HIDDEN-STATE GEOMETRY FOR REASONING SHARPENING VIA LAYER-WISE TUNING

Anonymous authors
Paper under double-blind review

ABSTRACT

A wrong answer can be informative about what a reasoning model still needs to learn. This is especially true when the model assigns almost the same score to a correct answer and an incorrect competitor, yet prefers the latter. We study these small-gap errors as a problem of learning a specific distinction. Candidate diagnostics show close competitions in the inspected pools, with incorrect predictions nearer the support boundary than correct predictions. This suggests using the competing answer to guide correction, rather than supplying the correct response alone. We introduce Layer-wise Tuning for Reasoning Sharpening (LTRS): select a hidden layer on held-out candidate data, build a fixed correct-minus-wrong direction from training pairs, and train a pairwise hidden margin alongside an anchored answer-learning objective. The comparison is internal, but its purpose is practical: improve the returned answer without adding a hidden-space selector at inference. Mathematical and medical evaluations support the procedure, while separate diagnostics examine adaptation, layer choice, and direction. A close wrong answer is not only a failure to discard; it can identify the distinction that training should address.

1 INTRODUCTION

Large language models have shown strong potential in mathematical reasoning through prompting and post-training. Representative approaches include chain-of-thought prompting (Wei et al., 2022), process supervision (Lightman et al., 2023; Uesato et al., 2022), and outcome-filtered self-training (Zelikman et al., 2022; Singh et al., 2024). Their effectiveness depends on how reasoning examples are constructed and selected. Human feedback can be costly, while self-generated rationales reduce dependence on human-written solutions. Yet selecting a correct response does not identify which incorrect alternative the model finds difficult to reject. This matters when an error reflects close competition rather than an unavailable answer.

In close-call reasoning, a correct answer and its strongest incorrect competitor receive nearly identical answer-content scores. Candidate-based evaluation exposes competition among plausible answers to the same question (Wang et al., 2023b; Cobbe et al., 2021). We focus on **small-gap errors**: a correct candidate is available, but its score falls just below that of the strongest wrong candidate. A small gap does not make the question easy—it identifies a close competition that the model resolves incorrectly. On such cases, the wrong answer is not merely a negative label; it identifies the specific alternative against which the correct answer needs to be distinguished.

Two related strategies motivate our approach. Inference-time methods score candidates with verifiers or intervene on hidden states (Cobbe et al., 2021; Li et al., 2023), introducing additional scoring or intervention at deployment. Training-time approaches learn from preferences or adapt internal representations (Rafailov et al., 2023; Wu et al., 2024). These methods show that both competing responses and hidden states can be useful. Our question is more specific: **which internal layer provides a useful correct–wrong comparison, and can that comparison become training supervision without adding an inference-time judge?**

We investigate whether hidden-state geometry can sharpen this distinction. The inspected candidate pools show close competition, and held-out readouts vary across layers. We therefore mean-pool candidate answer-content tokens and use the training pairs' average correct-minus-wrong contrast

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.

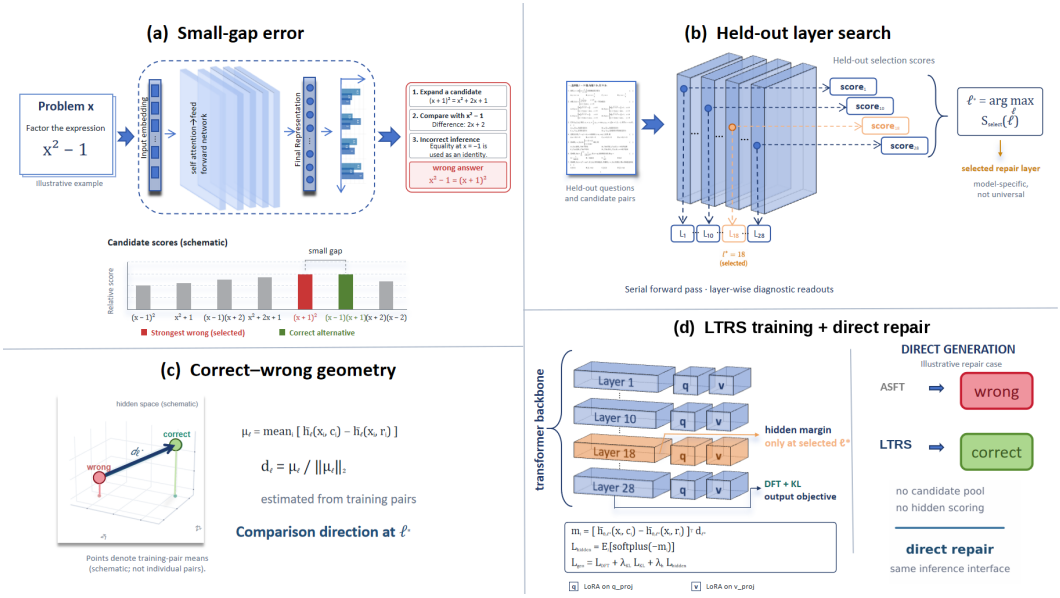

Figure 1: **LTRS: layer-wise supervision for reasoning sharpening.** A competitive wrong answer defines the comparison; a held-out diagnostic selects its layer; training-pair means define its direction. Hidden-margin supervision complements anchored answer learning, with adapters on matching query/value projections. Candidate bars, geometric points, and the repair case are schematic. No hidden-space selector is added at inference.

as a fixed direction. This is an empirical comparison of answer representations, not an assumed universal boundary between correct and incorrect reasoning.

We propose **Layer-wise Tuning for Reasoning Sharpening (LTRS)**. A held-out diagnostic selects the supervision layer; competitive answer pairs supply a pairwise margin along the fixed direction. Each instance contrasts a correct answer with a high-scoring incorrect competitor. The hidden margin is measured at the selected layer, while lightweight parameter updates span the backbone. Thus, *layer-wise* refers to the supervision location, not layer-restricted parameter updates. LTRS uses labeled pairs and retains answer-level learning, but adds no separate verifier or hidden-space judge at inference (Figure 1).

Experiments cover competition mathematics, scientific quantitative reasoning, and medical question answering. LTRS has higher reported macro scores than SFT at both scales in both domains; the mathematical baselines use a separate source protocol (Section 5.1). Layer, direction, and checkpoint analyses examine the internal comparison and training behavior. Our contributions are threefold:

1. We identify close competition between correct and wrong candidates as a useful focus for studying reasoning errors.
2. We turn this competition into layer-targeted hidden supervision through pairwise margin training, without adding an external reward model or inference-time intervention.
3. We evaluate the resulting procedure across mathematical and medical benchmarks and use targeted diagnostics to delimit its interpretation.

The central idea is that a model's strongest wrong answer can help specify what it has not yet learned to distinguish, and that this distinction can be sharpened through geometric supervision at a selected layer.

2 Related Work

Reasoning and answer selection. Chain-of-thought prompting elicits intermediate reasoning, while self-consistency aggregates the answers of sampled traces (Wei et al., 2022; Wang et al., 2023b). Least-to-most prompting and Tree of Thoughts organize the search over intermediate steps (Zhou

et al., 2023; Yao et al., 2023). Reasoning with Sampling and work on test-time compute further examine how alternative generations can improve the returned answer (Karan & Du, 2025; Snell et al., 2024). These approaches provide context for separating candidate availability from answer selection. LTRS uses candidate competition differently: a preferred wrong answer supplies training supervision, rather than another voting or search rule at inference.

Learning from responses and preferences. DFT reweights token-level learning, and ASFT adds a distributional anchor (Wu et al., 2026; Zhu et al., 2026). Preference optimization also learns from chosen and rejected responses (Rafailov et al., 2023). Learning from a negative is therefore not unique to our method. Our particular comparison is the projected difference between pooled answer representations at a selected layer, rather than an output likelihood ratio. The hidden comparison supplements the supplied token objective and reference anchor; it does not replace them.

Internal representations and adaptation. The tuned lens reads predictions at intermediate depths, and inference-time intervention acts on internal directions (Belrose et al., 2023; Li et al., 2023). Work on the geometry of truth and representation engineering further motivates examining information beyond the returned text (Marks & Tegmark, 2024; Zou et al., 2023). Here, a selected layer provides a location for training a comparison, not an additional deployment-time intervention. Standard low-rank adaptation supplies the trainable update (Hu et al., 2022; Dettmers et al., 2023); the contribution concerns its supervision rather than its factorization. Section 5.3 examines additional adaptation as a diagnostic, and Appendix F develops the broader connections.

Verification and feedback. Mathematical verifiers and process-based feedback assess answers or the steps leading to them (Cobbe et al., 2021; Lightman et al., 2023; Uesato et al., 2022). Self-Refine and Reflexion use feedback to revise a response or a later attempt (Madaan et al., 2023; Shinn et al., 2023), while studies of intrinsic self-correction caution that reconsideration alone is not sufficient in every setting (Huang et al., 2024). Our supervision comes from labeled correct/wrong pairs. It does not assume that the model can independently certify its mistakes, and the hidden margin is not a step-by-step verifier. Its role is to train an answer distinction that can influence the model's subsequent output.

Probing and relational supervision. Control-task probes and minimum-description-length probing emphasize that readout performance depends on the diagnostic as well as the representation (Hewitt & Liang, 2019; Voita & Titov, 2020). This motivates keeping fitting, selection, and audit data separate. Sentence-BERT, SimCSE, and supervised contrastive learning use relationships between examples to organize embeddings (Reimers & Gurevych, 2019; Gao et al., 2021; Khosla et al., 2020). They provide context for comparing representations, but semantic similarity is not answer correctness. Our comparison uses a labeled correct answer and the specific wrong candidate competing with it, rather than treating semantic proximity alone as the training target.

3 Small-Gap Errors

3.1 What is close, and what is wrong?

Let θ_0 be the frozen pre-repair model, x_i a question, and $\mathcal{A}(x_i)$ its candidate set. Let g_i be a gold-equivalent candidate. Answer equivalence is resolved before selecting negatives, so a valid alternative form is not treated as wrong. For answer-content token positions $T(a)$, define

$$s_{\theta_0}(a \mid x) = \frac{1}{|T(a)|} \sum_{t \in T(a)} \log p_{\theta_0}(y_t \mid y_{<t}, x). \tag{1}$$

The strongest wrong candidate and its advantage over the correct candidate are

$$w_i = \arg\max_{\substack{a \in \mathcal{A}(x_i) \\ a \not\equiv g_i}} s_{\theta_0}(a \mid x_i), \tag{2}$$

$$\delta_i = s_{\theta_0}(w_i \mid x_i) - s_{\theta_0}(g_i \mid x_i). \tag{3}$$

A *small-gap error* has an incorrect deployed answer, an available gold-equivalent candidate, and $0 < \delta_i \leq \tau$ for a specified diagnostic threshold τ. Exact ties use the evaluator's fixed rule. The score is a length-normalized log-probability, not a calibrated correctness probability. A correct final answer does not certify its derivation; medical options contain a correct choice by construction. The definition concerns answer preference, not proof of complete understanding.

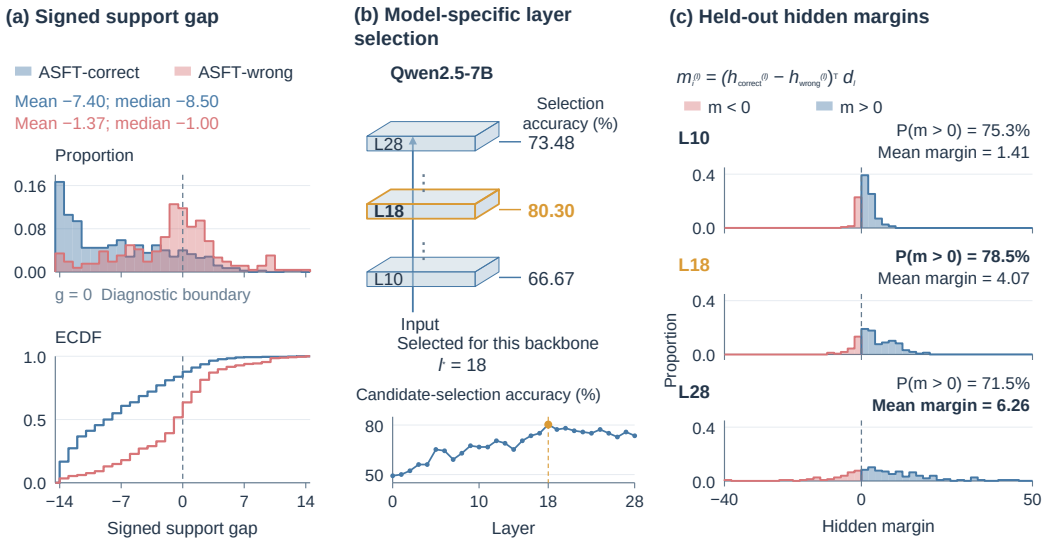

Figure 2: **Candidate competition and layer diagnosis.** (a) Empirical support-gap distributions and ECDFs. (b) Held-out layer selection. (c) A separate gold-centroid minus wrong-centroid margin audit. Red/blue mark sign regions of one paired distribution, not two populations. Definitions and source relationships remain separate (Appendix A).

3.2 Why examine this region?

Figure 2(a) examines questions with eligible candidate sets, separating correct and incorrect reference predictions. Under its signed support-gap statistic, the correct group has mean -7.40 and median -8.5, while the error group has mean -1.37 and median -1.0. Errors are closer to zero by both summaries. This support statistic is the highest wrong support count minus the highest correct support count; it is not the content gap in Eq. (3).

A separate support-switch audit examines cases in which the first sampled answer differs from the most-supported answer. Support gaps often fall in the interval $[1, 4)$, indicating close competition within that conditional set. The compared answers need not form a gold/wrong pair, so the observation is not a prevalence estimate for the content-gap errors in Eq. (3). Appendix A retains its full definition and population.

The two diagnostics motivate examining close competitors before interpreting every wrong answer as a lack of new knowledge. Their score units and denominators nevertheless remain separate. Sampled support measures how often an answer occurs, whereas Eq. (1) measures the model's score for its content. Neither diagnostic by itself determines the content gap of each corrected test example. Figure 2 therefore presents the observations as complementary evidence, not interchangeable measurements.

3.3 From competitor to training pair

An arbitrary wrong response may differ from the correct answer in ways already easy to recognize. A high-scoring negative instead identifies a distinction the model does not reliably express. We use pairs (g_i, w_i) to make that distinction explicit. Small gaps motivate the target, not a deployment-time gate. The archived support-global pair construction does not establish a fixed content-gap cutoff for every pair; the reported training is therefore not claimed to use an exclusively verified small-gap subset.

This distinction separates the motivation from the implementation. The small-gap view identifies a useful competition to investigate. The pairwise objective specifies how to learn from an available correct/wrong pair. It does not require a gap estimate at deployment, and candidate construction remains a training and diagnostic step rather than a new inference module.

4 LAYER-WISE TUNING FOR REASONING SHARPENING

Before training, a frozen model supplies representations for layer selection and direction estimation. During training, the adapted model re-encodes both answers and learns their ordering along that fixed direction. The comparison reference stays fixed while the representations can change.

4.1 SELECT THE SUPERVISION LAYER

At each candidate layer, we mean-pool answer-content hidden states and use the same diagnostic configuration. Fitting, layer selection, and final evaluation have separate data roles. The diagnostic readout is fitted on its training split, the layer is chosen on held-out selection data, and an independent audit examines the resulting comparison. This prevents the layer from being chosen by looking at final benchmark outcomes. The empirical selection record and the margin audit are described separately in Section 5.3. For held-out candidate-selection score $S_{\mathrm{sel}}(\ell)$, choose

$$\ell^* = \arg\max_{\ell \in \mathcal{L}} S_{\mathrm{sel}}(\ell). \tag{4}$$

When selection covers multiple datasets, the specified score is their unweighted mean accuracy. A single-split score remains candidate-selection accuracy, not benchmark macro accuracy. We read states along the same serial Transformer computation. Selecting a layer locates the auxiliary loss; it does not create independent layer branches or restrict all trainable modules to that layer. The selected location is an empirical choice for the backbone being repaired. It is not a claim that the same absolute index must work for a different architecture or that the final layer has lost all useful answer information.

4.2 ESTIMATE THE COMPARISON DIRECTION

Let $\bar{h}_{0,i}^{(\ell)}(a)$ denote the frozen model's pooled answer representation. From N training pairs, estimate

$$d_\ell = \frac{\frac{1}{N}\sum_{i=1}^{N}\left[\bar{h}_{0,i}^{(\ell)}(g_i) - \bar{h}_{0,i}^{(\ell)}(w_i)\right]}{\left\|\frac{1}{N}\sum_{i=1}^{N}\left[\bar{h}_{0,i}^{(\ell)}(g_i) - \bar{h}_{0,i}^{(\ell)}(w_i)\right]\right\|_2}. \tag{5}$$

The mean contrast must be nonzero. We keep d_{ℓ^*} fixed throughout adaptation; test answers do not enter the estimate. Averaging gives training pairs a shared comparison coordinate. During adaptation, the axis stays fixed while the two answer representations are recomputed under the current parameters. A positive projection describes their ordering on this axis; it is not a correctness certificate or a universal correctness direction. The pooled states represent answer-content tokens, excluding prompt and padding positions, with the full pooling rule in Appendix D.

The direction estimate and the loss also play different roles. The estimate summarizes the training contrast before adaptation; the loss asks the adapted model to express that contrast more consistently. Whether the shared coordinate helps the returned answer is tested by task evaluation, rather than inferred from separation on the data used to construct it.

4.3 TRAIN THE PAIRWISE HIDDEN MARGIN

Let θ denote the adapted model, with chosen answer g_i and rejected answer w_i. Re-encoding both responses gives

$$m_i(\theta) = \left[\bar{h}_{\theta,i}^{(\ell^*)}(g_i) - \bar{h}_{\theta,i}^{(\ell^*)}(w_i)\right]^\top d_{\ell^*}. \tag{6}$$

We minimize the smooth pairwise loss

$$\mathcal{L}_{\mathrm{hidden}} = \mathbb{E}_i\left[\mathrm{softplus}(-m_i(\theta))\right]. \tag{7}$$

This loss decreases as the correct-minus-wrong projection increases. It trains a relative relation without prescribing either answer's absolute location. Both responses are passed through the current model: the learned quantity is their difference, not a fixed target embedding for the chosen answer. The fixed direction prevents the comparison axis itself from changing during adapter optimization. We retain the DFT token objective and the KL reference anchor (Wu et al., 2026; Zhu et al., 2026):

$$\mathcal{L}_{\mathrm{geometry}} = \mathcal{L}_{\mathrm{DFT}} + \lambda_{\mathrm{KL}}\mathcal{L}_{\mathrm{KL}} + \lambda_h\mathcal{L}_{\mathrm{hidden}}. \tag{8}$$

Each term has a distinct role. DFT teaches the chosen response at the token level. KL anchors the output distribution to the frozen reference. The hidden loss explicitly compares the chosen response with the competitive negative at the selected layer. Thus ordinary answer learning remains in place while the internal distinction becomes an additional target. Appendix D specifies pooling, detached token weights, and reference-to-adapter KL.

In LTRS, layer selection localizes the hidden loss, not all trainable parameters. Standard LoRA modules update matching query and value projections throughout the backbone (Hu et al., 2022). Hidden-loss gradients flow through the selected representation and its upstream computation; output losses also train later layers. At inference, the fixed direction and hidden loss are removed. Mathematics retains direct generation, and medical evaluation retains its fixed-option content scorer. Neither adds a hidden-space ranking module. In particular, the sampled training pool is not searched again to answer a new mathematical problem. The medical setting still scores its provided options, but does not add a projection-based judge to that scorer. The method changes training supervision while keeping these task-specific deployment interfaces intact.

Training workflow. Freeze the reference, select the supervision layer on held-out candidates, and estimate its direction from training pairs. Re-encode both answers during adaptation and optimize the anchored token and hidden objectives. Detailed configurations remain in Appendix D.

5 Experiments

5.1 Evaluation setup

The mathematical backbone is Qwen2.5-7B. Table 1 uses SFT as its reporting baseline, followed by SFT+KL, DFT, ASFT, and LTRS. The eight non-LTRS mathematical rows reproduce the supplied ASFT-paper table (Zhu et al., 2026); they are not new runs in our evaluator. Its ASFT rows are distinct from the archived ASFT checkpoint used before LTRS.

Mathematical evaluation spans competition problems and scientific quantitative reasoning. AIME24 and AMC23 test competition-style problem solving (Mathematical Association of America, n.d.a; n.d.b); MATH-OAI draws on the MATH collection, Minerva Math adds scientific reasoning, and OlympiadBench supplies olympiad-level problems (Hendrycks et al., 2021a; Lewkowycz et al., 2022; He et al., 2024). Task characteristics and reference-model accuracy guide our discussion of difficulty. Macro accuracy is the unweighted mean across benchmarks. The full-suite 10K/30K records remain separate from the updated diagnostic in Section 5.3.

Medical evaluation uses MedQA, MMLU-Medical, and MedMCQA (Jin et al., 2020; Hendrycks et al., 2021b; Pal et al., 2022). Each reference/adapter pair uses the same full-option content scorer. A/B/C/D letter scoring is a separate reproduction check. The 10K/30K labels identify training settings, not test sizes. The appendices retain evaluation composition, configurations, and diagnostic records.

LTRS macro accuracy is the unweighted mean within each suite, never pooled across domains. The mathematical source averages are transcribed unchanged. Three do not match their displayed component means and are marked in Table 1. The source averages 16 decoding runs; our archived endpoints retain their own evaluator. These differences preclude a matched-run interpretation of the imported comparison.

Section 5.2 first examines challenging-task outcomes; Section 5.3 then examines adaptation, direction, and layer controls. Task utility and component attribution are evaluated separately.

5.2 Improvements on challenging tasks

Competition mathematics. Table 1 reports the new source baselines beside the unchanged LTRS results. At 10K, SFT and LTRS score 1.24% and 10.00% on AIME24, and 17.03% and 47.50% on AMC23. At 30K, the corresponding endpoints are 2.71% and 6.67%, and 21.56% and 45.00%. These close-to-zero source baselines make the competition benchmarks informative, but the cross-source differences are not controlled repair gains.

The reported mathematical averages are 35.65% for LTRS versus 16.73% for SFT at 10K, and 33.84% versus 19.93% at 30K. Their numerical differences are 18.92 and 13.91 points. The source ASFT

		Mathematics: Qwen2.5-7B†						Medical option-content			
Scale	Method	AIME24	Math500/ MATH-OAI	Minerva	Olympiad	AMC23	Avg.	MedQA	MMLU-Med	Med-MCQA	Macro
10K	SFT	1.24	41.84	11.30	12.26	17.03	16.73	26.79	31.69	27.78	28.75
	SFT + KL	0.41 (−0.83)	42.21 (+0.37)	12.05 (+0.75)	12.08 (−0.18)	17.19 (+0.16)	16.79 (+0.06)	26.39 (−0.40)	30.78 (−0.91)	27.52 (−0.26)	28.23 (−0.52)
	DFT	4.18 (+2.94)	59.51 (+17.67)	17.10 (+5.80)	24.95 (+12.69)	31.09 (+14.06)	27.77‡ (+11.04)	25.14 (−1.65)	29.24 (−2.45)	26.92 (−0.86)	27.10 (−1.65)
	ASFT	3.33 (+2.09)	59.60 (+17.76)	19.91 (+8.61)	24.50 (+12.24)	36.41 (+19.38)	28.75 (+12.02)	26.87 (+0.08)	29.76 (−1.93)	27.44 (−0.34)	28.02 (−0.73)
	LTRS	**10.00** (+8.76)	**66.00** (+24.16)	**27.94** (+16.64)	**26.81** (+14.55)	**47.50** (+30.47)	**35.65** (+18.92)	**30.71** (+3.92)	**37.79** (+6.10)	**29.45** (+1.67)	**32.65** (+3.90)
30K	SFT	2.71	44.74	13.21	13.44	21.56	19.93‡	26.08	32.79	28.23	29.03
	SFT + KL	2.70 (−0.01)	44.91 (+0.17)	13.03 (−0.18)	13.48 (+0.04)	18.90 (−2.66)	18.60 (−1.33)	25.45 (−0.63)	32.20 (−0.59)	28.45 (+0.22)	28.70 (−0.33)
	DFT	3.34 (+0.63)	57.93 (+13.19)	23.28 (+10.07)	25.31 (+11.87)	28.44 (+6.88)	27.66 (+7.73)	24.04 (−2.04)	30.56 (−2.23)	27.73 (−0.50)	27.44 (−1.59)
	ASFT	5.81 (+3.10)	57.03 (+12.29)	20.61 (+7.40)	24.44 (+11.00)	30.00 (+8.44)	27.18‡ (+7.25)	27.26 (+1.18)	31.14 (−1.65)	28.78 (+0.55)	29.06 (+0.03)
	LTRS	**6.67** (+3.96)	**65.20** (+20.46)	**26.10** (+12.89)	**26.22** (+12.78)	**45.00** (+23.44)	**33.84** (+13.91)	**31.34** (+5.26)	**38.66** (+5.87)	**32.08** (+3.85)	**34.03** (+5.00)

Table 1: **Mathematical and medical accuracy (%).** Parentheses subtract SFT within each domain/scale (green/red: increase/decrease). † Mathematical baselines are reported by Zhu et al. (2026); LTRS retains its original evaluation, so cross-source differences are descriptive, not matched gains. ‡ Reported averages differ from displayed component means (Appendix A). Bold marks column maxima; shading identifies LTRS.

rows report 28.75% and 27.18%, giving descriptive differences of 6.90 and 6.66 points. Source-reported averages, including those flagged for arithmetic discrepancies, are retained rather than silently recomputed.

The archived paired evaluation still uses the original ASFT reference of 29.97%, not the imported ASFT rows. It gives gains of 5.68 and 3.87 points for LTRS (Appendix B). AIME24 and AMC23 contribute most to these paired gains. At 10K, MATH-OAI changes little and Minerva improves modestly; OlympiadBench decreases slightly. At 30K, the remaining benchmarks are unchanged or lower than the archived reference. Keeping these evaluations separate distinguishes reported task performance from the effect of repair under one evaluator.

Small-gap competition is a more specific target than difficulty itself: a difficult question may offer a useful learning contrast when a correct candidate closely competes with a preferred wrong answer. The competition gains make this target worth pursuing, but benchmark totals do not establish the pre-repair gap of every corrected example. Table 1 reports task performance; the gap diagnostics investigate why particular candidate distinctions may be useful.

Knowledge-intensive answer selection. In the medical block of Table 1, macro accuracy rises from SFT's 28.75% to 32.65% at 10K and from 29.03% to 34.03% at 30K, gains of 3.90 and 5.00 points. All three benchmarks improve over SFT at both scales; MMLU-Medical gains 6.10 and 5.87 points. Relative to the corresponding ASFT pre-repair models, the macro gains remain 4.63 and 4.97 points. LTRS has the highest macro among the reported methods in both settings. These results extend the procedure from mathematical answer generation to knowledge-intensive option selection; the shared element is the training procedure, not an unchanged direction transferred between domains.

These are within-protocol content-choice gains, not clinical-reliability claims. Content scoring compares option text; letter scoring compares answer labels under its own prompt and scoring convention. Absolute accuracies can differ for the same checkpoint, so a difference between the two evaluators is not itself a model improvement or decline. The separate letter-scoring check passes at 10K but remains below its historical reference at 30K. Appendix C retains both reproduction checks and competitive medical controls.

(a) Adaptation diagnostic	
Method	**Macro**
ASFT + LoRA, local official-code reproduction	31.82 (−2.65)
Random direction	33.35 (−1.12)
LTRS, original L18 adapter	**34.47**

(b) Layer and direction	
Method	**Macro**
ASFT	29.97
Alternative layer, L10	33.10 (+3.13)
Covariance-random	33.12 (+3.15)
LTRS, L18	**35.65** (+5.68)

Table 2: **Mathematical diagnostics (macro accuracy, %).** The blocks are separate evaluations. Parentheses compare with LTRS 34.47 in (a) and the archived ASFT 29.97 in (b), not the imported ASFT rows of Table 1. Matching and source details are in Appendices A–B.

Training duration. Figure 3 separates endpoint performance from checkpoint behavior. The displayed 10K run has its highest recorded score at step 200, whereas the 30K run peaks at step 1000. Intermediate scores are non-monotonic. Each line follows a fixed recorded run; isolated markers show other recorded checkpoints without joining different runs. Full values and run identifiers are retained in Appendix C.

The distinction between answer formats is important. In mathematics, a candidate pool is used to construct the training contrast and the adapted model later generates an answer. In medical evaluation, the fixed options supply the alternatives whose contents are scored. The latter does not show that a correct response had to be discovered through generation. It tests whether the same type of hidden comparison can complement answer learning when the alternatives are already given.

These traces describe the recorded optimization path, not an average over random seeds. Their role is to show how the selected checkpoints relate to nearby training budgets; the separate markers retain variation that a single final score would hide.

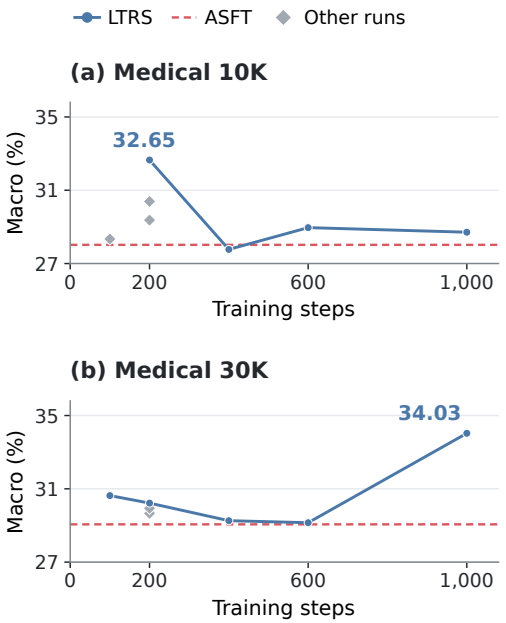

Figure 3: **Medical checkpoints.** Solid: LTRS within-run traces; dashed: matched ASFT references; gray: other runs, not error bars. Full records are in Appendix C.

5.3 Analysis: What does the geometry comparison add?

Additional adaptation is a control, not the method. Table 2(a) compares LTRS with a local reproduction of the official ASFT + LoRA recipe: 34.47 versus 31.82 macro, a 2.65-point (8.33% relative) advantage. Both use low-rank updates, so this is not merely the effect of adding an adapter. It supports the complete procedure, not isolated attribution to pair construction, layer choice, or hidden supervision. Such attribution requires matched starting checkpoints, data, budgets, and selection rules (Appendix A).

Direction and layer choice. Table 2(b) reports ASFT at 29.97, L10 at 33.10, covariance-random at 33.12, and selected L18 at 35.65. LTRS exceeds the latter two controls by 2.55 and 2.53 points. This evaluation is separate from block (a), where random direction is 1.12 points below LTRS. In Figure 2(b), L18 also leads the selection readout: 80.30%, versus 66.67% at L10 and 73.48% at L28. The centroid-margin audit gives positive frequencies of 78.5%, 75.3%, and 71.5%, respectively, whereas L28 has the largest mean margin. These different diagnostics identify a useful location, not a universal correctness layer.

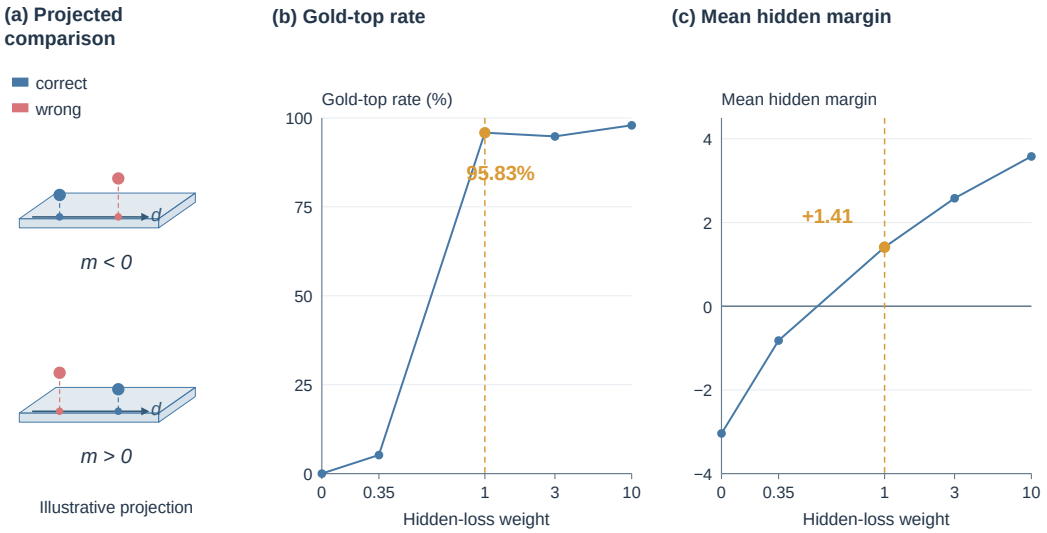

Figure 4: **Hidden ordering responds to its training weight.** (a) Illustrative projections. (b,c) Gold-top frequency and mean margin in a separate controlled listwise diagnostic, not final task accuracy. All weights and the non-monotonic response are retained. Both horizontal axes use symlog with linear threshold one.

The hidden objective changes its target relation. At hidden weight zero, the controlled ranking diagnostic gives 0.00% gold-top frequency and mean margin -3.04; at weight one, these become 95.83% and $+1.41$ (Figure 4). This is a separate listwise diagnostic, not the main pairwise objective. Task accuracy is already high at zero weight and does not increase monotonically with margin. The result verifies a trainable internal relation, not mediation of every benchmark gain. Appendix E retains the full sweep; neither gold-top frequency nor mean margin is final answer accuracy.

The two measurements answer different questions. Gold-top frequency records whether the correct candidate ranks first under the designated projection; mean margin records the signed strength of that comparison. Their joint role is to describe the internal relation being trained. The task table separately evaluates the answers returned by the adapted model under its original interface.

5.4 Scope of the evidence

The diagnostics support studying close competition, not a complete account of reasoning difficulty. Support gaps do not measure the prevalence of Eq. (3), and the archived pairs do not establish small-gap-only training. Updated aggregate comparisons lack a complete matched-run manifest and per-example predictions; no new confidence interval or isolated causal claim is supplied. Selected-checkpoint and variability records remain in the appendices. Medical gains concern benchmark content scoring, not clinical safety.

A direct mechanism test would link pre-repair content gap, representation change, and final correctness on the same held-out examples under matched training. Our separate diagnostics do not constitute that completed mediation test.

6 Conclusion

Small-gap errors expose a specific distinction to learn: a correct candidate is available, but a close wrong answer is preferred. LTRS selects an internal supervision location, estimates a fixed correct-minus-wrong direction, and trains a pairwise hidden margin alongside anchored answer learning. Competition-mathematics and medical results support its utility without making difficulty itself a guarantee of improvement. Adaptation, direction, and layer controls delimit the explanation. The main lesson is that a closely competing wrong answer can identify a concrete distinction for training, even when the underlying problem is difficult.

AI USE STATEMENT

AI tools assisted with writing and polishing selected sections, translation, literature retrieval, discussion of methodology and reported results, and preparation of plotting code and LaTeX. The authors are responsible for the final manuscript, citations, figures, and scientific claims.

REPRODUCIBILITY STATEMENT

Section 4 specifies the training objective and layer-selection procedure. Appendices A–E document evaluation populations, source distinctions, recorded configurations, and the available diagnostic sweeps. Appendix G records the provenance supplied with the final figures and distinguishes that record from computations reproduced during manuscript preparation. Missing matched-run metadata for the updated aggregate comparison are explicitly identified rather than inferred from other experiments.

ETHICS STATEMENT

The medical experiments assess benchmark answer-content scoring, not diagnosis, treatment, or deployment with patients. Improved benchmark accuracy does not establish clinical reliability. Candidate-score examples and geometric illustrations are labeled as schematics. Evaluation protocols and historical result series are kept separate to avoid turning figures or aggregate summaries into unsupported causal claims.

REFERENCES

Mohammad Gheshlaghi Azar et al. A General Theoretical Paradigm to Understand Learning from Human Preferences. arXiv preprint arXiv:2310.12036, 2023. URL https://arxiv.org/abs/2310.12036.

Nora Belrose et al. Eliciting Latent Predictions from Transformers with the Tuned Lens. arXiv preprint arXiv:2303.08112, 2023. URL https://arxiv.org/abs/2303.08112.

Tom B. Brown et al. Language Models are Few-Shot Learners. arXiv preprint arXiv:2005.14165, 2020. URL https://arxiv.org/abs/2005.14165.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering Latent Knowledge in Language Models Without Supervision. In *ICLR*, 2023. URL https://arxiv.org/abs/2212.03827.

Zixiang Chen et al. Self-Play Fine-Tuning Converts Weak Language Models to Strong Language Models. In *ICML*, 2024. URL https://arxiv.org/abs/2401.01335.

Karl Cobbe et al. Training Verifiers to Solve Math Word Problems. arXiv preprint arXiv:2110.14168, 2021. URL https://arxiv.org/abs/2110.14168.

DeepSeek-AI et al. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. arXiv preprint arXiv:2501.12948, 2025. URL https://arxiv.org/abs/2501.12948.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. QLoRA: Efficient Finetuning of Quantized LLMs. In *NeurIPS*, 2023. URL https://arxiv.org/abs/2305.14314.

Kawin Ethayarajh et al. KTO: Model Alignment as Prospect Theoretic Optimization. In *ICML*, 2024. URL https://arxiv.org/abs/2402.01306.

Jiahai Feng and Jacob Steinhardt. How do Language Models Bind Entities in Context? arXiv preprint arXiv:2310.17191, 2023. URL https://arxiv.org/abs/2310.17191.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In *EMNLP*, 2021. URL https://arxiv.org/abs/2104.08821.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer Feed-Forward Layers Are Key-Value Memories. In *EMNLP*, 2021. URL https://arxiv.org/abs/2012.14913.

Mor Geva, Avi Caciularu, Kevin Ro Wang, and Yoav Goldberg. Transformer Feed-Forward Layers Build Predictions by Promoting Concepts in the Vocabulary Space. In *EMNLP*, 2022. URL https://arxiv.org/abs/2203.14680.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On Calibration of Modern Neural Networks. In *ICML*, 2017. URL https://arxiv.org/abs/1706.04599.

Wes Gurnee and Max Tegmark. Language Models Represent Space and Time. arXiv preprint arXiv:2310.02207, 2023. URL https://arxiv.org/abs/2310.02207.

Soufiane Hayou, Nikhil Ghosh, and Bin Yu. LoRA+: Efficient Low Rank Adaptation of Large Models. arXiv preprint arXiv:2402.12354, 2024. URL https://arxiv.org/abs/2402.12354.

Chaoqun He et al. OlympiadBench: A Challenging Benchmark for Promoting AGI with Olympiad-Level Bilingual Multimodal Scientific Problems. In *ACL*, 2024. URL https://arxiv.org/abs/2402.14008.

Dan Hendrycks et al. Measuring Mathematical Problem Solving With the MATH Dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021a. URL https://arxiv.org/abs/2103.03874.

Dan Hendrycks et al. Measuring Massive Multitask Language Understanding. In *ICLR*, 2021b. URL https://arxiv.org/abs/2009.03300.

John Hewitt and Percy Liang. Designing and Interpreting Probes with Control Tasks. In *EMNLP-IJCNLP*, 2019. URL https://arxiv.org/abs/1909.03368.

Jiwoo Hong, Noah Lee, and James Thorne. ORPO: Monolithic Preference Optimization without Reference Model. arXiv preprint arXiv:2403.07691, 2024. URL https://arxiv.org/abs/2403.07691.

Neil Houlsby et al. Parameter-Efficient Transfer Learning for NLP. arXiv preprint arXiv:1902.00751, 2019. URL https://arxiv.org/abs/1902.00751.

Cheng-Yu Hsieh et al. Distilling Step-by-Step! Outperforming Larger Language Models with Less Training Data and Smaller Model Sizes. In *Findings of ACL*, 2023. URL https://arxiv.org/abs/2305.02301.

Edward J. Hu et al. LoRA: Low-Rank Adaptation of Large Language Models. In *ICLR*, 2022. URL https://arxiv.org/abs/2106.09685.

Jie Huang et al. Large Language Models Cannot Self-Correct Reasoning Yet. In *ICLR*, 2024. URL https://arxiv.org/abs/2310.01798.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What Disease does this Patient Have? A Large-scale Open Domain Question Answering Dataset from Medical Exams. arXiv preprint arXiv:2009.13081, 2020. URL https://arxiv.org/abs/2009.13081.

Saurav Kadavath et al. Language Models (Mostly) Know What They Know. arXiv preprint arXiv:2207.05221, 2022. URL https://arxiv.org/abs/2207.05221.

Aayush Karan and Yilun Du. Reasoning with Sampling: Your Base Model is Smarter Than You Think. arXiv preprint arXiv:2510.14901, 2025. URL https://arxiv.org/abs/2510.14901.

Prannay Khosla et al. Supervised Contrastive Learning. arXiv preprint arXiv:2004.11362, 2020. URL https://arxiv.org/abs/2004.11362.

Takeshi Kojima et al. Large Language Models are Zero-Shot Reasoners. In *NeurIPS*, 2022. URL https://arxiv.org/abs/2205.11916.

Dawid J. Kopiczko, Tijmen Blankevoort, and Yuki M. Asano. VeRA: Vector-based Random Matrix Adaptation. In *ICLR*, 2024. URL https://arxiv.org/abs/2310.11454.

Brian Lester, Rami Al-Rfou, and Noah Constant. The Power of Scale for Parameter-Efficient Prompt Tuning. In *EMNLP*, 2021. URL https://arxiv.org/abs/2104.08691.

Aitor Lewkowycz et al. Solving Quantitative Reasoning Problems with Language Models. In *NeurIPS*, 2022. URL https://arxiv.org/abs/2206.14858.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-Time Intervention: Eliciting Truthful Answers from a Language Model. In *NeurIPS*, 2023. URL https://arxiv.org/abs/2306.03341.

Xiang Lisa Li and Percy Liang. Prefix-Tuning: Optimizing Continuous Prompts for Generation. arXiv preprint arXiv:2101.00190, 2021. URL https://arxiv.org/abs/2101.00190.

Hunter Lightman et al. Let's Verify Step by Step. arXiv preprint arXiv:2305.20050, 2023. URL https://arxiv.org/abs/2305.20050.

Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring How Models Mimic Human Falsehoods. In *ACL*, 2022. URL https://arxiv.org/abs/2109.07958.

Shih-Yang Liu et al. DoRA: Weight-Decomposed Low-Rank Adaptation. In *ICML*, 2024. URL https://arxiv.org/abs/2402.09353.

Liangchen Luo et al. Improve Mathematical Reasoning in Language Models by Automated Process Supervision. arXiv preprint arXiv:2406.06592, 2024. URL https://arxiv.org/abs/2406.06592.

Trung Quoc Luong et al. ReFT: Reasoning with Reinforced Fine-Tuning. In *ACL*, 2024. URL https://arxiv.org/abs/2401.08967.

Aman Madaan et al. Self-Refine: Iterative Refinement with Self-Feedback. arXiv preprint arXiv:2303.17651, 2023. URL https://arxiv.org/abs/2303.17651.

Samuel Marks and Max Tegmark. The Geometry of Truth: Emergent Linear Structure in Large Language Model Representations of True/False Datasets. In *COLM*, 2024. URL https://arxiv.org/abs/2310.06824.

Mathematical Association of America. MAA Invitational Competitions. Mathematical Association of America, n.d.a. URL https://maa.org/maa-invitational-competitions/. Official competition description; accessed 17 September 2026.

Mathematical Association of America. American Mathematics Competitions. Mathematical Association of America, n.d.b. URL https://maa.org/student-programs/amc/. Official competition description; accessed 17 September 2026.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and Editing Factual Associations in GPT. In *NeurIPS*, 2022a. URL https://arxiv.org/abs/2202.05262.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. Mass-Editing Memory in a Transformer. arXiv preprint arXiv:2210.07229, 2022b. URL https://arxiv.org/abs/2210.07229.

Yu Meng, Mengzhou Xia, and Danqi Chen. SimPO: Simple Preference Optimization with a Reference-Free Reward. In *NeurIPS*, 2024. URL https://arxiv.org/abs/2405.14734.

Long Ouyang et al. Training language models to follow instructions with human feedback. arXiv preprint arXiv:2203.02155, 2022. URL https://arxiv.org/abs/2203.02155.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. MedMCQA: A Large-scale Multi-Subject Multi-Choice Dataset for Medical domain Question Answering. In *Proceedings of the Conference on Health, Inference, and Learning*, volume 174 of *Proceedings of Machine Learning Research*, pp. 248–260. PMLR, 2022. URL https://arxiv.org/abs/2203.14371.

Nina Panickssery et al. Steering Llama 2 via Contrastive Activation Addition. arXiv preprint arXiv:2312.06681, 2023. URL https://arxiv.org/abs/2312.06681.

Qwen Team. Qwen2.5 Technical Report. arXiv preprint arXiv:2412.15115, 2024. URL https://arxiv.org/abs/2412.15115.

Rafael Rafailov et al. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. In *NeurIPS*, 2023. URL https://arxiv.org/abs/2305.18290.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *EMNLP-IJCNLP*, 2019. URL https://arxiv.org/abs/1908.10084.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. Beyond Accuracy: Behavioral Testing of NLP models with CheckList. In *ACL*, 2020. URL https://arxiv.org/abs/2005.04118.

Zhihong Shao et al. DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. arXiv preprint arXiv:2402.03300, 2024. URL https://arxiv.org/abs/2402.03300.

Noah Shinn et al. Reflexion: Language Agents with Verbal Reinforcement Learning. arXiv preprint arXiv:2303.11366, 2023. URL https://arxiv.org/abs/2303.11366.

Avi Singh et al. Beyond Human Data: Scaling Self-Training for Problem-Solving with Language Models. *Transactions on Machine Learning Research*, 2024. URL https://arxiv.org/abs/2312.06585.

Karan Singhal et al. Large Language Models Encode Clinical Knowledge. arXiv preprint arXiv:2212.13138, 2022. URL https://arxiv.org/abs/2212.13138.

Karan Singhal et al. Towards Expert-Level Medical Question Answering with Large Language Models. arXiv preprint arXiv:2305.09617, 2023. URL https://arxiv.org/abs/2305.09617.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM Test-Time Compute Optimally can be More Effective than Scaling Model Parameters. arXiv preprint arXiv:2408.03314, 2024. URL https://arxiv.org/abs/2408.03314.

Alexander Matt Turner et al. Steering Language Models With Activation Engineering. arXiv preprint arXiv:2308.10248, 2023. URL https://arxiv.org/abs/2308.10248.

Jonathan Uesato et al. Solving math word problems with process- and outcome-based feedback. arXiv preprint arXiv:2211.14275, 2022. URL https://arxiv.org/abs/2211.14275.

Elena Voita and Ivan Titov. Information-Theoretic Probing with Minimum Description Length. arXiv preprint arXiv:2003.12298, 2020. URL https://arxiv.org/abs/2003.12298.

Peiyi Wang et al. Math-Shepherd: Verify and Reinforce LLMs Step-by-step without Human Annotations. arXiv preprint arXiv:2312.08935, 2023a. URL https://arxiv.org/abs/2312.08935.

Xuezhi Wang et al. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In *ICLR*, 2023b. URL https://arxiv.org/abs/2203.11171.

Jason Wei et al. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *NeurIPS*, 2022. URL https://arxiv.org/abs/2201.11903.

Yongliang Wu et al. On the Generalization of SFT: A Reinforcement Learning Perspective with Reward Rectification. In *ICLR*, 2026. URL https://arxiv.org/abs/2508.05629.

Zhengxuan Wu et al. ReFT: Representation Finetuning for Language Models. arXiv preprint arXiv:2404.03592, 2024. URL https://arxiv.org/abs/2404.03592.

Shunyu Yao et al. Tree of Thoughts: Deliberate Problem Solving with Large Language Models. In *NeurIPS*, 2023. URL https://arxiv.org/abs/2305.10601.

Weizhe Yuan et al. Self-Rewarding Language Models. In *ICML*, 2024. URL https://arxiv.org/abs/2401.10020.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. STaR: Bootstrapping Reasoning With Reasoning. arXiv preprint arXiv:2203.14465, 2022. URL https://arxiv.org/abs/2203.14465.

Qingru Zhang et al. AdaLoRA: Adaptive Budget Allocation for Parameter-Efficient Fine-Tuning. In *ICLR*, 2023. URL https://arxiv.org/abs/2303.10512.

Chujie Zheng et al. ProcessBench: Identifying Process Errors in Mathematical Reasoning. In *ACL*, 2025. URL https://arxiv.org/abs/2412.06559.

Denny Zhou et al. Least-to-Most Prompting Enables Complex Reasoning in Large Language Models. In *ICLR*, 2023. URL https://arxiv.org/abs/2205.10625.

He Zhu, Junyou Su, Peng Lai, Ren Ma, Wenjia Zhang, Linyi Yang, and Guanhua Chen. Anchored Supervised Fine-Tuning. In *ICLR*, 2026. URL https://arxiv.org/abs/2509.23753.

Andy Zou et al. Representation Engineering: A Top-Down Approach to AI Transparency. arXiv preprint arXiv:2310.01405, 2023. URL https://arxiv.org/abs/2310.01405.

A Evaluation Records and Comparison Scope

A.1 Updated scores and historical evaluations

The updated mathematical summary contains three author-supplied macro scores: original layer-18 LTRS, 34.47%; local official-code ASFT + LoRA reproduction, 31.82%; and a single random-direction run, 33.35%. Their differences from LTRS are 2.65 and 1.12 percentage points. Raw predictions, per-benchmark counts, and checkpoint identifiers for these updated rows were not included in the available record. Micro accuracies and confidence intervals are therefore not supplied.

The earlier 35.65% geometry result and the covariance-random comparison belong to separate archived evaluations. We do not transfer their sample counts, training scales, per-dataset scores, or uncertainty estimates to the updated comparison. The updated numbers are reported results, not experiments executed during manuscript preparation.

Results retain their recorded evaluation unit: selected-checkpoint accuracies are point estimates, not averages over unspecified repetitions. The archived studies have limited matched-seed coverage. Where seed-specific records are available, they are retained in Appendices B and C rather than treated as evidence for unmeasured cross-seed robustness.

Evidence record	**Available information**	**Use in this paper**
Updated mathematical summary	LTRS 34.47; ASFT + LoRA 31.82; random 33.35	Current aggregate comparisons; run manifest unavailable
Earlier full five-math evaluation	Per-dataset counts; geometry 35.65	Full-suite task results in Table 1; checkpoint context in Appendix B
Covariance-random comparison	Corrected mean 33.12; revised dispersion unavailable	Layer/direction block of Table 2, separate from the 33.35 run
LTRS layer diagnostic	Eligible pool 689; selection 132; audit 158	Separate diagnostic populations
Support-switch audit	409/665 in support-gap [1, 4)	Conditional frequency of close competition
Medical content-choice archive	Reference/adapter counts and checkpoint sweeps	Within-regime medical comparisons

Table A1: Evaluation records and their scope. Missing links between records are not filled with another run's values.

A.2 Baseline roles and corrected labels

The mathematical SFT, SFT+KL, DFT, and ASFT entries in Table 1 reproduce the eight 10K/30K rows in the supplied crop of Table 1 in Zhu et al. (2026). The source describes average mathematical accuracy over 16 stochastic decoding runs, with temperature 1.0 and maximum length 4096. These are reported source baselines, not checkpoints reevaluated in the LTRS pipeline. Source Math500 and archived MATH-OAI are labelled explicitly in the shared column; a sample-ID-level correspondence has not been newly verified. Source "ACM23" is shown under the existing AMC23 benchmark label. LTRS, medical, and independent diagnostic scores are unchanged.

Three source averages disagree with the arithmetic means of their five displayed component scores. DFT at 10K reports 27.77 rather than 27.366; SFT at 30K reports 19.93 rather than 19.132; ASFT at 30K reports 27.18 rather than 27.578. The differences exceed ordinary two-decimal rounding. The reported values are retained and marked ‡ in Table 1; the marks also qualify differences calculated from those averages. No component value is changed to force agreement.

The medical SFT, SFT+KL, and DFT rows remain the earlier author-provided summaries. Their displayed component means agree with the reported macro within rounding, but complete run manifests and per-example predictions were not supplied. Imported mathematical rows and medical summaries are not relabelled as matched single-factor ablations.

The mathematical backbone name follows the author's correction to Qwen2.5-7B. The source ASFT rows (28.75 and 27.18 average accuracy) do not replace the archived 29.97 reference used for the original paired mathematical evaluation. The same name identifies a training method, not a shared checkpoint identity across records.

All parenthetical differences in Table 1 subtract the displayed SFT score in the same domain and scale. For mathematics they are descriptive cross-source subtractions, not causal estimates or paired repair counts. Table 2, the medical curves, and historical paired tables retain their own stated references. The provenance of unrounded count differences remains in Appendix G.

A.3 Requirements for a matched ablation

To attribute a difference to one component, the comparison must match the starting checkpoint, tokenizer, prompt template, correct-response exposure, relevant negative pool, LoRA modules and rank, optimizer, learning rate, effective batch size, training budget, random seeds, checkpoint selection, and final evaluator. Using LoRA on both sides excludes a comparison against no adaptation; it does not automatically match these other factors. Table 2 is therefore a procedure-level diagnostic rather than a verified removal of only the hidden loss.

A.4 Distinct gap definitions

In the support-switch audit, let $b(x)$ be the first sampled answer, $q(x)$ the most-supported answer, and $c_x(a)$ the number of traces supporting a. The statistic is $g_{switch}(x) = c_x\left(q(x)\right) - c_x\left(b(x)\right)$ on examples with $q(x) \neq b(x)$. The 409/665 frequency, or 61.50%, concerns that conditional population, not all incorrect predictions and not necessarily gold/wrong pairs.

The geometry diagnostic's signed-support distribution is another archived statistic, with 426 correct and 263 wrong predictions among 689 eligible examples. Its means and medians retain their support-scale interpretation. Equation (3) instead measures a difference of answer-content log-probabilities. Thresholds and prevalence estimates cannot be moved between these statistics.

The original support-global pair record also uses a composite support-plus-hidden score for pair ranking. The content-score negative in Section 3 states the supplied method specification; it does not establish that every historical pair was mined solely with Eq. (1). A uniform mining protocol or a hard small-gap cutoff requires pair-level scoring records.

B Historical Mathematical Evaluations

B.1 Original per-benchmark results

The archived suite has 30 AIME24, 40 AMC23, 500 MATH-OAI, 272 Minerva Math, and 675 OlympiadBench examples. The LTRS rows of Table 1 retain these full-suite results; its non-LTRS mathematical rows now use the separate source in Appendix A. Table B1 retains the archived ASFT reference and original checkpoint labels. Neither table is a breakdown of the updated 34.47% score.

The AIME24 and AMC23 subsets differ in size from the other benchmarks. Their percentage changes should be interpreted with the test-set composition above; macro averaging gives each benchmark equal weight, not equal statistical precision.

Regime	Method	AIME24	AMC23	MATH-OAI	Minerva	Olympiad	Macro
Reference	ASFT	3.33	27.50	65.80	26.10	27.11	29.97
10K	LTRS, step 200	10.00 (+6.67)	47.50 (+20.00)	66.00 (+0.20)	27.94 (+1.84)	26.81 (−0.30)	35.65 (+5.68)
30K	LTRS, step 600	6.67 (+3.34)	45.00 (+17.50)	65.20 (−0.60)	26.10 (0.00)	26.22 (−0.89)	33.84 (+3.87)

Table B1: Historical mathematical accuracy (%), kept separate from the updated aggregate comparison. Parentheses give changes from ASFT in percentage points using displayed scores.

B.2 Updated diagnostics and older direction controls

The two diagnostic blocks are reported in the main text in Table 2; they are not repeated here.

The random-direction score of 33.35% in Table 2(a) and the covariance-random mean of 33.12% in Table 2(b) refer to different evaluations. The latter is the corrected value supplied for the previously

labelled ten-seed summary. Revised seed-level values and dispersion were not supplied, so the earlier standard deviation is not carried forward or recomputed from the superseded series.

The older layer comparison gives 35.65% macro at L18 and 33.10% at L10. It is consistent with layer-sensitive performance in that evaluation. It is not combined with the new 34.47% score to claim a new matched layer effect. Likewise, the updated ASFT + LoRA comparison is not relabeled as a verified hidden-loss-only ablation.

B.3 Checkpoints and larger holdout

The archived 30K geometry sweep gives macro scores of 32.86, 31.89, 33.84, and 29.74 at steps 200, 400, 600, and 1000. Step 600 is the best observed checkpoint in that sweep; it does not show monotonic improvement with more training. In a separate single-domain Numina evaluation, the reference and layer-18 adapter answer 4583/11000 and 4666/11000 examples correctly. These counts remain attached to that older evaluation and are not used as a replication or uncertainty estimate for the updated comparison.

C Medical Protocols and Selected Runs

C.1 Content scoring and letter-scoring checks

Content-choice scoring applies Eq. (1) to option texts. Official-letter scoring compares A/B/C/D answers under a separate prompt and scoring convention. The same checkpoint can obtain different absolute accuracies under these rules. This protocol difference does not explain a failure to match a historical reference evaluated under the same letter-scoring rule.

Check	MedQA	MMLU-Med	MedMCQA	Macro	Status
Medical 10K, seed 42	40.93	44.94	38.18	41.35	PASS
Medical 30K rerun, seed 42	38.73	42.61	39.66	40.33	Below historical reference

Table C1: Archived official-letter reproduction checks. These are not the content-choice baselines in the medical block of Table 1.

The checkpoint names are `main_asft_med10k_llama2_seed42` and `asft_med30k_llama2_seed42_rerun_for_lbr_cosine`. The content-choice experiment compares each adapter with its corresponding reference. The 30K reproduction status is retained without treating every checkpoint as a reproduction pass.

C.2 Checkpoint and seed context

Table C2 gives the complete checkpoint and seed records that support Figure 3. The figure plots the 10K seed20 and 30K seed18 sequences from that table. The 10K sequence starts at step 200; no earlier value from another run is inserted.

At Medical 30K and step 1000, the matched-budget random-direction run reaches macro 36.20%, exceeding the corresponding geometry score of 34.03%. This is a separate medical experiment, not the updated mathematical random score of 33.35%. The medical evidence supports improvement over the paired ASFT reference, not superiority over every directional control. Matching training budget alone also does not establish equality of all other conditions.

D Definitions and Implementation

D.1 Pooling and independent diagnosis

Answer representations are pooled only over answer-content tokens, excluding prompt and padding positions:

$$\overline{h}^{(\ell)}_{\theta,i}(a) = \frac{1}{|T(a)|} \sum_{t \in T(a)} h^{(\ell)}_{\theta,i,t}\,(x_i, a)\,. \tag{D1}$$

Medical 10K			Medical 30K		
Seed	**Step**	**Macro (%)**	**Seed**	**Step**	**Macro (%)**
18	100	28.34 (+0.32)	18	100	30.63 (+1.57)
18	200	30.39 (+2.37)	18	200	30.22 (+1.16)
19	200	29.37 (+1.35)	18	400	29.26 (+0.20)
20	200	32.65 (+4.63)	18	600	29.15 (+0.09)
20	400	27.77 (−0.25)	18	1000	34.03 (+4.97)
20	600	28.96 (+0.94)	19	200	29.66 (+0.60)
20	1000	28.71 (+0.69)	20	200	29.94 (+0.88)

Table C2: **Medical checkpoint and seed records.** Parentheses give macro changes from ASFT (10K: 28.02; 30K: 29.06). The two blocks list independent configurations, not row-wise pairs. Reported main results are selected checkpoints, not averages. Figure 3 follows seed 20 at 10K and seed 18 at 30K; the former starts at step 200.

Token boundaries, end-of-response handling, answer normalization, and gold-equivalence rules must remain fixed across runs. Directions are fitted on training data. Held-out positive-margin frequency is the fraction of audit pairs with a positive gold-minus-wrong projection; mean margin measures the size of the average projection. These are separate diagnostics.

On the same training pairs used to define the direction, the mean projection onto the normalized mean difference is nonnegative by construction. Training-set separation alone therefore does not establish a generalizable correctness signal. Selection and audit examples must not be used to fit the direction or probe being assessed.

D.2 Token objective and KL preservation

The chosen-response DFT loss uses a detached probability weight:

$$\mathcal{L}_{\text{DFT}} = \mathbb{E}_{t \in T(g)} \left[\text{sg} \left(p_\theta \left(y_t \mid y_{<t}, x \right) \right) \left(- \log p_\theta \left(y_t \mid y_{<t}, x \right) \right) \right]. \tag{D2}$$

The operator sg stops gradients through the weight. The reference-preservation term uses the reference-to-adapter KL orientation:

$$\mathcal{L}_{\text{KL}} = \mathbb{E}_t \left[D_{\text{KL}} \left(p_{\text{ref}} \left(\cdot \mid y_{<t}, x \right) \parallel p_\theta \left(\cdot \mid y_{<t}, x \right) \right) \right]. \tag{D3}$$

The reference distribution is frozen. Removing the hidden term leaves the DFT/KL objective; identifying an independently trained baseline with that exact ablation still requires the matching conditions in Appendix A.

D.3 Metrics and training exposure

For dataset k with n_k examples, Acc_k is the fraction of gold-equivalent predictions. The aggregate metrics are

$$\text{Macro} = \frac{1}{K} \sum_{k=1}^{K} \text{Acc}_k, \qquad \text{Micro} = \frac{\sum_{k=1}^{K} n_k \text{Acc}_k}{\sum_{k=1}^{K} n_k}. \tag{D4}$$

For secondary prediction-change audits, let b_i and a_i denote correctness before and after adaptation:

$$R = \sum_i \mathbf{1} \left[b_i = 0 \wedge a_i = 1 \right], \qquad D = \sum_i \mathbf{1} \left[b_i = 1 \wedge a_i = 0 \right]. \tag{D5}$$

The net change is $R - D$. These counts audit collateral effects without becoming a second research target. A macro confidence interval should resample examples within each benchmark before averaging accuracies; a pooled micro interval is not a substitute. No new interval is supplied for the updated aggregate results.

With device batch size B_d, accumulation G, and data-parallel world size W, nominal exposure is

$$B_{\text{eff}} = B_d GW, \qquad N_{\text{seen}} = \text{steps} \times B_{\text{eff}}, \qquad \text{Coverage} = \frac{N_{\text{seen}}}{N_{\text{pairs}}}. \tag{D6}$$

Coverage counts presentations, not necessarily unique examples. The archived single-process run uses $B_d = 1$ and $G = 4$. The configuration below describes that known geometry implementation, not unrecorded settings of the updated baseline.

Configuration	Value / setting
Backbone	Qwen2.5-7B
Candidate pairs	3,000
Hidden supervision	Layer 18 of 28
LoRA targets	Matching q_proj and v_proj modules throughout the model
Rank / alpha / dropout	16 / 32 / 0.05
KL weight / hidden weight	0.03 / 0.05
Optimization / checkpoint	200 steps / step 200
Direction	Fixed, fitted before adapter training
Inference	No added hidden scoring; task evaluator unchanged

Table D1: Archived original geometry configuration. This table is not a substitute for the missing matched-run manifest of the updated comparison.

E Full Auxiliary Hidden-Ranking Sweep

This controlled candidate-ranking experiment is separate from the large-model benchmark comparisons. The source notes accompanying Figure 4 identify it as a listwise hidden-ranking diagnostic, rather than an exact ablation of the pairwise objective in Eq. (7). It examines how hidden ordering responds to its training weight, not how common small-gap errors are.

Metric	Hidden-loss weight				
	0.0	0.35	1.0	3.0	10.0
Target acc. (%)	98.96	100.00 (+1.04)	98.96 (0.00)	95.83 (−3.13)	96.88 (−2.08)
Clean acc. (%)	99.77	100.00 (+0.23)	99.55 (−0.22)	99.32 (−0.45)	98.42 (−1.35)
Gold-top (%)	0.00	5.21 (+5.21)	95.83 (+95.83)	94.79 (+94.79)	97.92 (+97.92)
Mean margin	−3.04	−0.82 (+2.22)	+1.41 (+4.45)	+2.58 (+5.62)	+3.58 (+6.62)

Table E1: Separate single-run diagnostic on 96 target examples. Parentheses give changes from the zero-weight column: percentage points for accuracies and gold-top rate, original units for mean margin, using displayed summaries. Hidden ordering changes strongly, while task accuracy does not increase monotonically with margin magnitude.

At zero hidden weight, task accuracy is already high despite poor ranking under the selected projection. Projection ranking is therefore not necessary for correctness in every case. The sweep verifies that the diagnostic relation can be trained, not that it mediates the main benchmark gains. A stronger test would pair pre-repair content gaps, hidden-state changes, and final predictions on the same held-out examples under matched training conditions.

F Extended Related Work

F.1 Reasoning through prompting, search, and training

Few-shot learning (Brown et al., 2020) and chain-of-thought prompting (Wei et al., 2022; Kojima et al., 2022) show how the input can change the reasoning a language model expresses. Least-to-most prompting decomposes a problem (Zhou et al., 2023), while Tree of Thoughts searches over intermediate reasoning choices (Yao et al., 2023). Self-consistency (Wang et al., 2023b), Reasoning with Sampling (Karan & Du, 2025), and work on allocating test-time compute (Snell et al., 2024) examine how inference can use alternative generations. They provide context for distinguishing candidate availability from the answer returned. LTRS instead uses a competing answer to construct training supervision; it does not add a search or voting procedure at inference.

Other approaches change the training signal. Instruction following from human feedback (Ouyang et al., 2022), DeepSeekMath (Shao et al., 2024), and DeepSeek-R1 (DeepSeek-AI et al., 2025) use feedback or reinforcement learning to shape model behavior. STaR (Zelikman et al., 2022), ReST-style self-training (Singh et al., 2024), step-by-step distillation (Hsieh et al., 2023), and reinforced fine-tuning (Luong et al., 2024) use generated reasoning or rationale supervision in different ways. These methods locate our work within learning from responses, but do not establish the specific effect of our selected-layer objective. That effect must be assessed using the comparisons reported in this paper.

F.2 Verification and feedback

Training verifiers for mathematical answers (Cobbe et al., 2021) and comparing process-based with outcome-based feedback (Lightman et al., 2023; Uesato et al., 2022) distinguish answer assessment from reasoning supervision. Math-Shepherd (Wang et al., 2023a) and automated process supervision (Luo et al., 2024) develop ways to obtain step-level signals; ProcessBench tests the identification of reasoning errors (Zheng et al., 2025). Our hidden margin compares answer representations. It is not a step-by-step verifier, and a positive margin does not certify a derivation.

Self-Refine (Madaan et al., 2023) and Reflexion (Shinn et al., 2023) use feedback to revise responses or later attempts. The study of intrinsic self-correction by Huang et al. (Huang et al., 2024) cautions that asking a model to reconsider is not sufficient in every setting. Self-Rewarding Language Models uses model-produced evaluations in further training (Yuan et al., 2024). These distinctions matter here: our supervision comes from labeled correct/wrong pairs and a fixed training direction, not an assumption that the model can independently recognize every mistake.

F.3 Preference objectives and competitive negatives

DPO learns from preferred and rejected responses through an output-level objective (Rafailov et al., 2023). KTO (Ethayarajh et al., 2024), ORPO (Hong et al., 2024), and SimPO (Meng et al., 2024) use different formulations of preference learning, while Azar et al. provide a general framework for learning from human preferences (Azar et al., 2023). Self-Play Fine-Tuning uses comparisons between generated and demonstration responses (Chen et al., 2024). Learning from a negative response is therefore not unique to LTRS. Our specific choice is to compare pooled answer representations along a training-pair direction at a selected layer, while keeping the token objective and reference anchor.

F.4 Latent knowledge and answer correctness

TruthfulQA examines the reproduction of human falsehoods (Lin et al., 2022), and Kadavath et al. study models' assessments of their own knowledge and answer correctness (Kadavath et al., 2022). Unsupervised latent-knowledge discovery (Burns et al., 2023) and the geometry of true/false representations (Marks & Tegmark, 2024) further motivate inspecting information beyond the returned text. The tuned lens (Belrose et al., 2023) and inference-time intervention (Li et al., 2023) offer different ways to read or act on internal states. These results concern different tasks and labels. They do not imply that the direction estimated from our mathematical answer pairs is a universal axis of correctness.

F.5 Representation intervention and model editing

Representation Engineering (Zou et al., 2023), contrastive activation addition (Panickssery et al., 2023), and activation engineering (Turner et al., 2023) investigate control through internal activations. Representation Finetuning (Wu et al., 2024) learns interventions on representations, whereas ROME (Meng et al., 2022a) and MEMIT (Meng et al., 2022b) edit factual associations through model weights. LTRS uses hidden states to define an auxiliary training loss, rather than adding its direction to activations at inference. Its use of internal information should not be confused with either a deployment-time steering rule or a factual-memory editing algorithm.

Analyses of Transformer feed-forward layers as key-value memories (Geva et al., 2021) and as contributors to vocabulary predictions (Geva et al., 2022) show that internal computations can be examined at distinct depths and components. Studies of space and time representations (Gurnee

& Tegmark, 2023) and entity binding (Feng & Steinhardt, 2023) examine other forms of internal structure. These works motivate layer-aware analysis without supplying evidence that a particular layer must be optimal for our task. We select the supervision location empirically for the model being repaired.

F.6 Probing and contrastive representations

Control-task probes (Hewitt & Liang, 2019) and minimum-description-length probing (Voita & Titov, 2020) emphasize that diagnostic performance depends on what the probe learns, not only on what the representation contains. This is why fitting, selection, and audit data have separate roles in our protocol. Sentence-BERT (Reimers & Gurevych, 2019), SimCSE (Gao et al., 2021), and supervised contrastive learning (Khosla et al., 2020) use relational supervision to organize embeddings. They provide context for comparing representations, but semantic similarity is not answer correctness. Our contrast is defined by a labeled correct answer and the specific wrong candidate competing with it.

F.7 Parameter-efficient adaptation

Adapter modules (Houlsby et al., 2019), prefix-tuning (Li & Liang, 2021), and prompt tuning (Lester et al., 2021) adapt a model through different small sets of trainable parameters. LoRA (Hu et al., 2022) parameterizes weight changes with low-rank factors, and QLoRA (Dettmers et al., 2023) combines such updates with a quantized backbone. AdaLoRA allocates the adaptation budget (Zhang et al., 2023), DoRA separates weight magnitude and direction (Liu et al., 2024), VeRA learns scaling vectors over shared random matrices (Kopiczko et al., 2024), and LoRA+ changes the learning-rate relationship between the low-rank factors (Hayou et al., 2024). These are choices about parameterization or optimization. Our contribution concerns the supervision applied through an adapter. In particular, VeRA's random matrices are not the random hidden-direction control used in this paper, and this literature discussion is not a claim that we benchmarked against all these methods.

F.8 Evaluation and medical question answering

Calibration research distinguishes predictive scores from empirical correctness probabilities (Guo et al., 2017), while CheckList provides a framework for targeted behavioral tests (Ribeiro et al., 2020). These are reasons to label the content gap, sampled-answer support, and hidden margin as different quantities. Model documentation (Qwen Team, 2024), benchmark papers (Hendrycks et al., 2021a; Lewkowycz et al., 2022; He et al., 2024; Jin et al., 2020; Hendrycks et al., 2021b; Pal et al., 2022), and official competition descriptions (Mathematical Association of America, n.d.a; n.d.b) specify the evaluation context; none makes a close score difference a certificate of correctness.

Med-PaLM (Singhal et al., 2022) and Med-PaLM 2 (Singhal et al., 2023) examine medical question answering through benchmark evaluation and human assessment. They support treating medical answer quality as more than a single automatic score. Our medical experiments remain within their stated option-content scoring protocol. The reported improvements do not establish clinical reliability, and an option-content result should not be substituted for an answer-letter or clinical evaluation.

G Provenance of the Inserted Figures

Figures 2 and 4 retain the vector artwork exported from the supplied editable presentation. Its embedded notes identify upstream data files, diagnostic definitions, and previously performed checks. Figure 3 is typeset anew from the checkpoint values in Appendix C. Model/reference labels follow the author-provided correction; chart coordinates and experimental values are unchanged. No model evaluations or upstream hidden-state extractions were rerun. Figure 1 is the previously approved mechanism illustration.

G.1 Support-gap distribution and source populations

The recorded support gap is

$$g_{\mathrm{sup}}(x) = \max_{a \not\equiv g} s_{\mathrm{count}}(a \mid x) - \max_{a \equiv g} s_{\mathrm{count}}(a \mid x),$$

where s_{count} is the number of sampled traces supporting an answer. The source notes identify `fig2_gap_examples.csv`, a 689-row eligible table reconstructed from `candidate_features.jsonl` and the math10K base direct-generation evaluation. The candidate census is tagged as a 30K seed-44 artifact, whereas correctness grouping comes from the math10K base evaluation. This inherited cross-artifact construction is retained; it is not a single shared training run, and it is not the content gap in Eq. (3). The notes record that 828 of the 1,517 evaluation problems lack an eligible support-scored gold/wrong pair.

Histograms use common unit-width, integer-centered bins over the full observed range, separately normalized within each correctness group. ECDFs use observed jumps without fitting or resampling. The metadata report means -7.399061 and -1.372624, displayed as -7.40 and -1.37, with medians -8.5 and -1.0 for correct and wrong groups.

G.2 Layer readout and centroid-margin audit

The layer sweep uses the `candidate_tokens_mean_model_space` anchor in `per_anchor_layer_linear_metrics.csv`. A scalar linear readout is fitted on `probe_train`, selected on `probe_select`, and assessed separately. The recorded selection set has 132 questions, including 102 multicandidate questions; all positions 0–28 are present. Position zero denotes the input/embedding state, not an additional Transformer block. The notes record readout seed 20260804, zero-weight initialization, 500 epochs, and AdamW with learning rate 0.01 and weight decay 0.0001. Selection precedes the locked audit in the recorded manifest.

The margin histogram uses a distinct centroid comparison: the gold-candidate centroid minus the wrong-candidate centroid for each question. A normalized mean direction is fitted on 249 multicandidate training questions and evaluated on 158 multicandidate audit questions per layer. It is not the all-candidate selection accuracy, nor necessarily the chosen/highest-wrong pair used by the main training objective. The recorded means are 1.406819, 4.066576, and 6.264298 at L10, L18, and L28; strictly positive fractions are 75.316456%, 78.481013%, and 71.518987%. Common two-unit bins on $[-40, 50]$ retain negative values. The two colors represent sign regions of one paired distribution, not independently fitted correct/wrong densities.

G.3 Task-result endpoints and rounding

The source notes identify LTRS checkpoints `geometry_support_global_step200` for mathematics 10K, `geometry_seed20260818_step600` for mathematics 30K, `geometry_l18_seed20260820_step200` for medical 10K, and `geometry_l18_seed20260818_step1000` for medical 30K. They record direct generation for mathematics and `content_choice` for medicine; `official_letter` records are excluded from the medical block of Table 1. These are selected run/checkpoint records, not averages over unspecified repetitions.

Medical 10K MedQA is recorded as $342/1273 \to 391/1273$. Its exact difference rounds to $+3.85$ percentage points, whereas subtracting the displayed endpoints 26.87 and 30.71 gives $+3.84$. This is an ASFT-to-LTRS comparison retained as provenance. Table 1 now uses SFT as its reporting baseline: $30.71 - 26.79 = +3.92$ points. The change of comparator does not alter either recorded endpoint. Mathematics 30K AIME24 similarly uses $1/30 \to 2/30$: the exact increment is 3.333333 points, while rounded endpoints subtract to 3.34. This is also an ASFT-to-LTRS increment. Table 1 instead subtracts the imported SFT endpoint, $6.67 - 2.71 = +3.96$ points, without modifying the original counts. This is a cross-source difference, not a newly measured paired increment. Existing macro summaries retain their reported within-domain aggregates; imported baseline averages are kept exactly as supplied, with the three discrepancies documented in Appendix A.

G.4 Controlled hidden-weight response

Figure 4 uses the fields `weight`, `top_slice`, and `margin_slice` from the recorded `weight_sweep.json`. The source identifies diagnostic seed 20260902, train/test sizes 2160/540, a training target slice of 354, and 96 target-test examples. Each weight is a separate training configuration, not a time step. These records are not pooled with the distinct output-recovery experiment. No per-example response margins, error bars, or uncertainty bands are inferred from the aggregates.

Both response plots use the recorded base-10 symmetric-log transform with linear threshold one and linear scale one:

$$T(w) = \begin{cases} w/0.9, & |w| \le 1, \\ \operatorname{sign}(w)\big(1/0.9 + \log_{10}|w|\big), & |w| > 1. \end{cases}$$

The left-hand planes are illustrative ordering diagrams, not measured states or trajectories. They do not identify the controlled scorer with the Qwen L18 backbone.