Learning-Forgetting Optimality in Supervised Finetuning: A Cliff Perspective

Albert Catalan-Tatjer ALBERT.CATALAN-TATJER@TUE.ELLIS.EU
ELLIS Tübingen, Max Planck Institute for Intelligent Systems, Tübingen AI Center

Jonas Geiping JONAS@TUE.ELLIS.EU
ELLIS Tübingen, Max Planck Institute for Intelligent Systems, Tübingen AI Center

Abstract

Supervised finetuning (SFT) of pretrained language models trades off acquisition of new domain capabilities against retention of prior knowledge. Increasingly, recent works suggest that post-training quantization (PTQ) and forgetting from SFT are seen as a loss geometry problem, where flatness leads to lower degradation. In this work, we adopt a unified view of post-training perturbations. In particular, inspired by PTQ we propose **Scale Invariant Balancing (SIB)** a functionally equivalent parameterization of the model that flattens the loss landscape within the weight-space symmetries. Moreover, we extensively characterize the learning-forgetting trade-off of plain SFT, SIB, and various classical continual learning methods to find that, across models and methods, two regimes universally arise. Either baseline SFT performance appears as a gradual trade-off between learning and forgetting, in which case SIB can be applied to improve Pareto optimality. Alternatively, SFT trajectories develop into a sharp *cliff*: a sharp phase transition where training recipes flip from learning without forgetting into catastrophic forgetting without improvements in learning, in which case continual learning methods do not substantially intervene.

1. Introduction

Modern large language models are the result of a sequence of stages: pre-training, mid-training, supervised finetuning (SFT), and increasingly some form of reinforcement learning from human or verifiable feedback [21, 29]. The sequential nature of post-training induces a continual learning setting, wherein each successive stage has to balance learning a new domain while retaining previously learned behaviors. The central problem that arises when continually learning in neural networks however, is catastrophic forgetting [20], which remains a core challenge in modern post-training.

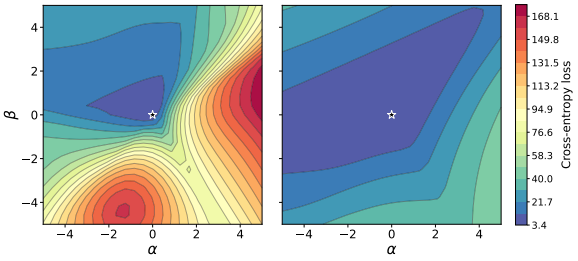


Figure 1: **Effect of Scale Invariant Balancing on the loss landscape.** Minimizing $||\nabla_W \mathcal{L}||_2^2$ on a scale-symmetry group, essentially stretches and shrinks directions proportional to their sensitivity to weight perturbations, relative to the symmetry.

Recent works point to the local geometry of the previous task loss landscape as a key factor to post-training *forgetting*. Catalan-Tatjer et al. [5] link flatness to post-training quantization (PTQ) degradation and Watts et al. [28] to forgetting under finetuning. From this perspective, a model is the joint product of its data, architecture, and training recipe, where the resulting weight-space geometry

shapes how gracefully it can absorb a new optimization stage. Interestingly, this connection hints that techniques that enable low-bit quantization may generalize to aid performance retention in other post-training perturbations.

In this paper, we investigate this unexplored connection. Inspired by Liu et al. [17] we leverage weight-space symmetries to minimize the norm of the gradient of the loss on a *retain dataset*. In doing so, we find a functionally equivalent parameterization of the model that minimizes sharpness w.r.t. a retain set. Notably, this reparameterization is a one-time step that requires no modification to the SFT pipeline, making it, in principle, compose with continual learning methods.

Finally, we analyze how baseline SFT, SIB, and classical continual learning methods, reshape the trade-off between domain adaptation and performance retention, by sweeping the learning rates and studying the Pareto-frontier. Our contributions are as follows:

1. We bridge the gap between post-training quantization and finetuning to propose **SIB**, a functionally equivalent reparameterization of any model that minimizes sharpness of the loss landscape on a chosen retention set.

2. We analyze the effect of SIB on the local geometry of the loss across different model families and sizes, confirming that SIB effectively minimizes $||\nabla_W \mathcal{L}||_2^2$.

3. We evaluate extensively baseline SFT, SIB, and other continual learning methods, where we uncover two regimes that universally arise during SFT: gradual trade off between learning and forgetting, where continual learning methods improve the Pareto frontier, or a sharp *cliff*: a sharp phase transition where training recipes flip from learning without forgetting into catastrophic forgetting without improvements in learning, in which case continual learning methods do not substantially intervene.

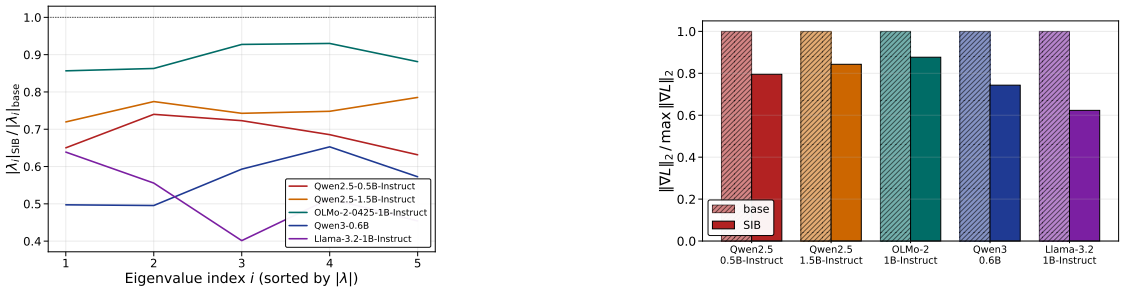
2. Related work

We focus on three threads of prior work that bear directly on this trade-off in modern LLMs: the structure of contemporary post-training pipelines, recent evidence tying forgetting to local loss geometry around pre-trained weights, and continual learning methods that attempt to mitigate or redirect that loss, many of which can be read as implicit approximations of this geometry. A comprehensive discussion of related work is provided in Section A.

3. Evaluating the Pareto frontier

Backbones. Generally, open-weights models provide a base and instruction-tuned version. We choose instruction-tuned models result from all post-training stages and are heavily optimized for benchmarks like GSM8K [12]. We work with models up to 8B parameters of the Qwen2.5, Qwen3, Llama-3.2, Llama-3.1, OLMo2 [11, 22, 23, 29] model families.

Learning. The domain adaptation task is **ChemL3**. This dataset is composed of the chemistry level-3 split from Feng et al. [8], a multiple-choice question answer dataset containing relatively niche chemistry questions, that is generally unknown to all of the backbones that we test. We refer to Section B for SFT implementation details.



(*a*) **Top 5 eigenvalues of the Hessian.** (*b*) **Norm of the gradient of the loss.**

Figure 2: **Loss geometry effects of SIB.** We observe that SIB reduces the magnitude of the top 5 by magnitude sharpest Hessian directions (left), as well as $||\nabla_W \mathcal{L}||_2^2$.

Forgetting. To measure retention, we measure performance degradation on GSM8K [6]. As mathematical reasoning is learned through substantial effort in pretraining and it represents a sufficiently complex proxy of task retention. Given that GSM8K is also evaluated through verification of model generations, it is further preferable over task retention measurements solely through multiple choice benchmarks, which do not measure whether generative capabilities have been retained. In all of our experiments and baseline models, we observe that GSM8K performance is uncorrelated to ChemL3, providing a clear learning-forgetting trade-off. Therefore, we choose the retain set $\mathcal{D}_R$ to be GSM8K's train split throughout.

4. Scale Invariant Balancing (SIB)

We introduce Scale Invariant Balancing (SIB) by first characterizing scale symmetries in open-weight language models, then we detail how to leverage them for a flatter loss landscape.

4.1. Common scale-symmetries in language models.

Weight-space symmetries have been used for improving model merging [1], accelerated optimization [31], and reduced quantization degradation [17]. We focus on *scale symmetries*: transformations $W \mapsto W/\alpha, W' \mapsto W' \operatorname{diag}(\alpha)$ on a pair of adjacent weight tensors, with $\alpha \in \mathbb{R}^d_{>0}$, that leave the network's input–output map exactly unchanged. We restrict our analysis to V/O and SwiGLU, more details and scale-symmetries that arise in modern transformers can be found in Appendix C.1.

- **V/O (per head):** $O \operatorname{diag}(\alpha) \operatorname{diag}(\alpha)^{-1} V = OV$, so we can rescale rows of V and columns of O within each head.
- **SwiGLU [25] (per neuron):** scaling rows of W_{up} by α and columns of W_{down} by $1/\alpha$ leaves $W_{\text{down}}\big(\sigma(W_{\text{gate}}x) \odot (W_{\text{up}}x)\big)$ unchanged, since the α factors out of the element-wise product.

4.2. Minimizing the norm of the gradient

Now, let $(W_{\text{div}}, W_{\text{mul}})$ be a scale-symmetric pair of tensors, and $\mathcal{D}_R$ a retention dataset whose loss landscape we want to flatten, we define the follow objective to minimize the gradient of $\mathcal{L}(\mathcal{D}_R)$,

$$\|g_{\text{div}}/\alpha\|^2 + \|\alpha\, g_{\text{mul}}\|^2, \qquad g_{\text{div}} = \nabla_{W_{\text{div}}} \mathcal{L}(\mathcal{D}_R),\ g_{\text{mul}} = \nabla_{W_{\text{mul}}} \mathcal{L}(\mathcal{D}_R), \tag{1}$$

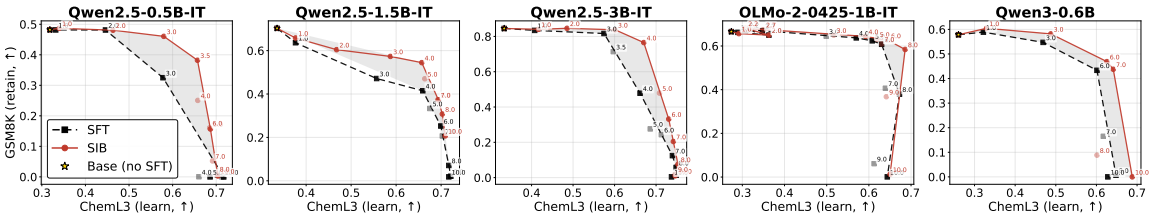


Figure 3: **Gradual learning-forgetting Pareto trade off**. We show the settings for which there is a gradual balance between learning and forgetting. Here, SIB dominates the Pareto frontier, but only moves it upward, 'sharpening' the cliff.

which admits the closed-form solution

$$\alpha^* = \left(\|g_{\text{div}}\|_2^2 / \|g_{\text{mul}}\|_2^2\right)^{1/4}, \qquad \alpha^* \in [1/R,\ R], \tag{2}$$

clamped to R (we use R=100 unless stated otherwise). The solution at the "balance" between $\|g_{\text{mul}}\|_2^2$ and $\|g_{\text{div}}\|_2^2$ gives name to this method.

Conceptually, this stretches the directions of higher weight sensitivity (as indicated by $\nabla_W \mathcal{L}(\mathcal{D}_R)$) and compresses the more robust, thereby protecting $\mathcal{L}(\mathcal{D}_R)$ from SFT weight updates. Conveniently, SIB is performed once between learning stages, to then proceed with vanilla SFT. Find more details as well as the step-by-step algorithm of SFT pipeline with SIB described in Algorithm 1.

5. Results and discussion

In this section, we characterize two regimes that arise in the Pareto trajectories of SFT. Firstly, gradual learning-forgetting trade off. Secondly, cliffs, i.e. sharp phase transitions from learning and not forgetting to catastrophic forgetting for no better learning. We report and discuss the role of continual learning methods in both.

5.1. Geometry effect of SIB

We show in Figure 2(*b*)subfigure the squared gradient norms $||\nabla\mathcal{L}(\mathcal{D}_R)||_2^2$ normalized by their maximum. Figure 2(*b*)subfigure confirms SIB reduces the gradient norm, lowering gradient norm values between $\tilde{1}$0pp and 35pp. Similarly, we study the 5 largest-by-magnitude eigenvalues of the hessian of $\mathcal{L}(\mathcal{D}_R)$ in Figure 2(*a*)subfigure [30]. Analyzing the ratio between the i-th base eigenvalue and its SIB counterpart, we see that it is lower than < 1, confirming that this notion of sharpness has also been decreased.

5.2. Learning-forgetting Pareto frontiers

To fabricate Pareto trajectories, we perform SFT with different learning rates. As expected, larger SFT weight updates result in more learning and more forgetting, the study of more model families and sizes reveals a more nuanced picture. In fact, we identify two Pareto regimes that emerge.

Gradual trade-off We begin by presenting the regime where there is a gradual exchange between performance acquisition and retention. In Figure 3 we exemplify this setting, showing plain SFT as the dashed black line. In this case, SIB (red) parameterizations approach the optimal corner by

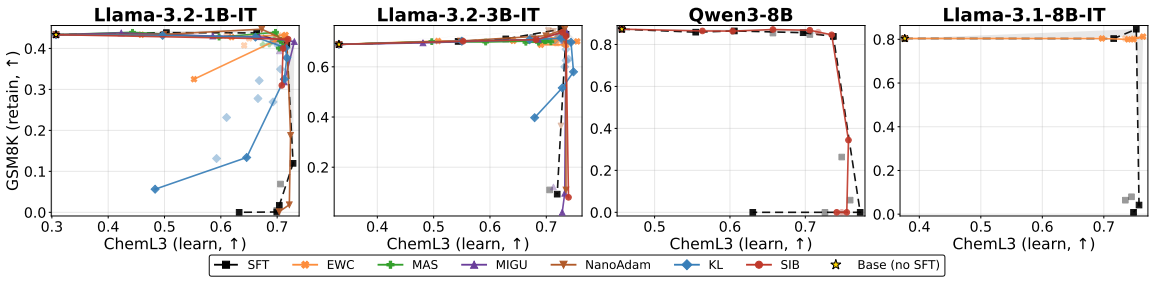


Figure 4: **Cliff regime**. These are some settings for which learning rates progress learning with minimal forgetting, until a sharp phase transition where there is minimal learning gain at catastrophic forgetting. We see that continual learning methods fail to substantially intervene in this situation.

retaining up to 35pp GSM8K accuracy (Qwen2.5-3B). Moreover, it is apparent that SIB does not substantially intervene in improving learning.

Cliffs Surprisingly, we identify a second regime. The sharp transition between learning and not forgetting, and catastrophically forgetting for no more learning draws a cliff-like contour. We show these *cliffs* in Figure 4, where SFT already approximates Pareto optimality. In this case, SIB and other continual learning methods (see Appendix D for method descriptions and hyperparameter sweeps). fail to offer better learning, indicating that these modulate forgetting rather than learning. So far, the results reported had fixed three epochs of SFT, we show in Figure 7 that in SFT with more epochs, and low learning rates cliffs also arise, indicating that the cliff regime can arise even with no continual learning method. Find the effect of the number of SFT epochs, LoRA, and more in Appendix E.

6. Conclusion

A unified perspective of post-training perturbations, opens up a stream of ideas from post-training quantization. In this work, we borrow ideas from Liu et al. [17], to leverage weight-space symmetries for a flatter parameterization of the base model. Because SIB preserves the network's function while modifying only the sharpness of the loss landscape, it serves as evidence that flatness directly mitigates catastrophic forgetting. Moreover, we uncover two regimes that emerge during SFT: one in which learning and forgetting are gradually traded-off, where SIB Pareto-dominates vanilla SFT, and a second regime where SFT already draws a cliff, i.e. approaches the optimal corner.

Overall, these findings point to two practical takeaways. First, if appropriately tuned, continual-learning techniques in the broad sense, including smaller learning rates, KL penalties, and explicit regularization, are employed, then forgetting can be almost entirely avoided for all large language models we investigate. The question of a trade-off between learning and forgetting is reframed to the question of *to what learning rate and hyperparameter setting can the model learn while not forgetting?* Second, across all methods we find surprisingly limited evidence for a successful increase in plasticity in LLMs, with no method substantially outperforming the baseline sweep of supervised finetuning in learning the new domain. Future work may show whether methods that truly increase learnability can be developed.

References

[1] Samuel K. Ainsworth, Jonathan Hayase, and Siddhartha Srinivasa. Git re-basin: Merging models modulo permutation symmetries. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://arxiv.org/abs/2209.04836.

[2] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget, 2018. URL https://arxiv.org/abs/1711.09601.

[3] Project Apertus, Alejandro Hernández-Cano, Alexander Hägele, Allen Hao Huang, Angelika Romanou, Antoni-Joan Solergibert, Barna Pasztor, Bettina Messmer, Dhia Garbaya, Eduard Frank Ďurech, and Ido Hakimi et al. Apertus: Democratizing open and compliant llms for global language environments, 2025. URL https://arxiv.org/abs/2509.14233.

[4] Dan Biderman, Jacob Portes, Mohammad Pezeshki, Georgi Froumusuzov, Mikel Artetxe, Sam Sanders, Jerry Tang, Joel Hestness, Mohammad Milani, et al. Lora learns less and forgets less. *arXiv preprint arXiv:2405.09673*, 2024.

[5] Albert Catalan-Tatjer, Niccolò Ajroldi, and Jonas Geiping. Training dynamics impact post-training quantization robustness, 2026. URL https://arxiv.org/abs/2510.06213.

[6] 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. *arXiv preprint arXiv:2110.14168*, 2021.

[7] Wenyu Du, Shuang Cheng, Tongxu Luo, Zihan Qiu, Zeyu Huang, Ka Chun Cheung, Reynold Cheng, and Jie Fu. Unlocking continual learning abilities in language models, 2024. URL https://arxiv.org/abs/2406.17245.

[8] Kehua Feng, Keyan Ding, Weijie Wang, Xiang Zhuang, Zeyuan Wang, Ming Qin, Yu Zhao, Jianhua Yao, Qiang Zhang, and Huajun Chen. Sciknoweval: Evaluating multi-level scientific knowledge of large language models, 2024.

[9] Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization, 2021. URL https://arxiv.org/abs/2010.01412.

[10] Gemma Team. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025. URL https://arxiv.org/abs/2503.19786.

[11] Aaron Grattafiori and Llama Team. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. URL https://arxiv.org/abs/2407.21783.

[12] Aryan Gulati, Brando Miranda, Eric Chen, Emily Xia, Kai Fronsdal, Bruno de Moraes Dumont, and Sanmi Koyejo. Putnam-AXIOM: A functional & static benchmark for measuring higher level mathematical reasoning in LLMs. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=kqj2Cn3Sxr.

[13] 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. *arXiv preprint arXiv:2106.09685*, 2021.

[14] Damjan Kalajdzievski. Scaling laws for forgetting when fine-tuning large language models, 2024. URL https://arxiv.org/abs/2401.05605.

[15] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017.

[16] Jiacheng Lin, Zhongruo Wang, Kun Qian, Tian Wang, Arvind Srinivasan, Hansi Zeng, Ruochen Jiao, Xie Zhou, Jiri Gesi, Dakuo Wang, Yufan Guo, Kai Zhong, Weiqi Zhang, Sujay Sanghavi, Changyou Chen, Hyokun Yun, and Lihong Li. Sft doesn't always hurt general capabilities: Revisiting domain-specific fine-tuning in llms, 2026. URL https://arxiv.org/abs/2509.20758.

[17] Zechun Liu, Changsheng Zhao, Igor Fedorov, Bilge Soran, Dhruv Choudhary, Raghuraman Krishnamoorthi, Vikas Chandra, Yuandong Tian, and Tijmen Blankevoort. Spinquant: Llm quantization with learned rotations. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://arxiv.org/abs/2405.16406.

[18] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations (ICLR)*, 2017. URL https://arxiv.org/abs/1608.03983.

[19] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019. URL https://arxiv.org/abs/1711.05101.

[20] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. *Psychology of learning and motivation*, 24:109–165, 1989.

[21] Team Olmo, :, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, and Hamish Ivison et al. Olmo 3, 2026. URL https://arxiv.org/abs/2512.13961.

[22] OLMo Team. 2 OLMo 2 furious. *arXiv preprint arXiv:2501.00656*, 2025. URL https://arxiv.org/abs/2501.00656.

[23] Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024. URL https://arxiv.org/abs/2412.15115.

[24] Mark Rofin, Aditya Varre, and Nicolas Flammarion. (how) learning rates regulate catastrophic overtraining, 2026. URL https://arxiv.org/abs/2604.13627.

[25] Noam Shazeer. Glu variants improve transformer, 2020. URL https://arxiv.org/abs/2002.05202.

[26] Chao-Hong Tan, Qian Chen, Wen Wang, Yukun Ma, Chong Zhang, Chong Deng, Qinglin Zhang, Xiangang Li, and Jieping Ye. Fggm: Fisher-guided gradient masking for continual learning, 2026. URL https://arxiv.org/abs/2601.18261.

[27] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca, 2023.

[28] Ishaan Watts, Catherine Li, Sachin Goyal, Jacob Mitchell Springer, and Aditi Raghunathan. Sharpness-aware pretraining mitigates catastrophic forgetting. In *ICLR 2026 Workshop on Geometry-grounded Representation Learning and Generative Modeling*, 2026. URL https://openreview.net/forum?id=B2qTJi5s0M.

[29] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, and Bo Zheng et al. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

[30] Zhewei Yao, Amir Gholami, Kurt Keutzer, and Michael Mahoney. Pyhessian: Neural networks through the lens of the hessian, 2020. URL https://arxiv.org/abs/1912.07145.

[31] Bo Zhao, Nima Dehmamy, Robin Walters, and Rose Yu. Symmetry teleportation for accelerated optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. URL https://arxiv.org/abs/2205.10637.

[32] Chunyuan Zhou, Thomas Jacobs, Advait Gadhikar, and Rebekka Burkholz. Pay attention to small weights. *arXiv preprint arXiv:2506.21374*, 2025.

Appendix A. Related work

Sequential finetuning of a pre-trained model forces a choice between absorbing a new task and preserving what was learned before, a tension that has been studied in many forms since McCloskey and Cohen [20] first articulated catastrophic forgetting. We focus on two threads of prior work that bear directly on this trade-off in modern LLMs: the structure of contemporary post-training pipelines, and recent evidence tying forgetting to local loss geometry around pre-trained weights.

A.1. Post-training of language models

Modern LLMs are the result of a sequence of stages: pre-training, mid-training, instruction tuning, preference alignment and, increasingly, reasoning-oriented post-training, each targeting a different capability [3, 21, 29]. SFT is the principal tool for injecting domain expertise, typically followed by RL stages that further refine the model. RL stages typically address forgetting through KL penalties between the updated and reference policies and very small learning rates [3, 21, 22], treating the magnitude of post-training updates as a central lever for retention.

A.2. Forgetting in LLMs

Recent studies in the relationship between pre-training and post-training, show growing evidence that forgetting in LLMs is tied to the local geometry of the loss of the pre-trained weights. Watts et al. [28] establish a connection between the flatness of the loss landscape and forgetting during SFT [9]. Previously, Catalan-Tatjer et al. [5] showed that PTQ robustness is governed by the same training dynamics suggesting that **robustness to weight perturbations** is a unifying lens for both quantization and post-training forgetting. PTQ has been successfully exploiting weight-space geometry: Liu et al. [17] show that, although the network is invariant to certain rotations of its weights, different parameterizations within these symmetries differ substantially in quantization error, an observation we build on to shape sensitivity to SFT updates. Empirically, forgetting has been observed under naive finetuning [14], where the most consistent practical lever in current recipes remains conservative learning rates, which have been shown to limit feature drift [16, 24].

A.3. Continual learning

Continual learning offers a long-standing toolbox for the stability-plasticity trade-off that re-emerges in LLM post-training. **Replay-based** methods mix examples from earlier stages into the new training mixture, directly addressing forgetting when prior data is available. **Regularization-based** methods penalize movement away from the pre-trained weights, weighted by an estimate of parameter importance: EWC [15] uses the diagonal Fisher information, MAS [2] uses the sensitivity of the model output, and KL penalties on logits effectively regularize functional distance from the base model. From a sharpness perspective, these methods can be seen as proxies for the curvature of the retention loss around the pre-trained weights, slowing updates along directions to which the retained capabilities are most sensitive. *Gradient-projection* methods restrict updates to subspaces orthogonal to those used by previous tasks, while more recent rehearsal-free, task-label-free approaches such as MIGU [7] mask updates by the magnitude of layer outputs, and FGGM [26] replaces this heuristic with a Fisher-based parameter-importance estimate. In LLM-specific settings, parameter-efficient methods such as LoRA [4, 13] have been proposed as a way to bound the magnitude of weight changes. Our methods are most directly related to the data-aware regularization line: rather

than penalizing motion of individual weights, we use the gradient of the retention loss to (i) reparameterize the model within its scale symmetries to flatten the retention landscape and (ii) allocate per-parameter learning-rate scales inversely proportional to retention sensitivity, both of which can be viewed as data-aware, geometry-based mechanisms for guiding SFT updates.

Appendix B. SFT implementation details.

Following common practice [27], we implement a standard instruction tuning training pipeline, including prompt loss masking, where the loss is computed only on the response tokens and not on the prompt and standard instruction message formats for each model. All finetuning runs use AdamW [19] with weight decay 0.01. We use a cosine learning-rate schedule [18] that warms up for the first 20 steps and then decays the peak learning rate to $0.1\times$ its maximum value, with an additional 50 cool-down steps at the end of training. To map out the retention-learning Pareto frontier per model, we exhaustively sweep the peak learning rate and the number of epochs for all models and methods. Unless indicated otherwise, we default to 3 epochs of SFT.

Appendix C. Scale Invariant Balancing

C.1. Scale-symmetries

We extend the scale-symmetries explained in the main text to a broader set of symmetries commonly present in open-weight language models. We restrict our analysis to V/O and SwiGLU MLP pairs because, first, they are present in all architectures we study, while the RMSNorm symmetry is broken in Gemma 3 [10] and OLMo 2, which parameterize the gain as $(1+\gamma)$; the additive constant destroys the homogeneity that the symmetry requires. Second, the two tensors in each V/O or SwiGLU pair have comparable parameter counts, whereas γ is a vector and its consumer is a matrix, which complicates questions of stability and normalization across the pair.

- **V/O (per head):** $O\,\mathrm{diag}(\alpha)\,\mathrm{diag}(\alpha)^{-1}V = OV$, so we can rescale rows of V and columns of O within each head.
- **SwiGLU [25] (per neuron):** scaling rows of W_{up} by α and columns of W_{down} by $1/\alpha$ leaves $W_{\text{down}}\big(\sigma(W_{\text{gate}}x)\odot(W_{\text{up}}x)\big)$ unchanged, since the α factors out of the elementwise product.
- **RMSNorm / linear:** $\mathrm{RMSNorm}(x)_i = \gamma_i\, x_i/\|x\|_{\text{RMS}}$ is degree 1 in γ, so γ absorbs into the columns of any downstream linear sharing that norm: $\gamma \mapsto \gamma/\alpha$, $W_{\text{down}} \mapsto W_{\text{down}}\,\mathrm{diag}(\alpha)$.
- **Q/K with RoPE (per kv-head):** attention logits depend on $Q^\top K$, so $Q \mapsto \alpha Q$, $K \mapsto K/\alpha$ is invariant. RoPE's block-diagonal rotation forces α to be a single scalar per kv-head.

C.2. Algorithm

Appendix D. Continual Learning methods

We compare against representative continual-learning methods that modulate SFT updates with respect to a retain dataset $\mathcal{D}_R$. These fall into two groups: methods that add a regularizer anchoring weights to their pre-trained values, and methods that reshape the update itself.

Algorithm 1: SFT with Scale-Invariant Balancing (SIB), V/O + MLP symmetry pairs

Data: Pre-trained weights $\theta^\star$; fine-tuning data $\mathcal{D}_{\mathrm{FT}}$; retention set $\mathcal{D}_R$; set of scale-invariant pairs $\mathcal{P} = \mathcal{P}_{\mathrm{V/O}} \cup \mathcal{P}_{\mathrm{MLP}}$, where

- $\mathcal{P}_{\mathrm{V/O}} = \{(W_V^{(\ell)},\, W_O^{(\ell)})\}_{\ell=1}^{L}$ (per-layer value $\to$ output projection pair);
- $\mathcal{P}_{\mathrm{MLP}} = \{(W_{\mathrm{up}}^{(\ell)},\, W_{\mathrm{down}}^{(\ell)}),\, (W_{\mathrm{gate}}^{(\ell)},\, W_{\mathrm{down}}^{(\ell)})\}_{\ell=1}^{L}$ (gate/up rows paired with down-projection columns);

clamp ratio R; FT optimizer Opt (e.g., AdamW); base LR η; epochs E.
Result: Fine-tuned weights θ
$\theta \leftarrow \theta^\star$
`// Phase 1 --- Geometry teleportation (one-shot reparameterization)`
foreach $(W_{\mathrm{div}}, W_{\mathrm{mul}}) \in \mathcal{P}$ **do**
  Compute gradient norms on retention set:
  $g_{\mathrm{div}} \leftarrow \left\|\nabla_{W_{\mathrm{div}}} \mathcal{L}_R(\theta)\right\|_2$
  $g_{\mathrm{mul}} \leftarrow \left\|\nabla_{W_{\mathrm{mul}}} \mathcal{L}_R(\theta)\right\|_2$
  `// Minimize squared gradient norm w.r.t. scale` α
  $\alpha \leftarrow \mathrm{clip}\left(\sqrt{g_{\mathrm{div}}/g_{\mathrm{mul}}},\ 1/R,\ R\right)$
  $W_{\mathrm{div}} \leftarrow W_{\mathrm{div}}/\alpha$
  $W_{\mathrm{mul}} \leftarrow W_{\mathrm{mul}} \cdot \alpha$
  `// Functionally equivalent transformation`
end
`// Phase 2 --- Standard SFT from the rebalanced weights`
for epoch $= 1$ **to** E **do**
  foreach *minibatch* $b \sim \mathcal{D}_{\mathrm{FT}}$ **do**
    $\theta \leftarrow \mathrm{Opt}(\theta,\, \nabla_\theta \mathcal{L}_{\mathrm{FT}}(\theta; b),\, \eta)$
  end
end
return θ

Importance-weighted regularization. A first family augments the SFT loss with a quadratic penalty that anchors each parameter to its pre-trained value $\theta_p^\star$, weighted by a per-parameter importance estimate Ω_p computed once on $\mathcal{D}_R$:

$$\mathcal{L}(\theta) \;=\; \mathcal{L}_{\mathrm{SFT}}(\theta) \;+\; \tfrac{\lambda}{2} \sum_p \Omega_p \,(\theta_p - \theta_p^\star)^2. \tag{3}$$

EWC [15] sets Ω_p to the diagonal of the empirical Fisher information, $\Omega_p = \mathbb{E}_{B\sim\mathcal{D}_R}\big[(\nabla_p \mathcal{L}(B))^2\big]$, while **MAS** [2] replaces it with the sensitivity of the model output, $\Omega_p = \mathbb{E}_{B\sim\mathcal{D}_R}\big[\,|\nabla_p \|f_\theta(B)\|^2|\,\big]$, which can be estimated without labels. Both can be read as second-order proxies for the curvature of the retention loss around $\theta^\star$, slowing motion along directions to which the retained capabilities are most fragile, and connecting naturally to the sharpness perspective discussed in Section A.2.

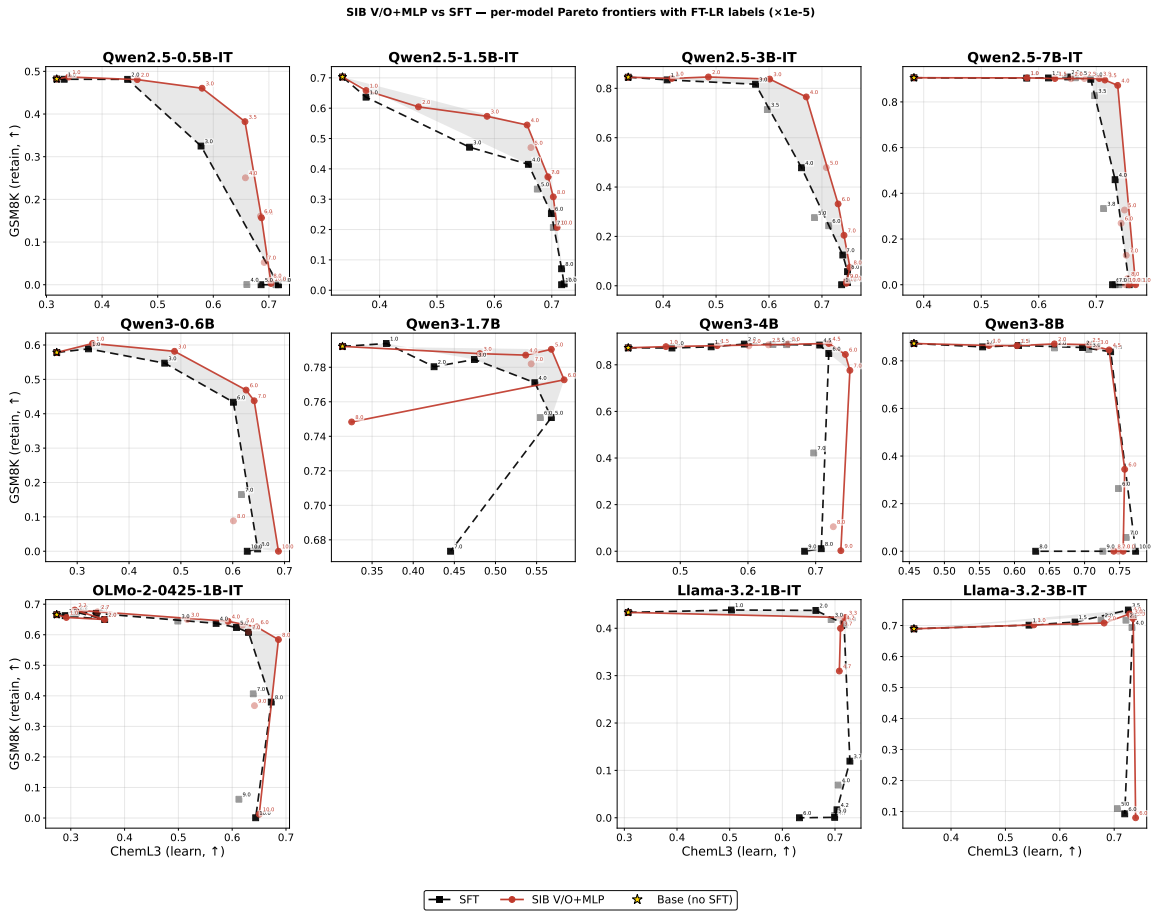


Figure 5: **Complete Pareto trajectories of SIB and SFT**. The text next to each point indicates its learning rate $\times 10^{-}5$.

Update-shaping methods. A second family does not modify the loss but reshapes the SFT update itself. **MIGU** [7] is a data-free approach that, at each step, computes the magnitude of the forward-pass activations and masks the gradient on parameters whose corresponding output activations fall below a per-layer percentile. **NanoAdam** [32] similarly keeps the standard Adam step but applies it only to the smallest-magnitude weights at each iteration, freezing large-magnitude weights on the assumption that they encode features acquired during pre-training. Unlike EWC and MAS, neither method makes use of $\mathcal{D}_R$, but modify the training using finetuning-time signals only, which makes them attractive when retain data is unavailable.

Appendix E. More results and discussion

Complete model grid for SFT and SIB In Figure 5 we show the comparison between SFT and SIB for the entire model grid that we have explored.

Complete model grid for continual learning methods We show the results for different continual learning methods for all of the model in Figure 6.

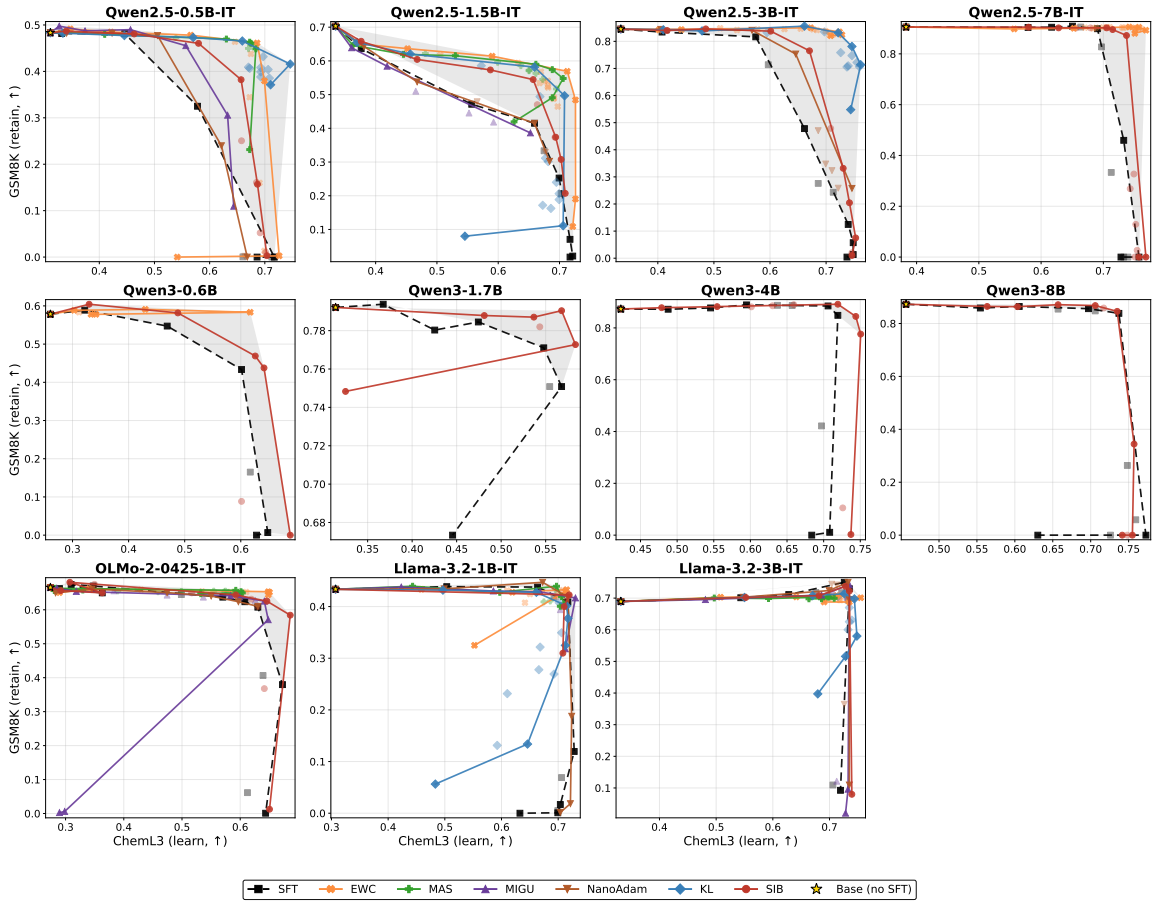


Figure 6: **Pareto optimality of continual learning methods**.

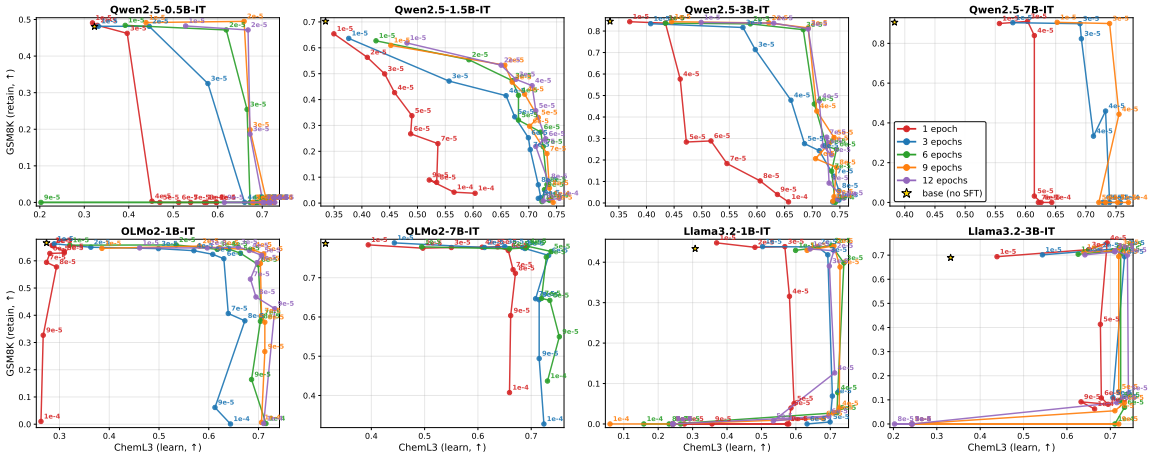


Figure 7: **Increasing the number of epochs creates cliffs.** We increase the number of epochs of SFT. More epochs increases the learning performance of small learning rates, which exhibit lower forgetting.

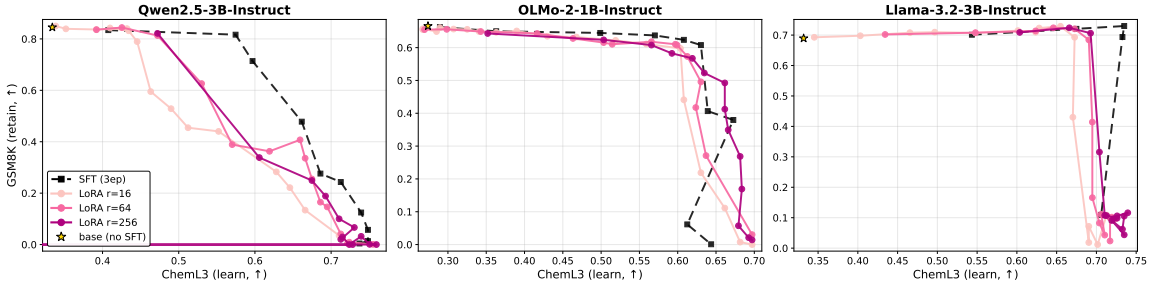


Figure 8: **Full finetuning Pareto-dominates LoRA**. We have found that LoRA provides a worse learning-forgetting trade-off than full finetuning across all model families and sizes, for most of the learning rates.

Number of epochs In Figure 7 we show the effect of increasing the number of epochs of SFT. We see that more epochs increases the learning performance of small learning rates, which exhibit lower forgetting.

LoRA We show in Figure 8 a comparison between LoRA and full-finetuning. We see that full-finetuning Pareto-dominates LoRA in all settings.

Model size The analysis of the learning–forgetting trade-off in Figure 9 enables a convenient comparison both across model sizes within a family and across families at comparable scales. In general, larger models exhibit a steeper trade-off, in which SFT often approaches the optimal top-right corner. Among the evaluated families, Qwen2.5 models display the most gradual trade-off during standard finetuning: all model sizes attain similar learning levels (0.5B and 7B models are within 4pp), where smaller variants sacrifice more GSM8K performance to achieve comparable learning. In contrast, for Llama3.X and OLMo2, ChemL3 accuracy is substantially more dependent on model size. In these families, ChemL3 performance mostly saturates before falling off the cliff

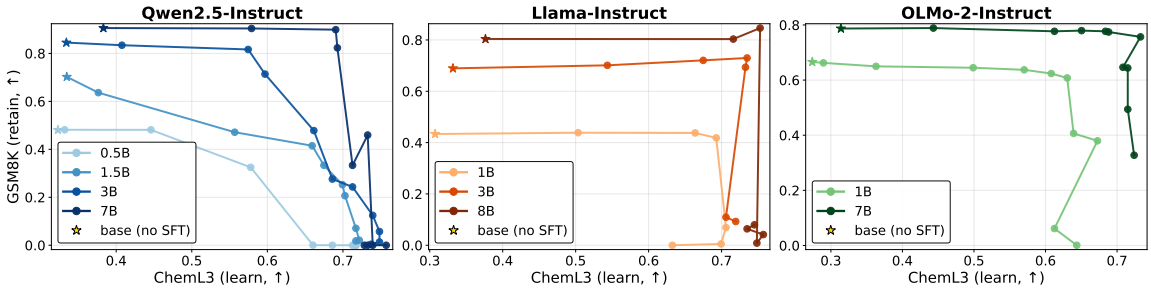


Figure 9: **Effect of model size and family on learning-forgetting trade-off.** Across model families and sizes, larger models generally show steeper trade-offs with SFT closer to the optimal regime. Qwen2.5 exhibits the most gradual trade-off with similar learning across sizes (< 4 pp difference), while Llama3.X and OLMo2 show stronger size dependence with early ChemL3 saturation and sharp GSM8K collapse.

where GSM8K accuracy collapses for no better learning. This is yet another data point of the exceptional fine-tunability of Qwen models.