Tracing the representation geometry of language models from pretraining to post-training

author names withheld

Under Review for the Workshop on High-dimensional Learning Dynamics, 2025

Abstract

The geometry of representations in a neural network can significantly impact downstream generalization. It is unknown how representation geometry changes in large language models (LLMs) over pretraining and post-training. Here, we characterize the evolving geometry of LLM representations using spectral methods (effective rank and eigenspectrum decay). With the OLMo and Pythia model families we uncover a consistent non-monotonic sequence of three distinct geometric phases in pretraining. An initial "warmup" phase sees rapid representational compression. This is followed by an "entropy-seeking" phase, characterized by expansion of the representation manifold's effective dimensionality, which correlates with an increase in memorization. Subsequently, a "compression-seeking" phase imposes anisotropic consolidation, selectively preserving variance along dominant eigendirections while contracting others, correlating with improved downstream task performance. We link the emergence of these phases to the fundamental interplay of cross-entropy optimization, information bottleneck, and skewed data distribution. Additionally, we find that in post-training the representation geometry is further transformed: Supervised Fine-Tuning (SFT) and Direct Preference Optimization (DPO) correlate with another "entropy-seeking" dynamic to integrate specific instructional or preferential data, reducing out-of-distribution robustness. Conversely, Reinforcement Learning with Verifiable Rewards (RLVR) often exhibits a "compression-seeking" dynamic, consolidating reward-aligned behaviors and reducing the entropy in its output distribution. This work establishes the utility of spectral measures of representation geometry for understanding the multiphase learning dynamics within LLMs.

1. Introduction

Loss curves during training offer an incomplete account of how large language models (LLMs) learn specific behaviors [8, 38]. While training loss typically decreases monotonically [16, 18], model capabilities and internal representational structures exhibit significant qualitative shifts [5, 31, 32]. This disconnect highlights a fundamental challenge: *How do high-dimensional distributed representations within LLMs evolve during training, and how do these representational transformations give rise to emergent capabilities?*

Here, we use spectral analyses to quantify the representation geometry in LLMs. We focus on the spectral properties of the last token's feature covariance matrix, computing effective rank ("RankMe") and spectral decay rate ("αReQ") to measure variance concentration, indicative of representational compression or expansion [1, 10]. These measures have been linked to generalization and learnability via gradient descent [2, 35]. Our analysis shows that LLM pretraining unfolds through a consistent sequence of distinct geometric phases (c.f. Figure 1):

- An initial **"warmup"** phase: rapid representational collapse coinciding with LR ramp-up.

© .

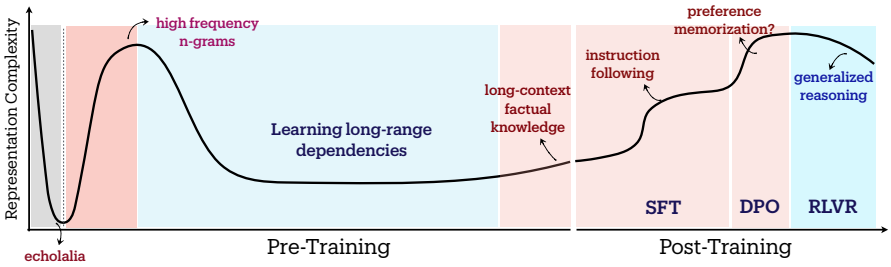

Figure 1: Spectral geometry reveals evolving representation complexity: Pretraining shows non-monotonic "entropy-seeking" (expansion for n-grams) and "compression-seeking" (consolidation for long-range dependencies) phases. Post-training (SFT, DPO, RLVR) further refines the geometry, most notably RLVR is "compression-seeking" (details in Figure A5)

- An **"entropy-seeking"** phase: manifold expansion, increased n-gram memorization.
- A **"compression-seeking"** phase: anisotropic consolidation, correlating with enhanced learning of long-range dependencies and robust generalization.

Our investigation of post-training stages shows analogous geometric shifts: Supervised Fine-Tuning (SFT) and Direct Preference Optimization (DPO) often induce an "entropy-seeking" -like manifold expansion for specific instruction assimilation, while Reinforcement Learning from Verifiable Rewards (RLVR) typically produces a "compression-seeking" -like contraction, consolidating reward-aligned behaviors. These findings offer a granular view of LLM training with practical implications.

2. Methods

2.1. Spectral Analysis of Representation Geometry

To quantitatively measure representation geometry, we perform spectral analysis of the feature covariance matrix $\hat{\Sigma}$. This matrix is derived from the last-layer representations $\mathbf{y}_N$ of the final token t_N from input sequences. The eigenspectrum $\{\sigma_i(\hat{\Sigma})\}_{i=1}^{d}$ of $\hat{\Sigma}$ measures information concentration along principal axes; a sharp decay indicates anisotropic geometry (compression), while a slow decay suggests significant variance across several directions (expansion). We use two key metrics:

- **Effective Rank (`RankMe`)**: Based on Von Neumann entropy, $\texttt{RankMe} := \exp(-\sum_{i=1}^{d} p_i \ln p_i)$, where $p_i = \sigma_i / (\sum_j \sigma_j)$ is the variance proportion along the i-th principal axis [10, 29].
- **Spectral Decay Rate (αReQ)**: LLM activation matrices often exhibit a powerlaw eigenvalue spectrum, i.e. $\sigma_i \propto i^{-\alpha\text{ReQ}}$ [11]. A smaller αReQ implies higher dimensionality, while larger αReQ indicates compactness [1, 34].

2.2. Quantifying Distributional Memorization

To differentiate distributional memorization from generalization, we measure alignment with n-gram frequencies in the pretraining corpus $\mathcal{D}$ using an ∞-gram language model [21]. The distributional memorization metric, Mem_∞, is the Spearman rank correlation (ρ_s) between ∞-gram LM outputs and LLM outputs for a target sequence y [36], given context u and input x:

$$Mem_\infty(LLM, \mathcal{D}, \mathcal{T}) := \rho_s \left(\bar{P}_{\infty,\mathcal{D}}(y|u \oplus x), \bar{P}_{LLM}(y|u \oplus x) \right). \quad (1)$$

2.3. Post-Training Methodologies and Evaluation

We analyze models undergoing: **Supervised Fine-Tuning (SFT)**, which adapts LLMs by minimizing negative log-likelihood on a curated dataset $\mathcal{D}_{\text{SFT}}$ of instruction-response pairs, and evaluate performance on In-Distribution (ID) and Out-of-Distribution (OOD) benchmarks [33]; **Direct Preference Optimization (DPO)** [27], which refines policy π_θ using a static preference dataset $\mathcal{D}_{\text{pref}} = \{(x, y_w, y_l)\}$ by minimizing a loss dependent on the log-ratio of probabilities $\hat{r}_\theta(x, y)$ against a reference policy π_{ref}; and **Reinforcement Learning from Verifiable Rewards (RLVR)** [20, 30], which optimizes π_θ to maximize expected cumulative reward $J(\theta)$ from verifiable properties of LLM outputs, with problem-solving efficacy evaluated using pass@k [6, 19, 40].

3. Experimental Setup

Our analysis relies on publicly available checkpoints from two primary model suites: the **Pythia Suite** (70M to 12B parameters) [3] and the **OLMo Framework** (e.g., OLMo-7B, OLMo-2 7B, 1B) [14, 20, 22]. The availability of numerous intermediate checkpoints from these suites is crucial for tracking training dynamics. For post-training analysis, we also examined Tülu-3.1 models [20] (which are LLaMA-based). Further details on architectures and training are provided in Appendix A.

4. Probing the Representation Geometry of Language Models

4.1. Phases of Pretraining: Non-Monotonic Changes in Representation Geometry

While standard pretraining metrics like loss typically decrease near-monotonically, offering limited insight into capability development, we find that representation geometry metrics show significant non-monotonic changes that correlate with downstream performance. Figure 2 illustrates this: measuring `RankMe` [10] and αReQ [1] on last-layer last-token representations (FineWeb dataset [24]), we identify three distinct pretraining phases:

- A **"warmup"** phase: rapid representational collapse during learning rate ramp-up.
- An **"entropy-seeking"** phase: characterized by manifold expansion.
- A **"compression-seeking"** phase: anisotropic consolidation along principal eigenvectors.

These phases are consistently observed across OLMo-2 and Pythia models and scales (Figure 2C-F), indicating robust non-monotonic geometric evolution.

4.2. Role of Learning Objective and Optimization in Learning Dynamics (Toy Model)

Having identified the distinct learning phases using the spectral geometry metrics, we now seek to understand the role of loss and optimization frameworks used in LLM pretraining in engendering these phases. Specifically, we studied the dynamics imposed by gradient descent while optimizing the cross-entropy loss in an analytically-tractable setting: the model $\mathrm{f}_\theta(x)$ is linear, i.e. $\mathrm{f}_\theta(x) = \theta x \in \mathcal{R}^d$, and logits are obtained (like in LLMs) as $z = W\mathrm{f}_\theta(x) \in \mathcal{R}^{|\mathcal{V}|}$. The outputs are obtained by applying a softmax operation on z (see Figure 3A). Extending the results of Pezeshki et al. [25], we found two key properties of gradient descent that contribute to the emergent geometric properties of the $\mathrm{f}_\theta(x)$

- **Primacy bias**: f_θ and W corresponding to high-frequency tokens are learned earlier in training.

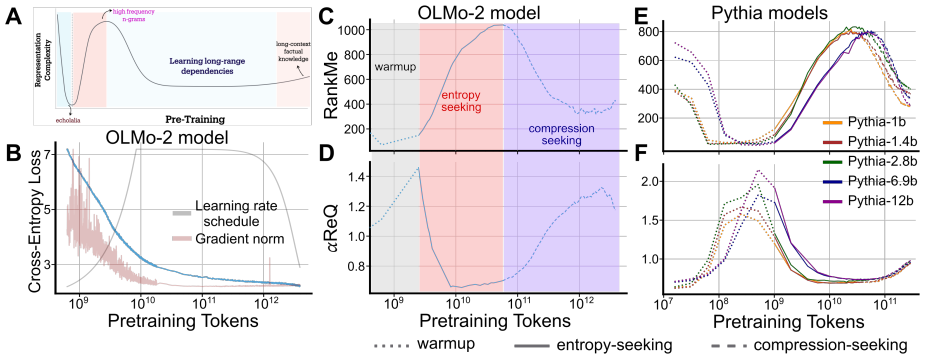

Figure 2: **Loss decreases monotonically, but representation geometry does not. (A)** Schematic from Fig 1, for the pretraining stage. **(B)** Cross-entropy loss, gradient norm and learning rate schedule during OLMo-2 7B model pretraining. **(C, D)** `RankMe` and αReQ, respectively, for OLMo-2 7B model vary non-monotonically across pretraining, demonstrating three key phases: "warmup" , "entropy-seeking" , and "compression-seeking" . **(E, F)** Same as C,D, but for Pythia models, demonstrating the consistent existence of the three phases across model families and scales.

- **Selection bias**: Under information bottleneck conditions, dominant directions in f_θ are more likely to be used for encoding new information.

We demonstrate (c.f. Figure 3) that two conditions are necessary for replicating the multiphase learning dynamics in our toy-model, as observed within LLMs: (1) non-uniform class distribution, and (2) information bottleneck ($d < |\mathcal{V}|$). These conditions are commonplace in LLM pretraining.

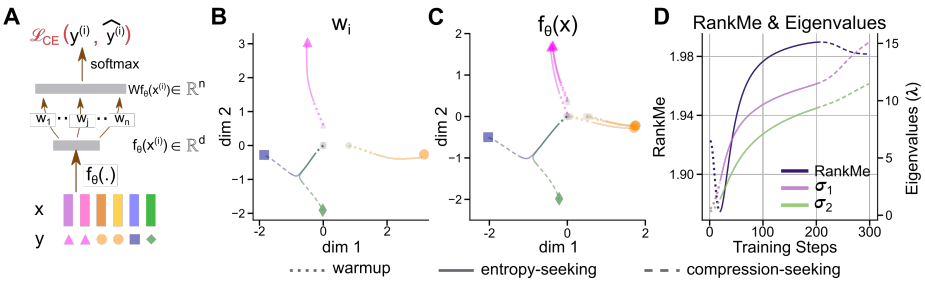

Figure 3: **Learning dynamics of cross-entropy loss replicate multiphase learning dynamics (Toy Model). (A)** Schematic of a model with feature extractor $f_\theta(\in \mathbb{R}^d)$, linear classifier $W(\in \mathbb{R}^{n\times d})$ and cross-entropy loss $\mathcal{L}_{CE}$. Skewed class distribution and information bottleneck ($d < n$) are critical. **(B, C)** W_i and $f_\theta(x)$ demonstrate distinctive trajectories analogous to "warmup" (dotted), "entropy-seeking" (solid), and "compression-seeking" (dashed) phases. **(D)** Quantitative spectral metrics RankMe and eigenvalues, σ_1, σ_2.

4.3. Representation Geometry Changes During Post-Training Stages

Post-training refines LLM capabilities and representation geometry. Analyzing Tülu-3.1 models [37] (LLaMA-3.1-8B base [13]) through SFT, DPO, and RLVR show distinct geometric shifts (Figure 4A).

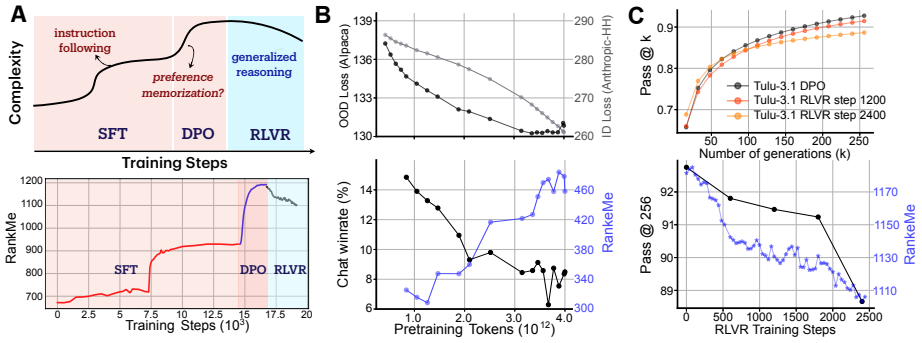

Figure 4: **Post-training induces distinct geometric transformations in model representations, aligned with specific behavioral changes.** **(A)** Conceptual overview of post-training (top), corresponding $RankMe$ metrics from Llama-3.1-Tülu-3.1-8B (bottom). **(B)** Impact of pretraining on OLMo-2-1B SFT (Anthropic-HH): (top) longer pretraining improves ID performance, while OOD generalization (Alpaca farm) saturates; (bottom) *Overtrained* models show distinct outputs after SFT on different datasets. **(C)** RLVR post-training narrows Llama-3.1-8B-Tülu-3-DPO's exploratory behavior on AMC-23 (e.g., at $k = 256$).

SFT & DPO exhibit "entropy-seeking" : SFT correlates with a monotonic `RankMe` increase (Figure 4A, bottom). This expansion aids instruction memorization for in-distribution (ID) examples but can reduce out-of-distribution (OOD) robustness, evidenced by SFT on OLMo2-1B showing improved ID loss but increased OOD loss with more base model pretraining (Figure 4B, top). The DPO objective is analogous to the contrastive visual learning loss function, and unsurprisingly demonstrates an increase in representation complexity [12, 41].

RLVR exhibits "compression-seeking" : RLVR, conversely, is associated with a monotonic `RankMe` decrease (Figure 4A, bottom). This "**"compression-seeking"** " correlates with constrained exploration; on the AMC-23 math benchmark (Figure 4C), RLVR (2400 steps) excels at `pass@16`, but the base DPO model performs better at `pass@256`, suggesting RLVR amplifies existing capabilities rather than broadening exploration [40, 42].

5. Discussion

Geometry of Pretraining: Memorization vs Generalization. We show that LLM pretraining is multiphasic, primarily characterized by **"entropy-seeking"** and **"compression-seeking"** phases. These geometric phases offer a quantitative framework to examine the interplay between memorizing short-context statistics (n-gram memorization during the **"entropy-seeking"** phase) and generalizing long-context information (promoted by the structured manifold of the **"compression-seeking"** phase). This geometric refinement aligns with phenomena like *grokking*.

Geometry of Post-Training: Alignment vs Exploration. Different post-training recipes induce distinct geometric shifts. Supervised Fine-Tuning (SFT) and DPO typically drive an **"entropy-seeking"** dynamic, expanding the representational manifold for specific instruction-response examples. This SFT manifold expansion is consistent with lazy-regime learning [28] and can improve in-distribution performance at the risk of overfitting. In contrast, Reinforcement Learning from Verifiable Rewards (RLVR) promotes a "compression-seeking" dynamic, refining representations towards reward-aligned directions.

References

[1] Kumar K Agrawal, Arnab Kumar Mondal, Arna Ghosh, and Blake Richards. α-req: Assessing representation quality in self-supervised learning by measuring eigenspectrum decay. *Advances in Neural Information Processing Systems*, 35:17626–17638, 2022.

[2] Peter L Bartlett, Philip M Long, Gábor Lugosi, and Alexander Tsigler. Benign overfitting in linear regression. *Proceedings of the National Academy of Sciences*, 117(48):30063–30070, 2020.

[3] Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pages 2397–2430. PMLR, 2023.

[4] Sid Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, et al. Gpt-neox-20b: An open-source autoregressive language model. *arXiv preprint arXiv:2204.06745*, 2022.

[5] Davis Brown, Charles Godfrey, Nicholas Konz, Jonathan Tu, and Henry Kvinge. Understanding the inner-workings of language models through representation dissimilarity. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

[6] 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. *arXiv preprint arXiv:2107.03374*, 2021.

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PmLR, 2020.

[8] Deep Ganguli, Danny Hernandez, Liane Lovitt, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova Dassarma, Dawn Drain, Nelson Elhage, et al. Predictability and surprise in large generative models. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, pages 1747–1764, 2022.

[9] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

[10] Quentin Garrido, Randall Balestriero, Laurent Najman, and Yann Lecun. Rankme: Assessing the downstream performance of pretrained self-supervised representations by their rank. In *International Conference on Machine Learning*, pages 10929–10974. PMLR, 2023.

[11] Arna Ghosh, Arnab Kumar Mondal, Kumar Krishna Agrawal, and Blake Richards. Investigating power laws in deep representation learning. *arXiv preprint arXiv:2202.05808*, 2022.

[12] Arna Ghosh, Kumar Krishna Agrawal, Shagun Sodhani, Adam Oberman, and Blake Richards. Harnessing small projectors and multiple views for efficient vision pretraining. *Advances in Neural Information Processing Systems*, 37:39837–39868, 2024.

[13] 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. *arXiv preprint arXiv:2407.21783*, 2024.

[14] Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, et al. Olmo: Accelerating the science of language models. *arXiv preprint arXiv:2402.00838*, 2024.

[15] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 297–304. JMLR Workshop and Conference Proceedings, 2010.

[16] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, pages 30016–30030, 2022.

[17] Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, 2017.

[18] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

[19] Sumith Kulal, Panupong Pasupat, Kartik Chandra, Mina Lee, Oded Padon, Alex Aiken, and Percy S Liang. Spoc: Search-based pseudocode to code. *Advances in Neural Information Processing Systems*, 32, 2019.

[20] Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, et al. T\" ulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*, 2024.

[21] Jiacheng Liu, Sewon Min, Luke Zettlemoyer, Yejin Choi, and Hannaneh Hajishirzi. Infini-gram: Scaling unbounded n-gram language models to a trillion tokens. In *First Conference on Language Modeling*, 2024.

[22] Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, et al. 2 olmo 2 furious. *arXiv preprint arXiv:2501.00656*, 2024.

[23] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[24] Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. The fineweb datasets: Decanting the

web for the finest text data at scale. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024. URL https://openreview.net/forum?id=n6SCkn2QaG.

[25] Mohammad Pezeshki, Oumar Kaba, Yoshua Bengio, Aaron C Courville, Doina Precup, and Guillaume Lajoie. Gradient starvation: A learning proclivity in neural networks. *Advances in Neural Information Processing Systems*, 34:1256–1272, 2021.

[26] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

[27] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741, 2023.

[28] Yi Ren and Danica J Sutherland. Learning dynamics of llm finetuning. *arXiv preprint arXiv:2407.10490*, 2024.

[29] Olivier Roy and Martin Vetterli. The effective rank: A measure of effective dimensionality. In *2007 15th European signal processing conference*, pages 606–610. IEEE, 2007.

[30] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

[31] Aaditya Singh, Stephanie Chan, Ted Moskovitz, Erin Grant, Andrew Saxe, and Felix Hill. The transient nature of emergent in-context learning in transformers. *Advances in Neural Information Processing Systems*, 36:27801–27819, 2023.

[32] Sidak Pal Singh, Bobby He, Thomas Hofmann, and Bernhard Schölkopf. Hallmarks of optimization trajectories in neural networks: Directional exploration and redundancy. *arXiv preprint arXiv:2403.07379*, 2024.

[33] Jacob Mitchell Springer, Sachin Goyal, Kaiyue Wen, Tanishq Kumar, Xiang Yue, Sadhika Malladi, Graham Neubig, and Aditi Raghunathan. Overtrained language models are harder to fine-tune. *arXiv preprint arXiv:2503.19206*, 2025.

[34] Carsen Stringer, Marius Pachitariu, Nicholas Steinmetz, Matteo Carandini, and Kenneth D Harris. High-dimensional geometry of population responses in visual cortex. *Nature*, 571 (7765):361–365, 2019.

[35] Vimal Thilak, Chen Huang, Omid Saremi, Laurent Dinh, Hanlin Goh, Preetum Nakkiran, Joshua M Susskind, and Etai Littwin. Lidar: Sensing linear probing performance in joint embedding ssl architectures. *arXiv preprint arXiv:2312.04000*, 2023.

[36] Xinyi Wang, Antonis Antoniades, Yanai Elazar, Alfonso Amayuelas, Alon Albalak, Kexun Zhang, and William Yang Wang. Generalization v.s. memorization: Tracing language models'

capabilities back to pretraining data. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=IQxBDLmVpT.

[37] Yizhong Wang, Ximing Li, Siqi Wang, Yash Khandwala, Aashi Anand, Yushi Yang, Sewon Lee, Hannaneh Hajishirzi, and Noah A Smith. Tülu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2404.07810*, 2024.

[38] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. *Transactions on Machine Learning Research*, 2022.

[39] Johannes Welbl, Nelson F Liu, and Matt Gardner. Crowdsourcing multiple choice science questions. *arXiv preprint arXiv:1707.06209*, 2017.

[40] Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in llms beyond the base model? *arXiv preprint arXiv:2504.13837*, 2025.

[41] Runtian Zhai, Bingbin Liu, Andrej Risteski, J. Zico Kolter, and Pradeep Kumar Ravikumar. Understanding augmentation-based self-supervised representation learning via RKHS approximation and regression. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=Ax2yRhCQr1.

[42] Rosie Zhao, Alexandru Meterez, Sham Kakade, Cengiz Pehlevan, Samy Jelassi, and Eran Malach. Echo chamber: Rl post-training amplifies behaviors learned in pretraining. *arXiv preprint arXiv:2504.07912*, 2025.

Appendix A. Experimental Setup and Model Details

This section provides detailed information about the datasets and model suites used in our study, as summarized in the main paper.

A.1. Studied Model Suites: Pythia & OLMo

This work analyzes checkpoints from two publicly released model suites:

- **Pythia Suite** [3]: Developed by EleutherAI, this suite consists of models ranging from 70M to 12B parameters, all trained on the Pile dataset [9] using the same data ordering and hyperparameters across scales. This allows for controlled study of scaling effects. Architecturally, they are based on the GPT-NeoX [4] design, featuring parallel attention/FFN layers and RoPE. Checkpoints are available at various training steps (e.g., 0, 1k, 2k, ..., 143k steps, where one step processes 2 million tokens).
- **OLMo Framework** [14, 20, 22]: Developed by AI2, OLMo provides a suite of truly open models (including training code, data, logs, and weights). We focus on the OLMo-7B model (and potentially others in the suite) trained on AI2's Dolma dataset. Architecturally, OLMo

makes specific choices like SwiGLU activation, RMSNorm, no biases in linear layers, and RoPE. Checkpoints are available at regular intervals throughout its 2.5T token training run. The OLMo-2 series, such as OLMo-2 7B and 1B models, were trained for approximately 4T tokens.

A.2. Post-Training Models: Tülu-3.1

For analyzing post-training stages, we utilized checkpoints from the Tülu-3.1 models developed by AI2 [20, 37]. These are LLaMA-based models (specifically LLaMA-3.1-8B [13]) that have undergone a sequential three-stage post-training recipe: Supervised Fine-Tuning (SFT), Direct Preference Optimization (DPO), and Reinforcement Learning with Verifiable Rewards (RLVR). We analyzed checkpoints from all post-training stages of these models.

A.3. Dataset for Spectral Analysis During Pretraining

Unless otherwise specified for a particular experiment (e.g., downstream task evaluations), the spectral metrics (`RankMe`, αReQ) during pretraining were measured by processing sequences from the FineWeb dataset [24].

Appendix B. Detailed Methodologies

This section provides an expanded description of the methodologies summarized in the main paper.

B.1. Detailed Spectral Analysis, Matrix Entropy, and Effective Rank

Last token representations in autoregressive language models: A rigorous understanding of LLM capabilities necessitates a precise characterization of the *geometry of their learned representations*. An autoregressive language model processes an input sequence of discrete tokens $\mathbf{s} = (t_1, t_2, \ldots, t_N)$, transforming each token t_k through its l layers (conditioned on preceding tokens $t_{<k}$) into a sequence of high-dimensional continuous vectors $\mathbf{f}_\theta^{(l)}(t_k|t_{<k})$. For autoregressive models, the representation of the final token (t_N) at the last layer, $\mathbf{y}_N \coloneqq \mathbf{f}_\theta^{(L)}(t_N|t_{<N})$, is particularly pivotal. Its significance stems from different factors: (i) it directly parameterizes the predictive distribution for the subsequent tokens $P(t_{N+1}|t_1, ..., t_N)$; (ii) it synthesizes information from the entire context $t_{\leq N}$ (or $t_{<N}$) to inform this prediction, meaning it inherently reflects the model's capacity for contextual understanding; and (iii) is often used as input to task-specific layers in downstream applications.

High-dimensional representation complexity metrics: To quantitatively measure representation geometry, we perform spectral analysis of the feature covariance matrix. Given a set of M input sequences, we form a feature matrix $\mathbf{F} \in \mathbb{R}^{M \times d}$; each row is a feature vector of the last token $\mathbf{y}_N$ for each input. Assuming the features are centered, the empirical covariance matrix is $\hat{\Sigma} \coloneqq \frac{1}{M}\mathbf{F}^T\mathbf{F}$. The eigenspectrum of $\hat{\Sigma}$, denoted by eigenvalues $\{\sigma_i(\hat{\Sigma})\}_{i=1}^d$, measures the concentration of information along the principal axes of variation. The distribution of $\{\sigma_i\}_{i=1}^d$ provides a quantitative description of feature geometry: a sharp decay indicates information compressed in a lower-dimensional subspace (anisotropic geometry), while a slow decay indicates a high-dimensional subspace is utilized.

This spectral perspective motivates using *matrix entropy* to measure the uniformity of the eigenvalue distribution. If $p_i = \sigma_i/(\sum_j \sigma_j)$ is the proportion of variance along the i-th principal

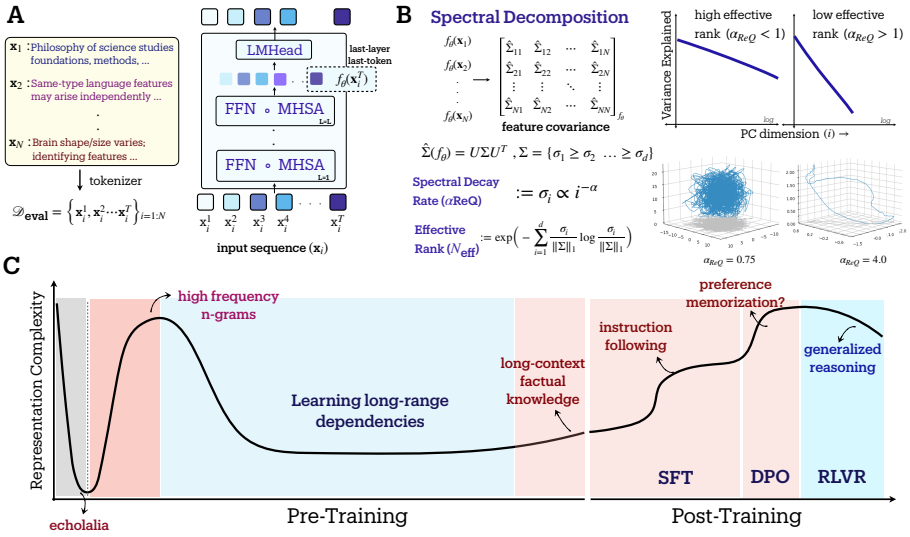

Figure A5: **Spectral framework of LLM feature geometry during training. (A)** LLM learning analyzed with empirical feature covariance $\hat{\Sigma}(f_\theta)$ of last-token representations $f_\theta(x_i)$. **(B)** Representation geometry quantified by covariance spectral properties: spectral decay rate (αReQ) for variance concentration, and effective rank (RankMe) for utilized dimensionality. **(C)** Spectral geometry reveals evolving representation complexity. Pretraining shows non-monotonic phases: post-echolalia, an "entropy-seeking" phase expands capacity (e.g., for recapitulating n-grams), then a "compression-seeking" phase consolidates representations for long-range dependencies. Post-training stages (SFT, DPO, RLVR) further refine geometry for instruction following and complex reasoning

axis, the Von Neumann *entropy-based effective rank* [10, 29] is defined as:

$$\texttt{RankMe} := \exp\left(S(\hat{\Sigma})\right) = \exp\left(-\sum_{i=1}^{d} p_i \ln p_i\right) \quad \in (0, d]. \tag{A2}$$

Low entropy indicates a skewed eigenvalue distribution, i.e. low-dimensional (anisotropic) representations, while high entropy implies a uniform spread, i.e. high-dimensional (isotropic) representations.

Our empirical studies also show that LLM activation matrices exhibit *heavy-tailed* eigenvalue spectra, i.e., a power law distribution where $\sigma_i \propto i^{-\alpha\text{ReQ}}$, where $\alpha\text{ReQ} > 0$ [11]. Slower decay or smaller αReQ implies a more uniform spread of σ_i's (higher dimensional), and thus higher $S(\hat{\Sigma})$ and RankMe. Conversely, faster decay or larger αReQ implies representations are compactly packed along fewer principal directions [1, 34], yielding lower entropy and smaller RankMe. αReQ and RankMe thus provide related metrics of representation geometry, though unlike RankMe, αReQ does not change with the model's feature dimensionality, d.

B.2. Detailed Quantification of Distributional Memorization

To dissect how LLMs utilize their pretraining corpus $\mathcal{D}$, we differentiate *distributional memorization*, i.e. how aligned are LLM output probabilities with n-gram frequencies in $\mathcal{D}$, from *distributional*

generalization, i.e. LLM capabilities beyond such statistics [21]. To quantify the alignment with n-gram statistics, we use the ∞-gram language model (LM) which uses the largest possible value of n for predicting the next token probability. Briefly, an ∞-gram LM can be viewed as a generalized version of an n-gram LM which starts with $n = \infty$, and then performs backoff till the n-gram count in $\mathcal{D}$ is non-zero [21]. Consequently, the output probability of the ∞-gram LM for each token is dependent on its longest existing prefix in $\mathcal{D}$.

The distributional memorization metric is defined as the spearman rank correlation (ρ_s) between the ∞-gram LM outputs and the LLM outputs for all tokens in a target sequence [36]. Formally, consider a concatenated sequence of instructions, u, question, x and target, y, from a question-answering task, $\mathcal{T}$. Then, the distributional memorization is computed as:

$$Mem_{\infty}(LLM, \mathcal{D}, \mathcal{T}) \coloneqq \rho_s \left(\bar{P}_{\infty,\mathcal{D}}(y|u \oplus x), \bar{P}_{LLM}(y|u \oplus x)\right) \tag{A3}$$

where $\bar{P}_{\cdot}(y|u \oplus x) \coloneqq \prod_{t_i \in y} P_{\cdot}(t_i|u \oplus x \oplus y_{[t_0:t_{i-1}]})$ denotes the joint likelihood of all tokens in y and $P_{\cdot}()$ is the next token prediction distribution, as described above.

B.3. Detailed Post-Training Methodologies and Evaluation

Supervised Fine-Tuning (SFT) adapts pre-trained LLMs by further training on a curated dataset $\mathcal{D}_{\text{SFT}} = \{(x_i, y_i)\}_{i=1}^{N_{\text{SFT}}}$ typically consisting of instruction-response pairs. The standard objective is to minimize the negative log-likelihood of the target responses, effectively maximizing $P_\theta(y|x)$ for examples in $\mathcal{D}_{\text{SFT}}$. We evaluate the robustness of the SFT model by contrasting its performance on held-out examples from $\mathcal{D}_{\text{SFT}}$ (In-Distribution, ID) with its performance on examples from a related but distinct dataset $\mathcal{D}_{\text{OOD}}$ (Out-of-Distribution, OOD), which may vary in task, style, or complexity not present in $\mathcal{D}_{\text{SFT}}$ [33].

Direct Preference Optimization (DPO) [27] refines an LLM policy π_θ based on a static dataset of human preferences $\mathcal{D}_{\text{pref}} = \{(x, y_w, y_l)\}$, where the response y_w is preferred over y_l for prompt x. It directly optimizes for preference satisfaction by minimizing the loss:

$$\mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}_{\text{pref}}} \left[\log \sigma \left(\hat{r}_\theta(x, y_w) - \hat{r}_\theta(x, y_l)\right)\right], \tag{A4}$$

where $\hat{r}_\theta(x, y) = \beta \log(\pi_\theta(y|x)/\pi_{\text{ref}}(y|x))$ represents the implicit log-ratio of probabilities scaled by β against a reference policy π_{ref}, and $\sigma(.)$ is the logistic function.

Reinforcement Learning from Verifiable Rewards (RLVR), as applied in works like Lambert et al. [20] and Shao et al. [30], optimizes the LLM's policy π_θ to maximize the expected discounted cumulative reward, $J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T} \gamma^t R_t\right]$, where $\tau = (s_0, a_0, \ldots, s_T, a_T)$ is a trajectory generated by actions $a_t \sim \pi_\theta(\cdot|s_t)$ in states s_t, $\gamma \in [0, 1]$ is a discount factor, and $R_t = R(s_t, a_t)$ is the reward at time t. This optimization is typically performed using policy gradient algorithms (e.g., PPO). Critically, the reward R_t in RLVR is derived from verifiable properties of the LLM's outputs, e.g. correctness on mathematical problems or passing unit tests.

Performance with pass@k: To evaluate problem-solving efficacy and generative exploration, particularly for RLVR-tuned models, we employ the pass@k metric [19]. For a given problem, k independent responses are stochastically generated from the model; the problem is deemed solved if at least one response constitutes a verifiable solution. Since direct estimation of pass@k can exhibit high variance, we utilize the unbiased estimator [6, 40]:

$$\texttt{pass@k} = \mathbb{E}_{P_i}\left[1 - \frac{\binom{N-c_i}{k}}{\binom{N}{k}}\right] \tag{A5}$$

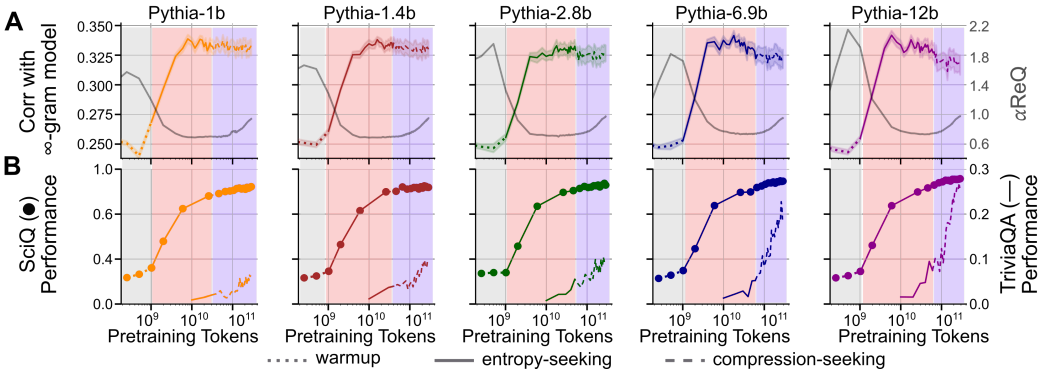

Figure A6: **Distinct learning phases are linked to different LLM capabilities. (A)** Memorization metric, i.e. spearman correlation between LLM and ∞-gram outputs, and representation geometry metric, αReQ, across Pythia models' (1–12B parameters) pretraining. Memorization peaks late in the "entropy-seeking" phase before plateauing or degrading slightly in the "compression-seeking" phase, suggesting that the former prioritizes capturing short-context n-gram statistics. **(B)** 0-shot performance on multiple-choice (SciQ) and factual question-answering (TriviaQA) tasks across pretraining. While accuracy on SciQ benefits from learning in both phases, accuracy on TriviaQA *groks* once the model learns long-context statistics, primarily in the "compression-seeking" phase.

where, N samples are generated for each problem P_i , and c_i denotes the count of correct solutions among them (parameters for this work are N=512 and k $\leq$ 256).

Appendix C. Supplementary Results and Analyses

This section provides additional details, figures, and discussions corresponding to the Results section of the main paper.

C.1. Memorization & beyond: Distributional memorization happens in entropy-seeking phase

In this section, we seek to associate the different geometric phases to specific LLM behaviors. Downstream tasks that test the LLM's factual reasoning and language understanding abilities seem to improve with more pretraining. However, it is unclear to what extent this increase is due to an improvement in the model's memorization ability, i.e. how good is the model in "regurgitating" short-context phrases from the pretraining dataset, as opposed to a general language understanding, i.e. leveraging long-context dependencies to generate reasonable output. We disentangle these two factors by using the distributional memorization metric Wang et al. [36] presented in eq. (A3) for Pythia models when processing sequences from the TriviaQA dataset [17].

Figure A6 illustrates the memorization metric and task performance over the course of pretraining for Pythia models of 5 different sizes – ranging from 1B to 12B. Across all models, the distributional memorization metric increased during the "entropy-seeking" phase and peaked towards the end of this phase. Intuitively, this result suggests that the "entropy-seeking" phase is particularly important for learning short-context statistics, e.g. high-frequency n-grams, present in the pretraining corpus. This intuition is also supported by Wang et al. [36] (c.f. Fig 12). Following this peak in the memorization

metric, it plateaued (or slightly decreased) during the "compression-seeking" phase, suggesting that the model's output in this phase is guided by factors beyond n-gram statistics. Notably, the 0-shot accuracy on multiple-choice question-answering tasks, e.g. SciQ [39], consistently improved throughout both the "entropy-seeking" and "compression-seeking" phases, potentially benefiting from both short- and long-context information learned in the respective phases.

However, 0-shot performance on factual question-answering tasks, e.g. TriviaQA [17], demonstrate a *grokking*-like behavior with the rise in accuracy closely aligned with the saturation of the memorization metric. Consequently, most of the improvement in task accuracy happens during the "compression-seeking" phase, potentially benefiting from the long-context statistics learned in this phase, which are crucial for this task. Taken together, these findings outline a distinct association between each phase and the emergence of different LLM capabilities: short-context n-gram modeling during the "entropy-seeking" phase and long-context information aggregation during the "compression-seeking" phase.

C.2. Role of Learning Objective and Optimization in Learning Dynamics (Toy Model)

Having demonstrated the existence and salience of distinct learning phases, we now seek to understand the role of loss and optimization frameworks used in LLM pretraining in engendering these phases. Specifically, we studied the gradient descent dynamics while optimizing the cross-entropy loss in an analytically-tractable setting — the model $\mathrm{f}_\theta(x)$ is linear, i.e. $\mathrm{f}_\theta(x) = \theta x \in \mathcal{R}^d$, and logits are obtained (like in LLM models) as $z = W\mathrm{f}_\theta(x) = W\theta x \in \mathcal{R}^{|\mathcal{V}|}$. The outputs are obtained by applying a softmax operation on z (see Figure A7A). We extended the results of Pezeshki et al. [25] to study how W and $\mathrm{f}_\theta(.)$ change when optimizing the loss using gradient descent. Notably, we found two key properties of gradient descent that contribute to the emergent geometric properties of the representation space

- **Primacy bias**: Representations and weights corresponding to high-frequency tokens are learned earlier in training.
- **Selection bias**: Dominant directions in the representation space are more likely to be used for encoding new information, i.e. $\Delta\sigma_i \propto \sigma_i$.

We demonstrate (c.f. Figure A7) that two conditions are necessary for replicating the multiphase learning dynamics in our toy-model, as observed within LLMs: (1) non-uniform class distribution, and (2) information bottleneck ($d < |\mathcal{V}|$). These conditions are common in LLM pretraining.

In this setup, $\mathrm{f}_\theta(.)$ and W for frequently-occurring classes separate during the initial "warmup" phase (Figure A7B, C, dotted lines), with alignment of weight and feature eigenvectors. An "entropy-seeking" phase follows, with volume expansion in $\mathrm{f}_\theta(.)$ and W spaces and increasing effective rank, leading to higher confidence for frequent classes (Figure A7B, C, solid lines). Subsequently, infrequent classes separate. Constrained by the bottleneck, the system reuses feature eigenvectors, encoding more information in dominant directions (σ_1 grows faster than σ_2, Figure A7D). This anisotropic encoding reduces `RankMe`, akin to the "compression-seeking" phase. These results suggest gradient-based cross-entropy optimization under these conditions can cause the observed non-monotonic geometric changes.

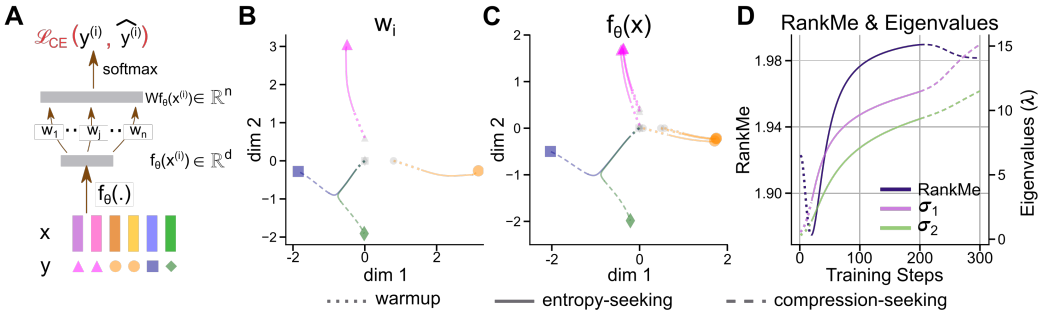

Figure A7: **Learning dynamics of cross-entropy loss replicate multiphase learning dynamics (Toy Model). (A)** Schematic of a model with feature extractor $f_\theta(\in \mathbb{R}^d)$, linear classifier $W(\in \mathbb{R}^{n\times d})$ and cross-entropy loss $\mathcal{L}_{CE}$. Skewed class distribution and information bottleneck ($d < n$) are critical. **(B, C)** Classifier weights (W_i) and feature representations ($f_\theta(x)$) demonstrate distinctive trajectories analogous to "warmup" (dotted), "entropy-seeking" (solid), and "compression-seeking" (dashed) phases. **(D)** Quantitative spectral metrics RankMe and eigenvalues, σ_1, σ_2.

C.3. Further Details on Post-Training Geometric Changes

SFT Details: The main paper discusses SFT exhibiting an "**"entropy-seeking"** " dynamic. This involves manifold expansion for instruction memorization on ID examples, potentially reducing OOD robustness (Figure 4B, top, in main paper). The chat winrates analysis (Figure 4B, bottom) for OLMo2-1B SFT on Anthropic-HH vs. Alpaca farm further shows that "overtrained" base models (higher pretraining `RankMe`) yield more distinguishable outputs on AlpacaEval, suggesting increased sensitivity to distribution shifts (e.g., winrate drop from 14% to 9%).

DPO as Contrastive Learning: The DPO loss (Equation A4) can be written as a Noise Contrastive Estimation (NCE) loss [15]:

$$\mathcal{L}_{DPO} = -\mathbb{E}_{x,y_w,y_l}\left[\log(\sigma(\hat{r}_\theta(x,y_w) - \hat{r}_\theta(x,y_l)))\right] = -\mathbb{E}_{x,y_w,y_l}\left[\log \frac{e^{\hat{r}_\theta(x,y_w)}}{e^{\hat{r}_\theta(x,y_w)} + e^{\hat{r}_\theta(x,y_l)}}\right] \quad \text{(A6)}$$

This formulation, with one "positive" (y_w) and one "negative" (y_l) example, is analogous to contrastive learning objectives in vision [7, 23, 26].

RLVR Details: For the RLVR analysis on the AMC-23 math benchmark (Figure 4C in main paper), Tülu-3.1-8B (post-DPO) and RLVR checkpoints (steps 1200, 2400; total 2440 steps) were used. `Pass@k` was evaluated for $k \in \{16, 32, \ldots, 256\}$. Better performance of the base model at higher k supports the idea of RLVR constraining exploration [40, 42].

Appendix D. Additional Discussion Points

D.1. Alternative Perspectives on Pretraining and Post-Training Geometry

Our spectral analysis reveals LLM pretraining as a structured sequence of geometric phases, not uniform optimization. An initial **"entropy-seeking"** phase expands dimensionality for local patterns (n-gram memorization), followed by a **"compression-seeking"** phase that reduces dimensionality,

packing information along dominant axes for long-range understanding. This suggests an iterative exploration-consolidation cycle. Post-training recipes also induce characteristic shifts. SFT's **"entropy-seeking"** dynamic expands the manifold for specific examples, enhancing ID performance but risking overfitting. RLVR's **"compression-seeking"** dynamic refines representations towards reward-aligned principles, potentially amplifying existing capabilities [42] by constraining to a structured subspace, thus reducing exploration [40].