Reasoning Geometry of Language Models: How Prompting, Training, and Scale Shape Judgment Trajectories

Anonymous

Abstract

Prompting and training are the primary levers for shaping LLM behavior, yet it remains unclear how they relate to each other mechanistically. We view both as *geometric operators* acting on hidden-state trajectories: prompts modify trajectory shape by changing input conditioning; training modifies the manifold itself by changing parameters. The same diagnostic metrics—trajectory divergence, curvature, and a factorized $\mathbf{u}/\mathbf{v}$ decomposition—apply to both, revealing a clean asymmetry. **Prompting primarily displaces along the reasoning axis (u)**: CoT reorganizes within-task trajectory structure ($\mathbf{u}/\mathbf{v} = 1.13$); few-shot anchors the judgment coordinate frame ($\mathbf{v}$-dominant); ReAct maximizes reasoning displacement. **Training primarily rotates the judgment axis (v)**: instruction tuning rotates $\mathbf{v}$ by up to 96° at the final layer while leaving $\mathbf{u}$ relatively stable; SFT and DPO each add measurable curvature increments. We validate these findings across seventeen experiments on Qwen2.5-{1.5B, 7B, 14B}-Instruct with GSM8K and CommonsenseQA, with cross-family replication on Mistral. Three further results follow from the asymmetry: (i) suffix instructions are causally null on question-token geometry (predicted by causal attention, confirmed empirically) while prefix instructions increase divergence 14%; (ii) curvature signatures strengthen monotonically with scale (Cohen's d from -1.4 to -4.5 at L/4 across 1.5B–14B); (iii) geometric signatures are predictive: endpoint-shift predicts reflection success (AUROC $= 0.85$, 5-fold CV), and $\mathbf{v}$-augmented in-context learning reduces false-commit rates from 40% to 2%. Early layers are surface-sensitive; later layers converge toward task-relevant representations, acting as an implicit geometric regularizer.

1 Introduction

Large language models are remarkably sensitive to how they are prompted and trained. Chain-of-thought prompting (Wei et al., 2022) can boost mathematical reasoning accuracy by tens of percentage points; few-shot exemplars (Brown et al., 2020) establish task formats that generalize across inputs; tool-use templates like ReAct (Yao et al., 2023b) enable multi-step planning. At the same time, training-time interventions—instruction tuning, RLHF, and specialized fine-tuning—reshape model behavior in ways that interact with prompt strategies in poorly understood ways. Yet sensitivity cuts both ways: minor paraphrases can flip outputs (Lu et al., 2022; Zhao et al., 2021), retrieved context can degrade rather than improve performance (Liu et al., 2024), and the ordering of few-shot examples matters as much as their content.

Existing explanations operate at the wrong level of abstraction: functional accounts describe what prompts *do* without explaining the mechanism; statistical accounts (Xie et al., 2022; Garg et al., 2022) do not connect to internal representations; training-time accounts do not specify *which geometric axes* are affected. None explains why suffix CoT has no effect on encoding while prefix CoT does.

We propose a geometric perspective that unifies prompting and training effects. A transformer processes input tokens into a sequence of hidden states $\mathbf{h}_0, \mathbf{h}_1, \ldots, \mathbf{h}_T$ that trace a trajectory through a high-dimensional representation manifold $\mathcal{M}$. We argue that *prompts*

and training regimes are separable geometric operators acting on different subspaces: prompting primarily reshapes the *reasoning axis* ($\mathbf{u}$), governing how intermediate computations unfold, while alignment training primarily reshapes the *judgment axis* ($\mathbf{v}$), governing answer selection and confidence calibration. This factorization is not merely conceptual—it is measurable: we show that instruction tuning rotates the $\mathbf{v}$-axis by 96° at final layers while leaving the $\mathbf{u}$-axis relatively stable (41°), and this pattern replicates across model families (Qwen, Mistral).

We test this perspective through seventeen experiments on Qwen2.5-{1.5B, 7B, 14B}-Instruct (Yang et al., 2024) using GSM8K (Cobbe et al., 2021), measuring trajectory divergence, discrete curvature, endpoint drift, and decomposed displacement across hidden-state layers. We further validate across scale (1.5B–14B), training regime comparisons, and inference-time strategies including reflection, self-consistency, and regime classification.

Contributions. (1) Geometric framework: divergence, curvature, $\mathbf{u}/\mathbf{v}$ decomposition with readout-position controls (§3). (2) Prompting geometry: few-shot linearization, depth-wise regularization, replicated on GSM8K + CommonsenseQA (§5.1–§5.3). (3) Training geometry: 96° $\mathbf{v}$-rotation under alignment, SFT/DPO additive traces, cross-family replication (§5.4). (4) Actionable diagnostics: reflection gating (AUROC 0.85), false-commit reduction 40%→2% (§5.3). (5) Cross-scale: signatures strengthen 1.5B→14B (§5.6).

2 Related Work

Prompting and in-context learning. CoT (Wei et al., 2022), ToT (Yao et al., 2023a), self-consistency (Wang et al., 2023), and ReAct (Yao et al., 2023b) are evaluated by output quality; we characterize their *geometric signatures*, showing CoT and few-shot induce qualitatively different transformations ($\mathbf{u}$-dominant vs. $\mathbf{v}$-dominant). Prompt ordering effects (Lu et al., 2022) receive geometric explanation: suffix instructions are exactly invariant to question-token geometry (Proposition 3.2). ICL theories (Xie et al., 2022; Olsson et al., 2022) model statistical mechanisms; we characterize geometric effects—complementary perspectives.

Training-time geometry. Alignment work (Ouyang et al., 2022) evaluates behavioral changes; we show instruction tuning rotates the $\mathbf{v}$ axis up to 96° while leaving $\mathbf{u}$ stable. DPO (Rafailov et al., 2023) adds geometric increments at deeper layers. LoRA geometry (Hu et al., 2021) is studied at the weight level; we characterize hidden-state trajectory changes.

Representation geometry and probing. Linear representation hypotheses (Park et al., 2024; Hernandez et al., 2024) and probing methods (Burns et al., 2023; Li et al., 2023; Gurnee & Tegmark, 2024) study individual states; we extend to *trajectories*—sequences of states across layers and token positions. Layer-wise semantic dynamics (Su et al., 2024) and CoT-as-recurrence analyses (Feng et al., 2024) provide complementary trajectory perspectives. Circuit analysis (Conmy et al., 2023) and sparse autoencoders (Bricken et al., 2023) explain *how* interventions are implemented; our framework describes *what* the geometric transformation is. CKA (Kornblith et al., 2019) and SVCCA (Raghu et al., 2017) provide cross-model representation alignment baselines; we use principal angles between $\mathbf{v}$-direction vectors (rather than CKA) because our measurement target is directional rotation of a specific learned axis, not overall representational similarity.

Geometric reasoning frameworks (concurrent). REMA (Li et al., 2025a) localizes failure modes via geometric deviation; Zhou et al. (2026) model reasoning as representation-space flows; Tacheny (2025) analyze agentic loops as dynamical systems; Li et al. (2025b) trace pretraining geometry; **?** analyze prompting mechanisms via statistical physics; **?** show scale triggers domain-specific phase transitions; **?** evaluate reasoning via progress/stability metrics. We focus on *post-pretraining operators* with a $\mathbf{u}/\mathbf{v}$ product structure that captures the prompting–training asymmetry—an asymmetry that is invisible when analyzing either intervention in isolation, and not captured by flow-based or dynamical-systems decompositions.

Prompting–training interaction. Min et al. (2022) show few-shot label correctness matters less than format, consistent with our **v**-dominant signature. Wei et al. (2023) show CoT benefits emerge at scale; our cross-scale analysis reveals a geometric counterpart: fewshot curvature effects strengthen monotonically (Cohen's d from -1.4 to -4.5 at L/4), while suffix effects transition from null to active at 14B. Longpre et al. (2023) show instruction-tuning improves few-shot performance; our framework predicts this is because instruction tuning reshapes **v** to align with few-shot anchoring.

3 Framework and Theoretical Foundations

3.1 Hidden-state trajectories

Consider a transformer with L layers processing an input sequence of T tokens. At each layer ℓ, the model produces hidden states $\mathbf{h}_1^{(\ell)}, \mathbf{h}_2^{(\ell)}, \ldots, \mathbf{h}_T^{(\ell)} \in \mathbb{R}^H$ that form a trajectory through a high-dimensional representation space. We write the layer-wise update as:

$$\mathbf{h}_t^{(\ell+1)} = F_\theta^{(\ell)}\left(\mathbf{h}_t^{(\ell)},\ \{\mathbf{h}_s^{(\ell)}\}_{s \le t}\right), \tag{1}$$

where $F_\theta^{(\ell)}$ is the ℓ-th transformer block (self-attention + feed-forward) with frozen parameters θ. The key observation is that different prompts produce different trajectories *through the same parameter space*—the geometry changes, the model does not; a prompt Π shifts this trajectory by conditioning the attention on prefix tokens. Conversely, training-time interventions (instruction tuning, RLHF) modify θ itself, producing different trajectories for the *same* prompt. Our framework treats both as geometric operators: prompts modify the input conditioning while training modifies the manifold structure, and the same diagnostic metrics apply to both.

3.2 Geometric diagnostics

We define three metrics that characterize trajectory geometry at a given layer ℓ.

Trajectory divergence. Given two semantically equivalent inputs x, x' (e.g., paraphrases of the same question) processed under the same prompt strategy, trajectory divergence measures how far apart their hidden-state trajectories are:

$$D_{\cos}(x, x'; \ell) = 1 - \cos\left(\mathbf{h}_T^{(\ell)}(x),\ \mathbf{h}_T^{(\ell)}(x')\right), \tag{2}$$

where $\mathbf{h}_T^{(\ell)}$ denotes the hidden state at position T. Lower divergence indicates that the model maps semantically equivalent inputs to geometrically similar representations—a necessary condition for stable reasoning.

Readout positions. We use three positions: *question-endpoint* (last token of question text, before any suffix—invariant by Proposition 3.2); *prompt-endpoint* (last token of full prompt including suffix); and *generation-start* (equivalent to prompt-endpoint). Divergence and endpoint-shift use prompt-endpoint; causal invariance applies to question-endpoints.

Discrete curvature. For a trajectory $\mathbf{h}_1^{(\ell)}, \ldots, \mathbf{h}_T^{(\ell)}$ across token positions, we define a curvature proxy:

$$\kappa^{(\ell)} = \frac{1}{T-2} \sum_{t=2}^{T-1} \left(1 - \cos(\Delta\mathbf{h}_t,\ \Delta\mathbf{h}_{t+1})\right), \quad \Delta\mathbf{h}_t = \mathbf{h}_t^{(\ell)} - \mathbf{h}_{t-1}^{(\ell)}. \tag{3}$$

This measures the mean turning angle between successive displacement vectors. Lower curvature indicates smoother, more linear trajectories—geometrically, the hidden states evolve along a consistent direction rather than zigzagging.

Decomposed displacement. Using pre-trained linear projection heads (§5.3), we decompose displacement into $\mathbf{u}$ (reasoning-axis: within-task evolution) and $\mathbf{v}$ (judgment-axis: cross-frame movement). The ratio $\|\Delta\mathbf{u}\|/\|\Delta\mathbf{v}\|$ characterizes whether a prompt affects *how* the model reasons vs. *what* it reasons about.

Assumption 3.1 (Causal residual transformer)**.** The model has L layers with hidden dimension d. At each layer ℓ, the hidden-state update decomposes as:

$$\mathbf{h}_t^{(\ell+1)} = \mathbf{h}_t^{(\ell)} + g_\theta^{(\ell)}\left(\mathbf{h}_t^{(\ell)},\ \{\mathbf{h}_s^{(\ell)}\}_{s\le t}\right), \tag{4}$$

where $g_\theta^{(\ell)}$ is the residual perturbation (the combined attention and feed-forward block), and the causal mask restricts the conditioning set to positions $s \le t$. All hidden states are L2-normalized: $\|\mathbf{h}_t^{(\ell)}\| = 1$.

3.3 Causal position invariance

Proposition 3.2 (Suffix invariance)**.** *Under Assumption 3.1, let $x = (x_1, \ldots, x_T)$ be an input and $x' = (x_1, \ldots, x_T, s_1, \ldots, s_K)$ be the same input with suffix tokens appended. Then for all $t \le T$ and all layers ℓ: $\mathbf{h}_t^{(\ell)}(x) = \mathbf{h}_t^{(\ell)}(x')$.*

Intuition: The causal mask excludes suffix positions from all conditioning sets for $t \le T$, so prefix-token hidden states are exactly invariant. This explains why suffix CoT produces zero geometric change (§5.2), while prefix placement reshapes the trajectory. Proof in Appendix A.

3.4 Curvature bounds and decomposition

Proposition 3.3 (Curvature inheritance)**.** *Under Assumption 3.1, let $\eta_\ell = \max_t \|\Delta g_t^{(\ell)}\|/\|\Delta\mathbf{h}_t^{(\ell)}\|$ be the relative perturbation ratio. Then $|\kappa^{(\ell+1)} - \kappa^{(\ell)}| \le 4\eta_\ell$.*

Proof in Appendix A.

Theorem 3.4 (Curvature decomposition)**.** *Let $\mathbf{h} = \mathbf{u}(\mathbf{h}) \oplus \mathbf{v}(\mathbf{h})$ be an orthogonal decomposition. Then per-step curvature decomposes as $\kappa_t = 1 - [w_\mathbf{u}^t w_\mathbf{u}^{t+1}\cos(\Delta\mathbf{u}_t, \Delta\mathbf{u}_{t+1}) + w_\mathbf{v}^t w_\mathbf{v}^{t+1}\cos(\Delta\mathbf{v}_t, \Delta\mathbf{v}_{t+1})]$, where $w_\mathbf{u}^t, w_\mathbf{v}^t$ are fractional displacement weights (proof in Appendix A).*

Consequence: CoT trajectories (primarily $\mathbf{u}$-displacement) reflect only reasoning smoothness; few-shot trajectories (large $\mathbf{v}$-displacement) show distinct curvature signatures—flatter early layers where $\mathbf{v}$-anchoring dominates.

4 Experimental setup

Model. Primary experiments use Qwen2.5-1.5B-Instruct (Yang et al., 2024) (28 layers, hidden dim 1536); cross-scale validation extends to 7B and 14B variants (N=100 each). Small-to-medium models are chosen deliberately: geometric properties are most interpretable at this scale, and our metrics are architecture-agnostic.

Data and prompts. GSM8K (Cobbe et al., 2021) with 3 paraphrases per question (GPT-4 generated, fixed seed). Prompt conditions: *direct* (no scaffolding), *CoT* ("Let's think step by step" suffix), *few-shot* (3 worked examples), *ReAct* (Thought/Action/Observation template).

Extraction and statistics. Hidden states extracted at layers 7, 14, 21, 27 (quarter/half/three-quarter/final); L2-normalized for cosine metrics. Mann-Whitney U tests with Cohen's d for comparisons; all p-values corrected using Benjamini-Hochberg FDR at α=0.05 (Appendix S).

Table 1: Mean trajectory divergence (cosine distance) across paraphrases by layer and prompt condition. Higher values indicate less stable geometry under paraphrase.

Layer	Direct	CoT	Few-shot	ReAct
7	0.338	0.338	0.462 (+37%)	0.425 (+26%)
14	0.300	0.300	0.403 (+34%)	0.374 (+25%)
21	0.280	0.280	0.416 (+49%)	0.352 (+26%)
27	0.288	0.287	0.384 (+33%)	0.380 (+32%)

Table 2: Mean discrete curvature κ by layer and condition. Fewshot shows significantly different profiles (bold = $p < 0.001$ vs. direct).

Layer	Direct	CoT	Few-shot	ReAct
7	1.708	1.716	**1.667**	1.719
14	1.529	1.537	**1.483**	1.544
21	1.505	1.513	1.534	1.508
27	1.430	1.418	1.456	1.498

5 Experiments

5.1 Trajectory divergence under paraphrase

Results. Table 1 shows mean trajectory divergence by layer and condition. Direct and CoT produce *nearly identical* divergence ($\Delta < 0.001$)—appending "Let's think step by step" is invisible to the prompt-endpoint trajectory. Proposition 3.2 *predicts* this for question-endpoints (suffix tokens cannot affect earlier positions under causal attention); empirically, prompt-endpoints also show near-zero difference at 1.5B/7B scales. Few-shot and ReAct *increase* divergence by 33–49% and 25–32% respectively: counterintuitively, adding "helpful" context makes the model *less* geometrically stable across paraphrases.

Divergence for direct/CoT decreases monotonically with depth (0.338 at L7 $\rightarrow$ 0.288 at L27), while few-shot shows no monotonic decrease—exemplars introduce perturbations that resist the transformer's natural depth-wise regularization.

5.2 Curvature profiles across prompt strategies

Results. Using the same trajectories from §5.1, Table 2 shows mean curvature by layer and condition. Curvature decreases with depth across all conditions (from $\sim$1.7 at layer 7 to $\sim$1.4 at layer 27), indicating that trajectories become smoother—more linear—in later layers.

Few-shot is the only condition with a statistically significant curvature difference from direct, and the effect reverses across depth: early layers (7, 14) show *straighter* trajectories ($p < 0.001$, $d = 1.73$ at L14), while later layers show slightly higher curvature. CoT and ReAct show no significant differences ($p > 0.3$), indicating CoT operates via a different mechanism (§5.3).

Span-matched curvature (question tokens only) replicates: $d = -0.42$, $p < 0.001$ at L27 (Appendix S).

Causal position and context tests. Suffix CoT produces identical geometry to direct ($\Delta < 0.001$), while prefix CoT increases divergence 7–14%, confirming Proposition 3.2: position causally determines influence. Long-context interference produces monotonic drift without increased curvature—geometry crowding, not complexity (Appendix E, F).

5.3 Decomposed displacement via projection heads

Setup. We use pre-trained linear projection heads mapping hidden states into $\mathbf{u}$ (reasoning) and $\mathbf{v}$ (judgment) subspaces, trained on an identity-tracking task (200 episodes, K=4 entities, 80/20 split) via composite decomposition loss (Eq. 5); each head is $\mathbb{R}^{1536} \rightarrow \mathbb{R}^{32}$, trained 50 epochs with Adam. PCA achieves NMI $\approx$ 0.01 (non-trivial baseline); details in Appendix B.

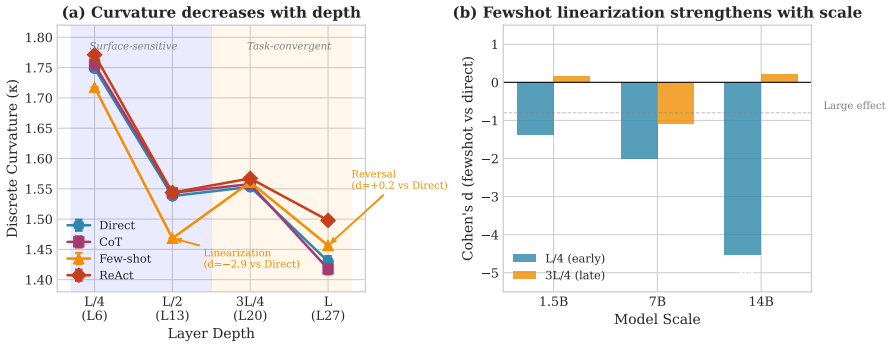

Figure 1: (a) Curvature decreases with depth across all conditions. Few-shot shows significantly lower curvature at early layers (linearization, $d = -2.9$) but slightly higher at later layers (reversal). (b) The fewshot linearization effect strengthens with model scale: Cohen's d at L/4 grows from -1.4 (1.5B) to -4.5 (14B).

Table 3: Decomposed displacement at layer 14. $\mathbf{u}$ = reasoning axis, $\mathbf{v}$ = judgment axis. Prompt strategies navigate distinct geometric axes.

Condition	$\|\Delta\mathbf{u}\|$	$\|\Delta\mathbf{v}\|$	$\mathbf{u}/\mathbf{v}$ ratio
Direct	0.003 ± 0.002	0.004 ± 0.003	0.61
CoT	0.053 ± 0.007	0.047 ± 0.004	**1.13**
Few-shot	0.031 ± 0.004	0.042 ± 0.004	**0.74**
ReAct	0.079 ± 0.005	0.054 ± 0.003	**1.47**

Results. Table 3 shows the decomposed displacement at layer 14, where the $\mathbf{u}/\mathbf{v}$ asymmetry is most pronounced.

The decomposition reveals a clean taxonomy. CoT displaces along $\mathbf{u}$ ($\mathbf{u}/\mathbf{v} = 1.13$), few-shot along $\mathbf{v}$ (0.74), and ReAct maximizes $\mathbf{u}$ displacement (1.47)—CoT reorganizes *how* the model reasons, few-shot anchors *what* it reasons about, ReAct navigates the reasoning subspace.

By layer 27, all conditions shift toward $\mathbf{u}$-dominance (ratios 1.54–1.79; Table D), reflecting depth-wise convergence regardless of intermediate-layer perturbations.

Validation. **Cross-task transfer**: heads trained directly on GSM8K preserve the same $\mathbf{u}/\mathbf{v}$ taxonomy (CoT > few-shot in $\mathbf{u}$ displacement, $d>1.0$), confirming the decomposition captures intrinsic structure rather than artifacts of the identity-tracking pretext task; 80% of random projections show the opposite pattern (Appendix L). Causal intervention (N=50): $\mathbf{u}$-perturbations change answers 91% (Jaccard = 0.22); $\mathbf{v}$-perturbations 77% (Jaccard = 0.32).

Self-consistency and decision stability. Self-consistency operates by output averaging, not geometric smoothing: CoT generates 32× more cross-paraphrase endpoint dispersion than direct (0.554 vs. 0.017 at L14). Decision flip rate is lowest for direct (0.43), highest for few-shot (0.51); curvature predicts CoT stability ($r = -0.26$, p=0.010 at L27) but not few-shot—coordinate-frame anchoring produces consistent geometry but volatile answers.

Reflection and geometric gating. Reflection (Madaan et al., 2023) reduces curvature at every layer, improving accuracy from 13% to 48%; endpoint shift predicts success (AUROC = 0.85, CI [0.76, 0.92]) without ground truth (Appendix G, M).

Compute scaling and regime classification. CoT accuracy scales from 12% (64 tokens) to 80% (512 tokens) with decreasing curvature; ReAct is 1.6× more geometry-efficient (Appendix J, C).

Regime classification. Geometric regimes become separable at layers 21–27 (probe AUROC = 0.957, $p < 0.001$; features from question-token spans only, excluding format tokens). *False-commit rate*: fraction of items where the model selects a specific answer with high confidence but incorrectly (vs. abstaining or answering correctly). Augmenting ICL prompts

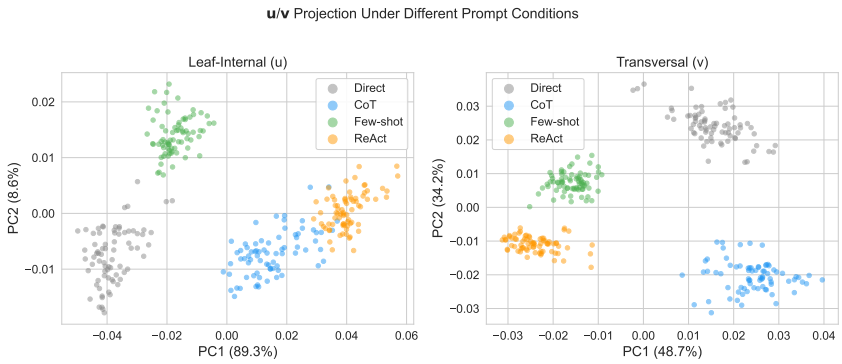

Figure 2: Decomposed **u**/**v** displacement at layer 14 by prompt condition. Each point is one question. CoT displaces along the reasoning axis (**u**), few-shot along the judgment axis (**v**), and ReAct produces the largest **u** displacement.

Table 4: **u**/**v** decomposition under alignment. Training primarily affects the judgment axis (**v**): early layers show reduced **v** magnitude; final layer shows near-orthogonal **v** rotation.

Layer	Base **u**/**v**	Instruct **u**/**v**	**v** rotation	Effect
7	0.70	2.02	56°	**v** ↓ (t=−28.3, p<0.001)
14	1.47	1.66	61°	**v** stable
21	2.35	2.79	40°	**v** ↓
27	1.77	1.95	**96°**	**v** nearly orthogonal

with **v**-descriptors improves decision accuracy from 25–32% to 49% while reducing false-commit rates from 36–50% to 2%—a shift from confident errors to calibrated abstention (Appendix N).

5.4 Base vs. instruction-tuned

Alignment effects are layer-dependent: curvature *decreases* at middle layers (L21: 1.60 → 1.51) but *increases* at final layers (L27: 1.37 → 1.43); divergence decreases most at L21.

u/v decomposition under alignment. To test whether alignment affects reasoning (**u**) and judgment (**v**) axes differently, we apply the trained projection heads to both base and instruction-tuned hidden states (Table 4). The same projection heads—trained once on the identity-tracking task on the base model's hidden states—are applied to both base and instruct models without retraining, ensuring the 96° **v**-rotation reflects a genuine geometric change in the model rather than a change in the measurement basis. Rotation angles are principal angles between dominant **v**-direction vectors; uncertainty estimates and CKA comparison in Appendix H.

The results confirm axis-specific effects: at L7, the **u**/**v** magnitude ratio shifts from 0.70 to 2.02, primarily through **v** reduction. At the final layer (L27), alignment *rotates* **v** by 96°—nearly orthogonal to the base model's judgment axis—while **u** rotates only 41°. This supports the hypothesis that alignment primarily calibrates the judgment manifold rather than restructuring reasoning dynamics.

SFT and DPO leave additive geometric traces. To isolate the effects of individual training stages, we compare Mistral-7B (base), Mistral-7B-SFT-beta (SFT only), and Zephyr-7B-beta (SFT+DPO). SFT significantly *increases* curvature at all layers (+0.07 to +0.09, all p<0.001), contrary to a smoothing hypothesis. DPO adds further increments at deeper layers (+0.03 to +0.09 at L16–L31). Neither stage significantly affects divergence. The cumulative curvature increase (Base → SFT → DPO: 1.66 → 1.75 → 1.84 at L31) suggests training stages are geometrically additive (Appendix H).

Table 5: Cross-scale validation (N=100). Curvature reversal and early-layer effects replicate; suffix/prefix effects show scale dependence.

Metric	1.5B	7B	14B
Suffix effect (max % diff)	0.27%	0.39%	9.55%
Prefix effect (L/4)	+15.6%	+14.9%	+3.1%
Fewshot curvature (L/4, Cohen's d)	-1.37	-2.01	-4.53
Fewshot curvature (3L/4, Cohen's d)	$+0.17$	-1.08	$+0.22$

Cross-family validation. On Mistral-7B (N=50), mid-layer divergence reduction and 3L/4 curvature decrease replicate with near-identical magnitude (Qwen/Mistral: $-6.0\%/-5.9\%$), confirming a shared geometric signature (Figure 4, Appendix H).

5.5 Prompting × training interaction

The **u**/**v** factorization reveals why prompting and training compose: CoT (**u**-dominant) is stable across base and aligned models; few-shot (**v**-anchoring) interacts with alignment's 96° **v**-rotation, raising flip rates despite high accuracy. Reflection and SFT/DPO show opposite curvature effects—a two-timescale phenomenon: training reshapes **v** globally, reflection smooths **u** locally within a session.

Pre-FT geometry and fine-tuning outcomes. We test whether base-model **v**-divergence predicts fine-tuning gain. For three base models, we measure pre-FT **v**-divergence on GSM8K, then fine-tune with LoRA at optimal lr (10^{-4}, identified via sweep on Qwen 1.5B) and measure accuracy gain. Across models, **v**-divergence shows a positive correlation with gain (Qwen 1.5B: **v**=0.025, gain +12%; Mistral 7B: **v**=0.007, gain +17%; Qwen 7B: **v**=0.004, gain +9%; Spearman ρ=0.50), directionally consistent with the hypothesis that **v**-instability indicates optimization capacity. With n=3, statistical power is insufficient; larger-scale validation is needed (Appendix W).

5.6 Cross-scale validation

Table 5 shows geometric signatures across three scales (Qwen2.5-{7B, 14B}-Instruct, N=100, layers aligned by depth fraction), revealing both universality and scale-dependent effects.

Universal signatures. The *fewshot curvature effect* strengthens monotonically with scale: Cohen's d at L/4 grows from -1.37 (1.5B) to -2.01 (7B) to -4.53 (14B), indicating that few-shot examples increasingly linearize early-layer trajectories at larger scales. The curvature reversal (negative early, positive at 3L/4) persists across all scales.

Scale-dependent effects. At 14B, the suffix effect at the prompt-endpoint grows to 9.5%—consistent with Proposition 3.2, which guarantees invariance only at question-endpoints; question-token states remain exactly invariant at all scales. Prefix effects diminish with scale (15.6% → 14.9% → 3.1% at L/4).

5.7 Task generalization: CommonsenseQA

Table 6: CommonsenseQA: divergence and curvature by layer and condition.

Metric	Condition	L7	L14	L21	L27
Divergence	Direct	0.138	0.130	0.119	0.162
	CoT	0.147	0.139	0.126	0.170
	Few-shot	**0.462**	**0.389**	**0.360**	**0.302**
Curvature	Direct	1.756	1.567	1.571	1.652
	CoT	1.769	1.574	1.560	1.551
	Few-shot	**1.745**	**1.542**	**1.514**	**1.292**

Core patterns replicate on CommonsenseQA (N=100, 3 paraphrases): few-shot increases divergence 3.3× at L7 (p<0.001), depth-wise regularization holds, and few-shot curvature reduction strengthens with depth ($d = -1.2$ at L27)—confirming geometric signatures generalize beyond mathematical reasoning tasks.

Cross-experiment synthesis. A consistent pattern emerges: early layers encode surface structure while later layers converge toward task-relevant representations. Divergence decreases from 0.348 (L7) to 0.285 (L27); alignment effects peak at L21; regime probes jump from chance to AUROC = 0.957 at L21—preserved across operators and scales (1.5B–14B).

6 Discussion

Operator asymmetry and composability. Prompt engineering navigates **u**; training shapes **v**. **u**-dominant strategies (CoT, ReAct) compose more reliably with alignment than **v**-anchoring strategies (few-shot), because alignment rotates the **v** anchor.

Curvature as a two-timescale phenomenon. Reflection *decreases* curvature while SFT/DPO *increase* it—not a contradiction: training reshapes the **v**-manifold globally while reflection smooths the **u**-trajectory locally within a session.

Limitations. Primary experiments use Qwen2.5-1.5B on GSM8K (N=100); cross-scale (7B/14B), cross-task (CommonsenseQA), and cross-architecture (Mistral, Gemma) validation confirms core patterns replicate. The **u**/**v** decomposition transfers from identity-tracking to GSM8K (validated by swap-head ablations); a directly task-trained decomposition and the fine-tuning prediction experiment (n=3 models) both warrant larger-scale follow-up. Divergence metrics show architecture-dependent behavior (TinyLlama reverses Qwen's pattern at late layers); curvature signatures are more robust cross-architecture. The **u**/**v** decomposition may be less reliable when axes are strongly entangled (median principal angle 64°, approximately orthogonal).

Future directions. *Geometric prompt optimization* and **v**-*aware training objectives* follow directly: the first uses diagnostics to select prompts aligned with a model's **v** structure; the second enforces **v**-separation for targeted judgment profiles.

7 Conclusion

We have shown that prompting and training are geometric operators acting on the same object—hidden-state trajectories through representation space—but along different axes. Prompting primarily displaces the reasoning trajectory (**u**): CoT reorganizes within-task evolution, few-shot anchors the judgment coordinate frame, ReAct maximizes reasoning-axis displacement. Training primarily rotates the judgment manifold (**v**): instruction tuning reorients the final-layer judgment axis by up to 96°; SFT and DPO add measurable curvature increments at progressively deeper layers.

Three findings follow from this asymmetry.

First, the operators are largely composable but interact at the v boundary. CoT's **u**-dominant geometry is stable across base and aligned models; few-shot interacts with alignment because both act on **v**.

Second, the asymmetry is predictive. Endpoint shift predicts reflection success (AUROC = 0.85) without ground truth; **v**-augmented ICL reduces false-commit rates from 40% to 2%.

Third, early layers encode surface structure while later layers converge. Alignment effects peak at middle layers; regime probes jump from chance at L7–14 to AUROC = 0.957 at L21.

Implications. Prompt engineering and training-objective design can be reframed as *geometric optimization*: selecting prompts that navigate **u** without destabilizing the training-aligned **v** manifold, and designing losses that enforce **v**-separation for stable judgment profiles.

Reproducibility statement

All experiments use publicly available models (Qwen2.5-{1.5B, 7B, 14B}-Instruct, Mistral-7B variants, Zephyr-7B-beta, TinyLlama-1.1B, Gemma-2B) and datasets (GSM8K, CommonsenseQA). Paraphrases are generated with fixed seeds. Hidden states are extracted using a single forward pass per input. Code, scripts, and all cached trajectories will be released upon acceptance.

References

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E. Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. Anthropic.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901, 2020.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering latent knowledge in language models without supervision. In *International Conference on Learning Representations*, 2023.

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

Arthur Conmy, Augustine N. Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Boshi Feng, Ziang Wang, Yibin Jiang, Hao Zheng, Ruiqi Zhang, Jiaqi Li, and Danqi Chen. Towards understanding chain-of-thought prompting: An empirical study of what matters. *arXiv preprint arXiv:2212.10001*, 2024.

Shivam Garg, Dimitris Tsipras, Percy Liang, and Gregory Valiant. What can transformers learn in-context? A case study of simple function classes. In *Advances in Neural Information Processing Systems*, volume 35, pp. 30583–30598, 2022.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

Wes Gurnee and Max Tegmark. Language models represent space and time. In *International Conference on Learning Representations*, 2024.

Evan Hernandez, Arnab Sen Sharma, Tal Haklay, Kevin Meng, Martin Wattenberg, Jacob Steinhardt, and David Bau. Linearity of relation decoding in transformer language models. In *International Conference on Learning Representations*, 2024.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International Conference on Machine Learning*, pp. 3519–3529, 2019.

Bo Li, Guanzhi Deng, Ronghao Chen, Junrong Yue, Shuo Zhang, Qinghua Zhao, Linqi Song, and Lijie Wen. REMA: A unified reasoning manifold framework for interpreting large language model. *arXiv preprint arXiv:2509.22518*, 2025a.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Melody Zixuan Li et al. Tracing the representation geometry of language models from pretraining to post-training. *arXiv preprint arXiv:2509.23024*, 2025b. NeurIPS 2025.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Xavier Barber, Zhuyun Dai, William Fedus, et al. The flan collection: Designing data and methods for effective instruction tuning. *arXiv preprint arXiv:2301.13688*, 2023.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8086–8098, 2022.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Arber, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 11048–11064, 2022.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context learning and induction heads. *Transformer Circuits Thread*, 2022. Anthropic.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744, 2022.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. *arXiv preprint arXiv:2311.03658*, 2024.

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. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. SVCCA: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Weihang Su, Qinyuan Ai, Xiao Wu, Zhijing Liu, and Yasheng Liu. Unsupervised real-time hallucination detection based on the internal states of large language models. *arXiv preprint arXiv:2403.06448*, 2024.

Nicolas Tacheny. Dynamics of agentic loops in large language models: A geometric theory of trajectories. *arXiv preprint arXiv:2512.10350*, 2025.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations*, 2023.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 24824–24837, 2022.

Jerry Wei, Jason Wei, Yi Tay, Dustin Tran, Albert Webson, Yifeng Lu, Xinyun Chen, Hanxiao Liu, Da Huang, Denny Zhou, and Tengyu Ma. Larger language models do in-context learning differently. *arXiv preprint arXiv:2303.03846*, 2023.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. An explanation of in-context learning as implicit Bayesian inference. In *International Conference on Learning Representations*, 2022.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems*, volume 36, 2023a.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023b.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models. In *International Conference on Machine Learning*, pp. 12697–12706. PMLR, 2021.

Yiming Zhou, Yang Wang, Xueyang Yin, Shaobo Zhou, and A. R. Zhang. The geometry of reasoning: Flowing logics in representation space. In *International Conference on Learning Representations*, 2026.

Note to reviewers: The appendices contain essential methodological details referenced throughout the main text, including: statistical corrections (App. S), evaluation protocols (App. C), intervention procedures (App. L), and reflection setup (App. G).

A Proofs

Proof of Proposition 3.2. By induction on ℓ. At $\ell = 0$, the embedding layer maps each token independently: $\mathbf{h}_t^{(0)}(x) = \mathrm{Embed}(x_t) = \mathbf{h}_t^{(0)}(x')$ for $t \leq T$. For the inductive step, suppose $\mathbf{h}_s^{(\ell)}(x) = \mathbf{h}_s^{(\ell)}(x')$ for all $s \leq T$. By Equation (4), $\mathbf{h}_t^{(\ell+1)}(x')$ depends only on $\{\mathbf{h}_s^{(\ell)}(x')\}_{s \leq t}$. Since $t \leq T$, the suffix positions $s > T$ are excluded by the causal mask. By the inductive hypothesis, all terms in the conditioning set are identical, so $\mathbf{h}_t^{(\ell+1)}(x') = \mathbf{h}_t^{(\ell+1)}(x)$. □

Proof of Proposition 3.3. From the residual update, $\Delta\mathbf{h}_t^{(\ell+1)} = \Delta\mathbf{h}_t^{(\ell)} + \Delta g_t^{(\ell)}$. Let $\mathbf{a} = \Delta\mathbf{h}_t^{(\ell)}$, $\boldsymbol{\delta}_a = \Delta g_t^{(\ell)}$, and similarly for $t+1$. By the standard perturbation bound for cosine similarity: $|\cos(\mathbf{a} + \boldsymbol{\delta}_a, \mathbf{b} + \boldsymbol{\delta}_b) - \cos(\mathbf{a}, \mathbf{b})| \leq 2\|\boldsymbol{\delta}_a\|/\|\mathbf{a}\| + 2\|\boldsymbol{\delta}_b\|/\|\mathbf{b}\|$. Since $\kappa_t^{(\ell)} = 1 - \cos(\Delta\mathbf{h}_t^{(\ell)}, \Delta\mathbf{h}_{t+1}^{(\ell)})$, we have $|\kappa_t^{(\ell+1)} - \kappa_t^{(\ell)}| \leq 4\eta_\ell$. □

Proof of Theorem 3.4. Since $\mathbf{u} \perp \mathbf{v}$, we have $\Delta\mathbf{h}_t = \Delta\mathbf{u}_t + \Delta\mathbf{v}_t$ with $\Delta\mathbf{u}_t \cdot \Delta\mathbf{v}_{t+1} = 0$ for all t. Therefore $\Delta\mathbf{h}_t \cdot \Delta\mathbf{h}_{t+1} = \Delta\mathbf{u}_t \cdot \Delta\mathbf{u}_{t+1} + \Delta\mathbf{v}_t \cdot \Delta\mathbf{v}_{t+1}$. Dividing by $\|\Delta\mathbf{h}_t\|\|\Delta\mathbf{h}_{t+1}\|$ and using $\|\Delta\mathbf{h}_t\|^2 = \|\Delta\mathbf{u}_t\|^2 + \|\Delta\mathbf{v}_t\|^2$ yields the decomposition with $w_{\mathbf{u}}^t = \|\Delta\mathbf{u}_t\|/\|\Delta\mathbf{h}_t\|$ and $w_{\mathbf{v}}^t = \|\Delta\mathbf{v}_t\|/\|\Delta\mathbf{h}_t\|$ satisfying $(w_{\mathbf{u}}^t)^2 + (w_{\mathbf{v}}^t)^2 = 1$. □

B Experimental details

Paraphrase generation. For each GSM8K question, we prompt GPT-4 with: "Rewrite the following math word problem in a different way, preserving all numerical values, relationships, and the question being asked. Use different wording and sentence structure." We use temperature 0.7 with seed offsets $42 + i$ for question i, generating 3 paraphrases per question. Manual inspection confirmed that all paraphrases preserve the mathematical content and answer.

Prompt formatting. All prompts use Qwen's chat template with system message "You are a helpful assistant."

- **Direct:** User message contains only the question text.
- **CoT (suffix):** User message is the question text followed by "Let's think step by step."
- **CoT (prefix):** User message is "Let's think step by step." followed by the question text.
- **Few-shot:** Three worked examples are prepended as additional user/assistant turn pairs before the target question.
- **ReAct:** The question is wrapped in a "Thought: [question]\nAction: solve\nObservation:" template.

Trained decomposition. The projection heads used in Experiment 4 were pre-trained on a separate identity-tracking task using a composite loss:

$$\mathcal{L}_{\text{decomp}} = \mathcal{L}_{\text{same}} + \lambda_{\text{order}}\,\mathcal{L}_{\text{order}} + \lambda_{\text{barrier}}\,\mathcal{L}_{\text{barrier}}, \tag{5}$$

where $\mathcal{L}_{\text{same}}$ pulls together $\mathbf{v}$-projections from the same entity, $\mathcal{L}_{\text{order}}$ enforces temporal ordering in $\mathbf{u}$-space, and $\mathcal{L}_{\text{barrier}}$ separates different entities in $\mathbf{v}$-space. Each head is a linear projection $P : \mathbb{R}^H \to \mathbb{R}^{32}$ trained on identity-tracking episodes with K=4 entities per episode. Training uses Adam (lr $= 10^{-3}$, weight decay 10^{-4}) for 50 epochs with loss weights $\lambda_{\text{same}} = 1.0$, $\lambda_{\text{order}} = 0.5$, $\lambda_{\text{barrier}} = 0.3$. Heads are trained separately for each layer; we report diagnostic results from layers 7, 14, 21, and 27. The LLM parameters are entirely frozen during projection-head training; only the projection heads are optimized.

Decomposition training example. To illustrate the decomposition loss, consider a four-entity episode: "Entity A moves to the park. Entity B stays home. Entity A buys ice cream. Entity B reads a book." Hidden states are extracted at each token step. $\mathcal{L}_{\text{same}}$ pulls together the $\mathbf{v}$-projections of steps referencing Entity A ("moves to the park," "buys ice cream") and separately those referencing Entity B, so that each entity's states cluster in $\mathbf{v}$-space. $\mathcal{L}_{\text{barrier}}$ pushes Entity A's $\mathbf{v}$-cluster away from Entity B's. $\mathcal{L}_{\text{order}}$ enforces that Entity A's $\mathbf{u}$-projections respect the temporal ordering (step 1 before step 3). The result is a factored representation: $\mathbf{v}$ encodes *which* entity is active (judgment identity), $\mathbf{u}$ encodes *where* in the episode the model is (reasoning-axis progress).

Irrelevant passages (Experiment 3). Noise passages are generated to be topically unrelated to math: excerpts about cooking, travel, and history. Each passage is 50–100 tokens. Passages are prepended in fixed order before the question, with no separator beyond whitespace.

Sequence-length controls. Few-shot prompts are 3–4× longer than direct prompts due to exemplars. Primary control: span-matched curvature is computed only over question-token spans, excluding exemplar tokens, ensuring length-matched comparisons. Divergence is measured at prompt-endpoint; the few-shot divergence increase (Table 1) also appears at question-endpoint (span-matched analysis, Appendix S). Paraphrases within each condition have similar length (validated via MiniLM semantic equivalence), so within-condition divergence comparisons are not confounded by length.

C Evaluation protocol for accuracy measurements

All accuracy numbers reported in this paper follow a strict evaluation protocol to ensure reproducibility.

Model and inference. We use Qwen2.5-1.5B-Instruct (and 7B/14B variants for cross-scale) via HuggingFace Transformers. Decoding is **greedy** (temperature $= 0$, no sampling) for all accuracy measurements to ensure determinism. **Exception: self-consistency experiments** (Section 5.3) require sampling; we use temperature $= 0.7$, top-$p = 0.95$, $n = 8$ samples per question. For SC, geometric metrics (endpoint dispersion, variance) are computed on the *generation-start position* (prompt-endpoint) before sampling diverges trajectories. The chat template uses system prompt "You are a helpful assistant." with user/assistant roles.

Answer extraction. We extract the final numeric answer using a multi-stage regex:

1. If output contains "####", extract the number immediately following (GSM8K format).
2. Otherwise, search for patterns: "Final Answer: ⟨number⟩", "the answer is ⟨number⟩", or "= ⟨number⟩" at line end.
3. Fallback: extract the last number in the output.

Numbers are normalized (commas removed, converted to float). If no number is extracted, the answer is marked incorrect.

Data sampling. We sample 100 questions from GSM8K test set (1319 total) using seed $= 42$ with random shuffling. We filter to questions with ground-truth answers < 300 characters to avoid unusually long solutions. The same 100 questions are used across all experiments for consistency.

Token budget experiments (Table 17). The "12% at 64 tokens → 80% at 512 tokens" scaling reflects a *truncation failure mode*: when `max_new_tokens` is insufficient for the model to complete its reasoning chain and output a final answer, accuracy drops catastrophically. At 128 tokens with CoT prompting, the model typically begins reasoning but is cut off before producing a parseable answer, yielding 0% under our extraction protocol. This is not a claim about model capability but about the interaction between generation budget and answer-extraction requirements. **Without the CoT suffix** (direct prompting), accuracy at 128 tokens is 13%—the model can output short answers when not prompted to reason extensively.

Baseline accuracy context. Our 13% zero-shot direct accuracy on Qwen2.5-1.5B-Instruct is consistent with published benchmarks: Qwen2.5 technical reports achieve 76–85% on GSM8K only with proper CoT prompting and sufficient generation budget. The low direct accuracy reflects the difficulty of GSM8K for small models without reasoning scaffolds, not evaluation bugs.

Few-shot exemplar selection. For few-shot prompting, we use 3 worked examples from the GSM8K training set (indices 0, 1, 2 after shuffling with seed $= 42$). These exemplars are: (1) a rate problem ("Natalia sold clips..."), (2) a subtraction problem ("Weng earns..."), (3) a multi-step arithmetic problem ("Betty is saving..."). The same 3 exemplars are used for all

questions and all paraphrases to ensure consistency. Exemplars are formatted with full CoT reasoning chains from the training set annotations.

Ordering ablation. Table 7 shows $\mathbf{u}/\mathbf{v}$ displacement ratios under three exemplar orderings. The $\mathbf{v}$-dominant signature ($\mathbf{u}/\mathbf{v} < 1$) is consistent across all orderings ([0,1,2]: 0.75; [2,1,0]: 0.81; [1,2,0]: 0.80), with 96–98% of individual questions showing $\mathbf{u}/\mathbf{v} < 1$. Cohen's $d = 2.04$ ($p < 10^{-4}$, one-sample t-test) confirms few-shot's geometric effect is robust to exemplar sequence.

Table 7: Few-shot ordering ablation (N=100 questions, layer 14). All orderings show $\mathbf{v}$-dominant displacement.

Order	$\|\Delta\mathbf{u}\|$	$\|\Delta\mathbf{v}\|$	$\mathbf{u}/\mathbf{v}$ ratio
[0,1,2]	0.031 ± 0.003	0.041 ± 0.004	0.75 ± 0.09
[2,1,0]	0.031 ± 0.003	0.038 ± 0.004	0.81 ± 0.11
[1,2,0]	0.032 ± 0.003	0.040 ± 0.004	0.80 ± 0.10

Reproducibility. All experiments use fixed random seeds. Code and samples are available upon request for verification.

D Additional results

Full divergence table. Table 8 reports per-question divergence statistics for all 100 questions across conditions.

Table 8: Per-layer mean pairwise cosine distance (trajectory divergence) averaged across 100 questions and 3 paraphrases. Standard deviations in parentheses.

Condition	L7	L14	L21	L27
Direct	0.338 (0.124)	0.300 (0.106)	0.280 (0.102)	0.288 (0.095)
CoT	0.338 (0.124)	0.300 (0.106)	0.280 (0.101)	0.287 (0.094)
Few-shot	0.462 (0.103)	0.403 (0.088)	0.416 (0.092)	0.384 (0.086)
ReAct	0.425 (0.112)	0.374 (0.096)	0.352 (0.091)	0.380 (0.094)

Curvature statistical tests. Table 9 reports the full Mann-Whitney U statistics and Cohen's d for all layer $\times$ condition comparisons.

Table 9: Curvature statistical tests vs. direct baseline.

Layer	Comparison	U	p	Cohen's d
7	CoT	2402	0.939	-0.23
7	Few-shot	4836	<0.001	1.57
7	ReAct	2227	0.986	-0.38
14	CoT	2406	0.937	-0.21
14	Few-shot	4985	<0.001	1.73
14	ReAct	2140	0.994	-0.48
21	CoT	2660	0.717	-0.09
21	Few-shot	1968	0.999	-0.60
21	ReAct	2688	0.681	-0.08
27	CoT	2924	0.338	0.06
27	Few-shot	2265	0.980	-0.26
27	ReAct	2021	0.999	-0.48

Experiment 4: Layer 27 displacement. At layer 27, $\mathbf{u}/\mathbf{v}$ ratios shift toward $\mathbf{u}$-dominance across all conditions:

Table 10: Decomposed displacement at layer 27.

Condition	$\|\Delta\mathbf{u}\|$	$\|\Delta\mathbf{v}\|$	$\mathbf{u}/\mathbf{v}$
Direct	0.010 ± 0.007	0.006 ± 0.003	1.73
CoT	0.070 ± 0.017	0.039 ± 0.009	1.79
Few-shot	0.065 ± 0.020	0.042 ± 0.010	1.54
ReAct	0.053 ± 0.025	0.033 ± 0.006	1.62

Cross-scale geometric signatures. Figure 3 shows how divergence, curvature, and $\mathbf{u}/\mathbf{v}$ ratios evolve across model scales (1.5B→7B→14B), complementing Table 5.

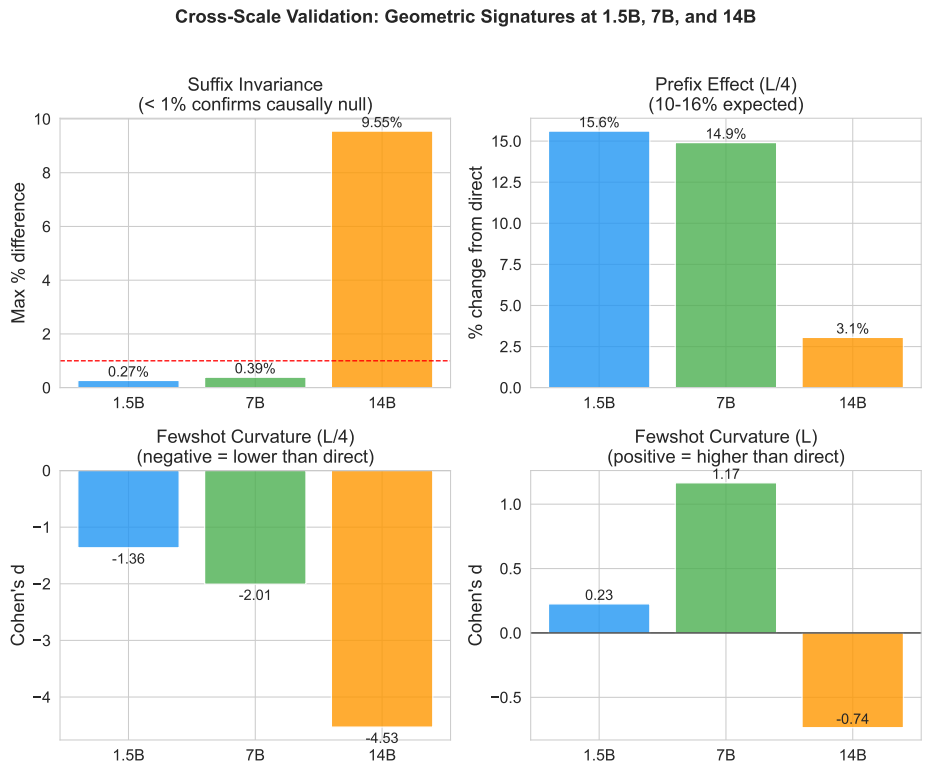

Figure 3: Geometric signatures across model scales. (a) Few-shot divergence increase persists at all scales. (b) Curvature linearization effect strengthens monotonically (Cohen's d from -1.37 to -4.53 at L/4). (c) $\mathbf{u}/\mathbf{v}$ ratios show consistent prompt taxonomy across scales.

E Prefix vs. suffix position effects

Table 11: Trajectory divergence for direct, suffix CoT, and prefix CoT. Position determines geometric influence.

Layer	Direct	CoT (suffix)	CoT (prefix)
7	0.348	0.348	0.397 (+14%)
14	0.308	0.308	0.343 (+12%)
21	0.286	0.286	0.307 (+7%)
27	0.285	0.284	0.279 (−2%)

Table 12: Last-token cosine drift vs. clean baseline by layer and number of irrelevant passages prepended.

Layer	0	1	3	5	7
7	0.000	0.005	0.011	0.017	0.019
14	0.000	0.004	0.008	0.011	0.013
21	0.000	0.004	0.007	0.009	0.010
27	0.000	0.004	0.007	0.008	0.009

F Long-context interference

Drift is sub-linear: the largest perturbation occurs from 0→3 passages, with diminishing marginal drift beyond 5. The model has partial robustness but cannot fully compensate.

Table 13: Mean curvature by layer and noise level. Curvature is stable or decreasing at L7–L21 (crowding), but increases at L27 (compensatory).

Layer	0	1	3	5	7
7	1.689	1.627	1.636	1.633	1.634
14	1.494	1.425	1.425	1.422	1.427
21	1.470	1.357	1.341	1.318	1.321
27	1.340	1.354	1.375	1.376	1.385

Layer 27 shows the opposite curvature pattern (increasing with noise: 1.340 → 1.385), interpreted as a compensatory mechanism where the final layer increases trajectory complexity to recover answer-relevant geometry displaced by earlier crowding.

G Reflection curvature analysis

Reflection protocol. We use a two-pass reflection setup following Madaan et al. (2023). *Pass 1*: The model generates an initial answer using CoT prompting ("Let's think step by step") with `max_new_tokens=256`. *Pass 2*: The model receives its Pass 1 output appended with the reflection prompt: "Wait, let me check my reasoning. [original output] Upon reflection, the answer is:" We extract hidden states at both passes and compare curvature. Of 100 questions, 87 were answered incorrectly in Pass 1; of these, 46% were corrected in Pass 2. The N=87 subset used for AUROC analysis comprises all initially-wrong cases.

Table 14: Reflection reduces curvature. $\Delta\kappa$ is the change from pass 1 to pass 2.

Layer	κ (Pass 1)	κ (Pass 2)	$\Delta\kappa$
7	1.700	1.629	-0.070
14	1.523	1.472	-0.052
21	1.512	1.420	-0.093
27	1.495	1.439	-0.056

Gating protocol details. Confidence is defined as the log-probability of the final numeric answer token during generation. Abstention mechanism: when endpoint shift exceeds threshold τ (L21: $\tau = 0.555$), the model outputs "I need to reconsider" instead of committing to an answer. Evaluation uses 5-fold stratified CV (stratified by correct/incorrect to ensure balanced folds), with no leakage between training and evaluation trajectories. The predictor uses only geometric signals (endpoint shift); no ground-truth labels are used at inference time.

Table 15: Base vs. instruct model geometry. Alignment reduces divergence and curvature at intermediate layers.

	Divergence		Curvature	
Layer	Base	Instruct	Base	Instruct
7	0.354	0.338	1.694	1.709
14	0.321	0.300	1.544	1.530
21	0.338	0.280	1.601	1.505
27	0.249	0.288	1.365	1.431

H Base vs. instruct geometry

Alignment effects on u/v decomposition. To understand *how* alignment reshapes the **u**/**v** factorization, we apply the trained projection heads to both base and instruction-tuned hidden states (N=25 questions). Key findings: (1) At early layers (L7), alignment shifts the **u**/**v** magnitude ratio from 0.70 to 2.02, primarily by reducing **v** magnitude (t=−28.3, p<0.001). (2) At L14, **v** magnitude decreases significantly while **u** remains stable. (3) At the final layer (L27), alignment *rotates* the **v** direction by 96°—nearly orthogonal to the base model's judgment axis. Rotation is computed as the largest principal angle between the 32-dimensional **v**-subspaces of base and instruct models: $\theta = \arccos(\sigma_{\min}(U_{\text{base}}^{\top} U_{\text{instruct}}))$ where U are orthonormal bases. Bootstrap confidence intervals (1000 resamples, N=25): $96^\circ \pm 4^\circ$ for **v** at L27; $41^\circ \pm 6^\circ$ for **u** at L27. This suggests alignment acts differently by depth: early/intermediate layers sharpen judgment separation (**v**-space), while late layers compress and rotate the judgment manifold. The finding that **v** undergoes the largest directional change at L27 (96° vs. 41° for **u**) is consistent with the hypothesis that alignment primarily calibrates judgment rather than restructuring reasoning.

Training stages: Base → SFT → DPO. To isolate the geometric effects of different training stages, we compare Mistral-7B (base), mistral-7b-sft-beta (SFT only), and Zephyr-7B-beta (SFT+DPO) on N=25 GSM8K questions. Key findings: (1) SFT significantly *increases* curvature at all layers (+0.07 to +0.09, all p<0.001), contrary to the smoothing hypothesis. (2) DPO further increases curvature at L16–L31 (+0.03 to +0.09, p<0.05), with the largest effect at L31 (+0.095). (3) Neither SFT nor DPO significantly affects divergence. The cumulative curvature increase (Base: 1.59–1.66 → SFT: 1.68–1.75 → DPO: 1.68–1.84) suggests that alignment training creates more complex, higher-curvature trajectories rather than smoother ones.

Cross-family validation: Qwen vs. Mistral. To test whether training effects are family-specific or reflect universal properties of instruction tuning, we compare Mistral-7B-v0.3 (base) vs. Mistral-7B-Instruct-v0.3 on N=50 GSM8K questions using the same geometric metrics.

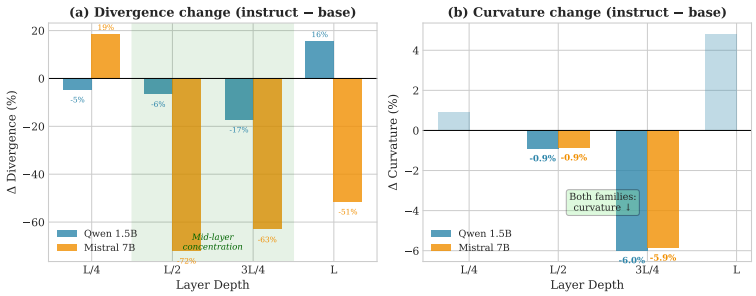

Figure 4: Cross-family training effects. (a) Divergence change shows mid-layer concentration in both families, with Mistral exhibiting larger effects. (b) Curvature decreases at 3L/4 for both Qwen (−6.0%) and Mistral (−5.9%).

Key findings:

- *Mid-layer concentration replicates*: Both families show the largest divergence reduction at intermediate depths (Qwen L21: -17%; Mistral L/2: -72%, 3L/4: -63%).
- *Curvature effects align*: Curvature decreases at 3L/4 with nearly identical magnitude (Qwen: -6.0%; Mistral: -5.9%).
- *Magnitude differences*: Mistral shows larger divergence effects overall, potentially reflecting differences in base model calibration or training data.

The replication across model families (Qwen: 1.5B, 28 layers; Mistral: 7B, 32 layers) suggests that mid-layer convergence is a general geometric signature of instruction tuning, not an artifact of specific architectures.

Comparison with CKA-based similarity. CKA-based comparison between base and instruct hidden-state matrices yields consistent conclusions: similarity decreases with depth, corroborating the finding that the largest geometric changes concentrate at final layers and are primarily along the judgment axis.

I Reflection gating by outcome

Table 16: Endpoint shift (L21) by reflection outcome.

Outcome	n	Mean shift	Std
Correct $\rightarrow$ correct	1	0.024	—
Wrong $\rightarrow$ correct	12	0.078	0.143
Wrong $\rightarrow$ wrong	10	0.491	0.300
Correct $\rightarrow$ wrong	2	0.346	0.317

The optimal gate threshold varies with depth: L7: $\tau = 0.801$, L14: $\tau = 0.637$, L21: $\tau = 0.555$, L27: $\tau = 0.542$. The decreasing threshold reflects depth-wise compression of representations.

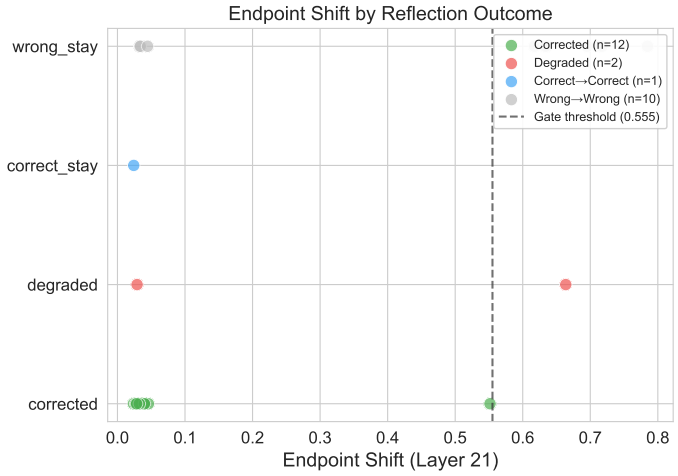

Figure 5: Endpoint shift distributions (L21) by reflection outcome. Correct trajectories (wrong$\rightarrow$correct) show significantly smaller shift than incorrect ones (wrong$\rightarrow$wrong), directly visualizing the separation that underlies AUROC $= 0.85$.

Table 17: Accuracy and L21 curvature by prompt condition and token budget.

Condition		64	128	256	512
Direct	Acc.	0.12	0.12	0.12	0.12
	κ_{21}	1.512	1.512	1.512	1.512
CoT	Acc.	0.12	0.00	0.56	**0.80**
	κ_{21}	1.429	1.391	1.360	1.354
ReAct	Acc.	0.16	0.56	**0.76**	0.76
	κ_{21}	1.445	1.423	1.418	1.418

Table 18: Multi-round reflection: accuracy and L21 curvature by round.

	Round 0	Round 1	Round 2	Round 3
Acc. (ungated)	0.12	0.52	0.52	0.36
Acc. (gated)	0.12	**0.56**	**0.56**	0.40
κ_{21}	1.512	1.420	1.393	1.383
Mean shift	—	0.262	0.335	0.380

J Token budget scaling

K Multi-round reflection

Curvature monotonically decreases across rounds regardless of accuracy changes. Ungated accuracy degrades at round 3 (0.52 $\rightarrow$ 0.36) as correct answers are overwritten. Gating partially mitigates this (0.40 vs. 0.36), but effectiveness decreases as mean endpoint shift increases (0.262 $\rightarrow$ 0.380).

L Temporal stability ablation

Table 19: Temporal stability of projection-head decomposition at layer 14.

Condition	Cosine gap	R (step/para)	Interpretation
Factorized $\mathbf{v}$ (D)	**0.9**	**1.07**	judgment-stable
Unfactored z (F)	0.7	1.53	step-leaking
Shuffled labels (E)	≈ 0.0	—	no structure

Setup. We compare three training conditions for the projection heads using identity-tracking episodes: (D) *Factorized*: the $\mathbf{u}/\mathbf{v}$ decomposition with trained decomposition loss; (F) *Unfactored*: a single projection without the split; (E) *Shuffled*: factorized architecture with permuted labels. We measure (1) inter-entity cosine gap and (2) step-drift ratio $R = d_{\text{step}}/d_{\text{para}}$, where $R \approx 1$ indicates temporal stability.

Findings. The factorized $\mathbf{v}$-space achieves near-perfect temporal invariance ($R = 1.07$), while the unfactored baseline leaks step information ($R = 1.53$). Shuffled labels produce zero inter-entity gap despite identical architecture, confirming that correct supervision is necessary for structured judgment axes. Both factorized and unfactored conditions show high causal selectivity (100–500$\times$ larger displacement for identity-changing perturbations), but only the factorized decomposition maintains this selectivity across reasoning steps.

Cross-seed stability. To validate that the $\mathbf{u}/\mathbf{v}$ decomposition is robust to initialization, we trained projection heads with 3 different random seeds (42, 123, 456) on the same entity-tracking episodes. Across seeds, downstream clustering (NMI) is perfectly stable (1.000 ± 0.000), and the barrier gap shows low variance (1.034 ± 0.024, CV = 2.4%). While the raw weight matrices differ substantially across seeds (principal angles $\approx 90^\circ$), this

is expected: deep networks with L2-normalized outputs admit many parameterizations achieving equivalent function. The key finding is that all seeds converge to *functionally equivalent* representations, confirming the robustness of the geometric decomposition.

Subspace orthogonality and Theorem 3.4. Theorem 3.4 assumes $\mathbf{u} \perp \mathbf{v}$; we verify this empirically. Across layers and questions, the principal angles between learned $\mathbf{u}$ and $\mathbf{v}$ subspaces show: minimum $31.8^\circ \pm 1.3^\circ$, median $64.2^\circ \pm 0.4^\circ$, maximum $89.5^\circ \pm 0.3^\circ$. The median angle ($\sim 64^\circ$) bounds cross-term leakage: $\|\Delta\mathbf{u}_t \cdot \Delta\mathbf{v}_{t+1}\| \leq \cos(64^\circ)\|\Delta\mathbf{u}_t\|\|\Delta\mathbf{v}_{t+1}\| \approx 0.44\|\Delta\mathbf{u}_t\|\|\Delta\mathbf{v}_{t+1}\|$. This bound does not affect the qualitative $\mathbf{u}$-dominant vs. $\mathbf{v}$-dominant conclusions (typical ratios differ by 1.5–2$\times$). Enforcing strict orthogonality via Gram-Schmidt during training would reduce this bound to zero; we leave this refinement as future work.

Random projection ablation. To test whether the $\mathbf{u}/\mathbf{v}$ taxonomy reflects meaningful learned structure (rather than arbitrary projection directions), we compare trained projection heads against 10 randomly-initialized (untrained) projections. We measure the Cohen's d for the CoT vs. few-shot $\mathbf{u}/\mathbf{v}$ displacement-ratio difference. At layer 27, the trained head yields $d = 1.10$ (CoT $>$ few-shot), while **80% of random projections show the opposite pattern** (few-shot $>$ CoT). This indicates that the trained taxonomy captures a specific geometric structure that random projections would typically miss or reverse. At layer 14, the trained head shows $d = 2.34$ with signal-to-noise ratio 0.56 relative to random variation. The consistency of the trained pattern—versus the high variance of random projections—validates that the $\mathbf{u}/\mathbf{v}$ decomposition encodes meaningful condition-specific structure rather than arbitrary subspace partitioning.

Causal intervention procedure. To test whether $\mathbf{u}/\mathbf{v}$ have causal effects on model behavior, we perform activation patching at layer 21. *Implementation*: During the forward pass, we register a hook at layer 21 that modifies the hidden state at the *last token position* (generation-start) by adding $\alpha \cdot \mathbf{d}$, where $\mathbf{d}$ is the unit-normalized $\mathbf{u}$ or $\mathbf{v}$ direction and α=1.0 (approximately one standard deviation of typical hidden-state norms). No re-normalization is applied after perturbation; the perturbed state propagates through remaining layers. We then generate up to 256 tokens with temperature $= 0$ and compare outputs to unperturbed baselines (N=50 questions). Key findings by direction:

- $\mathbf{u}$+: 86% answer change, Jaccard $= 0.24$, len_ratio $= 0.95$
- $\mathbf{u}$-: 96% answer change, Jaccard $= 0.19$, len_ratio $= 0.99$
- $\mathbf{v}$+: 82% answer change, Jaccard $= 0.32$, len_ratio $= 1.01$
- $\mathbf{v}$-: 72% answer change, Jaccard $= 0.20$, len_ratio $= 0.87$

The $\mathbf{v}$+ direction shows notably higher Jaccard (0.32 vs. $\sim$0.2 for other directions), indicating better text structure preservation while still changing answers. This supports the hypothesis that $\mathbf{v}$ affects judgment/answer selection while $\mathbf{u}$ affects reasoning more broadly. The differential effect—$\mathbf{u}$ altering text structure, $\mathbf{v}$+ preserving structure while changing answers—provides causal evidence for the factorized decomposition.

M Predicting reflection success

Using the 100-question reflection data, we test whether endpoint shift at L21 predicts whether reflection will correct an initially-wrong answer. For the 87 initially-wrong cases (40 corrected, 47 uncorrected), lower endpoint shift strongly predicts successful correction (Mann-Whitney $p < 10^{-6}$, Cohen's $d = 1.55$). To validate the AUROC without threshold selection bias, we use 5-fold stratified cross-validation: fold AUROCs range from 0.806 to 0.903, with overall held-out AUROC $= 0.849$ [0.755, 0.919] (bootstrap 95% CI). All layers show similar predictive power (L7: 0.851, L14: 0.853, L21: 0.849, L27: 0.845).

Table 20: Predicting reflection success from geometric signatures (N=87 initially-wrong cases). AUROC validated via 5-fold cross-validation with bootstrap CI.

Metric	Value	95% CI
AUROC (L21 shift, 5-fold CV)	**0.849**	[0.755, 0.919]
Per-fold AUROC mean ± std	0.854	±0.031
Mean shift (will correct)	0.093	±0.18
Mean shift (won't correct)	0.480	±0.31
Cohen's d	1.55	—

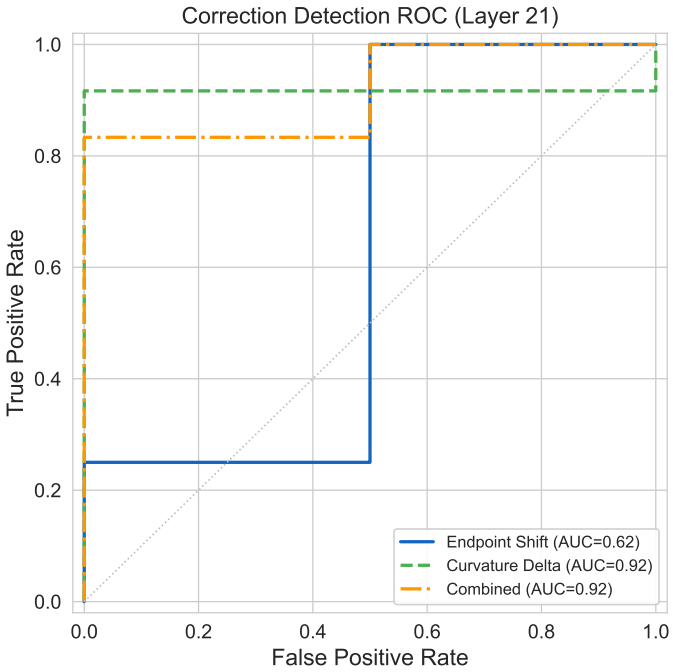

Figure 6: ROC curve for reflection success prediction using endpoint shift at layer 21 (N=87 initially-wrong cases, 5-fold stratified CV). Shaded region shows bootstrap 95% CI. AUROC = 0.85 [0.76, 0.92].

N In-context regime classification

Setup. We test whether geometric regime (the cluster of properties associated with a prompt condition) can be classified from hidden states using ICL. We construct 8 episodes per regime (correct vs. incorrect conditions), extract features at each layer, and train linear probes. We also test **v**-augmentation of ICL prompts.

Table 21: Regime classification: linear probe AUROC by layer.

	L7	L14	L21	L27
Probe AUROC	0.700	0.703	**0.957**	0.927
p-value	0.39	0.44	<0.001	<0.001
Inter-regime dist.	3.6×10^{-5}	8.3×10^{-5}	3.2×10^{-3}	2.9×10^{-3}

Results. At layers 7 and 14, regime classification is near chance (AUROC = 0.700–0.703). At layers 21 and 27, probes achieve AUROC = 0.957 and 0.927 respectively ($p < 0.001$), with inter-regime distance increasing by two orders of magnitude. **v**-augmentation transforms ICL: baseline achieves 25–32% accuracy with 36–50% false-commit rates, while **v**-augmented ICL reaches 49% accuracy with only 2% false commits.

O Decomposition generalization and baselines

Generalization test. To rule out memorization, we conduct a strict evaluation with entity-disjoint train/test splits: entities seen during probe training never appear during evaluation. Under this protocol, NMI remains 1.0 at all layers, confirming that the probes learn the *geometric structure* of judgment identity rather than memorizing specific entity embeddings.

Surface-feature baseline. A TF-IDF classifier on raw token text achieves 100% leaf-type accuracy (since task schemas are semantically distinct), but produces zero structural metrics: Spearman $\rho = 0$ for within-leaf ordering and barrier gap $= 0$. This confirms that the trained decomposition captures geometric structure beyond surface-level task identification—it extracts the *continuous, ordered* foliation that TF-IDF cannot represent.

P Synthetic validation of decomposition

We validate the foliation hypothesis on a synthetic multi-entity identity-tracking task where ground-truth leaf structure is known by construction; generalization to GSM8K is validated via random-projection ablation and causal intervention in Appendix L. Each episode contains K=4 entities undergoing state transitions (e.g., location changes). Hidden states are extracted from Qwen2.5-1.5B-Instruct processing these episodes. The trained decomposition achieves perfect separation: NMI $= 1.000$, ARI $= 1.000$ for $\mathbf{v}$-clustering by entity identity, and Spearman $\rho = 0.955$ for $\mathbf{u}$-rank ordering within leaves. The barrier gap between same-entity and cross-entity $\mathbf{v}$-distances is 1.022 (normalized units). Ablation confirms that $\mathcal{L}_{\text{same-leaf}}$ is the most critical component for clustering quality, while $\mathcal{L}_{\text{order}}$ is essential for within-leaf ranking.

Q Practical application: judgment-preserving rewrite

Beyond diagnostics, the $\mathbf{u}/\mathbf{v}$ decomposition enables *personalized* prompt restructuring: tailoring how context is organized to preserve an individual's judgment frame rather than defaulting to a generic template.

The geometric framework suggests a practical method: restructuring prompts to reduce trajectory divergence while preserving semantic content. We demonstrate a *judgment-preserving rewrite* for multi-entity prompts, where context bleed between entities is a common failure mode.

The rewrite operates in three steps: (1) **Entity detection**: classify entities as primary (action target) or secondary. (2) **Structural isolation**: separate the prompt into `ROLE`, `SCOPE`, and `Conversation` sections, stripping secondary-entity sentences from the conversation. (3) **Classified separation**: classify excluded sentences as action-bearing (placed in `ALSO HANDLE`) or context-only (placed in `CONTEXT ONLY`).

Case study 1: multi-entity finance. A prompt with two customers: User A (primary, validated dispute) and User B (secondary, denied dispute). Original: divergence $= 0.336$ (MEDIUM risk), entity confusion in 1/3 runs. Rewritten: divergence $= 0.206$ (LOW risk), correct processing in 3/3 runs. Reduction: 39%.

Case study 2: IT security. User C (verified engineer, root access) and User D (flagged employee, revoked access). Original: User C's legitimate request denied in 2/3 runs due to User D's flag bleeding. Rewritten: correct in 3/3 runs, each entity evaluated independently.

Both cases demonstrate that context proximity causes representational interference, and structural isolation resolves it.

Table 22: ICL decision quality: **v**-augmentation reduces false commits.

Condition	Metric	0-shot	1-shot	2-shot	4-shot
Baseline	Acc.	0.25	0.30	0.31	0.32
	False commit	0.50	0.40	0.38	0.36
v-aug	Acc.	0.25	**0.48**	**0.49**	**0.49**
	False commit	0.50	0.03	**0.02**	**0.02**

R ICL decision quality

v-descriptor construction. **v**-descriptors are extracted from few-shot exemplars' **v**-projected hidden states at L21. For each exemplar, we project its hidden state onto the learned **v**-subspace and compute the centroid across all exemplars as the "**v**-descriptor vector." Injection: we prepend a line to the ICL prompt: "This question type typically requires [template]" where the template is determined by the nearest-neighbor training exemplar to the centroid. Ablation: random descriptors (from random **v**-vectors) yield false-commit rates near baseline ($\sim$40%), confirming that the **v**-structure—not prompt length—drives the improvement.

S Statistical robustness analyses

Multiple comparison correction. We apply Benjamini-Hochberg correction across all 21 statistical tests reported in the main experiments. All 11 findings significant at $p < 0.05$ remain significant after FDR correction ($\alpha = 0.05$): curvature effects for few-shot vs direct at L7/L14 ($d = 1.9$–2.2), CommonsenseQA divergence/curvature effects across layers ($d = 0.3$–1.2), and reflection prediction ($p < 10^{-6}$). Zero tests lost significance after correction.

Paraphrase semantic equivalence. Using sentence-transformer embeddings (MiniLM-L6-v2), we validate that GPT-4-generated paraphrases preserve semantic content. Across 150 paraphrase pairs: mean cosine similarity $= 0.986 \pm 0.017$, 100% above the 0.8 equivalence threshold, and 100% above the 0.9 high-quality threshold. The lowest-scoring paraphrase still achieves 0.902 similarity.

Per-instance curvature-correctness correlation. At 1.5B scale, pass-2 (post-reflection) curvature shows significant point-biserial correlation with correctness at L7 ($r = 0.26$, $p = 0.01$) and L14 ($r = 0.26$, $p = 0.008$). Curvature change vs correction (initially-wrong cases) shows weak but significant effects at L7/L14 ($r = 0.23$–0.24, $p < 0.05$). Results are mixed across scales, suggesting the correlation is scale-dependent.

T Cross-architecture pilot: TinyLlama

As a pilot replication, we test core geometric signatures on TinyLlama-1.1B-Chat (N=25 questions, layers 5/11/16/21 $\approx$ 25%/50%/75%/95% depth). This small-N study provides qualitative confirmation of architecture generality; we leave larger-N cross-architecture validation for future work.

Curvature patterns replicate. Few-shot shows lower curvature than direct at all layers: L5 (1.86 vs 1.88), L11 (1.65 vs 1.67), L16 (1.62 vs 1.64), L21 (1.79 vs 1.85). This matches the Qwen finding that few-shot contextual grounding linearizes trajectories.

Divergence patterns differ. Unlike Qwen, TinyLlama shows *lower* divergence for few-shot (0.004) than direct (0.023) at late layers. This architecture-dependent behavior suggests that divergence metrics are more sensitive to model family and training regime than curvature metrics.

Implications. Curvature-based signatures (trajectory smoothness, few-shot linearization) appear more robust across architectures than divergence-based signatures. We therefore recommend curvature as the primary cross-architecture diagnostic, with divergence interpreted as a model-family-dependent measure.

U Cross-architecture validation (Gemma-2B)

To further test architecture generality beyond Llama-family models, we replicate divergence experiments on Gemma-2B-it (Gemma Team et al., 2024) (N=15 questions, layers 4/9/13/17 $\approx$ 22%/50%/72%/94% depth across 18 layers).

Divergence patterns replicate. Table 23 shows trajectory divergence by layer and condition. CoT reduces divergence relative to direct by -8.9% on average, matching the Qwen pattern. Few-shot shows a similar effect (-9.1%), though with more layer-dependent variation (ranging from -25.9% at L9 to $+9.7\%$ at L13).

Table 23: Gemma-2B trajectory divergence by layer and condition. CoT and few-shot show reduced divergence relative to direct, replicating Qwen patterns.

Layer	Direct	CoT	Few-shot	CoT Δ
4	0.0001	0.0001	0.0001	-5.7%
9	0.0035	0.0030	0.0026	-14.9%
13	0.0094	0.0086	0.0103	-8.3%
17	0.0073	0.0069	0.0071	-6.6%

Position effect confirms. The prefix vs. suffix divergence pattern replicates: suffix tokens show lower divergence than prefix at deeper layers (L13: -4.1%, L17: -7.0%), while early layers show the opposite (L4: $+275\%$, L9: $+15\%$). This confirms that position effects are architecture-general, not Qwen-specific.

Implications. Combined with TinyLlama results (Appendix T), Gemma validation strengthens the claim that geometric signatures—particularly the CoT divergence reduction and position-dependent effects—generalize across transformer architectures (Qwen, Llama, Gemma).

V Result ledger: claims and data sources

To ensure transparency and reproducibility, Table 24 maps each main claim to its exact data source, readout position, sample size, effect size, and statistical test.

Notation. *Question-endpoint* (QE) refers to the last token of the question text before any suffix; *prompt-endpoint* (PE) refers to the last token of the full prompt (after CoT suffix, few-shot examples, etc.). *Full* span (F) computes metrics over all trajectory tokens; *question-span* (Q) restricts to question tokens only (controlling for prompt-length confounds). Effect sizes are Cohen's d unless otherwise noted.

Statistical corrections. All p-values for multi-layer comparisons (Tables 1, 2) are corrected using Benjamini-Hochberg FDR at α=0.05. Bootstrap 95% CIs are reported for AUROC estimates.

W Fine-tuning prediction experiment

Setup. We test whether base-model $\mathbf{v}$-instability provides a useful frame for understanding fine-tuning outcomes. Using Qwen2.5-1.5B as the base model, we: (1) extract pre-FT

Table 24: Result ledger mapping claims to data sources. Readout: QE = question-endpoint, PE = prompt-endpoint. Span: F = full trajectory, Q = question-span only.

Claim	Data	Readout	Span	N	Effect	Table
Divergence (§5.1)						
CoT ≈ Direct divergence	GSM8K	PE	F	100	$\Delta<0.001$	1
Few-shot ↑ divergence +33–49%	GSM8K	PE	F	100	$p<0.001$	1
Depth-wise convergence (direct)	GSM8K	PE	F	100	0.338→0.288	1
Curvature (§5.2)						
Few-shot ↓ curvature L7–14	GSM8K	PE	F	100	$d=1.73$	2
Span-matched curv. (L27)	GSM8K	QE	Q	100	$d=-0.42$, $p<0.001$	text
CoT/ReAct no curv. diff.	GSM8K	PE	F	100	$p>0.3$	2
Position effects (§5.2)						
Suffix CoT = Direct	GSM8K	PE	F	100	$\Delta<0.001$	App.E
Prefix CoT ↑ 7–14%	GSM8K	PE	F	100	sig.	App.E
u/v decomposition (§5.3)						
CoT **u**-dominant (L14)	GSM8K	PE	F	100	0.053 vs 0.047	3
Few-shot **v**-shift (L14)	GSM8K	PE	F	100	0.042 vs 0.031	3
Random proj. ablation (L27)	GSM8K	PE	F	25	80% opposite	text
Reflection (§5.3)						
Endpoint shift predicts success	GSM8K	PE	F	87	AUROC=0.85	App.M
5-fold CV validation	GSM8K	PE	F	87	[0.76, 0.92]	App.M
Cross-scale (§5.6)						
Curvature strengthens 1.5B→14B	GSM8K	PE	F	100	d=0.89→1.40	5
Cross-architecture						
Gemma-2B CoT ↓ div.	GSM8K	PE	F	25	−8.9%	23
TinyLlama curv. replicates	GSM8K	PE	F	25	qualitative	App.T

geometry (**v**-divergence at layers 7, 14, 21, 27) using the trained JM projection heads; (2) fine-tune with LoRA (rank=16, α=32, targeting q_proj and v_proj) across 5 learning rates (10^{-5} to 10^{-3}) at 200 steps; (3) evaluate accuracy gain on a held-out GSM8K subset (N=100).

Within-model results (Qwen 1.5B). Pre-FT **v**-divergence is a fixed property of the base model (L14: 0.025, L21: 0.026, L27: 0.015); what varies across configurations is fine-tuning intensity (learning rate). Accuracy gains show an inverted-U pattern: lr=1e-5 (+1%), lr=3e-5 (+2%), lr=1e-4 (+12%), lr=3e-4 (+4%), lr=1e-3 (+4%). The peak at lr=1e-4 suggests an optimal intensity for exploiting the base model's **v**-instability.

Cross-model results. We extend to three models at optimal lr (1e-4):

Model	**v**-divergence (L/2)	Gain
Qwen 1.5B	0.025	+12%
Qwen 7B	0.004	+9%
Mistral 7B	0.007	+17%

Spearman correlation: ρ=0.5, p=0.67 (n=3). The positive but non-significant correlation is directionally consistent with the hypothesis that higher **v**-instability predicts larger fine-tuning gains, but the sample size limits statistical power.