm 0.17}$	$0.987_{\pm 0.032}$
HRA	$0.625_{\pm 0.12}$	$0.0264_{\pm 0.0013}$	$0.00804_{\pm 0.0008}$
(b) Pairwise-cosine RMS change ($\times 10^{-4}$)			
LoRA	$0.906_{\pm 0.015}$	$10.7_{\pm 0.15}$	$2.51_{\pm 0.11}$
OFT	$0.372_{\pm 0.00052}$	$0.35_{\pm 2.3\times 10^{-5}}$	$0.48_{\pm 0.022}$
DoRA	$2.26_{\pm 0.034}$	$10.6_{\pm 0.069}$	$4.65_{\pm 0.08}$
PiSSA	$5.33_{\pm 0.045}$	$5.98_{\pm 0.011}$	$17_{\pm 0.1}$
MiLoRA	$1.12_{\pm 0.0044}$	$10.6_{\pm 0.1}$	$4.11_{\pm 0.19}$
HRA	$0.307_{\pm 0.001}$	$0.35_{\pm 1.6\times 10^{-5}}$	$0.407_{\pm 5.9\times 10^{-6}}$

C.4 Component effects, interactions and task performance

Restoring dominant components lowers general-text NLL at every capacity and gives the largest reduction among the three groups in 106/108 checkpoints. The exceptions to the largest reduction are Qwen DoRA r14/seed 73 and MiLoRA r28/seed 13. Dominant restoration remains strongest in every configuration mean, supporting the consistent forgetting response across language capacities.

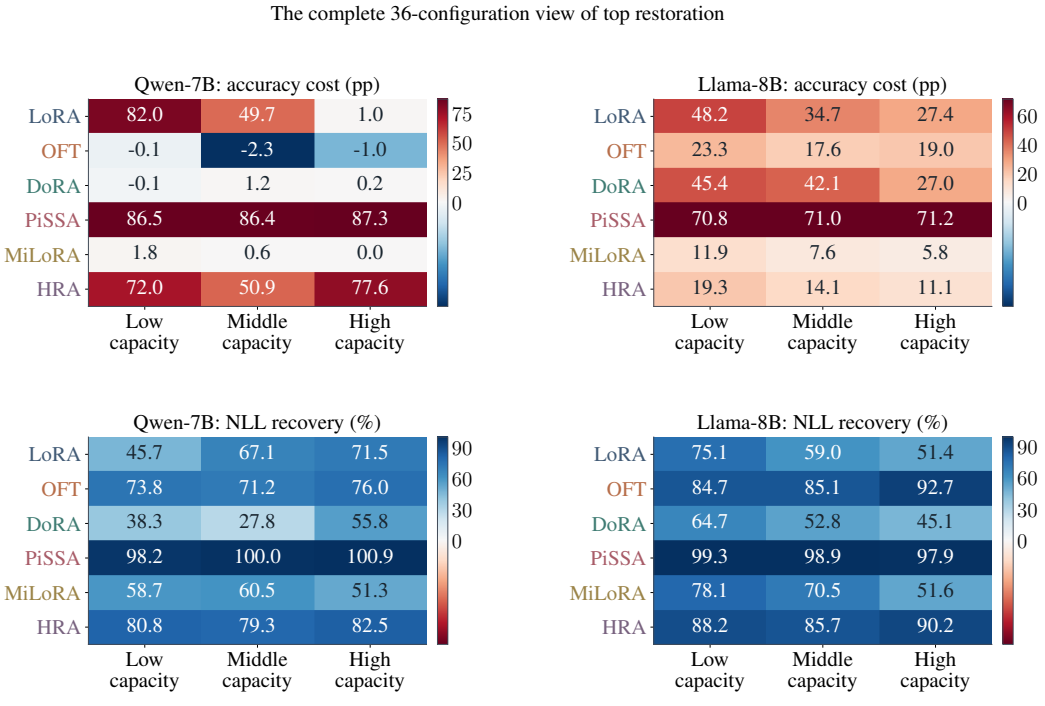

Figure 10: Dominant-component restoration at every selected configuration and seed. The retention response persists across the grid, while task-accuracy responses vary. Recovery is normalized by each checkpoint's trained-model NLL increase.

For a set A of adapted component groups, define

$$\boldsymbol{W}(A) = \sum_{B \in A} \sum_{i \in B} s_{0,i} \boldsymbol{u}_{*,i} \boldsymbol{v}_{*,i}^{\top} + \sum_{B \notin A} \sum_{i \in B} s_{0,i} \boldsymbol{u}_{0,i} \boldsymbol{v}_{0,i}^{\top}.$$

Its endpoints are the pretrained-component reconstruction and the full value-restored model. For general NLL $f(A)$, the effect of adapting group B is $f(A \cup \{B\}) - f(A)$. We evaluate it under all four combinations of pretrained or adapted components in the other groups. Adapting dominant components increases NLL in every combination across all 108 checkpoints. Intermediate and trailing effects change sign across combinations in 38 and 42 checkpoints, using a 10^{-4} tolerance. Thus, the dominant-component forgetting response persists as the other components change.

All 36 configurations include evaluations of every component combination for all three seeds.

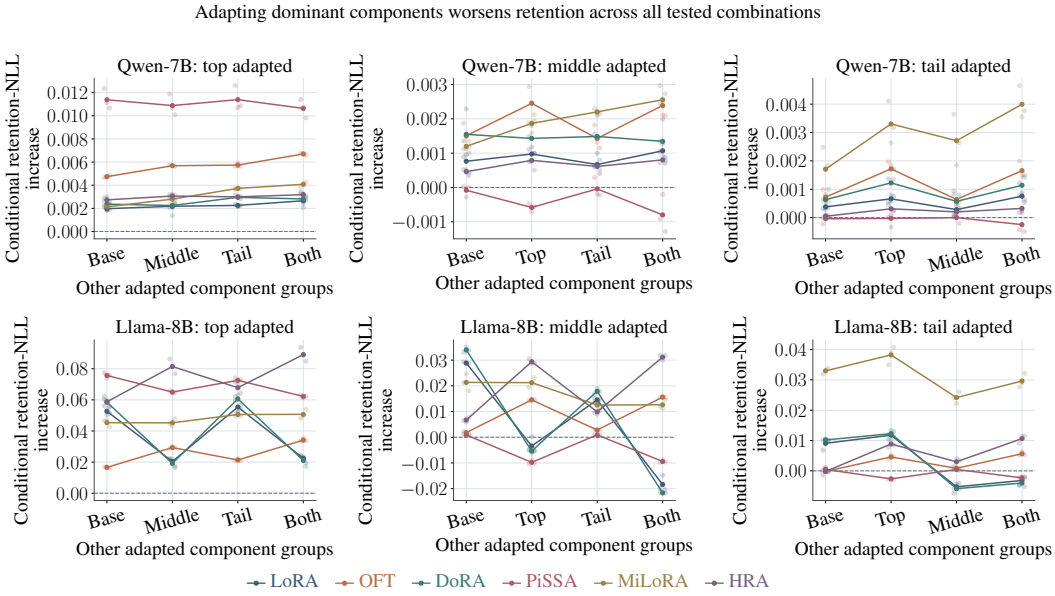

Figure 11: Effects of adapting each component group under all combinations of the other groups. Dominant-component adaptation consistently increases general NLL, while intermediate and trailing effects can change sign. Both NLL evaluations agree on the dominant-component direction.

Task performance across selected capacities. Table 19 extends the largest-capacity comparison in Figure 12. Qwen LoRA retains task performance with dominant components alone at small capacity, with neither component set alone at medium capacity, and with either set alone at large capacity. Llama LoRA's outcomes also vary across capacities and seeds. PiSSA retains the dominant-only pattern at every language capacity, supporting its consistent concentration of task gains in dominant components. The other methods illustrate the setting-dependent component contributions described in the main text, alongside the consistent forgetting response.

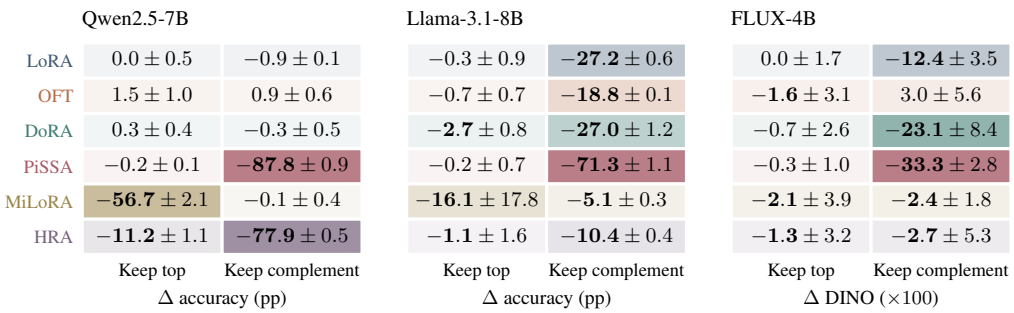

Figure 12: Task performance when retaining only some adapted components. Cells show mean ± SD changes from $\boldsymbol{W}_R$ (all singular values restored) at the largest capacities. Columns retain the adapted dominant components alone or the intermediate and trailing components together. Bold means mark losses exceeding 1 pp for LLMs or 1 DINO×100 unit for FLUX.

Table 19: Task sufficiency across all selected language capacities. Entries count seeds with accuracy loss $\leq$ 1 pp relative to full value restoration, out of three seeds per configuration. Top and complement retain those adapted components on a pretrained background. Tables 12 and 15 give capacities and selected LRs.

Model	Method	Small		Medium		Large	
		Top	Complement	Top	Complement	Top	Complement
Qwen2.5-7B	LoRA	3/3	0/3	0/3	0/3	3/3	3/3
	OFT	3/3	3/3	3/3	3/3	3/3	3/3
	DoRA	3/3	3/3	3/3	1/3	3/3	3/3
	PiSSA	3/3	0/3	3/3	0/3	3/3	0/3
	MiLoRA	0/3	0/3	0/3	3/3	0/3	3/3
	HRA	0/3	0/3	2/3	0/3	0/3	0/3
Llama-3.1-8B	LoRA	2/3	0/3	1/3	0/3	3/3	0/3
	OFT	1/3	0/3	0/3	0/3	2/3	0/3
	DoRA	1/3	0/3	0/3	0/3	0/3	0/3
	PiSSA	3/3	0/3	3/3	0/3	3/3	0/3
	MiLoRA	0/3	0/3	0/3	0/3	0/3	0/3
	HRA	1/3	0/3	2/3	0/3	2/3	0/3

C.5 Consistency of the dominant-component response

The comparison covers 42 selected configurations and 126 checkpoints across all six methods, all tested language capacities and the largest FLUX budget for each method. Restoring dominant components gives the largest mean reduction in general-text NLL or general-image drift in all 42 configurations and a larger reduction than restoring either intermediate or trailing components in 123/126 checkpoints. These counts support the recurring dominant-component forgetting response across models and tasks.

Consistency with smaller edits. We measure edit norm as the square root of the summed squared Frobenius changes from trained weights across all adapted matrices, using the stored weights. We select cases where restoring intermediate or trailing components produces a larger edit norm than restoring dominant components. Configuration comparisons use seed means of norms and forgetting reductions. Dominant restoration still gives the largest mean forgetting reduction among all three restorations in 17/17 eligible configurations, comprising 8 Qwen, 6 Llama and 3 FLUX configurations. At the checkpoint level, the count is 48/51, with 22/24 for Qwen, 18/18 for Llama and 8/9 for FLUX. Each configuration or checkpoint is counted once, including when both alternatives have larger norms. Requiring both alternatives to have larger norms gives 4/4 configurations and 10/12 checkpoints. The eligible cases involve LoRA, DoRA and MiLoRA. These counts show that

the dominant-component advantage can persist with smaller edits, so edit magnitude alone does not determine the ordering of forgetting reductions.

C.6 Native OFT orthogonality and polar projection

OFTv2 approximates the Cayley inverse with a truncated Neumann series (Qiu et al., 2025b). We measure $\|\boldsymbol{R}\boldsymbol{R}^\top - \boldsymbol{I}\|_{\mathrm{F}}$, its maximum absolute entry, and $\|\boldsymbol{R} - \widehat{\boldsymbol{R}}\|_{\mathrm{F}}/\|\boldsymbol{R}\|_{\mathrm{F}}$, where $\widehat{\boldsymbol{R}}$ is the polar factor (Higham, 1986). The relative distance expresses the deviation from orthogonality on a scale normalized by the transform norm. These diagnostics cover 21 checkpoints each for Qwen and Llama, and 33 for FLUX. The maximum native-to-polar distances are about 0.154%, 0.094% and 11.0%, respectively (Table 20). Task and retention evaluations use native transforms. Polar projection measures their deviation from an exact orthogonal transform, clarifying the numerical precision of the geometric-preservation comparison.

Table 20: Native OFT-transform diagnostics. Maxima are taken over all matrices and checkpoints in each row. Frobenius and entrywise maxima describe different norms.

Model	Checkpoints	Max. $\|\boldsymbol{R}\boldsymbol{R}^\top - \boldsymbol{I}\|_{\mathrm{F}}$	Max. entry	Max. polar residual
FLUX-4B	33	8.04	0.136	0.11
Llama-8B	21	0.118	0.00528	0.000936
Qwen-7B	21	0.42	0.0233	0.00154

D FLUX spectral interventions and numerical controls

D.1 Complete six-method FLUX interventions

The FLUX cat intervention study contains 18 checkpoints and 396 evaluations, covering all six methods at their largest tested budgets. LR maximizes mean step-700 validation DINO, with ties favoring lower LR. Interventions use the final checkpoints and jointly replace all 80 adapted matrices.

Table 21: FLUX intervention operating points selected by mean validation DINO. Parameters are in millions. Displacements pool all adapted matrices before normalization. All edits act jointly on 80 matrices.

Method	Capacity	LR	Params. (M)	$100d_W$	$100d_S$	$100d_S/d_W$
LoRA	32	5×10^{-5}	38.34	$0.783_{\pm 0.029}$	$0.0088_{\pm 0.0005}$	$1.12_{\pm 0.09}$
OFT	256	1×10^{-4}	39.17	$10.587_{\pm 0.085}$	$0.0071_{\pm 0.0013}$	$0.07_{\pm 0.01}$
DoRA	32	1×10^{-4}	39.23	$1.586_{\pm 0.062}$	$0.0204_{\pm 0.0007}$	$1.28_{\pm 0.02}$
PiSSA	32	5×10^{-5}	38.34	$2.477_{\pm 0.027}$	$0.1434_{\pm 0.0021}$	$5.79_{\pm 0.04}$
MiLoRA	32	1×10^{-4}	38.34	$2.019_{\pm 0.046}$	$0.1763_{\pm 0.0029}$	$8.73_{\pm 0.06}$
HRA	124	3×10^{-4}	38.09	$11.744_{\pm 0.260}$	$0.0017_{\pm 0.0000}$	$0.01_{\pm 0.00}$

Table 22: FLUX value-restoration controls. DINO scores and paired differences are multiplied by 100. The last column compares restoration with reconstruction through the same SVD/storage path.

Method	Trained	Reconstructed	Restored	Values only	Δ vs. recon.
LoRA	$68.06_{\pm 0.78}$	$67.84_{\pm 0.47}$	$68.12_{\pm 0.45}$	$36.24_{\pm 1.13}$	$0.28_{\pm 0.04}$
OFT	$59.04_{\pm 5.51}$	$59.33_{\pm 5.91}$	$59.18_{\pm 5.42}$	$36.17_{\pm 1.09}$	$-0.15_{\pm 0.57}$
DoRA	$70.47_{\pm 3.96}$	$70.95_{\pm 4.05}$	$70.59_{\pm 3.75}$	$36.24_{\pm 1.11}$	$-0.36_{\pm 0.36}$
PiSSA	$68.88_{\pm 3.45}$	$69.07_{\pm 2.91}$	$69.03_{\pm 2.96}$	$36.21_{\pm 1.24}$	$-0.05_{\pm 0.83}$
MiLoRA	$66.13_{\pm 2.10}$	$66.03_{\pm 2.17}$	$66.06_{\pm 2.15}$	$36.34_{\pm 1.35}$	$0.03_{\pm 0.60}$
HRA	$63.75_{\pm 5.76}$	$63.69_{\pm 5.78}$	$63.54_{\pm 5.99}$	$36.19_{\pm 1.06}$	$-0.15_{\pm 0.27}$

Evaluation and numerical references. The BF16 evaluator generates 512×512 images with 20 sampling steps and guidance 3.5. Test DINO averages four held-out image/prompt pairs with three generation repeats. The drift bank has five class prompts and five general prompts, with general drift and CLIP using the latter. Drift is $(1 - \text{DINO cosine})/2$ relative to base-generated images.

Every change is paired within a checkpoint before computing the mean and sample SD. Component effects use the same-run merged trained model, restoration controls use reconstruction through the same SVD/storage path, and comparisons of retained task performance use full value restoration. Same-run references account for this difference. All 396 evaluations succeed, with every restored matrix matched and within the recorded numerical tolerance.

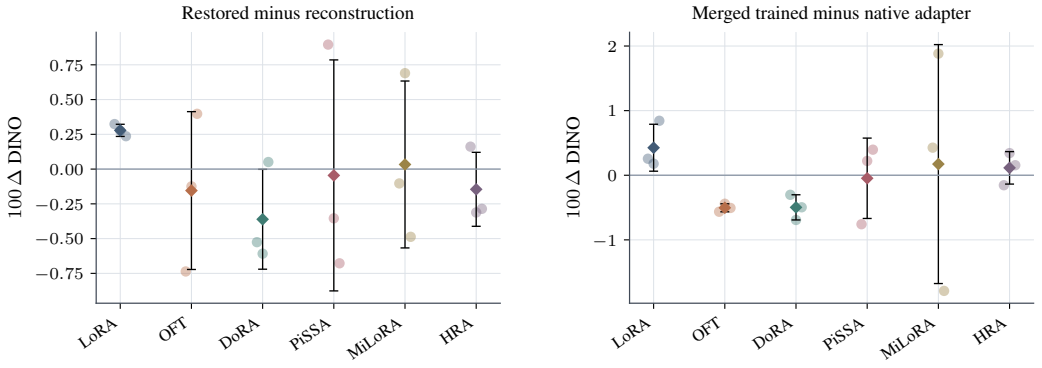

Figure 13: FLUX numerical controls. The left panel compares value restoration with reconstruction. The right panel compares the same-run merged trained model with the earlier native-adapter evaluation. Diamonds and bars show means and sample SDs, with faint points for individual checkpoints. Differences are in DINO scores $\times 100$.

Values and adapted bases. Value restoration nearly preserves fidelity for every method, whereas learned values on pretrained singular bases yield near-base scores (Table 22). This extends the main restoration finding to diffusion and points to the adapted bases as the principal source of task gains. Spectral displacement concentrates in PiSSA's dominant components and MiLoRA's trailing components (Figure 14).

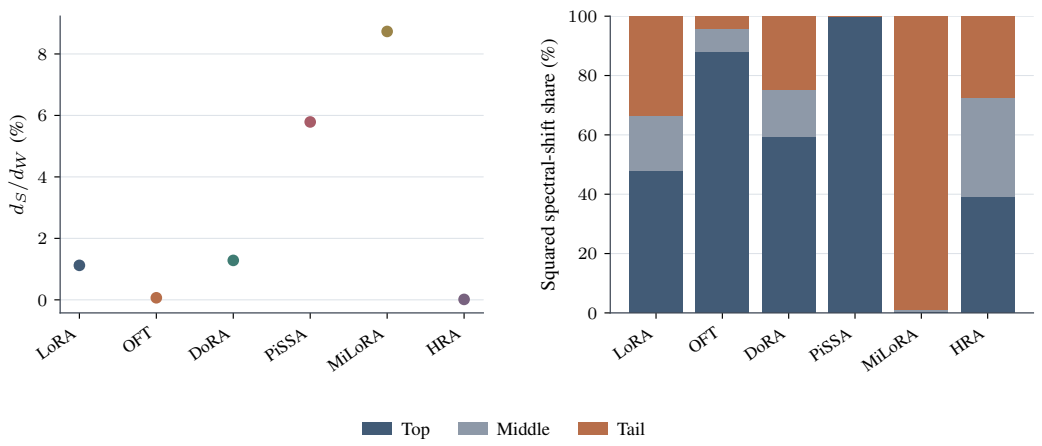

Figure 14: FLUX spectral displacement. The left panel shows the pooled spectral-to-weight displacement ratio with sample SDs. The right panel shows mean shares of squared singular-value displacement across component groups, with boundaries adjusted to pretrained spectral gaps. Shares are computed from all measured spectral vectors.

D.2 Task performance and forgetting across spectral components

The dominant, intermediate and trailing component groups are approximate thirds adjusted to nearby pretrained spectral gaps. Restore and keep operations replace complete rank-one contributions, using pretrained singular values as in Eq. (4). The measured response includes these interactions between components.

Restoring dominant components reduces general drift in every checkpoint and gives the largest mean reduction for each method. It gives the largest individual reduction in 17 of 18 checkpoints. MiLoRA seed zero has a slightly larger trailing-component response. Class-prompt drift also decreases in every checkpoint, with dominant restoration giving the largest reduction. Both drift measures therefore support the recurring dominant-component response. CLIP responses differ, with LoRA's mean general-prompt alignment decreasing. OFT gains fidelity on average but loses fidelity in one checkpoint.

Keeping only the adapted dominant components nearly preserves mean fidelity for LoRA, DoRA and PiSSA. LoRA and DoRA reduce general drift in every checkpoint, whereas PiSSA retains essentially all its drift. These responses show that similar retained task performance can accompany different retention outcomes. Fidelity losses vary across seeds, including a larger DoRA loss. We report DINO changes directly and use the 1 pp sufficiency rule only for language accuracy. OFT/HRA have greater trained drift than PiSSA in these selected settings.

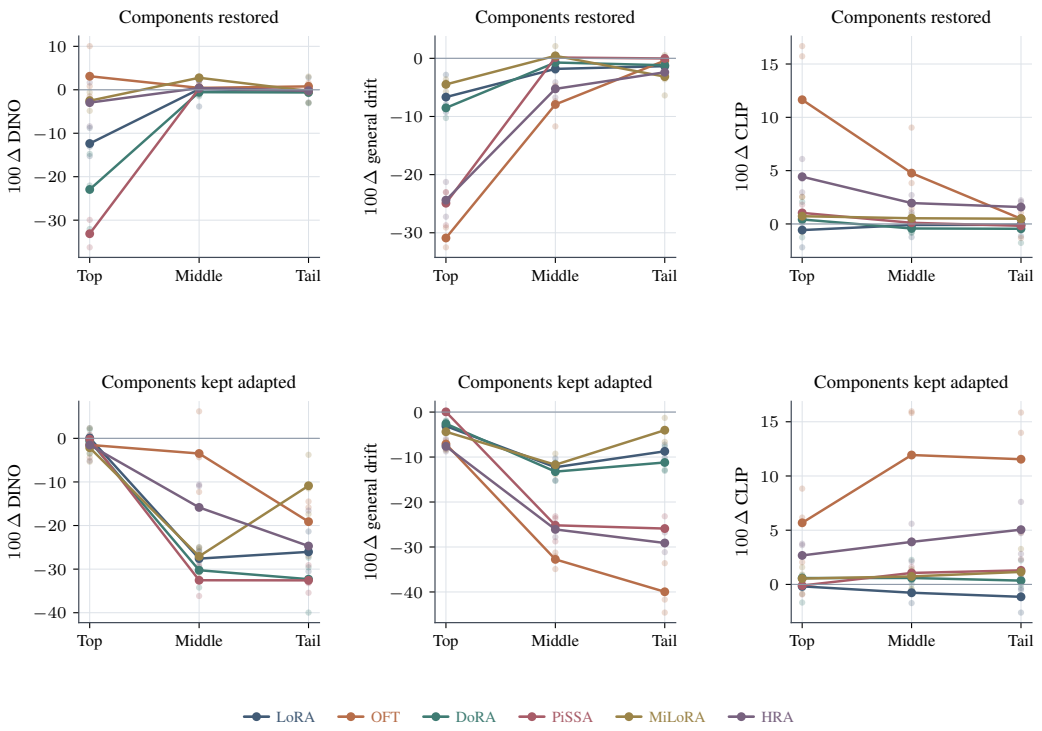

Figure 15: FLUX component-restoration responses. The top row restores the named component group, and the bottom row keeps only that adapted group. All changes use the same-run trained model. Lines show method means and faint points show individual checkpoints.

Summary. The FLUX results extend the singular-value restoration and dominant-component forgetting findings to image personalization.

E Geometry of Additive Updates

E.1 Distance to a Two-Sided Orthogonal Transformation

Write $q = \min(m, n)$ and let $O(k)$ denote the real $k \times k$ orthogonal group. The orbit $\{\boldsymbol{Q}_L \boldsymbol{W}_0 \boldsymbol{Q}_R^\top : \boldsymbol{Q}_L \in O(m), \boldsymbol{Q}_R \in O(n)\}$ contains exactly the matrices with the same singular values as $\boldsymbol{W}_0$. It includes reflections as well as rotations. A local smooth change of singular bases is generated by skew-symmetric matrices.

Proof of Proposition 3.1. **Step 1. Bound the distance from below.** Let $\boldsymbol{X} = \boldsymbol{Q}_L \boldsymbol{W}_0 \boldsymbol{Q}_R^\top$ be any matrix in the orbit. Orthogonal multiplication preserves singular values, so $\boldsymbol{X}$ has singular values $\boldsymbol{s}_0$ and $\|\boldsymbol{X}\|_{\mathrm{F}} = \|\boldsymbol{s}_0\|_2$. Write $\langle \boldsymbol{A}, \boldsymbol{B} \rangle_{\mathrm{F}} = \mathrm{tr}(\boldsymbol{A}^\top \boldsymbol{B})$. The singular-value trace inequality (Mirsky, 1975) bounds this inner product by the sum of the paired, decreasing singular values, giving

$$\langle \boldsymbol{W}_*, \boldsymbol{X} \rangle_{\mathrm{F}} \le \sum_{i=1}^{q} s_{*,i} s_{0,i}.$$

Expanding the squared distance therefore gives

$$\begin{aligned} \|\boldsymbol{W}_* - \boldsymbol{X}\|_{\mathrm{F}}^2 &= \|\boldsymbol{s}_*\|_2^2 + \|\boldsymbol{s}_0\|_2^2 - 2\langle \boldsymbol{W}_*, \boldsymbol{X} \rangle_{\mathrm{F}} \\ &\ge \sum_{i=1}^{q} (s_{*,i} - s_{0,i})^2 = \|\boldsymbol{s}_* - \boldsymbol{s}_0\|_2^2. \end{aligned}$$

Step 2. Attain the bound by aligning the singular bases. Take full SVDs $\boldsymbol{W}_j = \boldsymbol{U}_j \boldsymbol{\Sigma}_j \boldsymbol{V}_j^\top$ for $j \in \{0, *\}$. Here $\boldsymbol{U}_j$ and $\boldsymbol{V}_j$ are square orthogonal completions of the thin bases, and $\boldsymbol{\Sigma}_j \in \mathbb{R}^{m \times n}$

is rectangular diagonal. Choose $\boldsymbol{Q}_L = \boldsymbol{U}_* \boldsymbol{U}_0^\top$ and $\boldsymbol{Q}_R = \boldsymbol{V}_* \boldsymbol{V}_0^\top$. These matrices are orthogonal and give

$$\boldsymbol{Q}_L \boldsymbol{W}_0 \boldsymbol{Q}_R^\top = \boldsymbol{U}_* \boldsymbol{\Sigma}_0 \boldsymbol{V}_*^\top = \boldsymbol{W}_R.$$

Thus $\boldsymbol{W}_R$ keeps the base singular values while using the adapted singular bases. Orthogonal invariance of the Frobenius norm yields

$$\|\boldsymbol{W}_* - \boldsymbol{W}_R\|_{\mathrm{F}} = \|\boldsymbol{\Sigma}_* - \boldsymbol{\Sigma}_0\|_{\mathrm{F}} = \|\boldsymbol{s}_* - \boldsymbol{s}_0\|_2.$$

This equals the lower bound from Step 1, proving the minimum. No positivity or gap assumption was used, so the result also covers rank-deficient matrices and repeated singular values.

Step 3. Relate the distance to d_S/d_W. For nonzero base and update norms, the common normalization $\|\boldsymbol{s}_0\|_2 = \|\boldsymbol{W}_0\|_{\mathrm{F}}$ cancels, giving

$$\frac{d_S}{d_W} = \frac{\|\boldsymbol{s}_* - \boldsymbol{s}_0\|_2}{\|\boldsymbol{W}_* - \boldsymbol{W}_0\|_{\mathrm{F}}} = \frac{\|\boldsymbol{W}_* - \boldsymbol{W}_R\|_{\mathrm{F}}}{\|\boldsymbol{W}_* - \boldsymbol{W}_0\|_{\mathrm{F}}}.$$

For multiple matrices, summing the squared residuals before taking square roots gives the same interpretation for the pooled ratio. □

E.2 First-order spectral and basis motion

Proof of Proposition 3.2. **Step 1. Express the derivative in singular coordinates.** The positive, distinct singular values allow a smooth local choice of singular vectors (Stewart, 1990). Complete them to smooth square orthogonal bases and write $\boldsymbol{W}_0 + \epsilon \boldsymbol{E} = \boldsymbol{U}(\epsilon)\boldsymbol{\Sigma}(\epsilon)\boldsymbol{V}(\epsilon)^\top$. A dot denotes differentiation at $\epsilon = 0$. Define

$$\boldsymbol{C} = \boldsymbol{U}_0^\top \boldsymbol{E} \boldsymbol{V}_0, \qquad \boldsymbol{\Omega}_U = \boldsymbol{U}_0^\top \dot{\boldsymbol{U}}, \qquad \boldsymbol{\Omega}_V = \boldsymbol{V}_0^\top \dot{\boldsymbol{V}}.$$

Differentiating $\boldsymbol{U}^\top \boldsymbol{U} = \boldsymbol{I}$ gives $\boldsymbol{\Omega}_U^\top = -\boldsymbol{\Omega}_U$. The same argument applies to $\boldsymbol{\Omega}_V$. Now differentiate the SVD and multiply on the left by $\boldsymbol{U}_0^\top$ and on the right by $\boldsymbol{V}_0$ to obtain

$$\boldsymbol{C} = \boldsymbol{\Omega}_U \boldsymbol{\Sigma}_0 + \dot{\boldsymbol{\Sigma}} - \boldsymbol{\Sigma}_0 \boldsymbol{\Omega}_V,$$

where $\dot{\boldsymbol{\Sigma}}$ is rectangular diagonal with entries $\dot{s}_i$. Since skew-symmetric matrices have zero diagonal, the two basis terms contribute nothing to C_{ii}. Hence

$$\dot{s}_i = C_{ii} = \boldsymbol{u}_{0,i}^\top \boldsymbol{E} \boldsymbol{v}_{0,i}, \qquad 1 \le i \le q.$$

Step 2. Show that the other coordinates describe basis motion. Holding the singular values fixed leaves $\boldsymbol{\Omega}_U \boldsymbol{\Sigma}_0 - \boldsymbol{\Sigma}_0 \boldsymbol{\Omega}_V$, the tangent directions to the orbit in singular coordinates. These directions have zero diagonal. Conversely, for $i \neq j \le q$, the equations for any pair of off-diagonal entries are

$$C_{ij} = (\Omega_U)_{ij} s_{0,j} - s_{0,i} (\Omega_V)_{ij}, \qquad C_{ji} = -(\Omega_U)_{ij} s_{0,i} + s_{0,j} (\Omega_V)_{ij}.$$

Solving this two-variable system gives

$$(\Omega_U)_{ij} = \frac{s_{0,j} C_{ij} + s_{0,i} C_{ji}}{s_{0,j}^2 - s_{0,i}^2}, \qquad (\Omega_V)_{ij} = \frac{s_{0,i} C_{ij} + s_{0,j} C_{ji}}{s_{0,j}^2 - s_{0,i}^2}.$$

The denominators are nonzero because the singular values are distinct. If $m > n$, the extra rows satisfy $C_{aj} = (\Omega_U)_{aj} s_{0,j}$ for $a > q$. If $n > m$, the extra columns satisfy $C_{ib} = -s_{0,i} (\Omega_V)_{ib}$ for $b > q$. These equations are solvable because each $s_{0,i} > 0$. Thus every coordinate outside the first q diagonal entries can be produced by a change of orthogonal bases.

Step 3. Separate the normal and tangent components. The matrices $\boldsymbol{u}_{0,i} \boldsymbol{v}_{0,i}^\top$ are Frobenius-orthonormal because their pairwise inner products are $(\boldsymbol{u}_{0,i}^\top \boldsymbol{u}_{0,j})(\boldsymbol{v}_{0,i}^\top \boldsymbol{v}_{0,j}) = \delta_{ij}$. By Step 2 they span the orthogonal complement of the tangent space. The normal projection of $\boldsymbol{E}$ is therefore

$$\boldsymbol{E}_N = \sum_{i=1}^{q} \dot{s}_i \boldsymbol{u}_{0,i} \boldsymbol{v}_{0,i}^\top, \qquad \boldsymbol{E}_T = \boldsymbol{E} - \boldsymbol{E}_N.$$

The remainder $\boldsymbol{E}_T$ is tangent and changes only the bases to first order. Orthogonality gives

$$\|\boldsymbol{E}\|_{\mathrm{F}}^2 = \|\boldsymbol{E}_N\|_{\mathrm{F}}^2 + \|\boldsymbol{E}_T\|_{\mathrm{F}}^2 = \|\dot{\boldsymbol{s}}\|_2^2 + \|\boldsymbol{E}_T\|_{\mathrm{F}}^2.$$

Step 4. Average the first-order spectral energy. Let $\boldsymbol{z}_i = \mathrm{vec}(\boldsymbol{u}_{0,i}\boldsymbol{v}_{0,i}^\top)$, so $\|\boldsymbol{z}_i\|_2 = 1$ and $\dot{s}_i = \boldsymbol{z}_i^\top \mathrm{vec}(\boldsymbol{E})$. The assumed second moment yields

$$\mathbb{E}[\dot{s}_i^2] = \boldsymbol{z}_i^\top \mathbb{E}[\mathrm{vec}(\boldsymbol{E})\,\mathrm{vec}(\boldsymbol{E})^\top]\boldsymbol{z}_i = \frac{\|\boldsymbol{z}_i\|_2^2}{mn} = \frac{1}{mn}.$$

Summing these q equal contributions gives

$$\mathbb{E}\|\dot{\boldsymbol{s}}\|_2^2 = \frac{q}{mn} = \frac{1}{\max(m,n)}.$$

Since $\|\boldsymbol{E}\|_{\mathrm{F}} = 1$, Step 3 gives $\mathbb{E}\|\boldsymbol{E}_T\|_{\mathrm{F}}^2 = 1 - 1/\max(m,n)$ for the complementary tangent energy.

Step 5. Pass from the derivative to d_S/d_W. The base norms cancel as in Proposition 3.1, and $\|\epsilon\boldsymbol{E}\|_{\mathrm{F}} = |\epsilon|$. Thus, for $\epsilon \neq 0$,

$$\left(\frac{d_S}{d_W}\right)^2 = \frac{\|\boldsymbol{s}(\boldsymbol{W}_0 + \epsilon\boldsymbol{E}) - \boldsymbol{s}_0\|_2^2}{\epsilon^2}.$$

With $\boldsymbol{W}_0$ fixed, positivity and the nonzero singular-value gaps give the expansion

$$\boldsymbol{s}(\boldsymbol{W}_0 + \epsilon\boldsymbol{E}) - \boldsymbol{s}_0 = \epsilon\dot{\boldsymbol{s}} + O(\epsilon^2),$$

whose remainder is bounded uniformly over $\|\boldsymbol{E}\|_{\mathrm{F}} = 1$. Substitution into the ratio therefore gives $(d_S/d_W)^2 = \|\dot{\boldsymbol{s}}\|_2^2 + O(|\epsilon|)$ with a uniform error. Taking expectations and applying Step 4 gives

$$\lim_{\epsilon\to 0} \mathbb{E}\left[\left(\frac{d_S}{d_W}\right)^2\right] = \mathbb{E}\|\dot{\boldsymbol{s}}\|_2^2 = \frac{1}{\max(m,n)}.$$

Taking the square root proves the stated RMS limit. □

Corollary E.1 (Isotropic fixed-rank perturbations)**.** *Let $\boldsymbol{W}_0$ satisfy Proposition 3.2 and let $\boldsymbol{E} = \boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{Q}^\top/\|\boldsymbol{\Lambda}\|_{\mathrm{F}}$, where $\boldsymbol{\Lambda} \in \mathbb{R}^{r\times r}$ is fixed, nonzero diagonal and $\boldsymbol{P} \in \mathbb{R}^{m\times r}, \boldsymbol{Q} \in \mathbb{R}^{n\times r}$ are independent uniform orthonormal frames, with $1 \le r \le q$. Then Equation (3) holds.*

Proof. **Step 1. Check the norm.** Since $\boldsymbol{P}^\top\boldsymbol{P} = \boldsymbol{Q}^\top\boldsymbol{Q} = \boldsymbol{I}_r$, multiplication by these frames preserves the Frobenius norm of $\boldsymbol{\Lambda}$. Hence $\|\boldsymbol{E}\|_{\mathrm{F}} = 1$.

Step 2. Compute the second moment. Uniformity and independence of the frames make the distribution of $\boldsymbol{E}$ unchanged by any fixed left or right orthogonal multiplication. In particular, a row sign flip shows $\mathbb{E}[E_{ab}E_{cd}] = 0$ when $a \neq c$. When $a = c$ but $b \neq d$, a column sign flip gives the same conclusion. Row and column permutations make all $\mathbb{E}[E_{ab}^2]$ equal. Their sum is $\mathbb{E}\|\boldsymbol{E}\|_{\mathrm{F}}^2 = 1$, so each equals $1/(mn)$. Consequently,

$$\mathbb{E}[\mathrm{vec}(\boldsymbol{E})\,\mathrm{vec}(\boldsymbol{E})^\top] = \frac{\boldsymbol{I}_{mn}}{mn}.$$

Step 3. Apply Proposition 3.2. Steps 1 and 2 verify its two assumptions on $\boldsymbol{E}$, so both conclusions in Equation (3) follow. □

Corollary E.2 (Pooling matrices)**.** *For a finite collection of matrices satisfying the isotropic assumptions of Proposition 3.2, let $\boldsymbol{W}_{*,j} = \boldsymbol{W}_{0,j} + \epsilon a_j \boldsymbol{E}_j$, with fixed $a_j \ge 0$ and $\sum_j a_j^2 > 0$. The pooled displacements satisfy*

$$\lim_{\epsilon\to 0} \mathbb{E}\left[\left(\frac{d_S}{d_W}\right)^2\right] = \frac{\sum_j a_j^2/\max(m_j,n_j)}{\sum_j a_j^2}.$$

Proof. **Step 1. Write the pooled ratio.** Let $\boldsymbol{s}_{*,j}$ and $\boldsymbol{s}_{0,j}$ be the singular-value vectors of the adapted and base matrices. The common base-norm normalization cancels, while $\|\boldsymbol{E}_j\|_{\mathrm{F}} = 1$ fixes each update norm. Therefore

$$\left(\frac{d_S}{d_W}\right)^2 = \frac{\sum_j \|\boldsymbol{s}_{*,j} - \boldsymbol{s}_{0,j}\|_2^2}{\sum_j \|\epsilon a_j \boldsymbol{E}_j\|_{\mathrm{F}}^2} = \frac{\sum_j \|\boldsymbol{s}_{*,j} - \boldsymbol{s}_{0,j}\|_2^2}{\epsilon^2 \sum_j a_j^2}.$$

Step 2. Expand each numerator. Write $\dot{\boldsymbol{s}}_j$ for the spectral derivative at $\boldsymbol{W}_{0,j}$ in direction $\boldsymbol{E}_j$. The uniform expansion from Proposition 3.2 gives, for fixed a_j,

$$\frac{\mathbb{E}\|\boldsymbol{s}_{*,j} - \boldsymbol{s}_{0,j}\|_2^2}{\epsilon^2} = a_j^2 \mathbb{E}\|\dot{\boldsymbol{s}}_j\|_2^2 + O(|\epsilon|) = \frac{a_j^2}{\max(m_j, n_j)} + O(|\epsilon|).$$

When $a_j = 0$, the corresponding term is identically zero.

Step 3. Sum and take the limit. There are finitely many matrices and $\sum_j a_j^2$ is fixed and positive. Substituting Step 2 into Step 1 yields

$$\lim_{\epsilon \to 0} \mathbb{E}\left[\left(\frac{d_S}{d_W}\right)^2\right] = \frac{\sum_j a_j^2 / \max(m_j, n_j)}{\sum_j a_j^2}.$$

Only linearity of expectation is used, so independence between matrices is unnecessary. The result is an average of the individual squared-ratio scales, weighted by their update energies a_j^2. □

Under the stated isotropic assumptions, the inverse-dimension scale explains how an additive update can have small relative spectral displacement. This motivates the empirical geometry measurements, while restoration tests the functional role of the spectral changes.