NuRA: Numerical Manifolds in Pretrained LLMs Guide Test-Time Adaptation

Anonymous authors
Paper under double-blind review

Abstract

Despite advances in mathematical reasoning, reliable numerical prediction remains a challenge for large language models (LLMs). We find that pretrained numeric output representations exhibit angular relationships associated with magnitude and digit patterns, which we describe as the Numerical Manifold. The patterns extend across tokenizers and model families, including vision-language and math-specialized models. In this work, we propose **Nu**merical **R**epresentation **A**daptation (**NuRA**), which turns pretrained numerical geometry into a label-free learning signal for test-time adaptation. NuRA adapts hidden representations by minimizing angular dispersion, encouraging probability to concentrate on aligned numerical directions without labeled test data or additional offline training. We analyze how pretrained numerical relationships shape probability redistribution during adaptation. Experiments across numerical tasks and model families show broad accuracy gains with updates to approximately 0.01% of MLP parameters.

1 Introduction

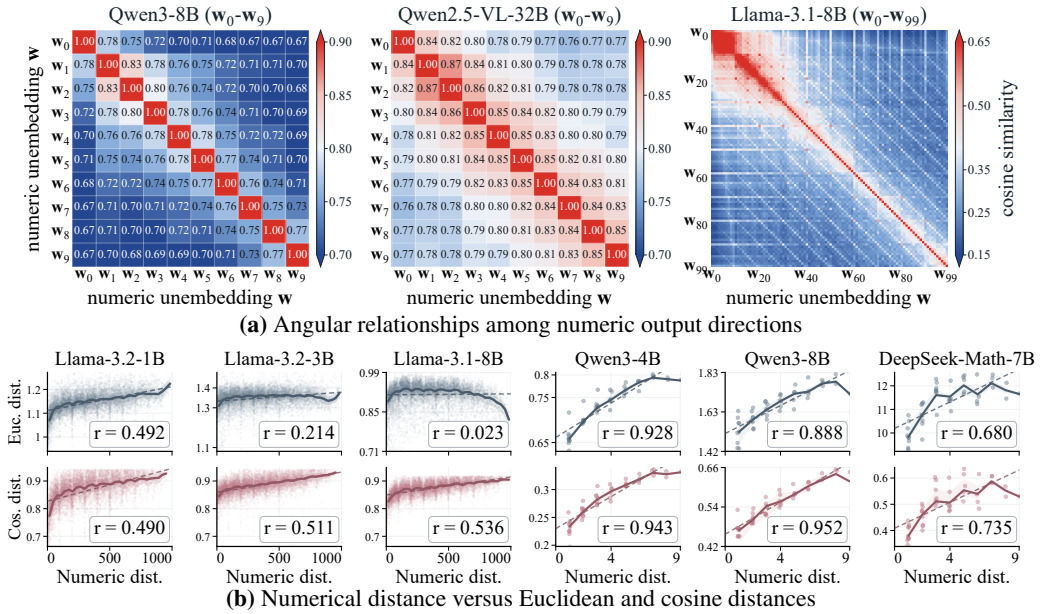
Mathematical reasoning with large language models (LLMs) has progressed to Olympiad-level problem solving and early contributions to mathematical research (Hubert et al., 2026; Feng et al., 2026). Accurate numerical prediction is essential to mathematical problem solving, but even models with strong reasoning performance struggle with basic numerical tasks (Yang et al., 2025; Li et al., 2025). Numerical errors can occur even when the model uses the correct operations, and accuracy can also vary when only the numerical values in a problem are changed (Gao et al., 2023; Mirzadeh et al., 2025). These findings highlight numerical reliability as a critical challenge in LLMs.

Supervised training can improve numerical prediction by using correct target values as a learning signal (Yang et al., 2025; Zausinger et al., 2025). However, gains vary across tasks, and generalization to longer numbers remains challenging (Yang et al., 2025). At inference time, tool-assisted methods improve calculation accuracy through code execution, calculators, or symbolic solvers (Gao et al., 2023; Chen et al., 2023; Schick et al., 2023), but repeated tool calls can increase latency and computational cost (Gao et al., 2025; Su et al., 2026). Test-time adaptation offers another approach by updating pretrained parameters for the current input (Hardt & Sun, 2024; Hu et al., 2025). Results in language modeling show that such updates can improve performance even on domains represented in the pretraining data (Hardt & Sun, 2024), suggesting that adaptation to the current input can provide benefits beyond pretraining. For numerical prediction, these updates require a learning signal that can be evaluated without knowing the correct value. This raises a central question:

What information learned during pretraining can guide reliable numerical prediction at test time?

Prior work shows that numerical values and individual digits can be decoded from hidden representations (Štefánik et al., 2026; Levy & Geva, 2025). Interestingly, we find that numerical structure is also reflected in angular relationships among *numeric output directions*, the normalized output-head vectors for numeric tokens (Figure 1). Similar patterns appear across model families and numerical tokenizers, including in vision-language and math-specialized models. Further examples across six additional models show local numerical structure with model-dependent similarity patterns (Appendix N.1). In the Llama and Qwen comparisons, larger models show stronger correlations between cosine and numerical distances, with a larger difference between cosine and Euclidean correlations. We describe this pretrained angular structure as the **Numerical Manifold**, motivated by the hypothesis that the observed regularities reflect a low-dimensional numerical representation.

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.



(a) Angular relationships among numeric output directions

(b) Numerical distance versus Euclidean and cosine distances

Figure 1: **Numerical relationships in pretrained output representations.** The patterns appear across model families, sizes, and numerical tokenizers, including in language, vision-language, and math-specialized models. **(a)** Pairwise cosine similarity among numeric output directions, with rows and columns indexing numeric tokens in numerical order. Nearby single-digit values in Qwen3-8B and Qwen2.5-VL-32B tend to have more aligned directions, while tokens 0–99 in Llama-3.1-8B exhibit broader similarity bands and recurring digit-dependent patterns. **(b)** Numerical distance on the horizontal axis and Euclidean (top) or cosine (bottom) distance on the vertical axis, with correlations r shown. Cosine distance generally correlates more strongly with numerical distance. In Llama-3.1-8B, the Euclidean correlation is nearly zero ($r = 0.023$), while the cosine correlation ($r = 0.536$) exceeds those of the 1B and 3B models. In the Llama and Qwen comparisons, larger models show stronger cosine correlations and a wider gap relative to Euclidean correlations.

The Numerical Manifold provides a reference for assessing numerical uncertainty in the learned representation. A prediction can remain uncertain about the exact number while assigning most of its probability to nearby representations. To quantify this concentration, we build on uncertainty measures that account for similarities among predictions (Kuhn et al., 2023; Nikitin et al., 2024). Drawing on directional statistics (Banerjee et al., 2005), we use *angular dispersion* to measure the spread of probability mass around its mean direction. Lower angular dispersion indicates greater concentration on the Numerical Manifold and serves as an objective for test-time adaptation.

In this work, we propose **Nu**merical **R**epresentation **A**daptation (**NuRA**), which uses this pretrained geometry to obtain a learning signal without labeled test data or additional offline training. Following work on learning from unlabeled test inputs (Sun et al., 2020; Wang et al., 2021) and entropy minimization for reasoning (Agarwal et al., 2025), NuRA minimizes angular dispersion at numerical decoding steps. It updates selected MLP parameters to adapt the hidden representation. We evaluate NuRA on numerical calculation and number understanding, mathematical reasoning, and visual numerical prediction. Our contributions are as follows:

- We identify angular relationships associated with magnitude and digit structure in pretrained numerical output representations across model families and tokenization schemes.
- We formulate angular dispersion as a measure of geometric uncertainty and use it as the adaptation objective in NuRA. We establish its local geometric interpretation and derive how pretrained angular relationships influence numerical token preferences through parameter updates.
- We demonstrate accuracy gains across numerical tasks and model families by updating approximately 0.01% of MLP parameters. Further analysis shows that highly sparse updates can retain most of the full-update token-accuracy gain.

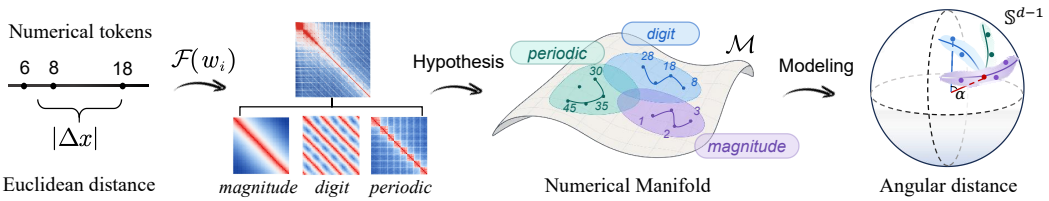


Figure 2: **From numerical values to angular geometry.** Normalizing pretrained output vectors reveals structured angular relationships, motivating the Numerical Manifold hypothesis. Angular distances on the surrounding unit sphere provide the basis for our adaptation objective. The pattern panels, manifold, and spherical arrangement are schematic.

2 Numerical Manifold and NuRA

Numerical relationships in a pretrained language model extend beyond proximity on the number line. Normalized output vectors reveal angular structure associated with both numerical magnitude and digit patterns. We interpret these regularities through the Numerical Manifold hypothesis and use the surrounding unit sphere to measure the geometric spread of numerical predictions (Figure 2). This leads to angular dispersion, a label-free objective for adapting predictions.

2.1 Numerical distributions and pretrained directions

A language model predicts the next token from its final normalized hidden representation $h_\theta(x) \in \mathbb{R}^d$. For context x and a vocabulary $\mathcal{V} = \{v_1, \ldots, v_M\}$ of M tokens, the prediction is

$$p_\theta(v_i \mid x) = \frac{\exp z_i(\theta)}{\sum_{j=1}^{M} \exp z_j(\theta)}, \qquad z_i(\theta) = w_i^\top h_\theta(x) + b_i. \tag{1}$$

Here $w_i^\top$ and b_i are the pretrained output-head row and bias for token v_i (Press & Wolf, 2017; Park et al., 2024). θ denotes the full model parameter vector used to compute $h_\theta(x)$.

NuRA constructs a *temperature-adjusted numeric distribution* over the numeric token set $\mathcal{N} \subset \mathcal{V}$ from the adapted logits:

$$q_i := q^{(\gamma)}_{\theta+\Delta\theta}(n_i \mid x) = \frac{\exp\big(\gamma z_i(\theta + \Delta\theta)\big)}{\sum_{n_j \in \mathcal{N}} \exp\big(\gamma z_j(\theta + \Delta\theta)\big)}, \qquad n_i \in \mathcal{N}. \tag{2}$$

The prediction probability q_i for numeric token $n_i = v_i$ depends on two adjustable quantities: the parameter increment $\Delta\theta$ and inverse temperature γ. The increment $\Delta\theta$ updates selected MLP parameters to change the hidden representation and logits. Increasing the inverse temperature $\gamma > 0$ gives greater relative weight to tokens with higher logits. At $\gamma = 1$, q equals the adapted model's distribution conditioned on the next token belonging to $\mathcal{N}$. During each episode, NuRA optimizes $\Delta\theta$ while keeping γ fixed. Distinct token strings remain separate entries even when they denote the same numerical value.

To represent the geometry of numerical predictions, we use the fixed pretrained output directions:

$$u_i = \mathcal{F}(w_i) := \frac{w_i}{\|w_i\|_2} \in \mathbb{S}^{d-1}, \qquad n_i \in \mathcal{N}. \tag{3}$$

The unit direction u_i is obtained by normalizing the pretrained output vector w_i associated with numeric token n_i. On the Riemannian sphere $\mathbb{S}^{d-1}$ with its standard metric, geodesic distance is angular distance. This sphere provides the ambient space for the *Numerical Manifold* model. Normalization is used only for the geometric objective, while logits are computed from the original output vectors.

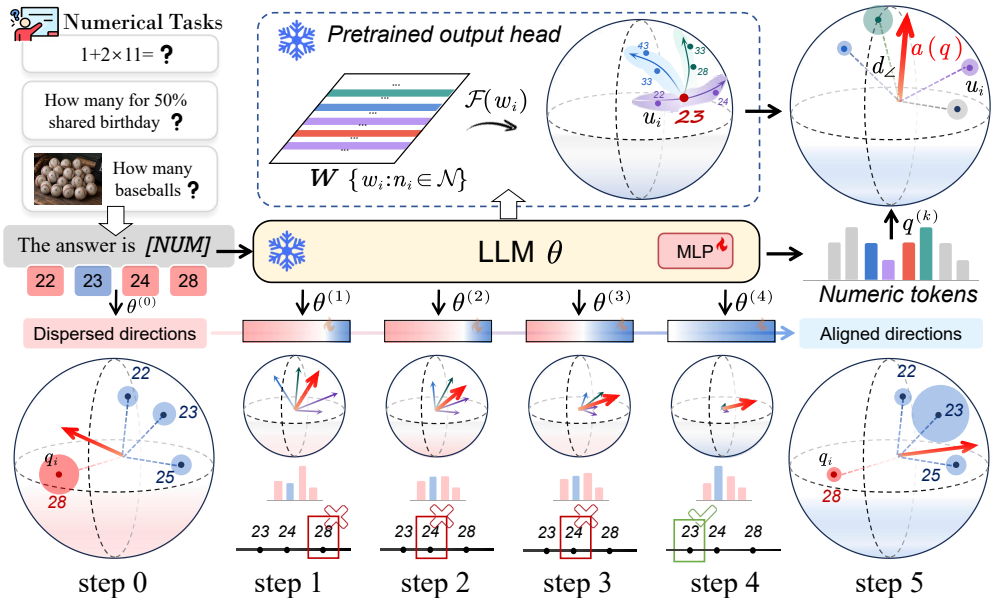


Figure 3: **Overview of** NuRA**.** Selected MLP updates minimize angular dispersion over fixed pretrained output directions. Distributions and trajectories are schematic.

2.2 Angular dispersion on the Numerical Manifold

The geometric regularities in Figure 1, including associations with numerical magnitude and recurring digit-dependent patterns, motivate a low-dimensional model of the pretrained directions u_i. Although numeric tokens are discrete, we model their pretrained directions u_i as points on a smooth embedded submanifold $\mathcal{M} \subset \mathbb{S}^{d-1}$, called the *Numerical Manifold*. Under the local assumptions of Proposition 1, the angular dispersion defined below approximates the probability-weighted mean deviation in local coordinates on $\mathcal{M}$ to first order.

Probability assigned to several tokens can be concentrated in a common region of this representation. To measure that concentration, we first form the normalized probability-weighted mean, following the mean-direction construction in directional statistics (Banerjee et al., 2005):

$$a(q) := \frac{\sum_{n_i \in \mathcal{N}} q_i u_i}{\left\| \sum_{n_i \in \mathcal{N}} q_i u_i \right\|_2} \in \mathbb{S}^{d-1}. \tag{4}$$

The *anchor* $a(q)$ summarizes the current numeric distribution q. Its separation from each numeric direction is measured by the angular distance $d_\angle(u, v) := \arccos(u^\top v)$ for $u, v \in \mathbb{S}^{d-1}$ (Boumal, 2023). We define *angular dispersion* as

$$D_\angle(q) := \sum_{n_i \in \mathcal{N}} q_i \, d_\angle\big(u_i, a(q)\big). \tag{5}$$

In Eq. (5), the anchor $a(q)$ serves as the reference direction determined by q, and $D_\angle(q)$ measures the probability-weighted angular deviation from it. A larger value indicates greater geometric spread of the numeric distribution around the anchor, while a smaller value indicates concentration on nearby directions, even when probability is shared among distinct numeric tokens. We therefore minimize angular dispersion during adaptation to encourage probability mass to concentrate on aligned numeric directions. This objective measures geometric concentration rather than correctness: a distribution concentrated around an incorrect candidate can also have low dispersion.

2.3 TEST-TIME ADAPTATION VIA ANGULAR DISPERSION MINIMIZATION

Let $\mathcal{L}(\theta; x)$ denote the angular dispersion of the numeric distribution induced by θ, and write $q^{(k)} := q^{(\gamma)}_{\theta^{(k)}}(\cdot \mid x)$ at iteration k. NuRA seeks a single-step increment $\Delta\theta$ on the selected MLP entries:

$$\min_{\Delta\theta} \mathcal{L}(\theta^{(k)} + \Delta\theta; x), \qquad \mathcal{L}(\theta^{(k)}; x) = D_{\angle}\big(q^{(k)}\big). \tag{6}$$

Starting from pretrained parameters $\theta^{(0)}$, we take one masked gradient step at each iteration:

$$\theta^{(k+1)} = \theta^{(k)} - \eta_k \, s \odot \nabla_\theta \mathcal{L}(\theta^{(k)}; x), \qquad k = 0, \dots, K-1, \tag{7}$$

where s is a binary mask and η_k is the effective step size. Among eligible MLP entries, s selects a prescribed fraction with the largest initial gradient magnitudes and remains fixed throughout the episode.

Each iteration recomputes $q^{(k)}$ and $a(q^{(k)})$ and differentiates through their dependence on the current parameters. After K updates, NuRA records the adapted full-vocabulary logits for decoding and restores $\theta^{(0)}$. The recorded logits are not rescaled by γ. Figure 3 illustrates the adaptation process, and Algorithm 1 summarizes the procedure.

3 UNDERSTANDING NUMERICAL ADAPTATION

We first relate angular dispersion to local concentration on $\mathcal{M}$, then analyze how one update in Eq. (7) changes token preferences. The two results use the following geometric and differentiability assumptions, respectively. Throughout, x, $\mathcal{N}$, γ, the output head, and s are fixed.

Assumption 1 (Local geometry)**.** Every $n_i \in \mathcal{N}$ has direction $u_i = \Phi(\xi_i)$, where $\xi_i \in \mathbb{R}^r$ is its local coordinate, $r = \dim \mathcal{M}$, and $\|\xi_i\|_2 \le \varepsilon$ for a radius $\varepsilon > 0$. The fixed embedding Φ from a neighborhood of 0 into $\mathcal{M} \subset \mathbb{S}^{d-1}$ is twice continuously differentiable (C^2). Its Jacobian satisfies $\Phi'(0)^\top \Phi'(0) = I_r$, the $r \times r$ identity.

Assumption 2 (Differentiable update)**.** At the current parameters, the computed angular loss is differentiable in the numeric logits, the logits are differentiable in θ, and, for the token pair considered, $z_i - z_j$ is C^2 nearby.

Proposition 1 (Dispersion in numerical coordinates)**.** *Under Assumption 1, for sufficiently small ε, the anchor exists and*

$$\left| D_{\angle}(q) - \sum_{n_i \in \mathcal{N}} q_i \|\xi_i - \bar{\xi}\|_2 \right| \le C\varepsilon^2, \qquad \bar{\xi} := \sum_{n_i \in \mathcal{N}} q_i \xi_i. \tag{8}$$

The constant C depends only on the fixed patch, uniformly over all probability weights q.

Thus, angular dispersion approximates the mean deviation from the weighted coordinate mean $\bar{\xi}$ to first order (proof in Appendix B.3).

To connect this geometric objective to prediction, consider one step of Eq. (7), with $\theta = \theta^{(k)}$, $\theta^+ = \theta^{(k+1)}$, and $\eta = \eta_k$. Let Δ denote the change from θ to θ^+.

Proposition 2 (Numerical preference under adaptation)**.** *Under Assumption 2, let $g_\ell = \partial\mathcal{L}/\partial z_\ell$ include the anchor derivative. For $v_i, v_j \in \mathcal{V}$, as $\eta \to 0$,*

$$\Delta \log \frac{p_\theta(v_i \mid x)}{p_\theta(v_j \mid x)} = -\eta \sum_{n_\ell \in \mathcal{N}} g_\ell \, \langle \nabla_\theta (z_i - z_j), \, s \odot \nabla_\theta z_\ell \rangle + O(\eta^2). \tag{9}$$

A positive left-hand side means increased preference for v_i over v_j. The inner products describe how selected parameters translate the geometric signal g_ℓ into this change. Appendix C gives the full gradient and proof; the result does not by itself guarantee improved accuracy.

Table 1: **Numerical calculation and number understanding.** Answer accuracy (A.Acc.) and token accuracy (T.Acc.), in percent. Best and second-best results are bold and underlined, respectively.

Method	Addition-50		NUPA		BB-Arith		NB-Mixed		NB-Arith		MATH401		Avg.	
	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.
Llama-3.1-8B-Instruct														
Greedy	16.00	35.59	19.39	37.24	68.03	72.02	4.50	4.00	17.98	46.42	21.95	30.70	24.64	37.66
Temp. Sampling	16.00	33.99	21.67	36.21	66.31	70.68	3.50	2.98	16.08	46.70	21.70	29.94	24.21	36.75
R2L Formatting	18.00	47.69	10.27	21.64	64.16	69.18	3.00	2.84	15.53	41.37	22.19	27.30	22.19	35.00
LTPO	14.00	34.58	16.54	32.39	66.52	70.08	3.50	3.00	14.99	42.50	20.95	30.01	22.75	35.43
Entropy Min.	20.00	44.96	19.77	37.42	66.95	67.41	4.50	4.09	15.26	45.23	21.70	29.45	24.70	38.09
Perplexity Min.	20.00	41.58	18.63	32.90	68.67	72.73	3.00	3.00	12.26	42.35	22.69	30.90	24.21	37.24
Real-Line Disp.	22.00	46.30	19.39	30.24	64.59	66.67	4.00	4.90	16.89	46.18	22.44	29.49	24.89	37.30
NuRA **(Ours)**	**24.00**	**56.39**	**24.14**	**41.87**	**69.10**	**75.45**	**5.00**	**5.65**	**20.44**	**48.26**	**22.94**	**33.08**	**27.60**	**43.45**
Qwen3-4B-Instruct-2507														
Greedy	42.00	42.65	33.84	43.53	76.39	81.36	7.00	18.54	47.14	41.29	17.96	31.80	37.39	43.20
Temp. Sampling	42.00	42.65	35.55	43.99	76.39	81.51	8.50	20.48	47.41	39.00	18.20	31.48	38.01	43.19
R2L Formatting	**44.00**	**56.00**	25.86	42.71	75.54	71.73	3.50	13.87	46.59	47.48	**18.45**	24.22	35.66	42.67
LTPO	42.00	43.48	31.94	42.81	77.04	82.66	**9.50**	23.27	46.05	55.94	18.20	31.85	37.46	46.67
Entropy Min.	40.00	39.38	34.60	43.25	76.18	80.12	7.00	17.45	47.68	41.25	18.20	31.81	37.28	42.21
Perplexity Min.	42.00	42.16	34.60	42.48	76.39	81.55	9.00	20.00	47.14	41.47	17.96	31.84	37.85	43.25
Real-Line Disp.	42.00	42.99	34.03	44.14	77.04	83.60	8.00	**23.96**	49.59	58.67	18.20	32.75	38.14	47.69
NuRA **(Ours)**	42.00	54.65	**36.69**	**44.24**	**77.25**	**85.39**	**9.50**	23.08	**53.68**	**63.55**	17.46	**33.45**	**39.43**	**50.73**

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

We evaluate six numerical calculation and number understanding datasets, four mathematical reasoning datasets, and three visual benchmarks across Llama, Qwen, DeepSeek-Math, DeepSeek-R1-Distill, Qwen2.5-VL, and InternVL (Appendix E). We report answer and token accuracy for numerical and visual prediction, and answer accuracy with mean generation length for reasoning. Token accuracy measures partial agreement within numerical answers. Baselines include greedy decoding, temperature sampling (Holtzman et al., 2020), R2L formatting (Singh & Strouse, 2024), self-consistency (Wang et al., 2023), LTPO (Ye et al., 2026), entropy minimization (Wang et al., 2021; Agarwal et al., 2025), input perplexity minimization (Hu et al., 2025), and real-line dispersion, which minimizes predictive numerical variance without labels (Appendix E.1).

4.2 MAIN RESULTS

Numerical prediction. NuRA achieves the highest average answer and token accuracy on both models, outperforming every baseline on all six Llama datasets (Table 1). On Qwen, higher token accuracy does not always improve complete-answer accuracy (Appendix F).

Mathematical reasoning. NuRA improves DeepSeek-Math-7B accuracy without longer outputs (Table 2). Perplexity minimization shortens outputs further but reduces accuracy. Whereas it optimizes input-sequence loss, NuRA refines each triggered numerical prediction (Appendix G).

Visual numerical prediction. NuRA achieves the highest answer accuracy across chart reading, visual mathematics, and counting on Qwen2.5-VL-3B (Table 3; Appendix H).

Numerical error detection. Before adaptation, angular dispersion improves AUROC and AURC over numeric entropy across all eight Llama-3.2-1B/3B dataset pairs under reference answer prefixes; AUROC gains are 0.0038–0.0444. Permuting directions among tokens worsens both metrics despite unchanged probabilities, supporting the value of learned token–direction correspondence (Appendix I).

Table 2: **Mathematical reasoning on DeepSeek-Math-7B.** Answer accuracy (A.Acc., %) and mean generation length (Len., tokens).

Method	GSM8K		MATH500		TabMWP		FinQA		Avg.	
	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.
Greedy	82.49	161.4	44.20	325.6	83.00	105.0	10.20	139.8	54.97	182.9
Temp. Sampling	77.94	161.8	34.60	350.1	80.80	104.7	10.90	140.5	51.06	189.3
Self-Consistency	82.03	166.6	42.60	354.0	**84.00**	105.1	9.94	139.2	54.64	191.2
LTPO	82.49	161.0	39.40	341.8	82.20	102.8	11.16	**127.8**	53.81	183.4
Entropy Min.	82.41	161.4	43.80	321.2	83.40	105.0	9.94	139.8	54.89	181.8
Perplexity Min.	78.17	**154.7**	40.00	**304.2**	82.60	**98.1**	10.20	140.6	52.74	**174.4**
Real-Line Disp.	82.49	168.1	44.20	324.2	83.00	105.3	10.90	138.1	55.15	183.9
NuRA	**82.64**	161.4	**44.40**	324.2	83.80	104.8	**12.03**	135.3	**55.72**	181.4

Table 3: **Visual numerical prediction.** Qwen2.5-VL-3B answer accuracy (A.Acc., %).

Method	ChartQA	MathVista	TallyQA	Avg.
Greedy	61.40	46.10	7.40	38.30
Temp. Sampling	61.00	39.80	5.60	35.47
Entropy Min.	63.60	47.80	13.60	41.67
Perplexity Min.	59.20	44.20	8.00	37.13
Real-Line Disp.	63.40	48.80	11.20	41.13
NuRA	**65.00**	**50.20**	**14.40**	**43.20**

Table 4: NuRA **design ablations.** Numerical token accuracy (%) on the 200-question mixture.

Variant	Llama-1B	Llama-3B	Qwen-4B
Before adaptation	20.36	38.23	66.49
Geometry and objective			
Permuted directions	20.36	36.25	65.26
Cosine dispersion	30.48	51.62	**70.67**
Fixed anchor	24.08	44.97	67.23
Full vocabulary	33.10	49.65	68.11
Parameter updates			
LayerNorm only	19.39	36.58	63.33
Last 4 MLP layers	21.71	38.52	66.57
NuRA **(full)**	**33.54**	**53.43**	68.77

Table 5: **Generation costs across calculation, reasoning, and visual tasks.** Total latency (s, mean ± std) and mean generated length (Len., tokens). Avg. uses all four datasets. Llama-3.2-3B-Instruct is used for the three text datasets and Qwen2.5-VL-3B-Instruct for ChartQA.

Method	Addition		MATH401		GSM8K		ChartQA		Avg.	
	Total (s)	Len.	Total (s)	Len.	Total (s)	Len.	Total (s)	Len.	Total (s)	Len.
Direct inference	0.138 ±0.001	3.5	0.124 ±0.003	3.1	0.106 ±0.000	2.7	0.282 ±0.000	3.5	0.162 ±0.001	3.2
Program-aided	2.333 ±0.007	60.5	2.638 ±0.039	65.2	–	–	–	–	–	–
Chain-of-Thought	–	–	–	–	6.714 ±0.043	169.6	8.220 ±0.071	133.8	–	–
Full MLP (100%)	2.407 ±0.031	3.5	1.403 ±0.023	3.1	1.185 ±0.004	2.7	4.227 ±0.019	3.5	2.306 ±0.017	3.2
Sparse (0.01%)	2.298 ±0.135	3.5	1.344 ±0.126	3.1	1.128 ±0.082	2.7	4.131 ±0.092	3.5	2.225 ±0.108	3.2
Sparse (0.0001%)	2.107 ±0.216	3.5	1.172 ±0.054	3.1	1.022 ±0.089	2.7	4.066 ±0.227	3.5	2.092 ±0.146	3.2

4.3 Adaptation analysis

Design ablations. Permuting directions lowers accuracy below the unadapted model by 1.98 and 1.23 points on Llama-3B and Qwen-4B (Table 4). Fixed anchors, full-vocabulary objectives, and last-four-layer updates fall below full NuRA on all three models; normalization-only updates harm accuracy. Angular dispersion exceeds cosine dispersion on Llama, but not Qwen (Appendix J).

Generation cost. Sparse updates preserve direct inference's mean output length on timed examples and take less time than the corresponding program-aided or chain-of-thought baseline on three of four datasets (Table 5). They remain slower than direct inference; forward and backward passes limit runtime savings from sparsity (Appendix P).

Sparse updates. On Llama-3.2-3B, fixed selection of $10^{-5}\%$ of candidate parameters nearly matches full-update token accuracy and performs similarly to dynamic selection (Figure 4). On DeepSeek-Math-7B evaluated on MATH401, sparse changes concentrated in early layers still shift later representations (Figure 5), consistent with effects propagating through fixed downstream parameters (Appendices K and L).

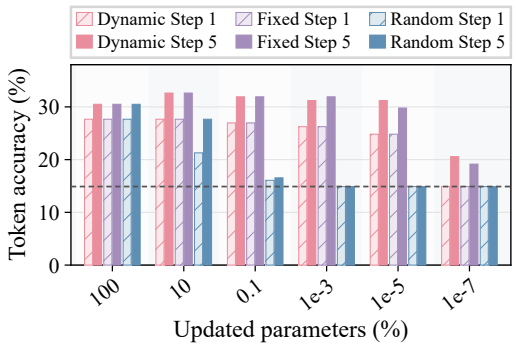


Figure 4: **Sparse updates on Llama-3.2-3B.** Token accuracy after one and five updates; the dashed line marks unadapted accuracy.

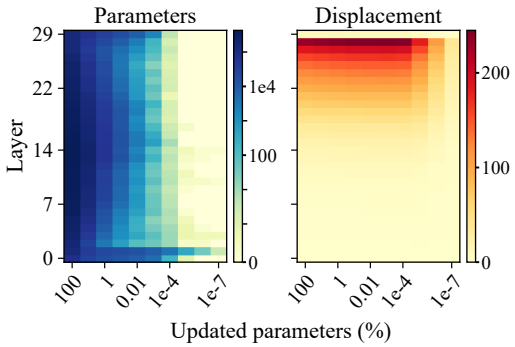
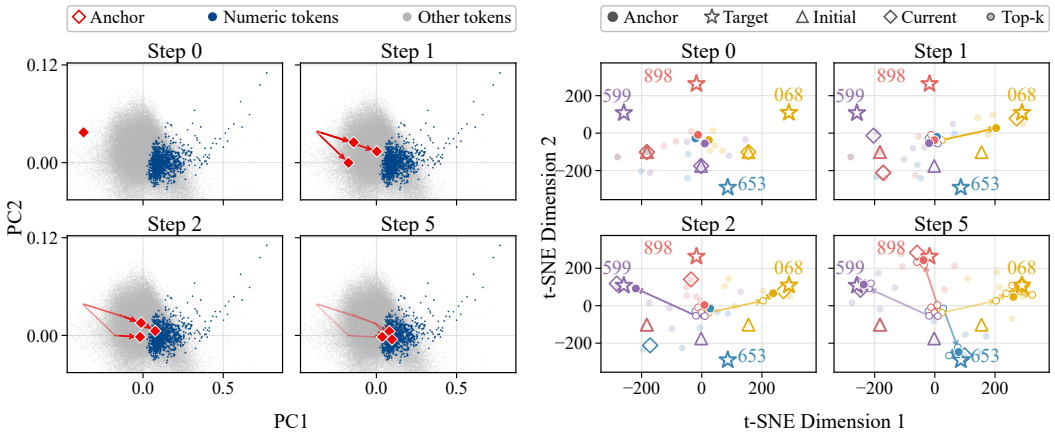


Figure 5: **Representation changes.** Layerwise parameter changes (left) and hidden-state L_2 shifts after five updates (right).

Anchor trajectories. Selected anchors move toward numerical directions and distinct correct targets during adaptation (Figure 6). Paired cases show that `NuRA` can correct predictions while entropy minimization reinforces incorrect candidates, suggesting that correction depends on where probability moves (Appendices N and O). The vocabulary and numerical views use fixed projections across updates, so the displayed movement reflects changes in posterior anchors rather than changes in the projection. The examples illustrate input-dependent refinement: initially similar anchors move toward different target directions. These projected paths are qualitative and do not establish distances or convergence in the original representation space.

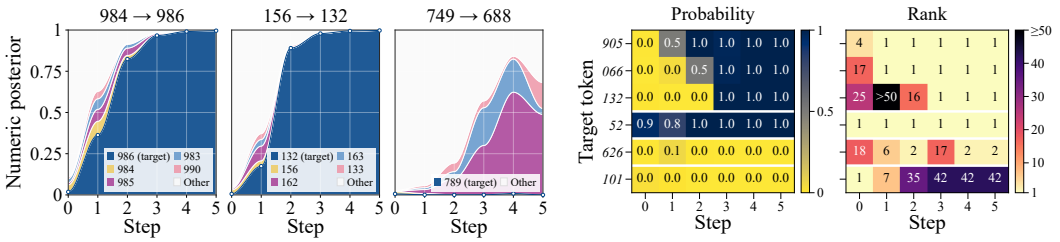


(a) Vocabulary-space trajectories

(b) Target-level trajectories

Figure 6: **Posterior anchor dynamics on Llama-3.2-3B.** (a) Local view of three selected anchors in a fixed full-vocabulary PCA projection; blue and gray points denote numerical and other output directions. (b) Four corrected positions in a shared t-SNE projection of numerical output directions; stars mark targets, triangles initial predictions, and diamonds current predictions. Both views show steps 0, 1, 2, and 5 from separate runs. Full trajectories and protocols are in Appendices N and O.

Redistribution and failures. Target probability rises in the corrected cases, but incorrect candidates can also attract later updates through the posterior-dependent anchor (Figure 7). Failed cases may improve the target rank only temporarily, exposing a limit of local refinement (Appendices O.1 and O.2). A higher target rank need not change the decoded answer if another candidate remains more probable. Tracking both quantities helps distinguish partial progress from correction.



(a) Numerical posterior

(b) Target probability and rank

Figure 7: **Numerical prediction dynamics on Llama-3.2-3B.** (a) Two corrections and one failure; targets are blue, and curves interpolate steps. (b) Rounded target probabilities and ranks from a separate run; colors saturate above rank 50 (Appendices O.1 and O.2).

The corrected cases show increasing target probability alongside better anchor alignment, consistent with numerical geometry guiding probability redistribution. The failed case illustrates a complementary limitation: because the anchor depends on the current distribution, a wrong candidate can remain influential during later updates. Their contrast with entropy minimization suggests that concentration alone is insufficient; which numerical directions receive probability also matters.

Hyperparameter sensitivity. The preferred inverse temperature depends on the task: GSM8K benefits from a lower value than Addition-50 and MATH401 (Figure 34; Appendix Q). A lower inverse temperature distributes adaptation weights more evenly across numerical candidates, allowing plausible alternatives to influence the anchor. Lowering it too far reduces accuracy, however, indicating that refinement also requires sufficient discrimination among candidates. Most gains occur within the first few updates, with limited benefit from additional steps over the tested range.

5 Related Work

Numerical representations and prediction. Analyses of pretrained LLMs identify numerical magnitude, digit structure, and sinusoidal patterns in their representations (Zhu et al., 2025; Levy & Geva, 2025; Kadlčík et al., 2025). Causal analyses connect arithmetic predictions to specific model components (Stolfo et al., 2023; Nikankin et al., 2025). Complementary approaches introduce numerical structure through continuous encodings, digit positions, Fourier features, or geometric embeddings (Golkar et al., 2023; McLeish et al., 2024; Zhou et al., 2026; Jin et al., 2026). Number Token Loss incorporates numerical distance into supervised training (Zausinger et al., 2025). The pretrained output head provides a reference for adapting numerical representations without labeled test answers.

Uncertainty and test-time adaptation. Semantic entropy and similarity-based uncertainty measures account for relationships among generated answers (Kuhn et al., 2023; Nikitin et al., 2024). Test-time learning uses objectives available without test labels, including self-supervised losses and prediction entropy (Sun et al., 2020; Wang et al., 2021). For LLMs, adaptation can use retrieved text or input perplexity (Hardt & Sun, 2024; Hu et al., 2025), while entropy minimization has been studied through finetuning, reinforcement learning, and logit adjustment (Agarwal et al., 2025). Appendix D expands on numerical evaluation, representation geometry, uncertainty estimation, and test-time learning.

6 Conclusion

In this work, we introduce `NuRA`, a test-time adaptation method that uses pretrained numerical geometry to refine predictions without labeled data. By minimizing angular dispersion through sparse MLP updates, `NuRA` improves accuracy across numerical tasks and model families. Our theoretical analysis connects this objective to local geometric concentration and changes in token preferences. These findings support pretrained geometry as an adaptation signal, while failures reveal the risk of reinforcing incorrect candidates.

AI USE STATEMENT

We used AI tools to assist with drafting and polishing the manuscript and to identify relevant prior work. The authors take responsibility for the final content.

ETHICS STATEMENT

This work introduces NuRA, a test-time adaptation method for numerical prediction with pretrained language models. Our evaluation of NuRA focuses on direct numerical prediction, mathematical and quantitative reasoning, and multimodal numerical question answering. The study examines numerical accuracy and the computational cost of test-time adaptation. Within this evaluation scope, we do not identify specific safety or ethical concerns associated with the proposed method or its evaluation.

REPRODUCIBILITY STATEMENT

We describe NuRA's numerical representations, angular dispersion objective, and test-time adaptation procedure in Section 2, including parameter selection, masked updates, and restoration of the pretrained parameters. Section 3 states the assumptions and main theoretical results. Appendices B and C provide the geometric derivations, full gradient, and proofs, with numerical safeguards documented in Appendix C. Appendix E presents the assumptions, proofs, and illustrative constructions for the conditional comparison with token-entropy adaptation.

REFERENCES

Manoj Acharya, Kushal Kafle, and Christopher Kanan. TallyQA: Answering complex counting questions. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):8076–8084, 2019. doi: 10.1609/aaai.v33i01.33018076. URL https://ojs.aaai.org/index.php/AAAI/article/view/4815.

Shivam Agarwal, Zimin Zhang, Lifan Yuan, Jiawei Han, and Hao Peng. The Unreasonable Effectiveness of Entropy Minimization in LLM Reasoning. In *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 107150–107180. Curran Associates, Inc., 2025. doi: 10.52202/085713-3573. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/9a14cf2ee36cf2524f54d8f196a6734c-Paper-Conference.pdf.

Arindam Banerjee, Inderjit S. Dhillon, Joydeep Ghosh, and Suvrit Sra. Clustering on the unit hypersphere using von Mises-Fisher distributions. *Journal of Machine Learning Research*, 6 (46):1345–1382, 2005. URL https://www.jmlr.org/papers/v6/banerjee05a.html.

Nora Belrose, Igor Ostrovsky, Lev McKinney, Zach Furman, Logan Smith, Danny Halawi, Stella Biderman, and Jacob Steinhardt. Eliciting Latent Predictions from Transformers with the Tuned Lens. *arXiv preprint arXiv:2303.08112*, 2023. URL https://arxiv.org/abs/2303.08112.

BIG-bench authors. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=uyTL5Bvosj.

Nicolas Boumal. *An Introduction to Optimization on Smooth Manifolds*. Cambridge University Press, 2023. doi: 10.1017/9781009166164. URL https://www.nicolasboumal.net/book/.

Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *Transactions on Machine Learning Research*, 2023. URL https://openreview.net/forum?id=YfZ4ZPt8zd.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan Routledge, and William Yang Wang. FinQA: A dataset of numerical reasoning over financial data. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical*

Methods in Natural Language Processing, pp. 3697–3711, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.300. URL https://aclanthology.org/2021.emnlp-main.300/.

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. URL https://arxiv.org/abs/2110.14168.

Josh Engels, Eric Michaud, Isaac Liao, Wes Gurnee, and Max Tegmark. Not All Language Model Features Are One-Dimensionally Linear. In *International Conference on Learning Representations*, volume 2025, pp. 84591–84622, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/d3221cdb27e49d9c1cd35ad254feccfe-Paper-Conference.pdf.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630, 2024. doi: 10.1038/s41586-024-07421-0. URL https://www.nature.com/articles/s41586-024-07421-0.

Tony Feng, Trieu H. Trinh, Garrett Bingham, Dawsen Hwang, Yuri Chervonyi, Junehyuk Jung, Joonkyung Lee, Carlo Pagano, Sang-hyun Kim, Federico Pasqualotto, Sergei Gukov, Jonathan N. Lee, Junsu Kim, Kaiying Hou, Golnaz Ghiasi, Yi Tay, YaGuang Li, Chenkai Kuang, Yuan Liu, Hanzhao Lin, Evan Zheran Liu, Nigamaa Nayakanti, Xiaomeng Yang, Heng-Tze Cheng, Demis Hassabis, Koray Kavukcuoglu, Quoc V. Le, and Thang Luong. Towards autonomous mathematics research. *arXiv preprint arXiv:2602.10177*, 2026. URL https://arxiv.org/abs/2602.10177.

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. PAL: Program-aided language models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 10764–10799. PMLR, 2023. URL https://proceedings.mlr.press/v202/gao23f.html.

Silin Gao, Jane Dwivedi-Yu, Ping Yu, Xiaoqing Ellen Tan, Ramakanth Pasunuru, Olga Golovneva, Koustuv Sinha, Asli Celikyilmaz, Antoine Bosselut, and Tianlu Wang. Efficient tool use with chain-of-abstraction reasoning. In Owen Rambow, Leo Wanner, Marianna Apidianaki, Hend Al-Khalifa, Barbara Di Eugenio, and Steven Schockaert (eds.), *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 2727–2743, Abu Dhabi, UAE, January 2025. Association for Computational Linguistics. URL https://aclanthology.org/2025.coling-main.185/.

Yonatan Geifman and Ran El-Yaniv. SelectiveNet: A deep neural network with an integrated reject option. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2151–2159. PMLR, 2019. URL https://proceedings.mlr.press/v97/geifman19a.html.

Mor Geva, Avi Caciularu, Kevin Wang, and Yoav Goldberg. Transformer Feed-Forward Layers Build Predictions by Promoting Concepts in the Vocabulary Space. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 30–45. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.emnlp-main.3. URL https://aclanthology.org/2022.emnlp-main.3/.

Siavash Golkar, Mariel Pettee, Michael Eickenberg, Alberto Bietti, Miles Cranmer, Geraud Krawezik, Francois Lanusse, Michael McCabe, Ruben Ohana, Liam Parker, Bruno Régaldo-Saint Blancard, Tiberiu Tesileanu, Kyunghyun Cho, and Shirley Ho. xVal: A Continuous Numerical Tokenization for Scientific Language Models. *arXiv preprint arXiv:2310.02989*, 2023. URL https://arxiv.org/abs/2310.02989.

Yves Grandvalet and Yoshua Bengio. Semi-supervised Learning by Entropy Minimization. In *Advances in Neural Information Processing Systems*, volume 17. MIT Press, 2004. URL https://proceedings.neurips.cc/paper_files/paper/2004/file/96f2b50b5d3613adf9c27049b2a888c7-Paper.pdf.

Moritz Hardt and Yu Sun. Test-Time Training on Nearest Neighbors for Large Language Models. In *International Conference on Learning Representations*, volume 2024, pp. 54625–54640, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/f02f1185b97518ab5bd7ebde466992d3-Paper-Conference.pdf.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Hkg4TI9xl.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring Mathematical Problem Solving With the MATH Dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper_files/paper/2021/file/be83ab3ecd0db773eb2dc1b0a17836a1-Paper-round2.pdf.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. In *International Conference on Learning Representations*, 2020. URL https://arxiv.org/abs/1904.09751.

Jinwu Hu, Zitian Zhang, Guohao Chen, Xutao Wen, Chao Shuai, Wei Luo, Bin Xiao, Yuanqing Li, and Mingkui Tan. Test-Time Learning for Large Language Models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 24823–24849. PMLR, 2025. URL https://proceedings.mlr.press/v267/hu25z.html.

Thomas Hubert, Rishi Mehta, Laurent Sartran, Miklós Z. Horváth, Goran Žužić, Eric Wieser, Aja Huang, Julian Schrittwieser, Yannick Schroecker, Hussain Masoom, Ottavia Bertolli, Tom Zahavy, Amol Mandhane, Jessica Yung, Iuliya Beloshapka, Borja Ibarz, Vivek Veeriah, Lei Yu, Oliver Nash, Paul Lezeau, Salvatore Mercuri, Calle Sönne, Bhavik Mehta, Alex Davies, Daniel Zheng, Fabian Pedregosa, Yin Li, Ingrid von Glehn, Mark Rowland, Samuel Albanie, Ameya Velingker, Simon Schmitt, Edward Lockhart, Edward Hughes, Henryk Michalewski, Nicolas Sonnerat, Demis Hassabis, Pushmeet Kohli, and David Silver. Olympiad-level formal mathematical reasoning with reinforcement learning. *Nature*, 651(8106):607–613, 2026. doi: 10.1038/s41586-025-09833-y. URL https://www.nature.com/articles/s41586-025-09833-y.

Shengkai Jin, Tianyu Chen, Chonghan Gao, and Jun Han. GeoNum: Bridging Numerical Continuity and Language Semantics via Geometric Embedding. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(27):22426–22434, 2026. doi: 10.1609/aaai.v40i27.39401. URL https://doi.org/10.1609/aaai.v40i27.39401.

Marek Kadlčík, Michal Štefánik, Timothee Mickus, Josef Kuchař, and Michal Spiegel. Pre-trained Language Models Learn Remarkably Accurate Representations of Numbers. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 26705–26714. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.1356. URL https://aclanthology.org/2025.emnlp-main.1356/.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2302.09664.

Amit Arnold Levy and Mor Geva. Language Models Encode Numbers Using Digit Representations in Base 10. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pp. 385–395. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.naacl-short.33. URL https://aclanthology.org/2025.naacl-short.33/.

Haoyang Li, Xuejia Chen, Zhanchao Xu, Darian Li, Nicole Hu, Fei Teng, Yiming Li, Luyu Qiu, Chen Jason Zhang, Li Qing, and Lei Chen. Exposing Numeracy Gaps: A Benchmark to Evaluate Fundamental Numerical Abilities in Large Language Models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 20004–20026. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-acl.1026. URL https://aclanthology.org/2025.findings-acl.1026/.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's Verify Step by Step. In *International Conference on Learning Representations*, volume 2024, pp. 39578–39601, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/aca97732e30bcf1303bc22ac3924fd16-Paper-Conference.pdf.

Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. Generating with Confidence: Uncertainty Quantification for Black-box Large Language Models. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=DWkJCSxKU5.

Pan Lu, Liang Qiu, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, Tanmay Rajpurohit, Peter Clark, and Ashwin Kalyan. Dynamic prompt learning via policy gradient for semi-structured mathematical reasoning. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://openreview.net/forum?id=DHyHRBwJUTN.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/663bce02a0050c4a11f1eb8a7f1429d3-Abstract-Conference.html.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 2263–2279, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-acl.177. URL https://aclanthology.org/2022.findings-acl.177/.

Sean McLeish, Arpit Bansal, Alex Stein, Neel Jain, John Kirchenbauer, Brian R. Bartoldson, Bhavya Kailkhura, Abhinav Bhatele, Jonas Geiping, Avi Schwarzschild, and Tom Goldstein. Transformers Can Do Arithmetic with the Right Embeddings. In *Advances in Neural Information Processing Systems*, volume 37, pp. 108012–108041. Curran Associates, Inc., 2024. doi: 10.52202/079017-3430. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/c35986bc1ee29b31c1011481b77fe540-Paper-Conference.pdf.

Iman Mirzadeh, Keivan Alizadeh-Vahid, Hooman Shahrokhi, Oncel Tuzel, Samy Bengio, and Mehrdad Farajtabar. GSM-Symbolic: Understanding the Limitations of Mathematical Reasoning in Large Language Models. In *International Conference on Learning Representations*, volume 2025, pp. 94743–94765, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/ec2e7a896f8250986b3907f57621ce94-Paper-Conference.pdf.

Jian Mu, Qixin Zhang, Zhiyong Wang, Menglin Yang, Shuang Qiu, Chengwei Qin, Zhongxiang Dai, and Yao Shu. Self-reflective generation at test time. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10221–10239, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.465. URL https://aclanthology.org/2026.acl-long.465/.

Yaniv Nikankin, Anja Reusch, Aaron Mueller, and Yonatan Belinkov. Arithmetic Without Algorithms: Language Models Solve Math with a Bag of Heuristics. In *International Conference on Learning Representations*, volume 2025, pp. 55939–55965, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/8c5f30296296d2ae402ebbd09aaa9c12-Paper-Conference.pdf.

Alexander Nikitin, Jannik Kossen, Yarin Gal, and Pekka Marttinen. Kernel Language Entropy: Fine-grained Uncertainty Quantification for LLMs from Semantic Similarities. In *Advances in Neural Information Processing Systems*, volume 37, pp. 8901–8929. Curran Associates, Inc., 2024. doi: 10.52202/079017-0283. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/10c456d2160517581a234dfde15a7505-Paper-Conference.pdf.

Kiho Park, Yo Joong Choe, and Victor Veitch. The Linear Representation Hypothesis and the Geometry of Large Language Models. In *Proceedings of the 41st International Conference on*

Machine Learning, volume 235, pp. 39643–39666. PMLR, 2024. URL https://proceedings.mlr.press/v235/park24c.html.

Kiho Park, Yo Joong Choe, Yibo Jiang, and Victor Veitch. The Geometry of Categorical and Hierarchical Concepts in Large Language Models. In *International Conference on Learning Representations*, volume 2025, pp. 76441–76463, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/be7430d22a4dae8516894e32f2fcc6db-Paper-Conference.pdf.

Ofir Press and Lior Wolf. Using the Output Embedding to Improve Language Models. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pp. 157–163. Association for Computational Linguistics, 2017. URL https://aclanthology.org/E17-2025/.

Yi Ren and Danica J. Sutherland. Learning dynamics of LLM finetuning. In *International Conference on Learning Representations*, volume 2025, pp. 70523–70563, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/afe1aa79e5eea7955f553c61a307273e-Abstract-Conference.html.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessi, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 68539–68551. Curran Associates, Inc., 2023. doi: 10.52202/075280-2997. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/d842425e4bf79ba039352da0f658a906-Paper-Conference.pdf.

Aaditya K. Singh and DJ Strouse. Tokenization counts: the impact of tokenization on arithmetic in frontier LLMs. *arXiv preprint arXiv:2402.14903*, 2024. URL https://arxiv.org/abs/2402.14903.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM Test-Time Compute Optimally Can be More Effective than Scaling Parameters for Reasoning. In *International Conference on Learning Representations*, volume 2025, pp. 10131–10165, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/1b623663fd9b874366f3ce019fdfdd44-Paper-Conference.pdf.

Michal Štefánik, Timothee Mickus, Marek Kadlčík, Bertram Højer, Michal Spiegel, Raúl Vázquez, Aman Sinha, Josef Kuchař, Philipp Mondorf, and Pontus Stenetorp. Language Models Learn Universal Representations of Numbers and Here's Why You Should Care. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 30663–30681. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.1415. URL https://aclanthology.org/2026.acl-long.1415/.

Alessandro Stolfo, Yonatan Belinkov, and Mrinmaya Sachan. A Mechanistic Interpretation of Arithmetic Reasoning in Language Models using Causal Mediation Analysis. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7035–7052. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.435. URL https://aclanthology.org/2023.emnlp-main.435/.

Qisheng Su, Shiting Huang, Zhen Fang, Ziyan Chen, Zehui Chen, and Feng Zhao. Beyond accuracy: Unveiling inefficiency patterns in tool-integrated reasoning. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7456–7477, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.339. URL https://aclanthology.org/2026.acl-long.339/.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-Time Training with Self-Supervision for Generalization under Distribution Shifts. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 9229–9248. PMLR, 2020. URL https://proceedings.mlr.press/v119/sun20b.html.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully Test-time Adaptation by Entropy Minimization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=uXl3bZLkr3c.

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. URL https://arxiv.org/abs/2203.11171.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed 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. Curran Associates, Inc., 2022. doi: 10.52202/068431-1800. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/9d5609613524ecf4f15af0f7b31abca4-Paper-Conference.pdf.

Haotong Yang, Yi Hu, Shijia Kang, Zhouchen Lin, and Muhan Zhang. Number Cookbook: Number Understanding of Language Models and How to Improve It. In *International Conference on Learning Representations*, volume 2025, pp. 3946–3984, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/0b8ccc9229328bdcedf54b989e7cc330-Paper-Conference.pdf.

Wengao Ye, Yan Liang, and Lianlei Shan. Thinking on the fly: Test-time reasoning enhancement via latent thought policy optimization. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/pdf/0e76aa031ef0c2effee86e9fea5f4fe79d0076ec.pdf.

Zheng Yuan, Hongyi Yuan, Chuanqi Tan, Wei Wang, and Songfang Huang. How well do Large Language Models perform in Arithmetic tasks? *arXiv preprint arXiv:2304.02015*, 2023. URL https://arxiv.org/abs/2304.02015.

Jonas Zausinger, Lars Pennig, Anamarija Kozina, Sean Sdahl, Julian Sikora, Adrian Dendorfer, Timofey Kuznetsov, Mohamad Hagog, Nina Wiedemann, Kacper Chlodny, Vincent Limbach, Anna Ketteler, Thorben Prein, Vishwa Mohan Singh, Michael Danziger, and Jannis Born. Regress, Don't Guess: A Regression-like Loss on Number Tokens for Language Models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 73995–74017. PMLR, 2025. URL https://proceedings.mlr.press/v267/zausinger25a.html.

Tianyi Zhou, Deqing Fu, Mahdi Soltanolkotabi, Robin Jia, and Vatsal Sharan. FoNE: Precise Single-Token Number Embeddings via Fourier Features. In *International Conference on Learning Representations*, volume 2026, pp. 28485–28516, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/30253ebdaa75eed08c4b10c75aa8f781-Paper-Conference.pdf.

Fangwei Zhu, Damai Dai, and Zhifang Sui. Language Models Encode the Value of Numbers Linearly. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 693–709. Association for Computational Linguistics, 2025. URL https://aclanthology.org/2025.coling-main.47/.

A NOTATION

All method and analysis expressions refer to one fixed context x. The numeric set and inverse temperature γ are fixed within an adaptation episode. The index k denotes an adaptation iteration, not a generation position. Vectors are columns, and $w_i^\top$ is a row of the output head. The parameters θ denote the full model parameter vector, as in Section 2.1. The mask s selects eligible MLP entries, so the output head and all unselected parameters remain fixed along each update. The parameter analysis uses the masked chain rule in Eq. (C.9). A superscript $+$ denotes the state after one update; q includes the fixed dependence on γ. Primes denote differentials: $f'(x)$ is represented by the Jacobian for a Euclidean map, $f'(x)[b]$ is its action on a direction, and $f''(x)[b, c]$ is the second differential evaluated on two directions. For a scalar-valued Euclidean function, $f'(x)[b] = \nabla f(x)^\top b$.

Symbol	Meaning
$\mathcal{V}, \mathcal{N}$	Vocabulary and numeric token subset.
x	Fixed prompt, preceding tokens, and any non-text inputs.
$\theta, \theta^{(k)}$	Full model parameters and their values after k updates.
$h_\theta(x) \in \mathbb{R}^d$	Hidden representation supplied to the output head.
$w_i \in \mathbb{R}^d, b_i$	Fixed output vector and optional bias for token i.
$z_i, p_\theta(v_i \mid x), q_i$	Logit, vocabulary probability, and temperature-adjusted numeric probability.
$u_i, a(q)$	Unit output direction and normalized probability-weighted mean.
$d_\angle, D_\angle, \mathcal{L}$	Angular distance, angular dispersion, and parameter objective.
γ	Inverse temperature used only in the adaptation weights.
s, η_k, K	Fixed binary mask, effective step size, and adaptation budget.
$\mathcal{M}, r$	Modeled numerical manifold and its dimension.
$\xi_i, \bar{\xi}, \varepsilon$	Local numerical coordinates, their probability-weighted mean, and patch radius.
α_i, ρ, g_i	Angle to the anchor, norm of the unnormalized mean, and loss derivative with respect to logit i.
$\Delta\theta, \Delta h$	Parameter increment supported on selected MLP entries and change in the final representation.

Table 6: Notation for the method and its analysis. Constants and auxiliary vectors used in an individual proof are defined in that proof.

When matrices are needed in the appendix, $W \in \mathbb{R}^{|\mathcal{V}| \times d}$ has rows $w_i^\top$, whereas $U \in \mathbb{R}^{d \times |\mathcal{N}|}$ has columns u_i in the same order as q. Thus $Wh_\theta(x)$ produces logits and Uq produces the weighted mean direction. The raw output vectors are not replaced by the normalized vectors when computing logits. Distinct token spellings remain distinct vocabulary entries even when they denote the same numerical value.

B GEOMETRY OF ANGULAR DISPERSION

The objective is defined on fixed pretrained directions and the probabilities assigned to them. We first express it through pairwise numerical relationships, then prove its local coordinate interpretation. Throughout this appendix, sums are over the fixed numeric set $\mathcal{N}$ unless otherwise indicated.

The angular distance $d_\angle$ need not equal the intrinsic distance on $\mathcal{M}$, and the anchor $a(q)$ need not belong to $\mathcal{M}$.

B.1 Dependence on pretrained numerical relationships

Let U collect the unit numeric directions as columns and let $G = U^\top U$. Writing $m = Uq$ and $\rho = \|m\|_2$, direct expansion gives

$$\begin{aligned} \rho^2 &= \left(\sum_i q_i u_i\right)^\top \left(\sum_j q_j u_j\right) = \sum_{i,j} q_i q_j u_i^\top u_j = q^\top G q, \\ u_i^\top a(q) &= \frac{u_i^\top \sum_j q_j u_j}{\rho} = \frac{\sum_j G_{ij} q_j}{\sqrt{q^\top G q}}, \\ D_\angle(q) &= \sum_i q_i \arccos\left(\frac{(Gq)_i}{\sqrt{q^\top G q}}\right). \end{aligned} \tag{B.1}$$

Thus, for fixed probabilities, the objective is determined by the pairwise relationships among numeric directions. A common orthogonal change of coordinates preserves G and the loss, whereas reassigning probabilities to different numerical directions can change the loss. The Gram matrix is an analytical expression and need not be formed when computing NuRA.

B.2 Normalization of the mean direction

The following elementary lemma is used to control the anchor in the coordinate expansion, differentiate it, and analyze the temperature limits.

Lemma B.1 (Normalization). *For $N(y) = y/\|y\|_2$ and nonzero $y, b \in \mathbb{R}^d$,*

$$\|N(y) - N(b)\|_2 \le \frac{2\|y-b\|_2}{\|b\|_2}, \qquad N'(y)[v] = \frac{I - N(y)N(y)^\top}{\|y\|_2}\, v. \tag{B.2}$$

Proof. Add and subtract $y/\|b\|_2$, and apply the reverse triangle inequality:

$$\begin{aligned} \left\| \frac{y}{\|y\|_2} - \frac{b}{\|b\|_2} \right\|_2 &\le \left\| \frac{y}{\|y\|_2} - \frac{y}{\|b\|_2} \right\|_2 + \frac{\|y-b\|_2}{\|b\|_2} \\ &= \frac{|\|b\|_2 - \|y\|_2| + \|y-b\|_2}{\|b\|_2} \le \frac{2\|y-b\|_2}{\|b\|_2}. \end{aligned} \tag{B.3}$$

For the differential, the quotient and chain rules give

$$\begin{aligned} N'(y)[v] &= \frac{v}{\|y\|_2} - \frac{y}{\|y\|_2^2}\frac{y^\top v}{\|y\|_2} \\ &= \frac{1}{\|y\|_2}\left(I - \frac{yy^\top}{\|y\|_2^2}\right) v. \end{aligned} \tag{B.4}$$

□

B.3 Proof of Proposition 1

A smooth embedded manifold admits local coordinates (Boumal, 2023, Theorem 3.12 and Section 8.1); a C^2 parameterization has a first-order Taylor approximation with a quadratic remainder. Under Assumption 1, ξ_i describes the learned representation of token n_i, rather than its numerical value on the real line. These coordinates interpret the objective; NuRA evaluates Eq. (5) directly from the pretrained directions. We write $\Phi'(\xi)$ for the Jacobian and $\Phi''(\xi)[b, c]$ for the second differential applied to b, c.

The proof expands the directions and their normalized mean in the same coordinates, then bounds the difference between chord length and angular distance. The local assumption covers every direction receiving positive probability; concentration of only a subset requires an additional bound on the remaining mass.

Proof. Let Φ be a C^2 embedding defined on an open neighborhood of $\overline{B}_r(0, \varepsilon_0)$, with $u_i = \Phi(\xi_i)$, $u_0 = \Phi(0)$, and $E = \Phi'(0)$. Choose coordinates with $E^\top E = I_r$, as allowed by a linear coordinate change for an embedded manifold (Boumal, 2023, Theorem 3.12 and Section 8.1). Since $\|\Phi(\xi)\|_2^2 = 1$, differentiating at the origin also gives $u_0^\top E = 0$. On the closed coordinate ball, choose finite constants B, L such that

$$\|\Phi''(\xi)[b, b]\|_2 \le B\|b\|_2^2, \qquad \|\Phi'(\xi)\|_{\mathrm{op}} \le L. \tag{B.5}$$

The same linear map E represents the individual directions and their weighted mean. Taylor expansion with an integral remainder yields

$$\begin{aligned} u_i &= \Phi(0) + \Phi'(0)[\xi_i] + \int_0^1 (1-t)\, \Phi''(t\xi_i)[\xi_i, \xi_i]\, \mathrm{d}t \\ &= u_0 + E\xi_i + R_i, \\ \|R_i\|_2 &\le \int_0^1 (1-t) B\|\xi_i\|_2^2\, \mathrm{d}t = \frac{B}{2}\|\xi_i\|_2^2. \end{aligned} \tag{B.6}$$

Write $\bar{\xi} = \sum_i q_i \xi_i$, $\bar{R} = \sum_i q_i R_i$, and $\tau^2 = \sum_i q_i \|\xi_i\|_2^2 \le \varepsilon^2$. Averaging the expansion and using Jensen's inequality gives

$$\begin{aligned} m = \sum_i q_i u_i &= u_0 + E\bar{\xi} + \bar{R}, \\ \|\bar{R}\|_2 &\le \sum_i q_i \|R_i\|_2 \le \frac{B}{2}\tau^2, \\ \|\bar{\xi}\|_2^2 &= \left\| \sum_i q_i \xi_i \right\|_2^2 \le \sum_i q_i \|\xi_i\|_2^2 = \tau^2. \end{aligned} \tag{B.7}$$

To control the normalization of m, put $b = u_0 + E\bar{\xi}$. Orthogonality implies

$$\begin{aligned} \|b\|_2^2 &= 1 + \|\bar{\xi}\|_2^2 \ge 1, \\ \|m\|_2 &\ge \|b\|_2 - \|\bar{R}\|_2 \ge 1 - \frac{B}{2}\varepsilon^2, \\ \|b\|_2 - 1 &= \frac{\|\bar{\xi}\|_2^2}{\sqrt{1 + \|\bar{\xi}\|_2^2} + 1} \le \frac{1}{2}\tau^2. \end{aligned} \tag{B.8}$$

Hence the anchor exists when $B\varepsilon^2/2 < 1$. Lemma B.1 now bounds its deviation from the same linearized mean:

$$\begin{aligned} \|a(q) - (u_0 + E\bar{\xi})\|_2 &\le \|N(m) - N(b)\|_2 + \|N(b) - b\|_2 \\ &\le \frac{2\|\bar{R}\|_2}{\|b\|_2} + \big(\|b\|_2 - 1\big) \\ &\le \left(B + \frac{1}{2}\right)\tau^2. \end{aligned} \tag{B.9}$$

Subtract this mean from each direction. Since E preserves lengths in the coordinate space, the reverse triangle inequality gives

$$\begin{aligned} \big|\|u_i - a(q)\|_2 - \|\xi_i - \bar{\xi}\|_2\big| &= \big|\|u_i - a(q)\|_2 - \|E(\xi_i - \bar{\xi})\|_2\big| \\ &\le \|R_i\|_2 + \|a(q) - (u_0 + E\bar{\xi})\|_2, \\ \sum_i q_i \big|\|u_i - a(q)\|_2 - \|\xi_i - \bar{\xi}\|_2\big| &\le \frac{B}{2}\tau^2 + \left(B + \frac{1}{2}\right)\tau^2 = \frac{3B+1}{2}\tau^2. \end{aligned} \tag{B.10}$$

It remains to replace chord lengths by the angular distances in the loss. The derivative bound in Eq. (B.5) and Lemma B.1 imply

$$\begin{aligned} \|u_i - u_0\|_2 &\le L\varepsilon, \\ \|a(q) - u_0\|_2 &\le 2\|m - u_0\|_2 \le 2\sum_i q_i \|u_i - u_0\|_2 \le 2L\varepsilon, \\ \|u_i - a(q)\|_2 &\le 3L\varepsilon. \end{aligned} \tag{B.11}$$

For two unit vectors, chord length s and angle α satisfy $s = 2\sin(\alpha/2)$ and $\alpha \le \pi s/2$. Using $\sin t \ge t - t^3/6$ for $0 \le t \le \pi/2$, we obtain

$$\begin{aligned} 0 \le \alpha_i - \|u_i - a(q)\|_2 &= \alpha_i - 2\sin(\alpha_i/2) \\ &\le \frac{\alpha_i^3}{24} \le \frac{(3\pi L/2)^3}{24}\varepsilon^3. \end{aligned} \tag{B.12}$$

Combining the two distance comparisons returns to the original objective:

$$\begin{aligned} \left| D_\angle(q) - \sum_i q_i \|\xi_i - \bar{\xi}\|_2 \right| &\le \sum_i q_i \left|\alpha_i - \|u_i - a(q)\|_2\right| \\ &\quad + \sum_i q_i \left| \|u_i - a(q)\|_2 - \|\xi_i - \bar{\xi}\|_2 \right| \\ &\le \frac{(3\pi L/2)^3}{24}\varepsilon^3 + \frac{3B+1}{2}\tau^2 \\ &\le \left[\frac{3B+1}{2} + \frac{(3\pi L/2)^3}{24}\varepsilon_0 \right] \varepsilon^2. \end{aligned} \tag{B.13}$$

Choose ε_0 small enough that $B\varepsilon_0^2/2 < 1$. The bracketed quantity is then a finite constant independent of q, proving Proposition 1. In particular, this bound applies to any of the temperature-adjusted weights while the stated patch condition holds. □

B.4 Probability weighting and the anchor

The inverse temperature changes the probabilities used in the geometric reference, without changing the pretrained directions. With $Z = \sum_{j\in\mathcal{V}} e^{z_j}$, normalizing powers of the model probabilities gives

$$\begin{aligned} \frac{p_\theta(i \mid x)^\gamma}{\sum_{j\in\mathcal{N}} p_\theta(j \mid x)^\gamma} &= \frac{(e^{z_i}/Z)^\gamma}{\sum_{j\in\mathcal{N}} (e^{z_j}/Z)^\gamma} \\ &= \frac{e^{\gamma z_i}}{\sum_{j\in\mathcal{N}} e^{\gamma z_j}} = \mathrm{softmax}(\gamma z_{\mathcal{N}})_i. \end{aligned} \tag{B.14}$$

Thus $\gamma = 1/T$ corresponds to softmax temperature T. We write $q(\gamma)$ only in this subsection, where γ varies and the logits are fixed. The following consequence of Eq. (B.14) and Lemma B.1 makes the two limiting regimes explicit.

Corollary 1 (Sharp and uniform weighting)**.** *Fix finitely many numeric logits and unit directions. If the largest numeric logit belongs to a unique token $i_\star$, then*

$$q_{i_\star}(\gamma) \longrightarrow 1, \qquad a(q(\gamma)) \longrightarrow u_{i_\star} \quad \text{as } \gamma \to \infty. \tag{B.15}$$

As $\gamma \downarrow 0$, the weights tend to the uniform distribution. If $\sum_i u_i \neq 0$, then

$$q_i(\gamma) \longrightarrow \frac{1}{|\mathcal{N}|}, \qquad a(q(\gamma)) \longrightarrow \frac{\sum_i u_i}{\|\sum_i u_i\|_2}. \tag{B.16}$$

The latter limit is independent of the input logits.

Proof. For a unique maximizer, put $\Delta = z_{i_\star} - \max_{j\neq i_\star} z_j > 0$ and $n = |\mathcal{N}| > 1$. The probability outside $i_\star$ is bounded by

$$\begin{aligned} 1 - q_{i_\star}(\gamma) &= \frac{\sum_{j\neq i_\star} \exp[-\gamma(z_{i_\star} - z_j)]}{1 + \sum_{j\neq i_\star} \exp[-\gamma(z_{i_\star} - z_j)]} \\ &\le \sum_{j\neq i_\star} e^{-\gamma\Delta} = (n-1)e^{-\gamma\Delta}, \\ \|m(\gamma) - u_{i_\star}\|_2 &= \left\| \sum_{j\neq i_\star} q_j(\gamma)(u_j - u_{i_\star}) \right\|_2 \\ &\le 2\big(1 - q_{i_\star}(\gamma)\big). \end{aligned} \tag{B.17}$$

Consequently $m(\gamma) \neq 0$ for all sufficiently large γ. Applying Lemma B.1 with $b = u_{i_\star}$ gives

$$\begin{aligned} \|a(q(\gamma)) - u_{i_\star}\|_2 &\le 2\|m(\gamma) - u_{i_\star}\|_2 \\ &\le 4(n-1)e^{-\gamma\Delta}, \\ d_\angle\big(a(q(\gamma)), u_{i_\star}\big) &\le \frac{\pi}{2}\|a(q(\gamma)) - u_{i_\star}\|_2 \le 2\pi(n-1)e^{-\gamma\Delta}. \end{aligned} \tag{B.18}$$

This proves the first limit and gives an explicit rate. For the uniform limit, let $\bar{z} = n^{-1}\sum_i z_i$. Expanding the finite exponentials at zero yields

$$\begin{aligned} q_i(\gamma) &= \frac{1 + \gamma z_i + O(\gamma^2)}{n + \gamma\sum_j z_j + O(\gamma^2)} \\ &= \frac{1}{n} + \frac{\gamma}{n}(z_i - \bar{z}) + O(\gamma^2), \\ m(\gamma) &= \frac{1}{n}\sum_i u_i + \frac{\gamma}{n}\sum_i (z_i - \bar{z})u_i + O(\gamma^2). \end{aligned} \tag{B.19}$$

The remainder constants depend on the fixed logits and directions. If $m_0 = n^{-1}\sum_i u_i \neq 0$, the normalization bound implies $\|a(q(\gamma)) - N(m_0)\|_2 = O(\gamma)$ and proves Eq. (B.16). For $n = 1$, both claims hold directly. □

With tied maximum logits, the sharp limit is uniform over the maximizing tokens, and its anchor is their normalized mean when that mean is nonzero. The corollary characterizes which directions influence the anchor; the directions themselves and the dimension of the representation model do not change with γ.

C Full Gradient and Numerical Preference

The calculation below differentiates through both the probability weights and their mean direction. The numeric support and γ are fixed, the weighted mean is nonzero, and all angles lie strictly between zero and π. These are the smoothness conditions for the exact-angle objective used in the method and analysis. Numerical evaluation floors normalization denominators at 10^{-4} and clips cosine arguments to $[-1 + 10^{-4}, 1 - 10^{-4}]$ before applying $\arccos$.

C.1 Differentiating the probability weights and their mean

Let $m = \sum_i q_i u_i$, $\rho = \|m\|_2$, and $a = m/\rho$. By Lemma B.1,

$$\begin{aligned} \mathrm{d}m &= \sum_i u_i \,\mathrm{d}q_i, & \mathrm{d}\rho &= a^\top \mathrm{d}m, \\ \mathrm{d}a &= \frac{\mathrm{d}m}{\rho} - \frac{m}{\rho^2}\,\mathrm{d}\rho = \frac{I - aa^\top}{\rho}\sum_i u_i \,\mathrm{d}q_i. \end{aligned} \tag{C.1}$$

Write $\alpha_i = \arccos(u_i^\top a)$ and introduce the auxiliary directions

$$e_i = \frac{u_i - \cos\alpha_i\, a}{\sin\alpha_i}, \qquad \bar{e} = \sum_i q_i e_i. \tag{C.2}$$

Each e_i is perpendicular to a, as is $\mathrm{d}a$. These directions have a precise sphere-geometric interpretation: Boumal's Example 10.21, Eq. (10.6), gives $\mathrm{Log}_a(u_i) = \alpha_i e_i$. For fixed q, the sphere gradient of $\sum_i q_i d_\angle(u_i, a)$ is $-\bar{e}$ by Definition 3.58 (Boumal, 2023). Differentiating the angles and then their

weighted average therefore gives the continuous chain

$$\begin{aligned}
\mathrm{d}\alpha_i &= -\frac{u_i^\top \mathrm{d}a}{\sqrt{1-(u_i^\top a)^2}} = -e_i^\top \mathrm{d}a, \\
\mathrm{d}D_\angle &= \sum_i \alpha_i \,\mathrm{d}q_i + \sum_i q_i \,\mathrm{d}\alpha_i \\
&= \sum_i \alpha_i \,\mathrm{d}q_i - \bar{e}^\top \mathrm{d}a \\
&= \sum_i \alpha_i \,\mathrm{d}q_i - \frac{\bar{e}^\top (I - aa^\top)}{\rho} \sum_i u_i \,\mathrm{d}q_i \\
&= \sum_i \left(\alpha_i - \frac{u_i^\top \bar{e}}{\rho}\right) \mathrm{d}q_i.
\end{aligned} \tag{C.3}$$

The anchor contribution has zero weighted mean, since

$$\sum_i q_i \frac{u_i^\top \bar{e}}{\rho} = \frac{(\sum_i q_i u_i)^\top \bar{e}}{\rho} = \frac{m^\top \bar{e}}{\rho} = a^\top \bar{e} = 0. \tag{C.4}$$

The scaled softmax in Eq. (2) satisfies

$$\begin{aligned}
\frac{\partial q_j}{\partial z_i} &= \frac{\gamma \mathbf{1}\{i=j\} e^{\gamma z_j} \sum_\ell e^{\gamma z_\ell} - \gamma e^{\gamma z_j} e^{\gamma z_i}}{(\sum_\ell e^{\gamma z_\ell})^2} \\
&= \gamma q_j \big(\mathbf{1}\{i=j\} - q_i\big).
\end{aligned} \tag{C.5}$$

Substituting this differential into Eq. (C.3) and using Eq. (C.4) produces

$$\begin{aligned}
g_i = \frac{\partial D_\angle}{\partial z_i} &= \gamma \sum_j q_j \left(\alpha_j - \frac{u_j^\top \bar{e}}{\rho}\right) \big(\mathbf{1}\{i=j\} - q_i\big) \\
&= \gamma q_i \left[\alpha_i - \frac{u_i^\top \bar{e}}{\rho} - \sum_j q_j \left(\alpha_j - \frac{u_j^\top \bar{e}}{\rho}\right)\right] \\
&= \gamma q_i \left[\alpha_i - D_\angle(q) - \frac{u_i^\top \bar{e}}{\rho}\right].
\end{aligned} \tag{C.6}$$

Expanding $\bar{e}$ makes the pairwise contribution to the anchor derivative explicit:

$$\begin{aligned}
u_i^\top \bar{e} &= \sum_j q_j \frac{u_i^\top (u_j - \cos\alpha_j \, a)}{\sin\alpha_j} \\
&= \sum_j q_j \frac{u_i^\top u_j - \cos\alpha_i \cos\alpha_j}{\sin\alpha_j}.
\end{aligned} \tag{C.7}$$

$$g_i \coloneqq \frac{\partial D_\angle}{\partial z_i} = \gamma q_i \left[\alpha_i - D_\angle(q) - \frac{1}{\rho} \sum_{n_j \in \mathcal{N}} q_j \frac{u_i^\top u_j - \cos\alpha_i \cos\alpha_j}{\sin\alpha_j}\right]. \tag{C.8}$$

Increasing a token's probability changes both its contribution to the average distance and the anchor. The first two terms compare its angular distance with the current average; the last term accounts for anchor movement through pairwise relationships among pretrained directions. Thus, assigning the same probability values to a different configuration of directions can change the adaptation signal.

The resulting logit gradient sums to zero. Through the fixed output head, the same signal reaches the hidden representation and MLP parameters by

$$\begin{aligned}
\nabla_h \mathcal{L} &= \sum_{n_i \in \mathcal{N}} g_i w_i, \\
s \odot \nabla_\theta \mathcal{L} &= s \odot \left[\left(\frac{\partial h_\theta(x)}{\partial \theta}\right)^\top \sum_{n_i \in \mathcal{N}} g_i w_i\right] = \sum_{n_i \in \mathcal{N}} g_i \big(s \odot \nabla_\theta z_i\big).
\end{aligned} \tag{C.9}$$

The fixed directions define the signal g_i, while the second line describes how the current model implements that signal through a parameter update.

C.2 Proof of Proposition 2

With the output head fixed, any representation change $\Delta h = h_{\theta^+}(x) - h_\theta(x)$ satisfies

$$\log \frac{p_{\theta^+}(v_i \mid x)}{p_{\theta^+}(v_j \mid x)} - \log \frac{p_\theta(v_i \mid x)}{p_\theta(v_j \mid x)} = (w_i - w_j)^\top \Delta h. \tag{C.10}$$

Combining this identity with a parameter update yields Proposition 2, following the step-wise perspective of LLM learning-dynamics analysis (Ren & Sutherland, 2025). Assumption 2 supplies the chain rule at the current parameters and a quadratic Taylor remainder for the logit difference.

Proof. The full-vocabulary softmax normalizer cancels from any pair of token probabilities. With the output head fixed, an arbitrary update satisfies

$$\begin{aligned} \log \frac{p_{\theta^+}(v_i \mid x)}{p_{\theta^+}(v_j \mid x)} - \log \frac{p_\theta(v_i \mid x)}{p_\theta(v_j \mid x)} &= [z_i(\theta^+) - z_j(\theta^+)] - [z_i(\theta) - z_j(\theta)] \\ &= (w_i - w_j)^\top \big(h_{\theta^+}(x) - h_\theta(x)\big). \end{aligned} \tag{C.11}$$

This identity is exact, independent of update size. To evaluate the effect of a gradient update, set $f_{ij}(\theta) = z_i(\theta) - z_j(\theta)$ and $d = s \odot \nabla_\theta \mathcal{L}(\theta; x)$, so $\theta^+ = \theta - \eta d$. Taylor's formula along this parameter-space segment gives

$$\begin{aligned} f_{ij}(\theta - \eta d) - f_{ij}(\theta) = &- \eta \langle \nabla_\theta f_{ij}(\theta), d \rangle \\ &+ \eta^2 \int_0^1 (1-t)\, d^\top \nabla_\theta^2 f_{ij}(\theta - t\eta d) d \, \mathrm{d}t. \end{aligned} \tag{C.12}$$

Substitute the masked chain rule from Eq. (C.9) into the leading term:

$$\begin{aligned} -\eta \langle \nabla_\theta f_{ij}, d \rangle &= -\eta \, \langle \nabla_\theta (z_i - z_j),\, s \odot \nabla_\theta \mathcal{L} \rangle \\ &= -\eta \left\langle \nabla_\theta (z_i - z_j),\, s \odot \sum_{n_\ell \in \mathcal{N}} g_\ell \nabla_\theta z_\ell \right\rangle \\ &= -\eta \sum_{n_\ell \in \mathcal{N}} g_\ell \, \langle \nabla_\theta (z_i - z_j),\, s \odot \nabla_\theta z_\ell \rangle \, . \end{aligned} \tag{C.13}$$

Since f_{ij} is C^2 near the current parameters, its Hessian is bounded on a sufficiently short closed update segment. If the operator norm is at most L_{ij} on that segment, the remainder in Eq. (C.12) obeys

$$\begin{aligned} |R_{ij}| &\le \eta^2 \int_0^1 (1-t) \left| d^\top \nabla_\theta^2 f_{ij}(\theta - t\eta d) d \right| \, \mathrm{d}t \\ &\le \eta^2 \int_0^1 (1-t) L_{ij} \|d\|_2^2 \, \mathrm{d}t \\ &= \frac{L_{ij}}{2} \eta^2 \|s \odot \nabla_\theta \mathcal{L}\|_2^2. \end{aligned} \tag{C.14}$$

Combining the exact probability ratio, the expanded leading term, and this bound yields

$$\begin{aligned} \log \frac{p_{\theta^+}(v_i \mid x)}{p_{\theta^+}(v_j \mid x)} - \log \frac{p_\theta(v_i \mid x)}{p_\theta(v_j \mid x)} = &- \eta \sum_{n_\ell \in \mathcal{N}} g_\ell \, \langle \nabla_\theta (z_i - z_j),\, s \odot \nabla_\theta z_\ell \rangle + R_{ij}, \\ |R_{ij}| &\le \frac{L_{ij}}{2} \eta^2 \|s \odot \nabla_\theta \mathcal{L}\|_2^2. \end{aligned} \tag{C.15}$$

The expansion is in the effective step size η, at fixed context, starting parameters, mask, and γ. Its constant may depend on these fixed quantities. All anchor terms are retained in g_ℓ. For a stabilized angular computation, the proof is unchanged with g_ℓ replaced by its actual derivative, provided the computed objective is differentiable with respect to the numeric logits and the logits are differentiable with respect to the parameters at the starting point. □

The coefficients g_ℓ encode the current angular configuration, while the inner products in Eq. (9) describe how the selected parameters couple a change in token n_ℓ to the preference between v_i and v_j. Numerical refinement depends on both this geometric signal and the model's response. The formula does not ensure that a correct token gains probability; that conclusion additionally requires favorable alignment between the signal and the selected parameters' logit responses.

The same reasoning gives an explicit response for any recorded parameter change:

$$\begin{aligned} \Delta \log \frac{p_i}{p_j} &= \langle \nabla_\theta (z_i - z_j), \Delta\theta \rangle + R_{ij}, \\ |R_{ij}| &\le \frac{L_{ij}}{2} \|\Delta\theta\|_2^2. \end{aligned} \tag{C.16}$$

Here Δ denotes the endpoint difference in the same log probability ratio. This form also accounts for a recorded finite-precision update without identifying it with an exact real-arithmetic gradient step.

D Extended Related Work

Mathematical reasoning and numerical evaluation. GSM8K and MATH evaluate mathematical problem solving through word problems and competition problems (Cobbe et al., 2021; Hendrycks et al., 2021). MATH 401 focuses on arithmetic expressions, while Number Cookbook and NumericBench cover numerical operations in a broader range of settings (Yuan et al., 2023; Yang et al., 2025; Li et al., 2025). GSM-Symbolic tests sensitivity to controlled changes in problem statements (Mirzadeh et al., 2025). Chain-of-thought prompting, aggregation of sampled solutions, process supervision, and allocation of test-time computation address how solutions are generated and selected (Wei et al., 2022; Wang et al., 2023; Lightman et al., 2024; Snell et al., 2025).

Numerical representations and encodings. Numerical structure has been examined at several stages of model computation. Linear probes recover values from hidden states in arithmetic contexts (Zhu et al., 2025), and digit probes identify representations organized by decimal position (Levy & Geva, 2025). Sinusoidal probes recover values from pretrained input embeddings, and correcting embeddings that deviate from the fitted pattern can reduce arithmetic errors (Kadlčík et al., 2025). Related analyses examine the consistency of these patterns across models and layers, and steer activations toward representations of known target values (Štefánik et al., 2026). These interventions use numerical supervision to fit probes or select target representations. Causal interventions further link arithmetic predictions to transformer components and numerical heuristics (Stolfo et al., 2023; Nikankin et al., 2025).

Designed encodings introduce numerical structure explicitly. xVal scales a shared embedding by the numerical value and uses a numerical prediction head (Golkar et al., 2023). Abacus encodes positions within a number (McLeish et al., 2024), FoNE uses Fourier features (Zhou et al., 2026), and GeoNum integrates geometric embeddings through staged training (Jin et al., 2026). Number Token Loss retains discrete token outputs and adds a supervised penalty based on distance to the target value (Zausinger et al., 2025).

Output geometry and predictive uncertainty. Vocabulary projections trace how feed-forward layers contribute to token predictions (Geva et al., 2022), and the Tuned Lens learns affine maps from intermediate hidden states to output predictions (Belrose et al., 2023). Geometric analyses characterize concept directions, categorical hierarchies, and multidimensional features (Park et al., 2024; 2025; Engels et al., 2025). Relationships among outputs also inform uncertainty estimation. Semantic entropy aggregates probabilities over answers with equivalent meanings (Kuhn et al., 2023; Farquhar et al., 2024), while black-box estimators measure the semantic dispersion of sampled responses (Lin et al., 2024). Kernel Language Entropy encodes similarities among answers in a positive semidefinite kernel and measures its von Neumann entropy (Nikitin et al., 2024). For unit output directions, directional statistics provide a description of concentration around a mean direction (Banerjee et al., 2005). This connects uncertainty in numerical prediction to both token probabilities and the angular relationships in the pretrained representation.

Test-time learning in language models. Entropy regularization supplies a learning signal from unlabeled predictions (Grandvalet & Bengio, 2004). Test-time training constructs an auxiliary self-supervised objective (Sun et al., 2020), while Tent minimizes prediction entropy through normalization-layer updates (Wang et al., 2021). For LLMs, Hardt & Sun (2024) finetune on retrieved neighboring text, and TLM minimizes input perplexity using low-rank parameter updates (Hu et al., 2025). SRGen instead learns corrective vectors from the generated context at uncertain decoding steps (Mu et al., 2026). Agarwal et al. (2025) study entropy minimization through finetuning, reinforcement learning, and direct logit adjustment, with model parameters fixed in the last case. These methods differ in both the source of the learning signal and the quantities they update. Numerical output geometry supplies a reference for minimizing angular dispersion through selected MLP updates.

E Evaluation Tasks and Models

Table 7 lists the evaluated sample counts and sources. MATH500 denotes the 500-problem MATH evaluation set distributed with PRM800K. TabMWP uses questions grounded in tables, FinQA uses financial reports, and the visual suite covers chart questions, visual mathematical problems, and object counting.

Table 7: Evaluated sample counts and dataset sources.

Task	Dataset	Examples	Source
Numerical	Addition-50	50	Generated in this work; Appendix E
	NUPA	526	Yang et al. (2025)
	BB-Arith	466	BIG-bench arithmetic (BIG-bench authors, 2023)
	NB-Mixed	200	NumericBench (Li et al., 2025)
	NB-Arith	367	Li et al. (2025)
	MATH401	401	Yuan et al. (2023)
Mathematical	GSM8K	1,319	Cobbe et al. (2021)
	MATH500	500	Hendrycks et al. (2021); Lightman et al. (2024)
	TabMWP	500	Lu et al. (2023)
	FinQA	1,147	Chen et al. (2021)
Visual	ChartQA	500	Masry et al. (2022)
	MathVista	1,000	Lu et al. (2024)
	TallyQA	500	Acharya et al. (2019)

Addition-50. Following prior work on synthetic addition (McLeish et al., 2024), we construct 50 problems by randomly sampling pairs of positive integers and computing their exact sums programmatically. Each pair is presented with the fixed prompt "What is the sum of a and b?" Keeping the operation and wording fixed reduces variation from language comprehension and problem decomposition, allowing a focused assessment of numerical calculation.

NumericBench subsets. NB-Mixed denotes NumericBench-Mixed, the mixed-number-string subset of NumericBench (Li et al., 2025). NB-Arith denotes its arithmetic subset. The two subsets contain 200 and 367 evaluated examples, respectively.

Metric definitions. We score complete answers after answer extraction and normalization. We use symbolic equivalence for MATH500 and TabMWP, numerical tolerance for MATH401, and relaxed numerical accuracy for ChartQA. Token accuracy aligns tokenized, normalized answers by position and divides the total number of matches by the sum of the longer answer lengths; missing or extra tokens count as errors. This is a supplementary answer-agreement metric, not teacher-forced next-token accuracy. Its values depend on the model's tokenizer. Number Cookbook (Yang et al., 2025) similarly supplements answer correctness with digit-level agreement; our metric operates on model tokens. For reasoning tasks, mean generated length measures the number of output tokens per example. It excludes the additional computation used for adaptation. All averages give equal weight to the evaluated datasets.

Table 8: Model identifiers in the results. Abbreviated names are used in the tables.

Table name	Checkpoint identifier
Llama-3.2-1B	Llama-3.2-1B-Instruct
Llama-3.2-3B	Llama-3.2-3B-Instruct
Llama-3.1-8B	Llama-3.1-8B-Instruct
Qwen3-4B	Qwen3-4B-Instruct-2507
Qwen3-8B	Qwen3-8B
DeepSeek-Math-7B	DeepSeek-Math-7B-Instruct
DeepSeek-R1-Distill-Qwen-1.5B	DeepSeek-R1-Distill-Qwen-1.5B
Qwen2.5-VL-3B	Qwen2.5-VL-3B-Instruct
InternVL3.5-4B	InternVL3_5-4B-HF

E.1 Baseline definitions

Our comparisons use full-vocabulary Shannon entropy for *entropy minimization*, and causal language-model loss on the input for *perplexity minimization*. These baselines adopt the objectives of Tent (Wang et al., 2021) and TLM (Hu et al., 2025) for local adaptation. *Real-line dispersion* uses the variance of numeric token values under the numerical distribution. Like Number Token Loss (Zausinger et al., 2025), this objective uses numerical distances. Here the reference is the predictive mean, allowing adaptation without the target values required by the supervised loss. *R2L formatting* (Singh & Strouse, 2024) aligns numerical groups from the right; it does not reverse autoregressive generation. *Self-consistency* aggregates sampled answers, whereas LTPO optimizes latent thought vectors with the model weights frozen (Wang et al., 2023; Ye et al., 2026).

Adaptation protocols. In free generation, adaptation is triggered when the unadapted top prediction is a numeric token. Reference-prefix evaluations instead adapt at the designated numerical positions, and trajectory collection uses five updates. Reference answers determine evaluation contexts and scores where applicable; they do not enter the adaptation objective.

E.2 Adaptation algorithm

For a fixed context x, let $\mathcal{P}$ denote the eligible MLP coordinates and r their selected fraction. $\mathrm{TopFrac}_{\mathcal{P}}(|g|, r)$ returns a binary mask retaining the largest absolute gradients within $\mathcal{P}$ and zero elsewhere. η_k denotes the effective step size after clipping the masked gradient. The inverse temperature γ is held constant within each episode.

Algorithm 1 `NuRA` at one numerical prediction position

Require: Context x, pretrained parameters θ_0, numeric directions $\{u_i\}_{i\in\mathcal{N}}$, eligible coordinates $\mathcal{P}$, fraction r, inverse temperature γ, steps K

$\theta \leftarrow \theta_0$
for $k = 0, \ldots, K-1$ **do**
  $z \leftarrow z_\theta(x)$; $q \leftarrow \mathrm{softmax}(\gamma z_{\mathcal{N}})$
  $a \leftarrow (\sum_i q_i u_i)/\|\sum_i q_i u_i\|_2$
  $\mathcal{L} \leftarrow \sum_i q_i \arccos(u_i^\top a)$
  $g \leftarrow \nabla_\theta \mathcal{L}$ ▷ Differentiate through a
  if $k = 0$ **then**
    $s \leftarrow \mathrm{TopFrac}_{\mathcal{P}}(|g|, r)$
  end if
  $\theta \leftarrow \theta - \eta_k (s \odot g)$
end for
$z^\star \leftarrow z_\theta(x)$; $\theta \leftarrow \theta_0$
return $z^\star$

F Additional Numerical Prediction Results

We report answer and token accuracy for the five additional language models. Results for Llama-3.1-8B and Qwen3-4B are in Table 1.

Table 9: Numerical prediction on Llama-3.2-1B.

Method	Addition-50		NUPA		BB-Arith		NB-Mixed		NB-Arith		MATH401		Avg.	
	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.
Greedy	6.00	2.88	12.55	20.41	19.96	30.10	0.00	0.00	0.00	9.28	7.98	23.56	7.75	14.37
Temp. Sampling	6.00	2.66	8.56	12.18	18.67	27.85	0.00	0.00	**0.27**	9.94	7.73	23.40	6.87	12.67
R2L Formatting	6.00	2.71	6.84	9.33	15.45	24.06	0.00	0.00	0.00	8.08	7.73	22.61	6.00	11.13
LTPO	2.00	2.51	12.55	19.99	8.37	18.92	0.00	0.00	**0.27**	9.96	3.49	13.10	4.45	10.75
Entropy Min.	6.00	2.76	9.70	16.67	**23.61**	30.15	0.00	0.00	0.00	9.33	**8.73**	24.15	8.00	13.84
Perplexity Min.	6.00	2.56	11.98	21.36	20.17	28.78	0.50	0.98	0.00	9.66	6.73	16.49	7.56	13.30
Real-Line Disp.	6.00	2.68	4.18	6.43	23.18	30.70	0.00	0.00	0.00	10.11	7.98	23.92	6.89	12.31
NuRA	**10.00**	**6.55**	**13.12**	**24.95**	23.39	**34.04**	**1.50**	**1.96**	0.00	**10.64**	**8.73**	**24.42**	**9.46**	**17.09**

Table 10: Numerical prediction on Llama-3.2-3B.

Method	Addition-50		NUPA		BB-Arith		NB-Mixed		NB-Arith		MATH401		Avg.	
	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.
Greedy	16.00	13.03	12.17	24.11	51.93	62.36	**4.50**	3.96	16.08	31.45	7.98	28.74	18.11	27.27
Temp. Sampling	16.00	17.25	10.27	23.72	55.15	62.14	4.00	4.00	16.08	31.46	7.98	29.67	18.25	28.04
R2L Formatting	18.00	19.05	8.94	20.21	34.12	43.61	**4.50**	3.96	16.62	31.02	5.99	25.09	14.69	23.82
LTPO	16.00	13.44	11.41	21.13	45.71	54.25	3.50	2.97	17.17	30.57	4.99	19.22	16.46	23.60
Entropy Min.	16.00	16.85	12.36	25.07	54.29	64.65	4.00	3.00	**17.44**	31.96	7.73	29.81	18.64	28.56
Perplexity Min.	14.00	12.90	12.55	26.21	56.01	65.00	2.50	2.00	16.62	32.09	7.98	29.10	18.28	27.88
Real-Line Disp.	16.00	14.23	8.75	14.74	53.00	63.41	3.00	3.00	16.62	31.80	7.98	29.18	17.56	26.06
NuRA	**20.00**	**25.12**	**16.16**	**26.87**	**57.30**	**69.30**	**4.50**	**5.50**	**17.44**	**33.65**	**8.23**	**30.30**	**20.60**	**31.79**

Table 11: Numerical prediction on Qwen3-8B.

Method	Addition-50		NUPA		BB-Arith		NB-Mixed		NB-Arith		MATH401		Avg.	
	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.
Greedy	42.00	87.38	29.66	42.81	74.68	83.89	7.50	21.64	54.77	64.69	17.46	32.32	37.68	55.45
Temp. Sampling	44.00	87.91	28.52	**46.75**	**75.75**	83.82	4.50	19.35	54.77	64.27	17.21	32.16	37.46	55.71
R2L Formatting	40.00	80.78	23.95	41.05	73.18	82.19	5.50	21.25	51.23	61.41	17.21	31.68	35.18	53.06
LTPO	44.00	89.21	27.76	42.14	74.03	83.71	**9.00**	21.36	**56.13**	66.02	16.96	31.99	37.98	55.74
Entropy Min.	44.00	88.06	29.66	42.75	74.89	83.80	6.00	20.96	54.50	63.70	17.46	32.02	37.75	55.22
Perplexity Min.	38.00	87.07	31.18	43.29	75.32	82.56	4.50	14.45	**56.13**	64.48	17.46	31.98	37.10	53.97
Real-Line Disp.	44.00	89.98	29.85	44.95	75.54	84.10	**9.00**	21.75	54.77	64.40	17.46	32.54	38.43	56.29
NuRA	**46.00**	**90.16**	**32.32**	45.63	75.11	**84.30**	8.50	**22.50**	**56.13**	**68.12**	**19.95**	**44.18**	**39.67**	**59.15**

Table 12: Numerical prediction on DeepSeek-Math-7B.

Method	Addition-50		NUPA		BB-Arith		NB-Mixed		NB-Arith		MATH401		Avg.	
	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.
Greedy	24.00	85.40	12.93	28.80	21.03	27.02	1.50	4.27	11.44	18.72	13.22	25.26	14.02	31.58
Temp. Sampling	26.00	82.26	13.69	28.06	22.10	28.37	1.50	4.82	7.36	13.49	10.97	22.91	13.60	29.99
R2L Formatting	**36.00**	80.58	12.55	31.14	19.96	28.14	1.50	3.84	8.17	11.09	13.22	25.01	15.23	29.97
LTPO	26.00	79.72	13.31	30.60	13.95	24.56	1.50	2.40	3.27	3.96	3.99	12.13	10.34	25.56
Entropy Min.	24.00	85.63	13.31	29.29	20.82	27.16	**2.00**	4.27	11.44	18.72	12.97	21.51	14.09	31.10
Perplexity Min.	26.00	85.24	17.68	35.07	23.61	30.22	1.50	3.86	11.99	18.97	13.22	25.17	15.67	33.09
Real-Line Disp.	24.00	85.63	11.79	33.30	21.46	27.72	1.50	4.82	11.44	18.72	13.22	25.11	13.90	32.55
NuRA	24.00	**86.39**	**19.58**	**36.87**	**27.04**	**33.55**	**2.00**	**8.33**	**14.17**	**26.03**	**14.46**	**26.79**	**16.88**	**36.33**

Table 13: Numerical prediction on DeepSeek-R1-Distill-Qwen-1.5B.

Method	Addition-50		NUPA		BB-Arith		NB-Mixed		NB-Arith		MATH401		Avg.	
	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.
Greedy	12.00	36.53	18.06	35.48	28.11	40.23	**4.00**	2.25	2.45	14.94	5.99	6.29	11.77	22.62
Temp. Sampling	10.00	30.78	18.25	35.48	26.61	38.37	2.50	2.43	3.00	13.25	6.48	6.49	11.14	21.13
R2L Formatting	12.00	26.52	12.74	33.84	**44.85**	46.48	1.50	2.09	2.72	12.78	3.49	4.38	12.88	21.02
LTPO	12.00	36.53	16.92	33.59	28.97	40.48	3.00	2.16	2.45	11.96	6.23	6.50	11.60	21.87
Entropy Min.	10.00	35.13	18.25	36.25	28.11	40.23	3.00	2.22	**3.54**	14.39	6.23	7.36	11.52	22.60
Perplexity Min.	**16.00**	30.15	18.25	34.85	25.54	35.33	3.00	2.25	2.72	13.97	5.74	6.29	11.87	20.47
Real-Line Disp.	12.00	36.98	14.83	36.37	34.55	47.61	1.50	2.45	2.45	14.98	4.24	6.75	11.59	24.19
NuRA	14.00	**42.17**	**19.01**	**37.28**	42.49	**54.32**	**4.00**	**2.73**	1.36	**15.40**	**7.23**	**9.87**	**14.68**	**26.96**

G Additional Mathematical Reasoning Results

We report answer accuracy and mean generation length for the remaining language models. Table 2 presents DeepSeek-Math-7B.

Table 14: Mathematical reasoning on Llama-3.2-1B.

Method	GSM8K		MATH500		TabMWP		FinQA		Avg.	
	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.
Greedy	47.16	107.34	26.20	354.51	50.60	86.07	4.97	135.58	32.23	170.87
Temp. Sampling	42.15	104.75	23.80	342.40	45.60	93.91	4.01	127.11	28.89	167.04
Self-Consistency	48.98	**89.13**	28.80	342.47	52.20	**77.69**	3.84	112.89	33.45	**155.54**
LTPO	47.69	108.16	23.20	381.19	49.20	102.26	4.80	126.24	31.22	179.46
Entropy Min.	48.22	105.90	27.80	361.10	52.20	92.48	6.02	113.29	33.56	168.19
Perplexity Min.	48.37	108.33	23.60	387.57	50.80	86.05	2.09	**87.58**	31.22	167.38
Real-Line Disp.	47.16	108.96	24.60	371.62	52.40	86.65	2.88	118.14	31.76	171.34
NuRA	**52.92**	103.70	**33.40**	**333.99**	**52.80**	84.22	**6.10**	113.11	**36.31**	158.75

Table 15: Mathematical reasoning on Llama-3.2-3B.

Method	GSM8K		MATH500		TabMWP		FinQA		Avg.	
	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.
Greedy	76.88	120.41	47.80	332.86	69.00	96.97	12.99	126.19	51.67	169.11
Temp. Sampling	76.27	128.98	45.40	322.35	70.80	94.78	11.16	134.47	50.91	170.14
Self-Consistency	78.77	119.88	48.40	333.39	70.00	**82.33**	10.11	127.52	51.82	165.78
LTPO	77.56	128.54	46.40	336.40	66.20	127.32	12.12	146.64	50.57	184.73
Entropy Min.	78.17	119.97	47.80	333.01	67.60	97.10	12.82	120.18	51.60	167.56
Perplexity Min.	72.86	125.41	38.00	340.83	69.20	96.76	12.99	122.15	48.26	171.29
Real-Line Disp.	78.09	120.30	47.20	332.00	67.00	105.59	12.90	128.28	51.30	171.54
NuRA	**83.24**	**109.89**	**50.00**	**314.59**	**71.00**	93.38	**13.95**	**91.26**	**54.55**	**152.28**

Table 16: Mathematical reasoning on Llama-3.1-8B.

Method	GSM8K		MATH500		TabMWP		FinQA		Avg.	
	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.
Greedy	84.46	117.93	53.20	417.78	77.40	144.65	15.95	148.27	57.75	207.16
Temp. Sampling	82.64	121.60	49.00	407.06	74.60	109.45	17.09	130.34	55.83	192.11
Self-Consistency	85.52	117.60	53.40	415.78	75.20	110.42	17.18	144.71	57.82	197.13
LTPO	83.62	104.75	44.20	432.61	67.60	132.87	15.95	161.23	52.84	207.86
Entropy Min.	84.99	117.12	51.40	422.81	77.40	127.43	16.04	149.37	57.46	204.18
Perplexity Min.	83.62	118.06	52.40	**387.55**	75.20	**105.82**	15.87	155.37	56.77	191.70
Real-Line Disp.	84.38	137.37	51.80	424.76	78.00	139.24	16.04	123.85	57.56	206.30
NuRA	**87.49**	**101.37**	**55.00**	402.27	**78.20**	110.87	**18.83**	**120.79**	**59.88**	**183.82**

Table 17: Mathematical reasoning on Qwen3-4B.

Method	GSM8K		MATH500		TabMWP		FinQA		Avg.	
	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.
Greedy	93.10	198.28	56.80	**595.99**	98.40	**143.51**	22.84	275.21	67.79	303.25
Temp. Sampling	92.72	196.43	52.00	602.27	97.80	**143.51**	21.01	276.93	65.88	304.79
Self-Consistency	92.19	**194.32**	54.60	601.10	98.40	**143.51**	22.14	265.58	66.83	**301.13**
LTPO	91.58	217.56	48.20	623.47	94.00	153.79	22.14	286.90	63.98	320.43
Entropy Min.	93.18	198.28	56.80	**595.99**	97.60	**143.51**	22.84	274.61	67.60	303.10
Perplexity Min.	93.25	198.28	47.60	597.67	93.80	151.25	21.80	**261.29**	64.11	302.12
Real-Line Disp.	93.40	198.28	**57.00**	**595.99**	98.60	146.81	23.02	274.89	68.01	303.99
NuRA	**93.78**	195.15	**57.00**	**595.99**	**99.20**	147.10	**23.10**	274.61	**68.27**	303.21

Table 18: Mathematical reasoning on Qwen3-8B.

Method	GSM8K		MATH500		TabMWP		FinQA		Avg.	
	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.
Greedy	88.93	214.66	69.20	555.88	80.20	123.05	24.06	214.58	65.60	277.04
Temp. Sampling	86.50	216.32	72.80	561.61	77.20	134.03	25.11	**210.92**	65.40	280.72
Self-Consistency	85.44	218.09	**76.20**	556.21	79.80	132.22	25.02	212.07	66.62	279.65
LTPO	88.63	214.19	62.60	554.02	80.40	**117.59**	25.98	223.35	64.40	277.29
Entropy Min.	88.63	214.87	68.60	555.88	79.60	123.05	24.85	214.56	65.42	277.09
Perplexity Min.	**89.99**	**204.04**	67.40	**539.31**	79.60	118.24	26.85	211.96	65.96	**268.39**
Real-Line Disp.	88.70	214.96	68.80	555.88	80.20	123.05	25.11	214.56	65.70	277.11
NuRA	89.08	211.32	70.80	552.16	**82.00**	119.25	**26.94**	213.07	**67.21**	273.95

Table 19: Mathematical reasoning on DeepSeek-R1-Distill-Qwen-1.5B.

Method	GSM8K		MATH500		TabMWP		FinQA		Avg.	
	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.	A.Acc.	Len.
Greedy	77.03	**553.64**	34.80	899.70	63.00	539.23	**9.15**	758.46	46.00	687.76
Temp. Sampling	75.89	590.63	33.80	923.11	60.80	586.01	8.02	774.11	44.63	718.46
Self-Consistency	77.71	587.85	**42.60**	899.70	65.40	534.91	6.89	777.72	48.15	700.04
LTPO	76.80	585.53	37.20	897.53	63.20	539.23	8.98	819.42	46.55	710.43
Entropy Min.	77.48	587.85	35.40	899.70	63.20	523.37	9.07	753.16	46.29	691.02
Perplexity Min.	77.63	587.70	40.40	914.96	64.20	538.84	8.81	758.46	47.76	699.99
Real-Line Disp.	78.62	559.25	36.20	884.20	69.40	531.95	8.89	734.94	48.28	677.59
NuRA	**79.68**	558.84	42.20	**884.12**	**71.20**	**521.40**	**9.15**	**726.02**	**50.56**	**672.60**

H Additional Visual Numerical Prediction Results

We report answer and token accuracy for both vision-language models, including the token-level results omitted from Table 3.

Table 20: **Visual numerical prediction on Qwen2.5-VL-3B.** Answer accuracy (A.Acc.) and token accuracy (T.Acc.), in percent.

Method	ChartQA		MathVista		TallyQA		Avg.	
	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.
Greedy	61.40	57.23	46.10	45.96	7.40	22.00	38.30	41.73
Temp. Sampling	61.00	55.49	39.80	41.18	5.60	18.50	35.47	38.39
Entropy Min.	63.60	58.84	47.80	47.37	13.60	38.50	41.67	48.24
Perplexity Min.	59.20	54.73	44.20	42.31	8.00	22.50	37.13	39.85
Real-Line Disp.	63.40	58.91	48.80	**48.94**	11.20	39.00	41.13	48.95
NuRA	**65.00**	**64.76**	**50.20**	47.97	**14.40**	**39.50**	**43.20**	**50.74**

Table 21: Visual numerical prediction on InternVL3.5-4B.

Method	ChartQA		MathVista		TallyQA		Avg.	
	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.	A.Acc.	T.Acc.
Greedy	74.60	66.46	49.90	54.69	18.60	37.50	47.70	52.88
Temp. Sampling	73.60	66.46	48.20	53.12	13.40	34.00	45.07	51.19
Entropy Min.	74.20	65.84	47.80	54.97	14.60	38.00	45.53	52.94
Perplexity Min.	73.20	65.22	49.00	54.69	16.60	35.00	46.27	51.63
Real-Line Disp.	75.00	66.46	**50.10**	54.97	18.60	38.00	47.90	53.14
NuRA	**76.00**	**70.77**	50.00	**55.26**	**19.40**	**39.50**	**48.47**	**55.18**

I NUMERICAL PREDICTION ERROR DETECTION

Evaluation protocol. We evaluate frozen Llama-3.2-1B-Instruct and Llama-3.2-3B-Instruct on 50 Addition-50 questions and 100 questions each from BB-Arith (BIG-bench authors, 2023), MATH401 (Yuan et al., 2023), and NumericBench-Arith (Li et al., 2025). At each numerical token position in the reference answer, the model receives the question and preceding reference tokens. Its highest-probability numerical token is marked incorrect if it differs from the reference. All scores use the same positions and predictions, with no parameter updates.

Uncertainty scores. Let p be the full-vocabulary distribution and $q_i = p_i / \sum_{j \in \mathcal{N}} p_j$ its renormalization over numerical tokens $\mathcal{N}$. We compare angular dispersion with $H(q)$, $H(p)$, $1 - \max_{i \in \mathcal{N}} q_i$, and $1 - \max_{i \in \mathcal{V}} p_i$, where H denotes Shannon entropy. Maximum-probability scores follow a standard approach to error detection (Hendrycks & Gimpel, 2017). Angular dispersion uses the normalized output directions and posterior anchor in Eqs. equation 4–equation 5, with $\gamma = 1$. Permuted angular randomly reassigns these directions to numerical tokens and recomputes the anchor. It preserves q and the set of directions, changing only their correspondence to tokens.

Metrics. AUROC treats incorrect predictions as the positive class. For AURC, predictions are retained in increasing order of uncertainty, and risk is the error rate among retained predictions. The area under this risk–coverage curve measures selective prediction performance (Geifman & El-Yaniv, 2019). Higher AUROC and lower AURC are better. Table 22 reports AURC for each model and dataset; Table 23 reports AUROC.

Table 22: **Selective numerical prediction.** AURC (↓) under reference answer prefixes. AUROC is reported in Table 23. Bold and underlining mark the best and second-best scores within each model.

Uncertainty score	Llama-3.2-1B				Llama-3.2-3B			
	Addition-50	BB-Arith	MATH401	NB-Arith	Addition-50	BB-Arith	MATH401	NB-Arith
Angular dispersion	**0.8482**	**0.3198**	**0.3153**	0.9090	**0.6492**	**0.1260**	**0.0899**	**0.3651**
Numeric entropy $H(q)$	0.8889	<u>0.3414</u>	0.3268	0.9296	0.6671	0.1404	0.0967	0.3849
$1 - \max_i q_i$	<u>0.8867</u>	0.3431	0.3285	0.9270	0.6623	<u>0.1371</u>	<u>0.0921</u>	<u>0.3813</u>
Vocabulary entropy $H(p)$	0.8888	0.3478	0.3267	<u>0.9072</u>	0.6636	0.1406	0.0992	0.4417
$1 - \max_i p_i$	0.8876	0.3507	<u>0.3259</u>	**0.9047**	<u>0.6600</u>	0.1373	0.0975	0.4610
Permuted angular	0.9483	0.3770	0.3491	0.9746	0.7012	0.1481	0.1009	0.4020

Table 23: **Numerical prediction error detection.** AUROC (↑) under reference answer prefixes. AURC results and protocol are in Appendix I.

Uncertainty score	Llama-3.2-1B				Llama-3.2-3B			
	Addition-50	BB-Arith	MATH401	NB-Arith	Addition-50	BB-Arith	MATH401	NB-Arith
Angular dispersion	**0.9941**	**0.9612**	**0.9799**	<u>0.5884</u>	**0.9315**	0.9455	**0.9154**	<u>0.8781</u>
Numeric entropy $H(q)$	0.9722	<u>0.9504</u>	0.9755	0.5440	0.9175	0.9417	0.8917	0.8699
$1 - \max_i q_i$	<u>0.9837</u>	0.9503	0.9732	0.5645	<u>0.9274</u>	**0.9495**	<u>0.9084</u>	**0.8787**
Vocabulary entropy $H(p)$	0.9751	0.9424	0.9749	0.5841	0.9211	0.9412	0.8866	0.7894
$1 - \max_i p_i$	0.9822	0.9409	<u>0.9778</u>	**0.5920**	<u>0.9274</u>	<u>0.9490</u>	0.8950	0.7641
Permuted angular	0.8143	0.8241	0.8474	0.4937	0.8075	0.8446	0.7756	0.6982

J NuRA ABLATION DETAILS

The ablations use the same 200 questions drawn from Addition-50, NUPA, BB-Arith, NumericBench-Mixed, NumericBench-Arith, and MATH401, with dataset sources listed in Appendix E. Each prediction uses the reference answer prefix, and accuracy is measured over numerical token positions.

Geometry and objective. *Permuted directions* randomly reassigns output-head directions to numerical tokens. *Cosine dispersion* replaces angular distance with cosine distance. *Fixed anchor* computes the anchor before the first update at each prediction position and holds it fixed during adaptation. *Full vocabulary* constructs the adaptation objective over the entire vocabulary rather than numerical tokens alone.

Parameter updates. *LayerNorm only* updates normalization parameters across all layers. *Last 4 MLP layers* restricts parameter updates to the MLPs in the final four transformer layers. *Before adaptation* uses the original model, and NuRA *(full)* uses the complete method.

K SPARSE PARAMETER UPDATES

Protocol. We evaluate Llama-3.2-3B-Instruct on Addition-50 questions 10–29, predicting each numerical token from the question and preceding reference answer tokens. We restore original parameters at each position and take five updates with learning rate 0.02 and $\gamma = 1$, using all numerical tokens in the objective. The candidate set $\mathcal{P}$ comprises MLP and normalization parameters across all layers; all other parameters remain fixed. Full updates modify all of $\mathcal{P}$. Table 24 lists nominal update percentages and exact parameter counts.

Parameter selection. *Random fixed* samples parameters uniformly; *Gradient fixed* selects the largest absolute entries of the initial gradient. Both selections remain fixed within each prediction position, whereas *Gradient dynamic* reselects at every step. Random selection reports means and sample standard deviations over seeds 42–44; gradient-based strategies use seed 42. The unadapted and full-update results are shared references.

Metrics. We report accuracy over numerical positions and numerical equivalence of reconstructed answers, both under reference prefixes. For the gradient g before masking and clipping and selected coordinates $S \subseteq \mathcal{P}$, retained gradient energy is

$$E = \frac{\sum_{i \in S} g_i^2}{\sum_{i \in \mathcal{P}} g_i^2}. \tag{K.1}$$

We average E over steps within each position, positions within each question, and then questions.

Accuracy and gradient energy. At $10^{-5}\%$, fixed selection reaches 29.79% token accuracy, close to 30.50% for full updates, while random selection remains at the unadapted accuracy of 14.89% (Figure 8; Table 24). Fixed selection retains only 7.97% of the gradient energy, suggesting that most of the prediction gain can be recovered without preserving most of the full gradient. Larger update fractions retain more energy without consistently improving accuracy.

Fixed and dynamic selection. At $10^{-5}\%$, dynamic selection retains 17.58% of the gradient energy and reaches 31.21% accuracy. Despite retaining more energy, it provides no consistent accuracy advantage over fixed selection at larger fractions. Their similar performance supports reusing parameters selected by the initial gradient over subsequent updates. Complete-answer reconstruction remains at most 5%, so the gains primarily concern individual numerical predictions.

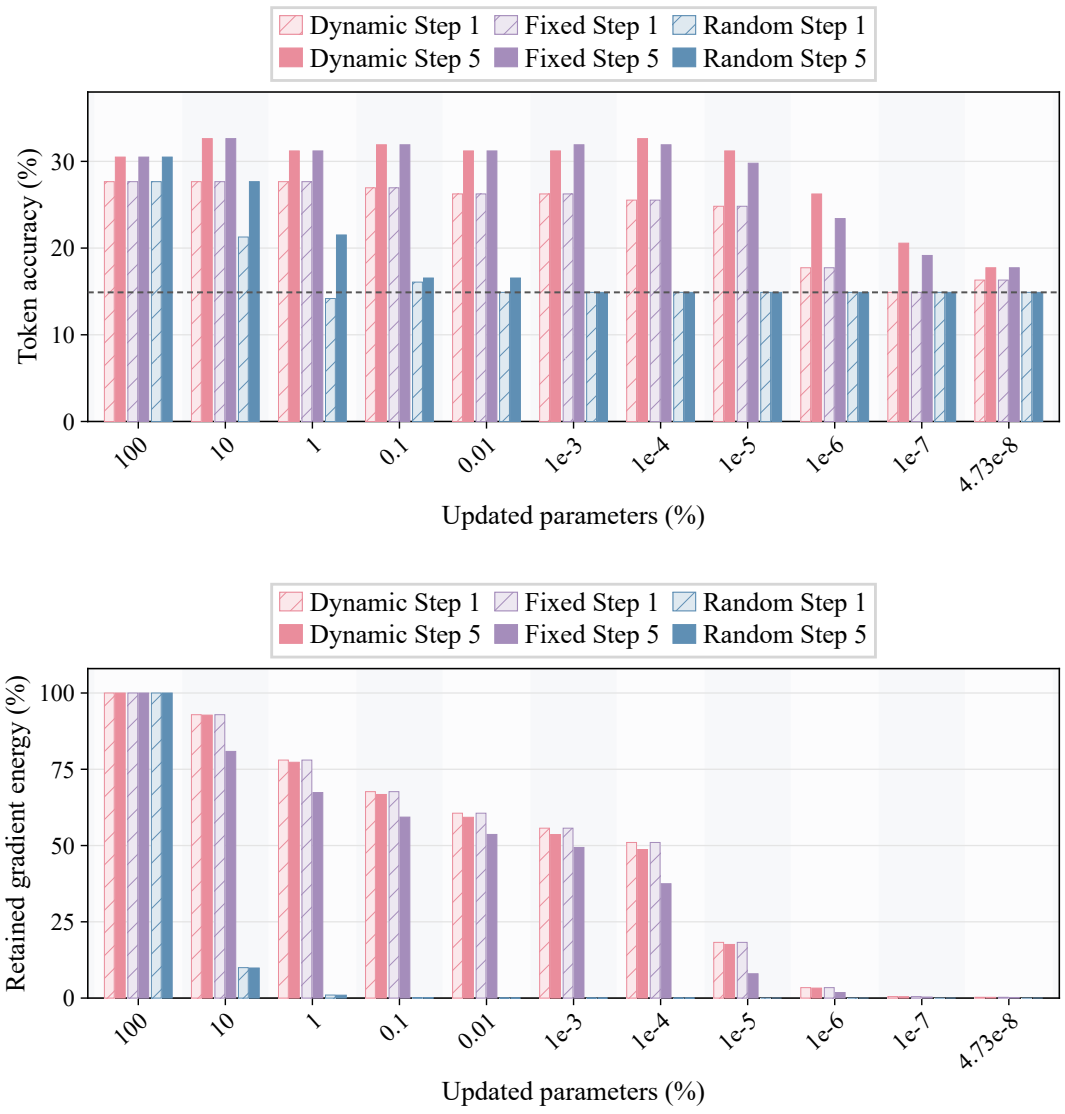


Figure 8: **Accuracy and retained gradient energy on Llama-3.2-3B.** Fixed-gradient selection approaches full-update token accuracy while retaining only a small fraction of gradient energy. Hatched and solid bars show one and five updates, respectively.

Table 24: **Sparse parameter updates on Llama-3.2-3B.** Results under reference answer prefixes. Bold and underlining mark the best and second-best accuracy at each update fraction.

Updated (%)	Parameters	Random fixed	Gradient fixed	Gradient dynamic
(a) Numerical token accuracy (%)				
0 (before)	0	14.89	14.89	14.89
4.73×10^{-8}	1	<u>14.89</u> ± 0.00	**17.73**	**17.73**
1×10^{-7}	2	14.89 ± 0.00	<u>19.15</u>	**20.57**
1×10^{-6}	21	14.89 ± 0.00	<u>23.40</u>	**26.24**
1×10^{-5}	211	14.89 ± 0.00	<u>29.79</u>	**31.21**
0.0001	2,114	14.89 ± 0.00	<u>31.91</u>	**32.62**
0.001	21,141	14.89 ± 0.00	**31.91**	<u>31.21</u>
0.01	211,410	<u>16.55</u> ± 0.41	**31.21**	**31.21**
0.1	2,114,101	<u>16.55</u> ± 0.82	**31.91**	**31.91**
1	21,141,012	<u>21.51</u> ± 1.48	**31.21**	**31.21**
10	211,410,125	<u>27.66</u> ± 0.00	**32.62**	**32.62**
100 (all)	2,114,101,248	30.50	30.50	30.50
(b) Answer reconstruction accuracy (%)				
0 (before)	0	0.00	0.00	0.00
4.73×10^{-8}	1	0.00 ± 0.00	0.00	0.00
1×10^{-7}	2	0.00 ± 0.00	0.00	0.00
1×10^{-6}	21	0.00 ± 0.00	0.00	0.00
1×10^{-5}	211	0.00 ± 0.00	0.00	0.00
0.0001	2,114	<u>0.00</u> ± 0.00	<u>0.00</u>	**5.00**
0.001	21,141	<u>0.00</u> ± 0.00	**5.00**	**5.00**
0.01	211,410	<u>0.00</u> ± 0.00	**5.00**	**5.00**
0.1	2,114,101	<u>0.00</u> ± 0.00	**5.00**	**5.00**
1	21,141,012	<u>0.00</u> ± 0.00	**5.00**	**5.00**
10	211,410,125	<u>0.00</u> ± 0.00	**5.00**	**5.00**
100 (all)	2,114,101,248	5.00	5.00	5.00
(c) Retained gradient energy (%)				
0 (before)	0	—	—	—
4.73×10^{-8}	1	1.778×10^{-8} ± 6.633×10^{-9}	0.19	0.25
1×10^{-7}	2	3.063×10^{-8} ± 1.096×10^{-8}	0.35	0.46
1×10^{-6}	21	4.601×10^{-7} ± 3.113×10^{-8}	1.82	3.27
1×10^{-5}	211	4.924×10^{-6} ± 6.447×10^{-7}	7.97	17.58
0.0001	2,114	9.071×10^{-5} ± 7.691×10^{-5}	37.49	48.70
0.001	21,141	1.160×10^{-3} ± 4.763×10^{-4}	49.38	53.61
0.01	211,410	8.202×10^{-3} ± 7.009×10^{-4}	53.64	59.27
0.1	2,114,101	0.10 ± 4.556×10^{-3}	59.32	66.74
1	21,141,012	0.96 ± 4.875×10^{-3}	67.37	77.30
10	211,410,125	9.86 ± 0.01	80.85	92.76
100 (all)	2,114,101,248	100.00	100.00	100.00

L Parameter Changes and Representation Response

Protocol. We evaluate Llama-3.2-3B, Qwen3-4B, and DeepSeek-Math-7B on a fixed set of five questions from each of Addition-50, MATH401, GSM8K, BB-Arith, NUPA, NB-Arith, and NB-Mixed. At the first numerical position in each reference answer, we use the question and preceding reference tokens as input. Original parameters are restored for each question and update fraction. We take five updates with learning rate 0.01 and $\gamma = 0.5$, using all numerical tokens in the objective. Parameter selection uses the largest absolute initial gradients and remains fixed across updates; the reference token is not used for selection or optimization. Selected gradients are clipped to a global L_2 norm of 1 before each update. The eligible pool includes MLP projections and normalization parameters within transformer blocks, including Qwen's query/key normalization. The update fractions range from 100% to $10^{-7}\%$ of this pool. These representation experiments are separate from the accuracy experiment in Appendix K.

Measurements. We count scalar parameters whose stored values actually change, averaging over updates and questions. Counts can be smaller than the selected set because updates are written back to FP16 parameters. For each layer, we measure the L_2 distance between the current hidden state and its initial value at the prediction position, then average over questions. Figure 5 sums parameter counts over modules and shows representation displacement at step 5. The appendix retains the module breakdown and all six representation measurements. Counts share an absolute color scale, with a linear region near zero and logarithmic scaling for larger values; displacement uses a linear scale shared within each model. Layers are indexed from zero, excluding the embedding output. The last hidden-state measurement includes the model's final normalization. Displacement measures the magnitude of change, not its direction or prediction accuracy.

Update fraction and propagation. Figures 9 and 10 compare update fractions on MATH401. At small fractions, changes mainly occur in MLP down projections, with the strongest concentration in different early layers across models. On DeepSeek-Math-7B, at $10^{-4}\%$, the first two layers account for 96.83% of the parameters changed per update, yet the mean displacement across nonfinal layers remains within 6% of the dense-update value. At $10^{-5}\%$, layers 18–29 have no parameter changes, while their representations still respond to adaptation. This separation between the location of parameter changes and the layers that respond is consistent with their effects propagating through subsequent computation. At the smallest fractions, the response weakens but still extends beyond the layers with parameter changes.

Variation across models and tasks. Figures 11 and 12 fix the update fraction at 0.01% and compare all seven tasks. The layer and module distributions vary across models and tasks, so the same update fraction does not imply the same intervention. Representation displacement also varies with the model and input. In the figures, G/U/D denote the MLP gate, up, and down projections; N1/N2 denote the two block normalization modules; Q/K denote query/key normalization. NumericBench and BB-Mixed denote NB-Arith and NB-Mixed, respectively.

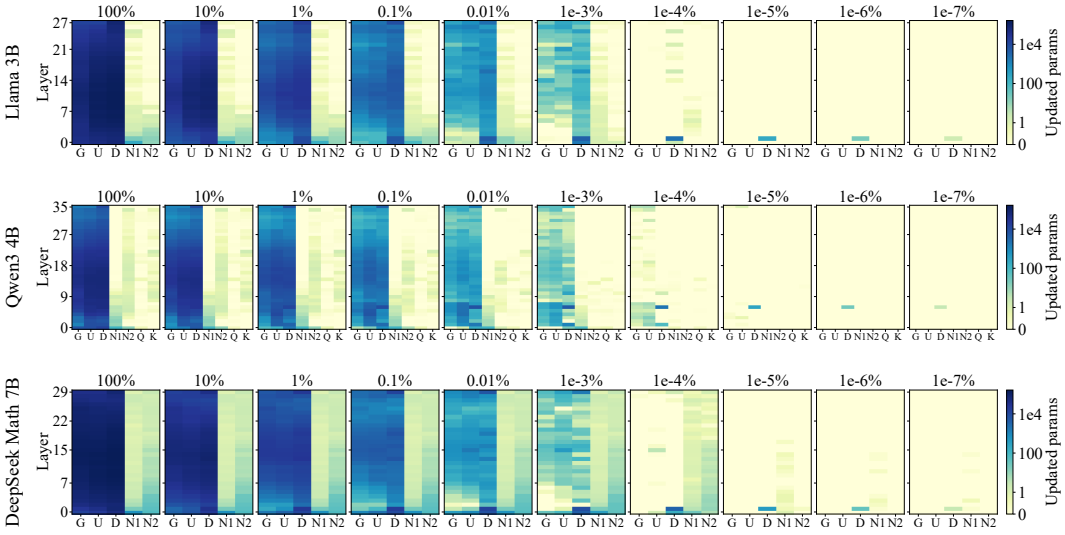


Figure 9: **Parameter changes across update fractions on MATH401.** Each panel shows mean actual changes per update by layer and module. Columns give the selected fraction of eligible parameters.

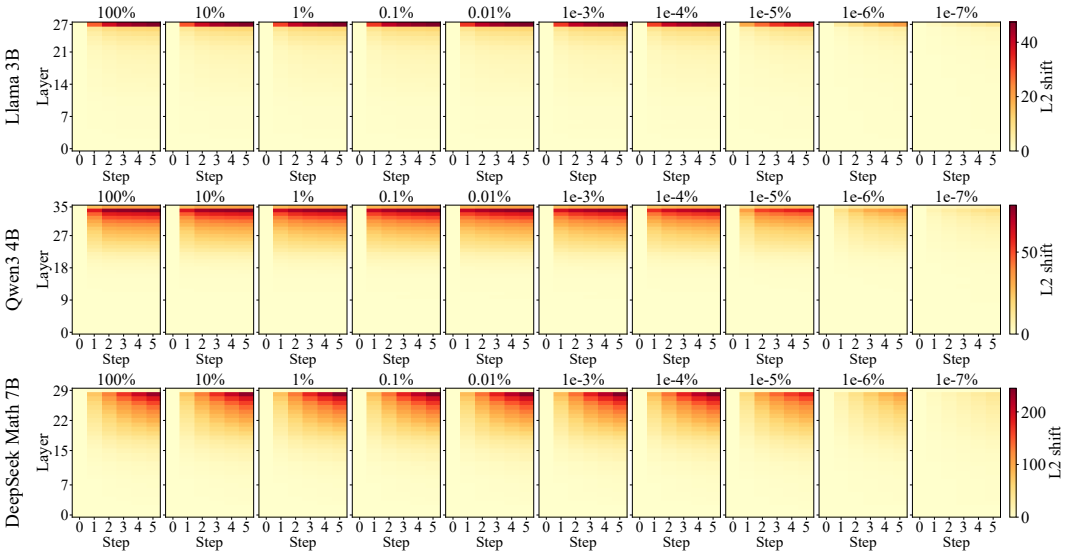


Figure 10: **Representation displacement across update fractions on MATH401.** Each panel shows mean hidden-state L_2 displacement from step 0 over the five updates. A common linear scale is used within each model.

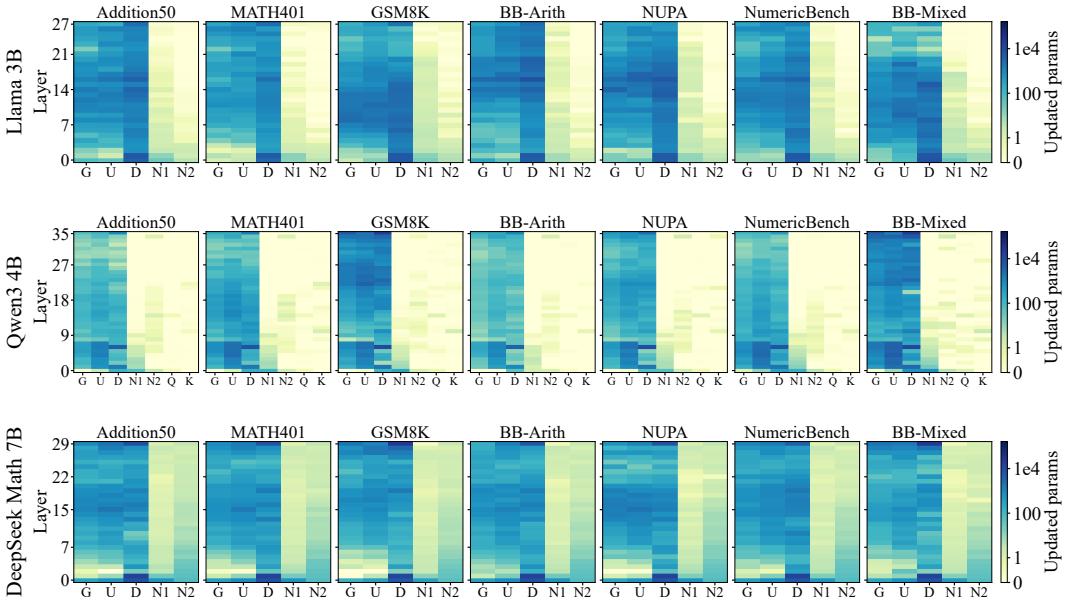


Figure 11: **Parameter changes across tasks at a** 0.01% **update fraction.** Each panel shows mean actual changes per update by layer and module. The same absolute count scale is used across models and tasks.

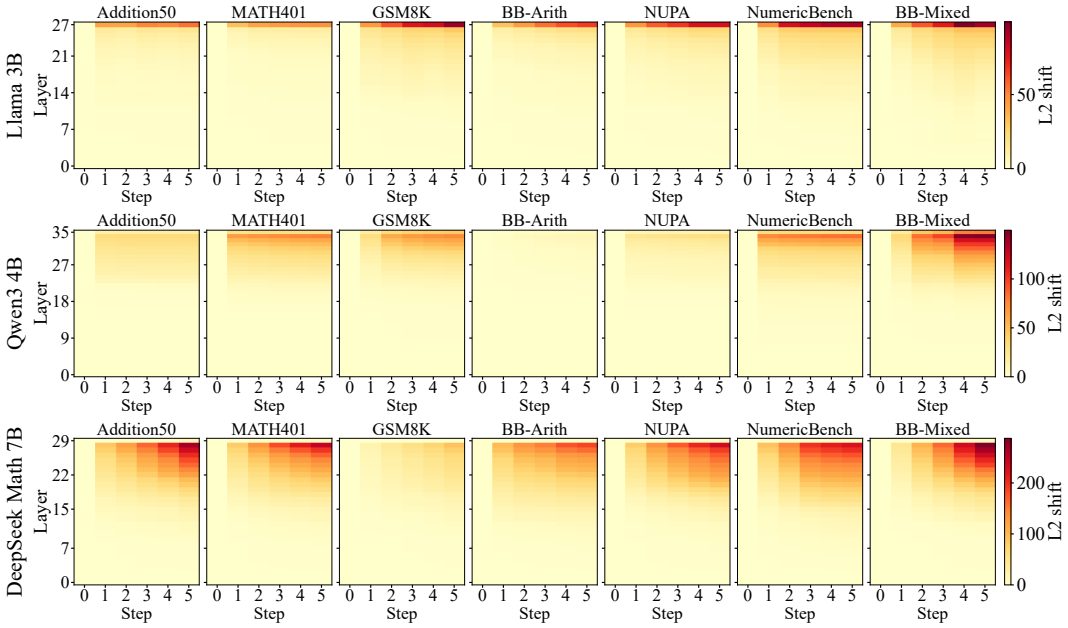


Figure 12: **Representation displacement across tasks at a** 0.01% **update fraction.** Each panel shows mean hidden-state L_2 displacement from step 0. Scales are shared across tasks within a model.

M Local Approximation of Numerical Prediction Changes

We examine the accuracy of the first-order parameter-response approximation for finite NuRA updates. Following the analysis of individual updates in LLM learning dynamics (Ren & Sutherland, 2025), we compare predicted and measured changes in the logit margin between a reference token and a competing numerical token.

Protocol. We evaluate Llama-3.2-1B-Instruct on 96 questions and Llama-3.2-3B-Instruct on 24 questions, sampling 32 and 8 questions, respectively, from each of Addition-50, MATH401, and BB-Arith. Each question contributes its first numerical answer position, conditioned on the preceding reference tokens. All MLP layers are eligible for adaptation; all other parameters remain fixed. With $\mathcal{L}$ denoting angular dispersion, the binary mask s retains the largest 0.01% of entries of $|\nabla_\theta \mathcal{L}|$. We rescale this masked gradient to each prescribed update norm δ,

$$v = s \odot \nabla_\theta \mathcal{L}, \qquad \Delta\theta = -\delta \frac{v}{\|v\|_2}. \tag{M.1}$$

Each update starts independently from the pretrained parameters, with the direction and mask fixed across the six norms. Evaluations use FP32 and $\gamma = 1$.

Predicted and measured response. Let y be the reference token and j the highest-logit numerical token other than y before adaptation. We keep j fixed and define $m(\theta) = z_y(\theta) - z_j(\theta)$. The predicted and measured changes are

$$\widehat{\Delta m} = \nabla_\theta m(\theta)^\top \Delta\theta, \qquad \Delta m = m(\theta + \Delta\theta) - m(\theta). \tag{M.2}$$

The reference token enters only this diagnostic, not the update objective or coordinate selection. We report mean absolute error (MAE) in logit units, sign agreement as a percentage of positions, and relative L_2 error,

$$\text{Rel. } L_2 \text{ error} = 100 \frac{\|\widehat{\boldsymbol{\Delta m}} - \boldsymbol{\Delta m}\|_2}{\|\boldsymbol{\Delta m}\|_2}, \tag{M.3}$$

where vectors collect all evaluated positions for one model and update norm.

Table 25: **First-order prediction of numerical margin changes.** Independent sparse updates from the original parameters. MAE is in logit units; relative L_2 error and sign agreement are percentages.

Update norm $\|\Delta\theta\|_2$	Llama-3.2-1B MAE ↓	Llama-3.2-1B Relative L_2 error (%) ↓	Llama-3.2-1B Sign agreement (%) ↑	Llama-3.2-3B MAE ↓	Llama-3.2-3B Relative L_2 error (%) ↓	Llama-3.2-3B Sign agreement (%) ↑
0.0005	0.0071	2.63	100.00	0.0144	4.06	100.00
0.0010	0.0301	5.63	98.96	0.0707	12.90	100.00
0.0020	0.1310	12.49	97.92	0.3807	56.59	100.00
0.0040	0.5737	28.09	94.79	1.3272	112.14	95.83
0.0080	2.1431	57.76	92.71	3.3851	140.69	95.83
0.0200	9.0379	136.36	89.58	12.4092	228.70	95.83

At the smallest norm, relative error is 2.63% on Llama-1B and 4.06% on Llama-3B, with 100% sign agreement on both (Table 25). Magnitude estimates become less accurate as the update grows. On Llama-3B, relative error exceeds 100% from a norm of 0.004, even though sign agreement remains 95.83%. These results support the local approximation at small update norms, with agreement in direction persisting over a wider range than agreement in magnitude. Sign agreement includes both margin increases and decreases; it measures approximation accuracy rather than improved token correctness.

N NUMERICAL DIRECTIONS IN THE VOCABULARY SPACE

We place numerical output directions in the context of the full vocabulary. The projections show their position within each model's pretrained geometry and how posterior anchors move during adaptation.

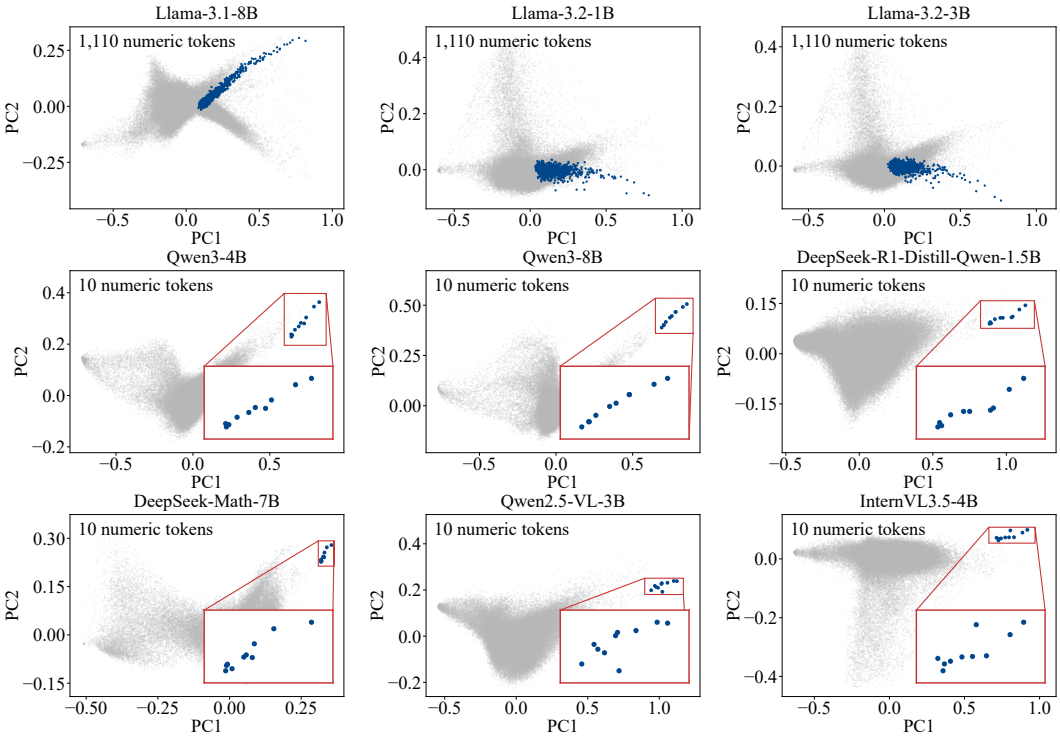


Figure 13: **Numerical output directions across nine models.** Blue points denote numerical tokens and gray points the remaining vocabulary. PCA is fitted independently to each model's normalized output vectors; coordinates are not comparable across panels. Red boxes enlarge numerical regions for models with ten numerical tokens.

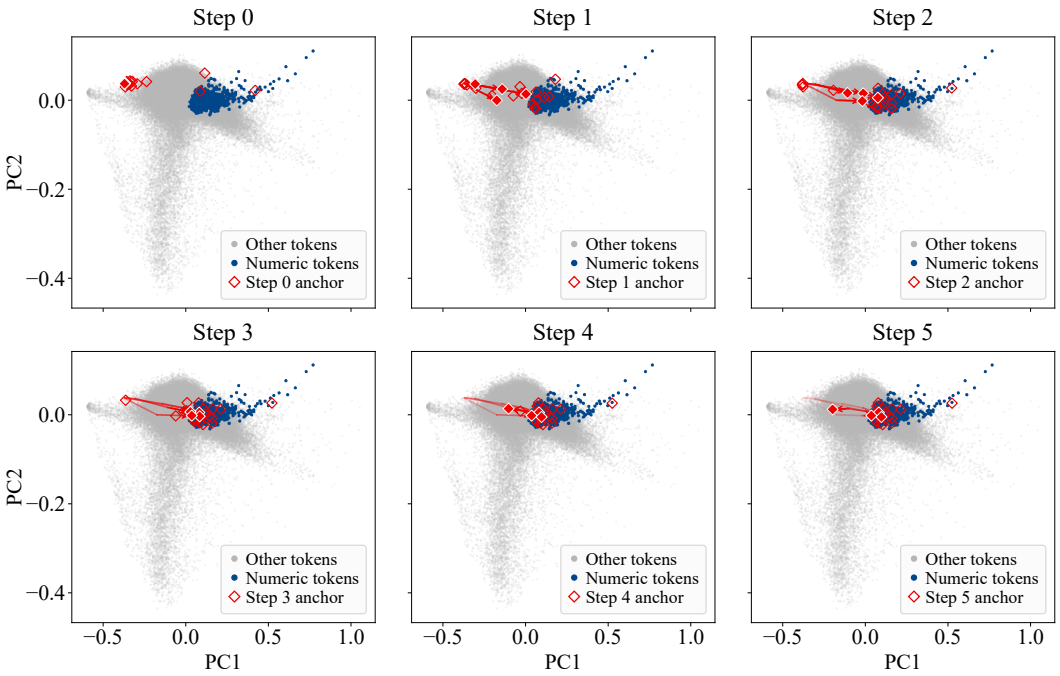


Figure 14: **Full vocabulary-space trajectories on Llama-3.2-3B.** All 24 recorded anchors at steps 0–5, projected with a single PCA basis fitted to the normalized vocabulary directions. Numerical tokens are blue, other tokens gray, and current anchors red. Paths join recorded updates.

The main-text view retains three illustrative trajectories from this run. These were selected from anchors with large initial projected distances to numerical points; the complete view retains all recorded trajectories. The first two components explain approximately 2.27% of the vocabulary variance, so apparent proximity in this view alone does not establish high-dimensional alignment or successful correction.

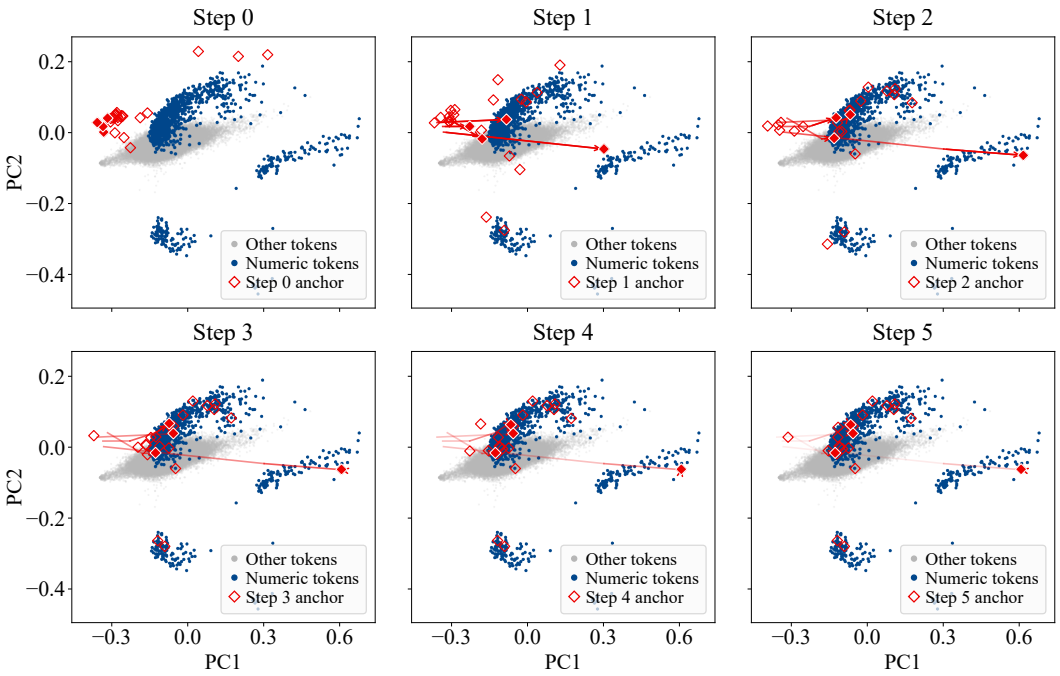


Figure 15: **Anchor trajectories in the numerical subspace on Llama-3.2-3B.** The same 24 anchors as in Figure 14, now projected using PCA fitted to numerical output directions. All steps share this basis and a fixed local view. Blue and gray points denote numerical and other vocabulary directions; red diamonds mark current anchors. Four trajectories with large initial projected distances to numerical points are highlighted.

This numerical-only PCA view resolves structure within the numerical region that is less visible in the full-vocabulary projection. Anchors approach different groups of numerical directions, illustrating input-dependent movement within the pretrained geometry. Projected proximity alone does not determine whether the selected token is correct.

N.1 Numerical Output Geometry Across Additional Models

Figure 16 extends the comparison to six additional models. Near-diagonal bands and recurring off-diagonal patterns remain visible, while their contrast and overall similarity vary across models. GLM-5.3 illustrates that local numerical structure can persist even when digit directions are highly aligned overall.

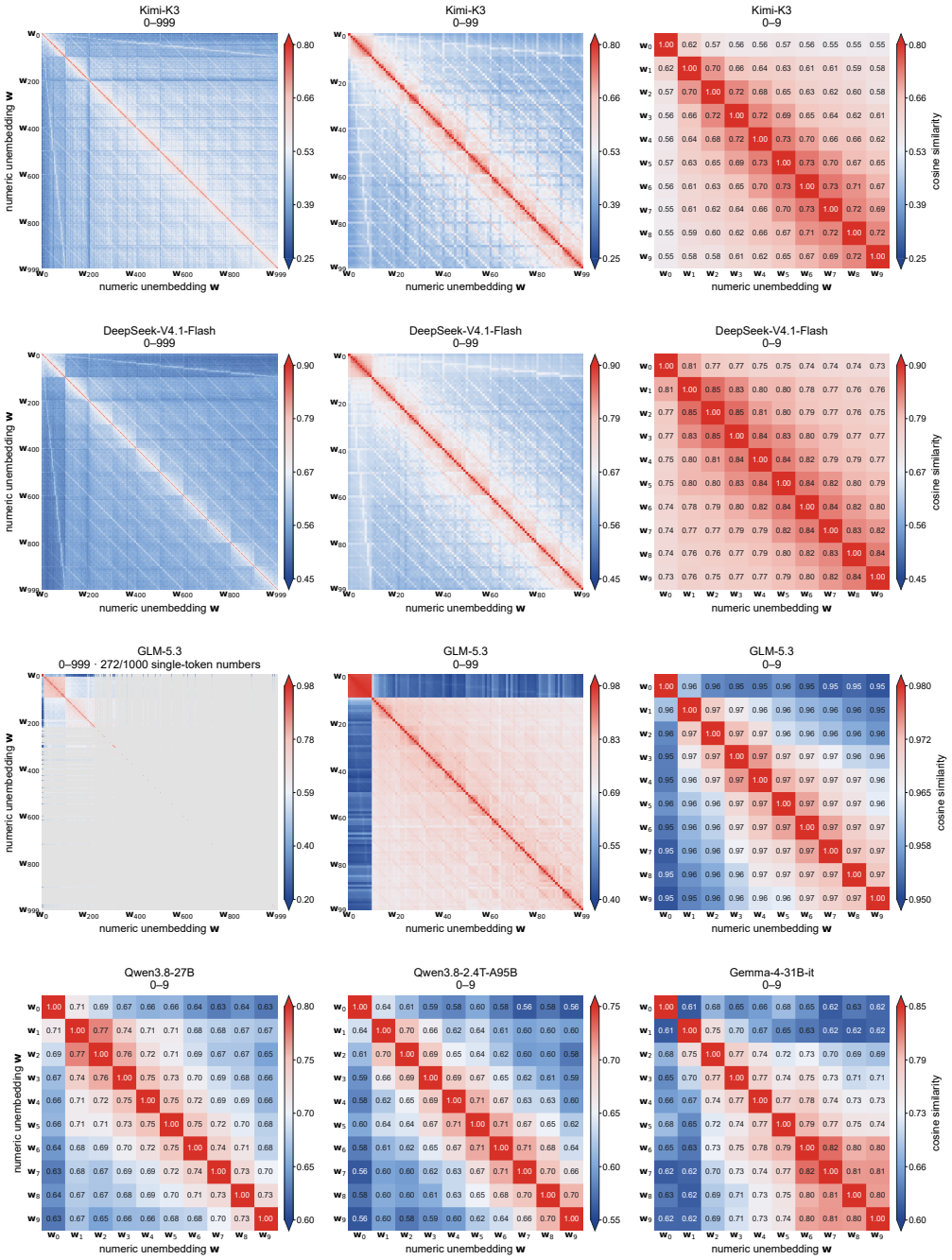


Figure 16: **Numerical output geometry across six additional models.** Pairwise cosine similarities for integers in numerical order, using ordinary decimal strings without leading zeros. The first three rows compare ranges 0–999, 0–99, and 0–9; the last row shows single digits. Gray rows and columns mark numbers without a single-token encoding, including 728 of 1,000 integers for GLM-5.3. Color limits vary by panel; triangular ends mark values beyond these limits.

O NUMERICAL PREDICTION DYNAMICS

Main-text trajectories. Figure 6b uses a separate Llama-3.2-3B adaptation run on Addition-50 with five updates and learning rate 0.02. The selected positions are corrected by the final step; Table 26 gives their predictions, target probabilities, and ranks. The displayed steps share the original six-step t-SNE projection. The trajectories represent posterior anchors.
The initially clustered anchors accompany diffuse posteriors, with numerical entropy between 6.77 and 6.86 nats. As adaptation proceeds, they approach distinct targets, including two corrections from 111 to 653 and 898. The recorded angles to the targets also decrease in the original representation space, supporting the alignment observed in the projection, although individual updates need not improve alignment monotonically.

Additional cases. Figures 17–24 show six groups of successful corrections and two groups of unsuccessful corrections. Each case predicts the next token from a fixed reference prefix; correctness therefore concerns that token, not a complete generated answer. These runs use five updates, learning rate 0.02, $\gamma = 0.5$, and 0.01% of MLP parameters selected by the initial gradient. The target is used only for evaluation. Predicted tokens, target probabilities, and ranks are measured over the full vocabulary, although the objective uses numerical tokens. Cases are selected by outcome and do not estimate correction rates.

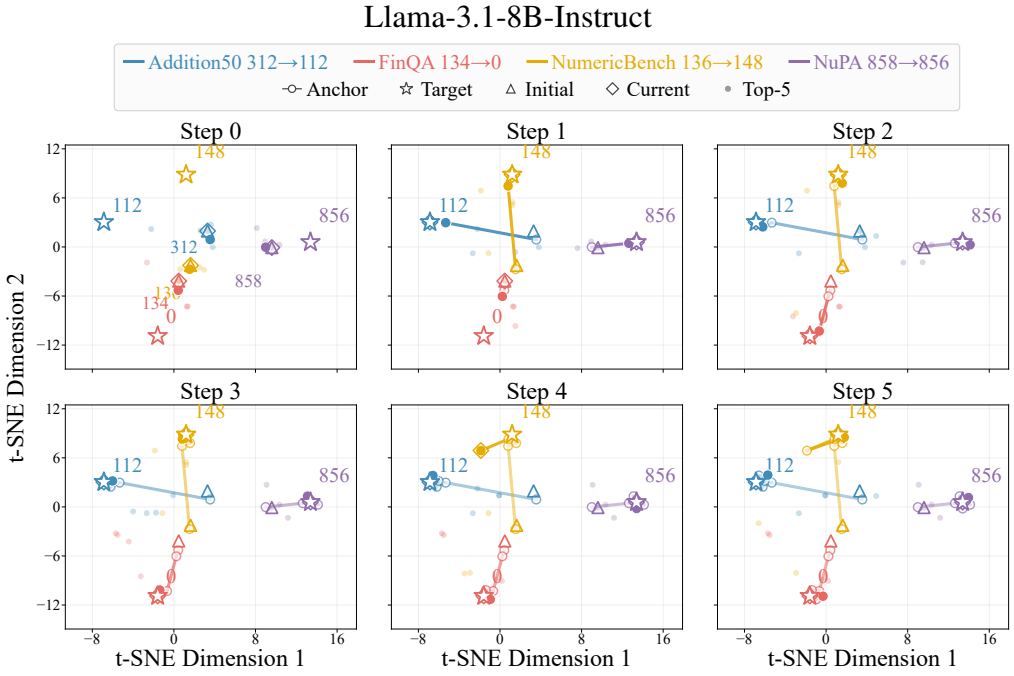
Paired objectives. Figures 25–27 compare `NuRA` with numerical Shannon-entropy minimization using the same input and numerical vocabulary. Each objective selects its own initial-gradient mask, held fixed during adaptation. The anchor for entropy minimization is computed only for visualization. In the Llama-1B example, entropy minimization ends with lower numerical entropy (0.64 versus 1.50) but predicts an incorrect token. Across the three paired examples, `NuRA` ends closer to the reference direction in the original space and corrects the prediction, while entropy minimization ends farther from it. Their initial anchors coincide, and the advantage does not necessarily appear at the first update. The different prediction outcomes thus accompany differences in how probability moves among numerical directions.

Unsuccessful corrections. Figures 23 and 24 show that concentration can also increase around an incorrect candidate. In several Qwen3-4B cases, the initially predicted digit remains unchanged as the posterior sharpens. Proximity to the reference direction in the projection is therefore insufficient for correction. A locally concentrated but incorrect posterior may provide little signal for moving to an alternative numerical region, which limits refinement without answer supervision.

Table 26: **Predictions underlying Figure 6.** Target probabilities and ranks use the full vocabulary.

	Step 0	Step 1	Step 2	Step 3	Step 4	Step 5
			111→653			
Predicted token	111	439	439	653	653	653
p(target)	6.10e-04	7.34e-04	0.0014	0.6833	0.9972	1.0000
Target rank	901	721	218	1	1	1
			111→898			
Predicted token	111	439	938	898	898	898
p(target)	8.32e-04	5.36e-04	0.0017	0.7514	0.9990	0.9999
Target rank	637	955	119	1	1	1
			855→068			
Predicted token	855	068	068	068	068	068
p(target)	7.08e-04	0.2049	0.6087	0.9979	1.0000	1.0000
Target rank	621	1	1	1	1	1
			743→599			
Predicted token	743	559	599	599	599	599
p(target)	8.61e-04	8.76e-04	0.7774	0.9951	1.0000	1.0000
Target rank	616	178	1	1	1	1

Figure 17. Successful corrections



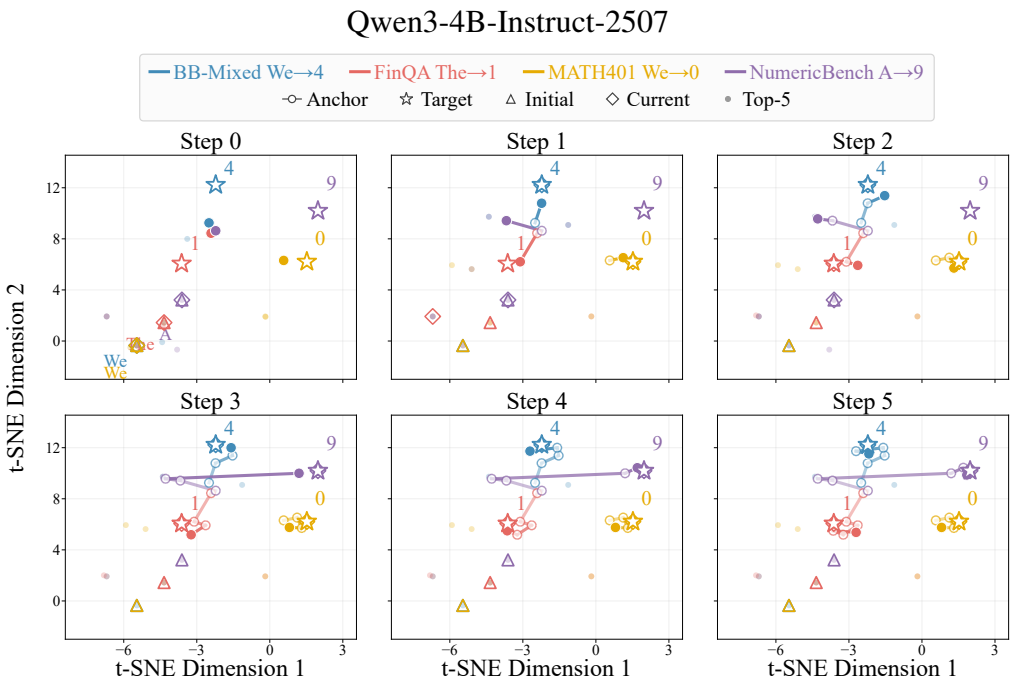
Case 1. Addition-50
Model. Llama-3.1-8B-Instruct
Question. What is the sum of 24688520783 and 87359791884?
Reference answer. 112048312667 **Token position.** 0 **Target token.** 112
Predicted token. 312 → 112 → 112 → 112 → 112 → 112
p(**target**). 0.0295 → 0.9928 → 0.9996 → 0.9901 → 1.0000 → 1.0000
Target rank. 6 → 1 → 1 → 1 → 1 → 1

Case 2. FinQA
Model. Llama-3.1-8B-Instruct
Question. what portion of the total full-time employees of mainline operations are pilots and flight crew instructors?
Reference answer. 0.13202 **Token position.** 0 **Target token.** 0
Predicted token. 134 → 134 → 0 → 0 → 0 → 0
p(**target**). 0.2317 → 0.3597 → 0.9979 → 1.0000 → 1.0000 → 1.0000
Target rank. 2 → 2 → 1 → 1 → 1 → 1

Case 3. NB-Arith
Model. Llama-3.1-8B-Instruct
Question. What is the result of A * B? Please round the answer to two decimal places. A: 449.577, B: -329496.081
Reference answer. -148133859.61 **Token position.** 1 **Target token.** 148
Predicted token. 136 → 148 → 148 → 148 → 149 → 148
p(**target**). 1.09e-04 → 0.7700 → 0.9425 → 0.9775 → 2.09e-06 → 0.9996
Target rank. 49 → 1 → 1 → 1 → 3 → 1

Case 4. NUPA
Model. Llama-3.1-8B-Instruct
Question. Directly return the answer as an integer without any comma separator, like 123 . Add two numbers: 85649647285288561093 + 7622094998724
Reference answer. 85649654907383559817 **Token position.** 0 **Target token.** 856
Predicted token. 858 → 856 → 856 → 856 → 856 → 856
p(**target**). 0.2178 → 0.9862 → 1.0000 → 1.0000 → 1.0000 → 1.0000
Target rank. 2 → 1 → 1 → 1 → 1 → 1

Figure 18. Successful corrections



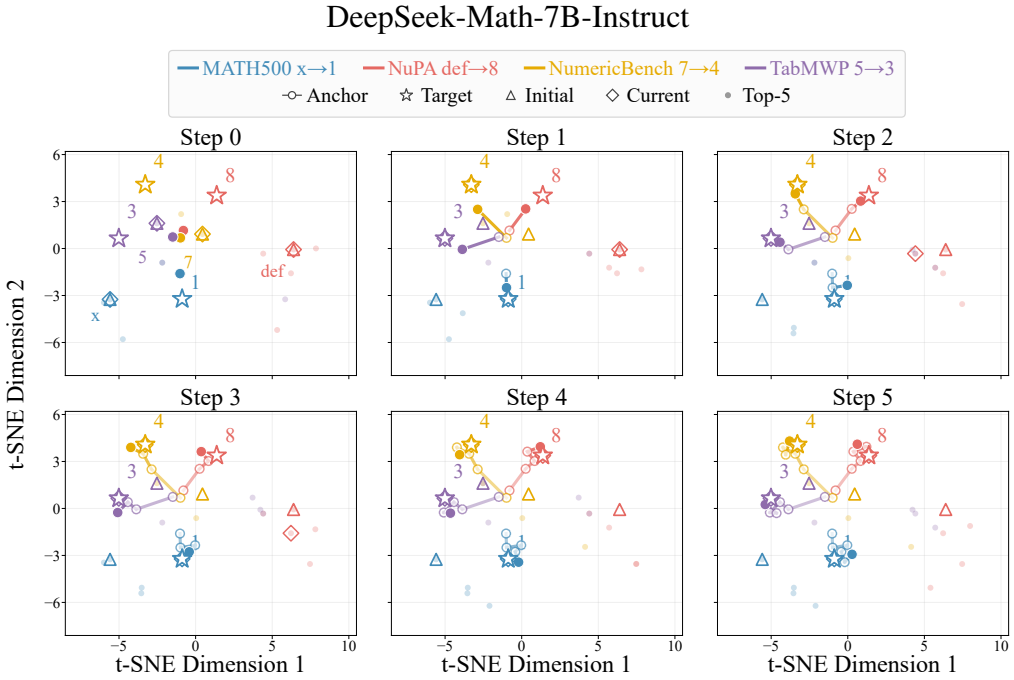
Case 1. NB-Mixed
Model. Qwen3-4B-Instruct-2507
Question. String: 5WK7JpBacBwngY2U5We5m95v7bhlj3H7jMzjGSZOj6P2s7bgn624V2i1k1dP67Z8 51N4iDl6T8n6B6ZO4Qj4C23J17FQy2X8KJrH7R1ae1cTOFz9jV47N3n9ZfQfncOa1c7pYG3mi4ZaaSAi 31Eyzp
How many numbers are there in the string? Note that a sequence like 'a243b' counts as a single number.
Reference answer. 41 **Token position.** 0 **Target token.** 4
Predicted token. We → 4 → 4 → 4 → 4 → 4
p**(target).** 3.38e-06 → 0.9257 → 0.9999 → 1.0000 → 1.0000 → 1.0000
Target rank. 4 → 1 → 1 → 1 → 1 → 1

Case 2. FinQA
Model. Qwen3-4B-Instruct-2507
Question. what is the net change in aon 2019s unpaid restructuring liabilities during 2006?
Reference answer. 18.0 **Token position.** 0 **Target token.** 1
Predicted token. The → To → 1 → 1 → 1 → 1
p**(target).** 1.10e-06 → 6.82e-04 → 0.7946 → 0.7993 → 0.8094 → 0.8117
Target rank. 5 → 3 → 1 → 1 → 1 → 1

Case 3. MATH401
Model. Qwen3-4B-Instruct-2507
Question. e^(iπ)+1=
Reference answer. 0 **Token position.** 0 **Target token.** 0
Predicted token. We → 0 → 0 → 0 → 0 → 0
p**(target).** 2.84e-08 → 1.0000 → 1.0000 → 1.0000 → 1.0000 → 1.0000
Target rank. 8 → 1 → 1 → 1 → 1 → 1

Case 4. NB-Arith
Model. Qwen3-4B-Instruct-2507
Question. What is the result of A + B? Please round the answer to two decimal places. A: -35585.313, B: 98320102.776
Reference answer. 98284517.46 **Token position.** 0 **Target token.** 9
Predicted token. A → A → A → 9 → 9 → 9
p**(target).** 1.70e-11 → 4.10e-09 → 2.93e-13 → 0.9021 → 1.0000 → 1.0000
Target rank. 24 → 13 → 49 → 1 → 1 → 1

Figure 19. Successful corrections



Case 1. MATH500
Model. DeepSeek-Math-7B-Instruct
Question. Find the maximum value of

$$\frac{x-y}{x^4+y^4+6}$$

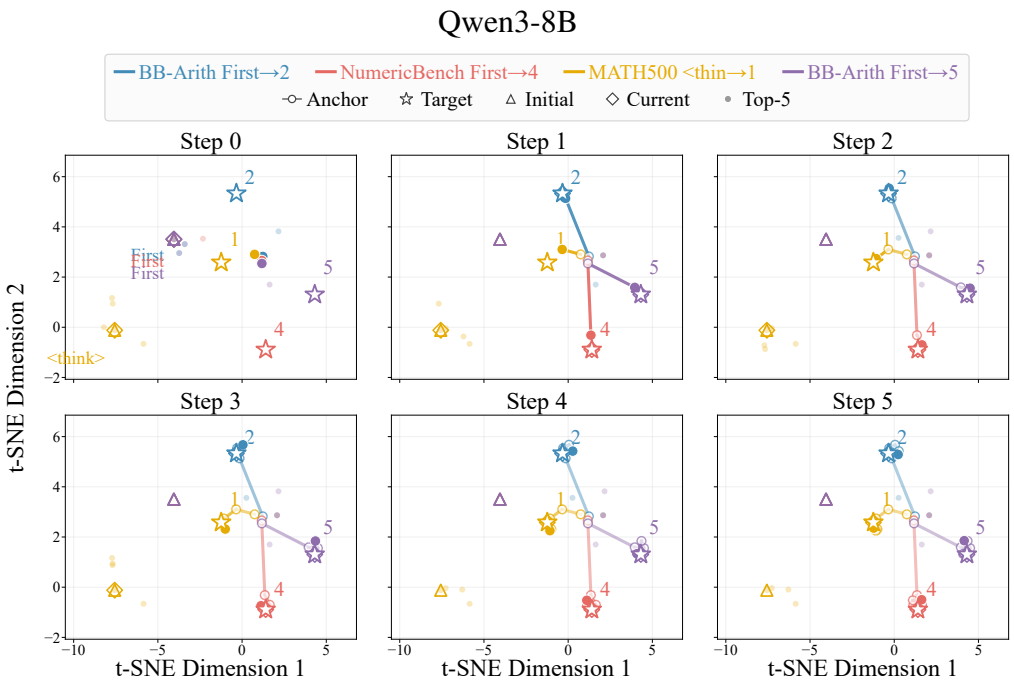
over all real numbers x and y.
Reference answer. $\frac{1}{4}$ **Token position.** 3 **Target token.** 1
Predicted token. x → 1 → 1 → 1 → 1 → 1
p**(target).** 0.1594 → 0.6693 → 0.9812 → 0.9985 → 1.0000 → 1.0000
Target rank. 2 → 1 → 1 → 1 → 1 → 1

Case 2. NUPA
Model. DeepSeek-Math-7B-Instruct
Question. Directly return the answer as an integer without any comma separator, like 123 . Get the maximal number: 8731227499629275265082 and 800332209419922329806549437069031651644
Reference answer. 800332209419922329806549437069031651644 **Token position.** 0 **Target token.** 8
Predicted token. def → def → → function → 8 → 8
p**(target).** 1.54e-06 → 1.17e-05 → 1.06e-05 → 0.0026 → 0.4773 → 0.9999
Target rank. 1048 → 355 → 205 → 16 → 1 → 1

Case 3. NB-Arith
Model. DeepSeek-Math-7B-Instruct
Question. What is the result of A - B? Please round the answer to two decimal places. A: 161.458, B: 49798.726
Reference answer. -49637.27 **Token position.** 1 **Target token.** 4
Predicted token. 7 → 4 → 4 → 4 → 4 → 4
p**(target).** 0.1859 → 0.6439 → 0.9971 → 1.0000 → 1.0000 → 1.0000
Target rank. 2 → 1 → 1 → 1 → 1 → 1

Case 4. TabMWP
Model. DeepSeek-Math-7B-Instruct
Question. Florence spent a week at the beach and recorded the number of shells she found each day. According to the table, what was the rate of change between Tuesday and Wednesday?
Reference answer. -3 **Token position.** 1 **Target token.** 3
Predicted token. 5 → 3 → 3 → 3 → 3 → 3
p**(target).** 0.0572 → 0.9486 → 0.9873 → 0.9979 → 0.9995 → 0.9989
Target rank. 5 → 1 → 1 → 1 → 1 → 1

Figure 20. Successful corrections



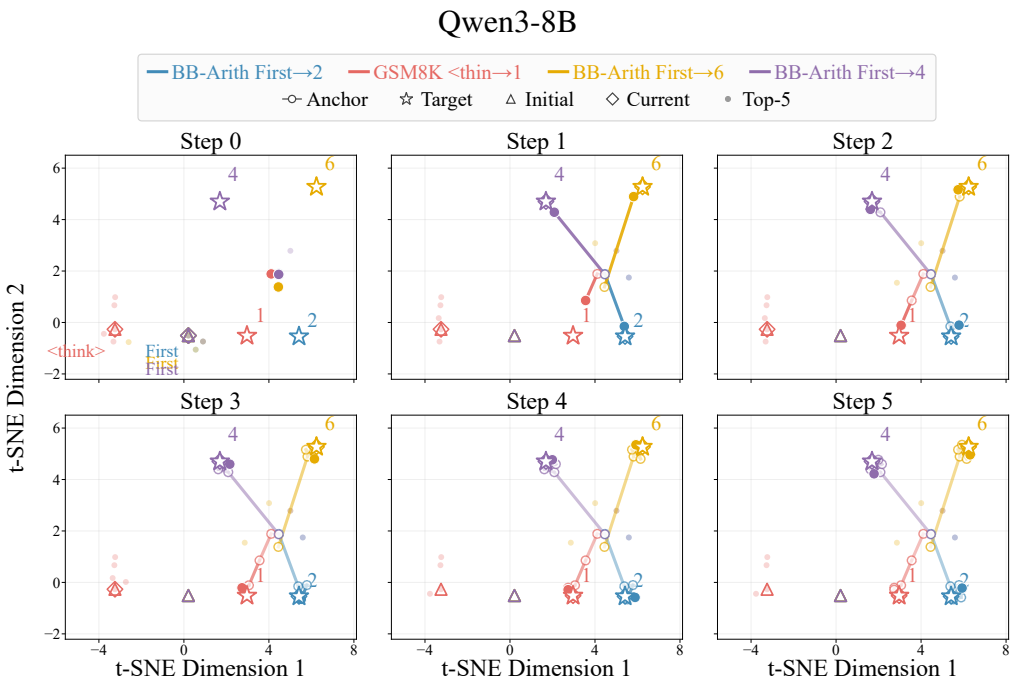
Case 1. BB-Arith
Model. Qwen3-8B
Question. What is 682 minus 897?
Reference answer. -215 **Token position.** 1 **Target token.** 2
Predicted token. First → 2 → 2 → 2 → 2 → 2
p**(target).** 0.0154 → 1.0000 → 1.0000 → 1.0000 → 1.0000 → 1.0000
Target rank. 6 → 1 → 1 → 1 → 1 → 1

Case 2. NB-Arith
Model. Qwen3-8B
Question. What is the result of A - B? Please round the answer to two decimal places. A: 161.458, B: 49798.726
Reference answer. -49637.27 **Token position.** 1 **Target token.** 4
Predicted token. First → 4 → 4 → 4 → 4 → 4
p**(target).** 0.1481 → 0.9998 → 1.0000 → 1.0000 → 1.0000 → 1.0000
Target rank. 2 → 1 → 1 → 1 → 1 → 1

Case 3. MATH500
Model. Qwen3-8B
Question. The product of a set of distinct positive integers is 84. What is the least possible sum of these integers?
Reference answer. 14 **Token position.** 0 **Target token.** 1
Predicted token. ¡think¿ → ¡think¿ → ¡think¿ → ¡think¿ → 1 → 1
p**(target).** 3.45e-15 → 9.52e-13 → 8.81e-09 → 7.98e-08 → 0.9072 → 0.9353
Target rank. 64363 → 6714 → 63 → 8 → 1 → 1

Case 4. BB-Arith
Model. Qwen3-8B
Question. What is 1395 minus 6647?
Reference answer. -5252 **Token position.** 1 **Target token.** 5
Predicted token. First → 5 → 5 → 5 → 5 → 5
p**(target).** 0.0387 → 1.0000 → 1.0000 → 1.0000 → 1.0000 → 1.0000
Target rank. 6 → 1 → 1 → 1 → 1 → 1

Figure 21. Successful corrections



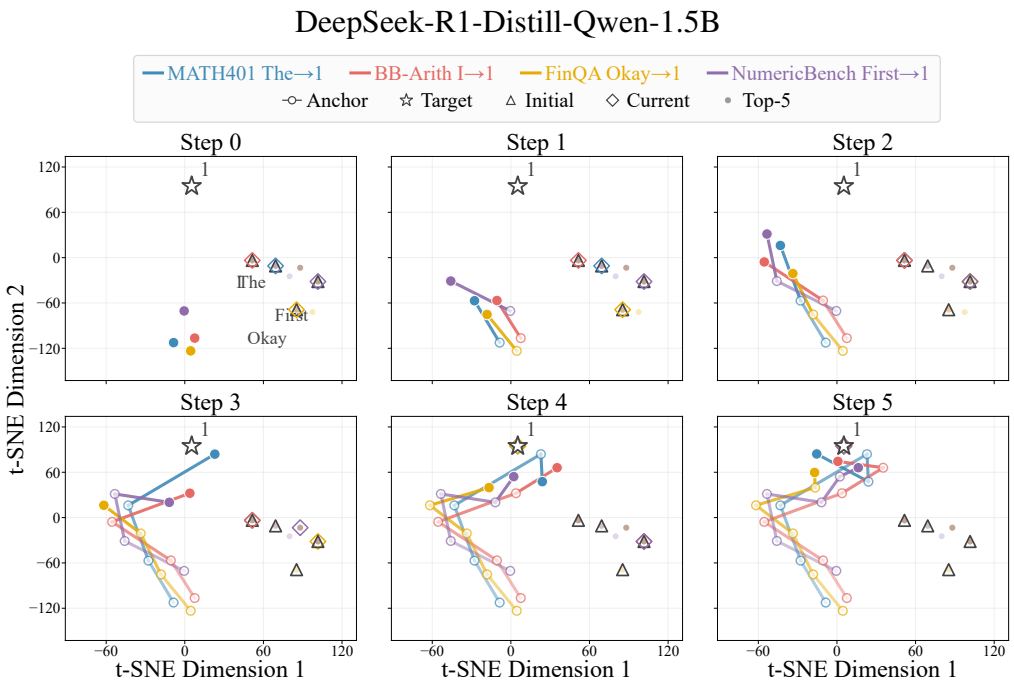
Case 1. BB-Arith
Model. Qwen3-8B
Question. What is 4620 minus 6628?
Reference answer. -2008 **Token position.** 1 **Target token.** 2
Predicted token. First → 2 → 2 → 2 → 2 → 2
p**(target).** 0.0074 → 0.9986 → 1.0000 → 1.0000 → 1.0000 → 1.0000
Target rank. 4 → 1 → 1 → 1 → 1 → 1

Case 2. GSM8K
Model. Qwen3-8B
Question. John is a carpenter. For his friend Ali, he manufactured 4 wooden tables for $20 each and 2 roof frames for $10 each. How much does Ali have to pay John?
Reference answer. 100 **Token position.** 0 **Target token.** 1
Predicted token. ¡think¿ → ¡think¿ → ¡think¿ → ¡think¿ → 1 → 1
p**(target).** 3.07e-16 → 1.24e-14 → 5.40e-08 → 5.80e-07 → 0.7656 → 0.7738
Target rank. 102426 → 37811 → 13 → 7 → 1 → 1

Case 3. BB-Arith
Model. Qwen3-8B
Question. What is 2950 minus 9438?
Reference answer. -6488 **Token position.** 1 **Target token.** 6
Predicted token. First → 6 → 6 → 6 → 6 → 6
p**(target).** 5.49e-04 → 1.0000 → 1.0000 → 1.0000 → 1.0000 → 1.0000
Target rank. 18 → 1 → 1 → 1 → 1 → 1

Case 4. BB-Arith
Model. Qwen3-8B
Question. What is 2402 minus 6658?
Reference answer. -4256 **Token position.** 1 **Target token.** 4
Predicted token. First → 4 → 4 → 4 → 4 → 4
p**(target).** 0.0029 → 1.0000 → 1.0000 → 1.0000 → 1.0000 → 1.0000
Target rank. 11 → 1 → 1 → 1 → 1 → 1

Figure 22. Successful corrections



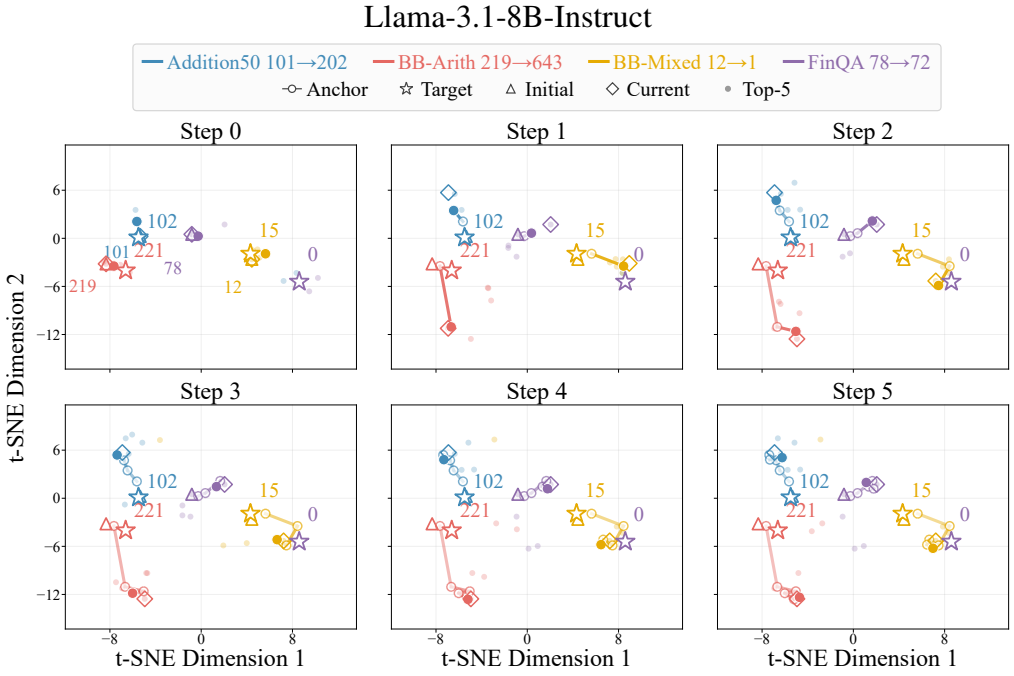
Case 1. MATH401
Model. DeepSeek-R1-Distill-Qwen-1.5B
Question. 501+515=
Reference answer. 1016 **Token position.** 0 **Target token.** 1
Predicted token. The → The → I → I → 1 → 1
p**(target).** 9.64e-08 → 1.87e-05 → 0.0010 → 0.1049 → 0.4050 → 0.5654
Target rank. 707 → 24 → 5 → 4 → 1 → 1

Case 2. BB-Arith
Model. DeepSeek-R1-Distill-Qwen-1.5B
Question. What is 73 plus 98?
Reference answer. 171 **Token position.** 0 **Target token.** 1
Predicted token. I → I → I → I → First → 1
p**(target).** 6.87e-07 → 1.31e-05 → 3.73e-04 → 0.0391 → 0.3230 → 0.5249
Target rank. 240 → 25 → 6 → 4 → 2 → 1

Case 3. FinQA
Model. DeepSeek-R1-Distill-Qwen-1.5B
Question. what is the total value of shares purchased during october 2018?
Reference answer. 13435200.0 **Token position.** 0 **Target token.** 1
Predicted token. Okay → Okay → First → First → 1 → 1
p**(target).** 5.19e-07 → 5.76e-04 → 0.0185 → 0.1292 → 0.4486 → 0.5820
Target rank. 212 → 8 → 7 → 4 → 1 → 1

Case 4. NB-Arith
Model. DeepSeek-R1-Distill-Qwen-1.5B
Question. What is the result of A divided by B? Please round the answer to two decimal places. A: 58278.465, B: 34253.351
Reference answer. 1.7 **Token position.** 0 **Target token.** 1
Predicted token. First → First → First → To → First → 1
p**(target).** 1.17e-04 → 0.0026 → 0.0070 → 0.0951 → 0.1056 → 0.3720
Target rank. 16 → 6 → 4 → 4 → 4 → 1

Figure 23. Failed corrections



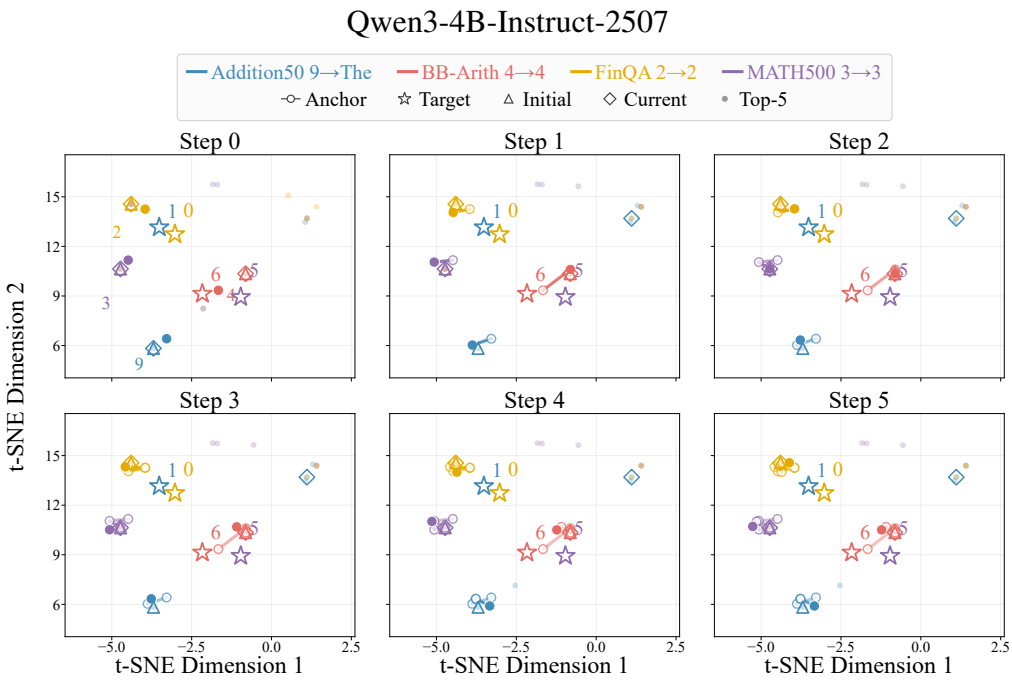
Case 1. Addition-50
Model. Llama-3.1-8B-Instruct
Question. What is the sum of 407053881524017593146 and 612958120259747478090?
Reference answer. 1020012001783765071236 **Token position.** 0 **Target token.** 102
Predicted token. 101 → 202 → 202 → 202 → 202 → 202
p**(target).** 0.1276 → 8.80e-04 → 5.66e-07 → 1.04e-07 → 6.31e-08 → 8.23e-12
Target rank. 4 → 4 → 5 → 19 → 3 → 6

Case 2. BB-Arith
Model. Llama-3.1-8B-Instruct
Question. What is 64379 minus 66595?
Reference answer. -2216 **Token position.** 1 **Target token.** 221
Predicted token. 219 → 642 → 643 → 643 → 643 → 643
p**(target).** 0.2717 → 3.57e-04 → 1.07e-11 → 7.59e-11 → 6.08e-14 → 1.48e-14
Target rank. 2 → 21 → 12196 → 164 → 9507 → 21403

Case 3. NB-Mixed
Model. Llama-3.1-8B-Instruct
Question. String: 4I23159vL1356363K5Yi2aD2k1g9m87A79r7iE74i3Qrp2f545
How many numbers are there in the string? Note that a sequence like 'a243b' counts as a single number.
Reference answer. 15 **Token position.** 0 **Target token.** 15
Predicted token. 12 → 3 → 1 → 1 → 1 → 1
p**(target).** 0.0788 → 2.06e-08 → 4.37e-12 → 2.59e-15 → 2.00e-16 → 1.38e-17
Target rank. 3 → 16 → 33 → 67142 → 74 → 217

Case 4. FinQA
Model. Llama-3.1-8B-Instruct
Question. as of december 312018 what was the percent of restricted cash and marketable securities dedicated to the support our insurance programs
Reference answer. 0.7271 **Token position.** 0 **Target token.** 0
Predicted token. 78 → 72 → 72 → 72 → 72 → 72
p**(target).** 0.0302 → 2.90e-04 → 8.63e-08 → 1.54e-15 → 1.39e-16 → 1.14e-16
Target rank. 5 → 8 → 5 → 407 → 132 → 132

Figure 24. Failed corrections



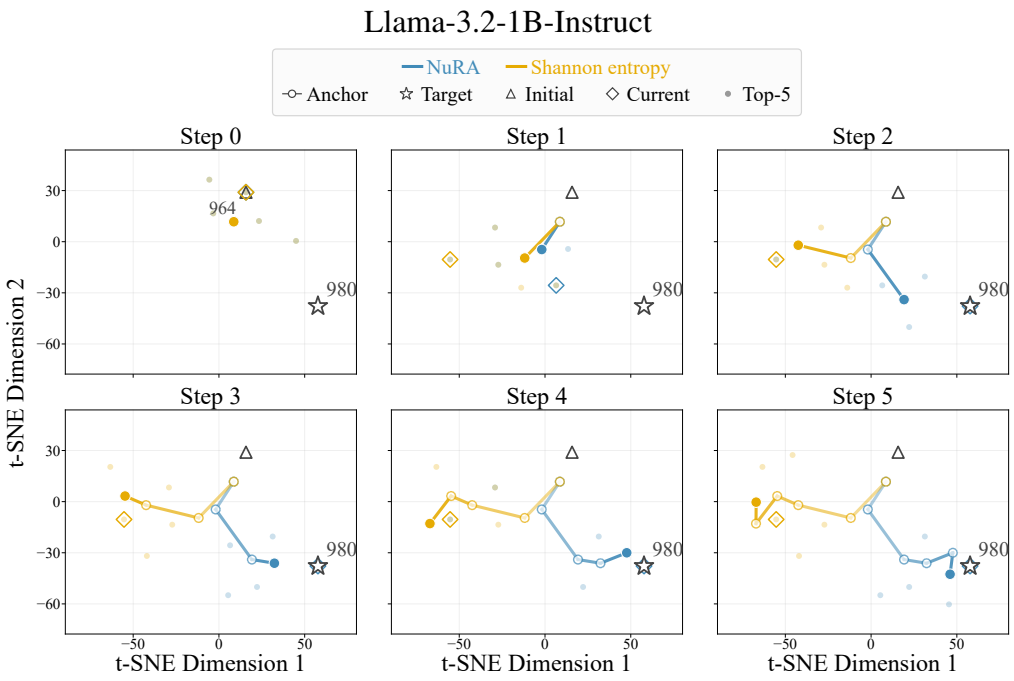
Case 1. Addition-50
Model. Qwen3-4B-Instruct-2507
Question. What is the sum of 9 and 8?
Reference answer. 17 **Token position.** 0 **Target token.** 1
Predicted token. 9 → The → The → The → The → The
p**(target).** 0.0099 → 7.35e-12 → 1.35e-11 → 2.02e-11 → 2.43e-11 → 2.94e-11
Target rank. 3 → 17 → 16 → 15 → 15 → 15

Case 2. BB-Arith
Model. Qwen3-4B-Instruct-2507
Question. What is 49 plus 20?
Reference answer. 69 **Token position.** 0 **Target token.** 6
Predicted token. 4 → 4 → 4 → 4 → 4 → 4
p**(target).** 0.4610 → 5.61e-12 → 5.61e-12 → 5.61e-12 → 5.70e-12 → 5.52e-12
Target rank. 2 → 4 → 4 → 4 → 4 → 4

Case 3. FinQA
Model. Qwen3-4B-Instruct-2507
Question. what was the lowest effective tax rate in the three year period?
Reference answer. 0.27 **Token position.** 0 **Target token.** 0
Predicted token. 2 → 2 → 2 → 2 → 2 → 2
p**(target).** 2.05e-10 → 1.30e-10 → 1.04e-10 → 8.23e-11 → 7.20e-11 → 6.00e-11
Target rank. 12 → 10 → 10 → 10 → 10 → 10

Case 4. MATH500
Model. Qwen3-4B-Instruct-2507
Question. Let $z = 1 + i$ and $w = \dfrac{3z+1}{5z+7}$. Find $|w|$.
Reference answer. $\frac{5}{13}$ **Token position.** 3 **Target token.** 5
Predicted token. 3 → 3 → 3 → 3 → 3 → 3
p**(target).** 4.45e-04 → 1.48e-08 → 6.66e-12 → 6.66e-12 → 6.56e-12 → 6.46e-12
Target rank. 6 → 8 → 8 → 8 → 8 → 8

Figure 25. Entropy comparison

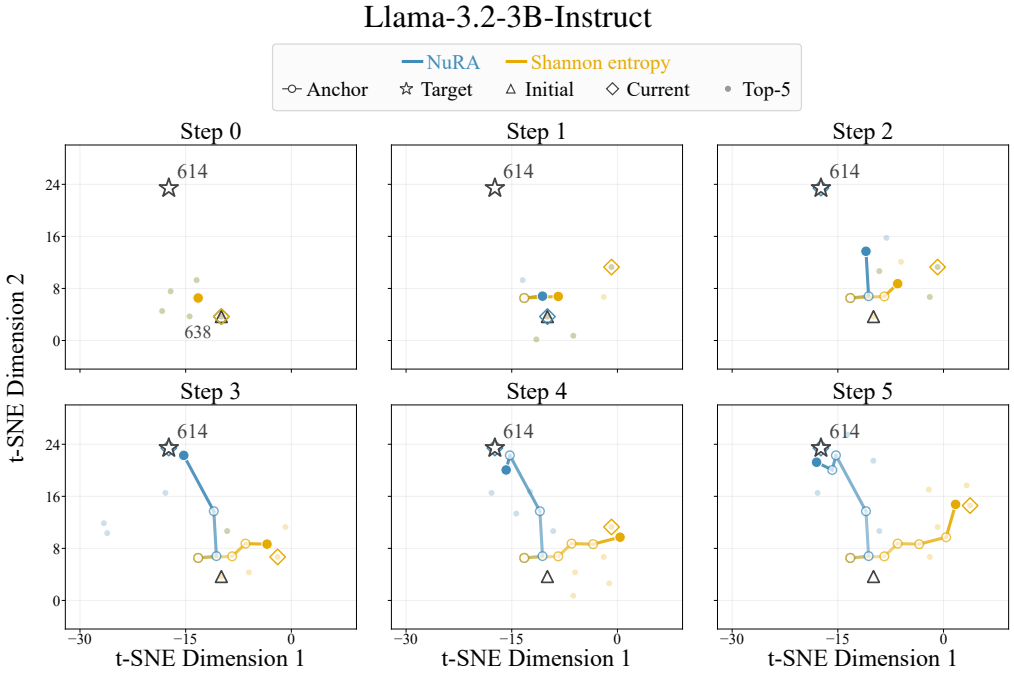


Addition-50
Model. Llama-3.2-1B-Instruct
Question. What is the sum of 8648302 and 1155236?
Reference answer. 9803538 **Token position.** 0 **Target token.** 980

NuRA
Predicted token. 964 → 969 → 980 → 980 → 980 → 980
p**(target).** 1.13e-04 → 0.0142 → 0.2714 → 0.6148 → 0.9096 → 0.9674
Target rank. 133 → 14 → 1 → 1 → 1 → 1

Shannon entropy
Predicted token. 964 → 880 → 880 → 880 → 880 → 880
p**(target).** 1.13e-04 → 0.0277 → 0.0024 → 2.24e-07 → 5.62e-04 → 1.13e-07
Target rank. 133 → 6 → 5 → 44 → 4 → 36

Figure 26. Entropy comparison

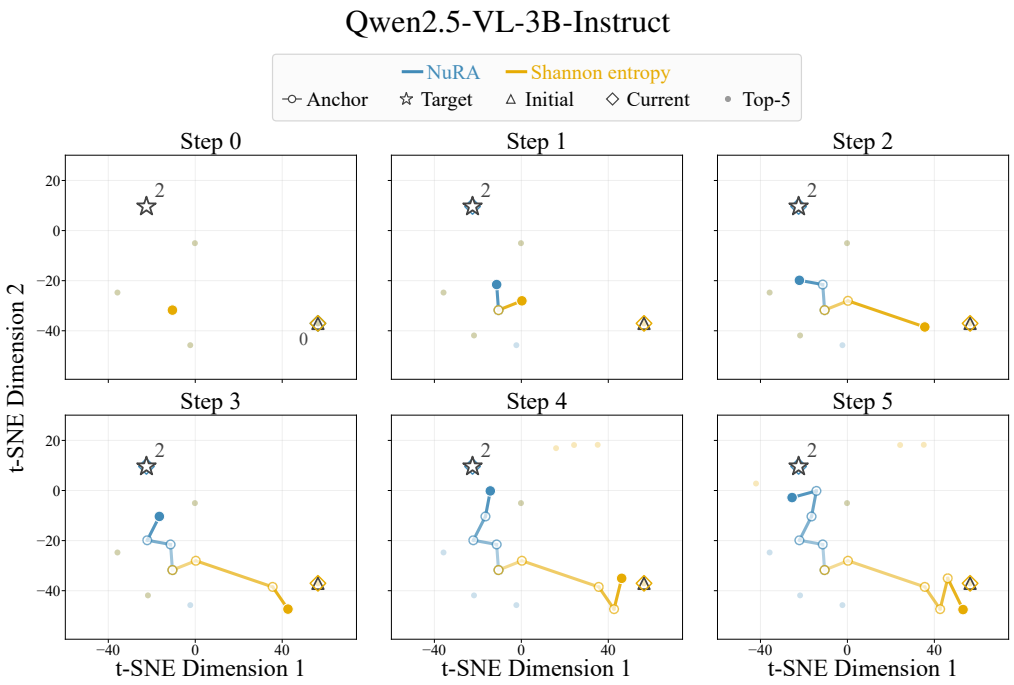


NUPA
Model. Llama-3.2-3B-Instruct
Question. Directly return the answer as an integer without any comma separator, like 123 . Add two numbers: 37861240 + 61444721436
Reference answer. 61482582676 **Token position.** 0 **Target token.** 614

NuRA
Predicted token. 638 → 638 → 614 → 614 → 614 → 614
p**(target).** 0.0024 → 4.93e-05 → 0.3305 → 0.9957 → 0.9781 → 1.0000
Target rank. 34 → 71 → 1 → 1 → 1 → 1

Shannon entropy
Predicted token. 638 → 629 → 629 → 628 → 629 → 129
p**(target).** 0.0024 → 8.61e-05 → 1.32e-04 → 6.19e-07 → 3.77e-12 → 5.32e-11
Target rank. 34 → 60 → 24 → 61 → 25694 → 127509

Figure 27. Entropy comparison



MathVista
Model. Qwen2.5-VL-3B-Instruct
Question. What is the age gap between these two people in image?
Reference answer. 28 **Token position.** 0 **Target token.** 2

NuRA
Predicted token. 0 → 2 → 2 → 2 → 2 → 2
p(**target**). 0.1727 → 0.3120 → 0.4486 → 0.6186 → 0.7526 → 0.8991
Target rank. 2 → 1 → 1 → 1 → 1 → 1

Shannon entropy
Predicted token. 0 → 0 → 0 → 0 → 0 → 0
p(**target**). 0.1727 → 0.1020 → 0.0024 → 3.53e-05 → 2.26e-06 → 9.78e-08
Target rank. 2 → 3 → 3 → 4 → 9 → 8

O.1 Numerical Posterior Redistribution

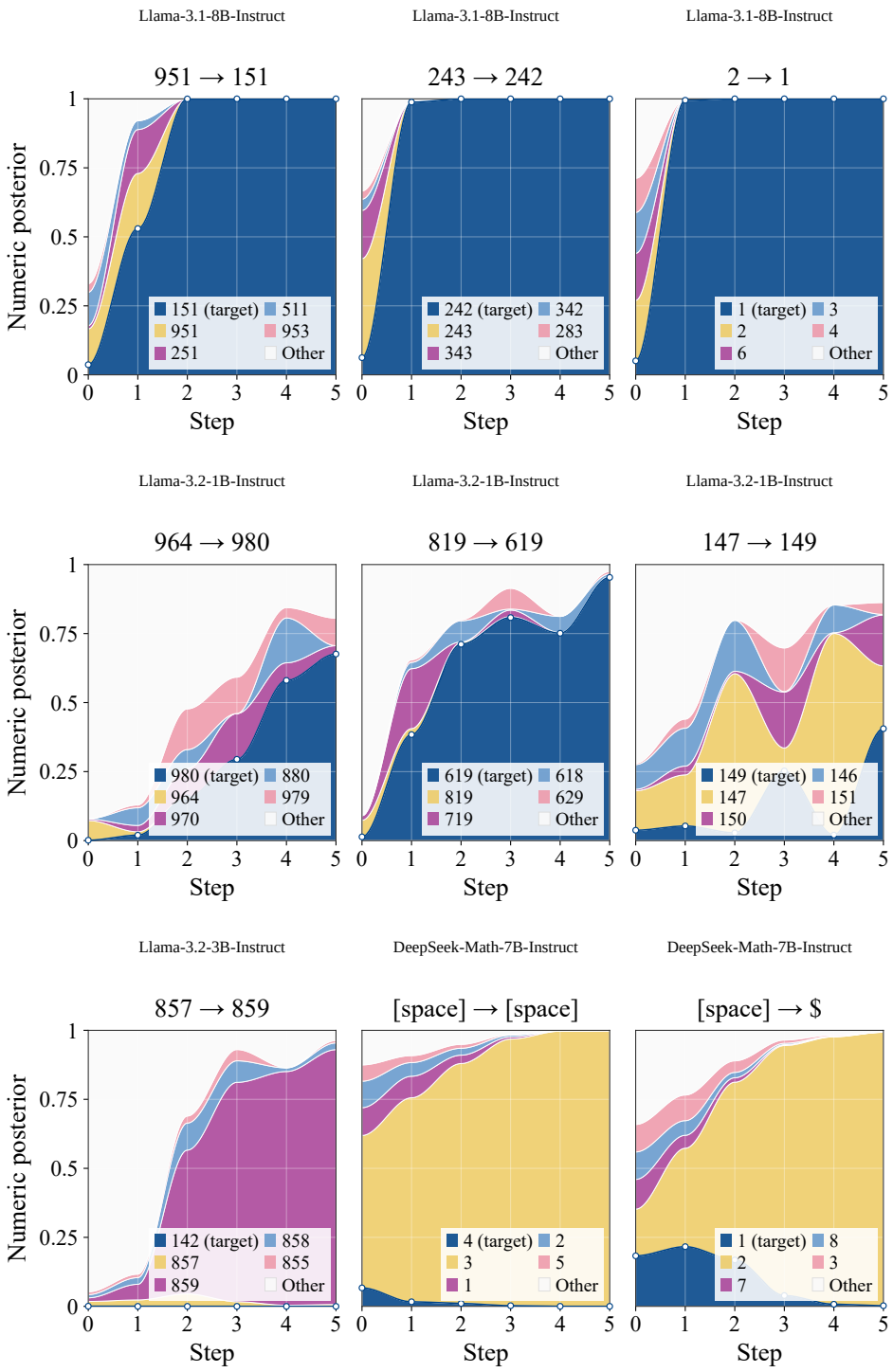


Figure 28: **Selected numerical posterior trajectories.** The first two rows show corrected predictions and the last row shows uncorrected predictions. Each panel names its model. Bands show the numerical posterior at $\gamma = 0.5$, with the reference token in blue and unshown numerical tokens in Other. Titles report initial and final full-vocabulary predictions; [space] denotes a whitespace token. Markers show measured updates and curves between them are interpolated. The main-text cases and the bottom-left case use fixed generated prefixes; the remaining cases use reference prefixes.

O.2 Target Probability and Rank During Adaptation

We visualize how the reference numerical token's probability and rank change during adaptation across four models. The cases show successful corrections, predictions that remain correct or incorrect, and errors introduced by adaptation.

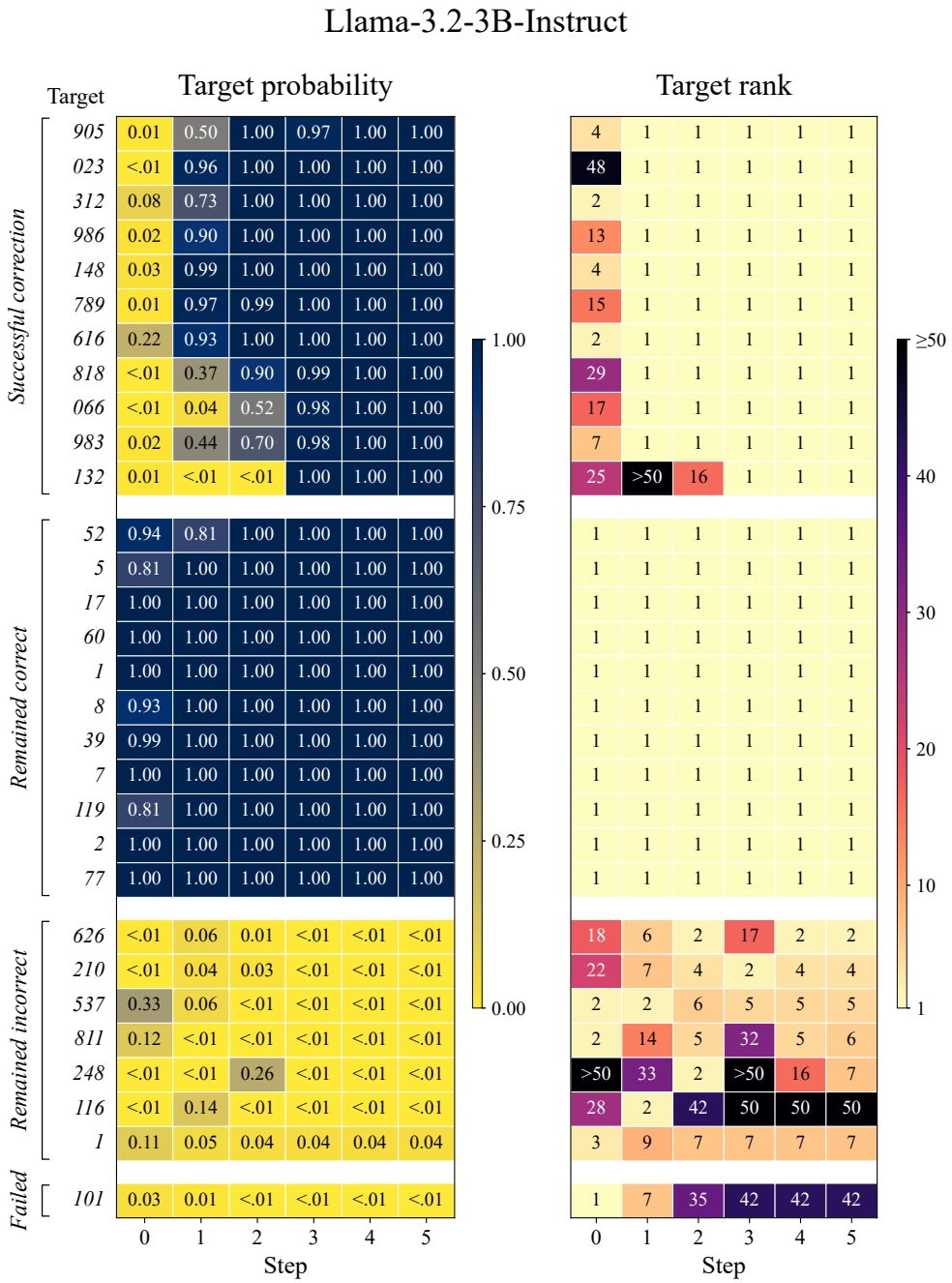


Figure 29: **Target probability and rank on Llama-3.2-3B-Instruct.** Rows are selected numerical positions, grouped by initial and final full-vocabulary correctness; Failed means correct-to-incorrect. This Addition-50 run retains its historical numerical-subset measurements. Ranks above 50 are clipped in color only. Selected cases do not estimate correction rates.

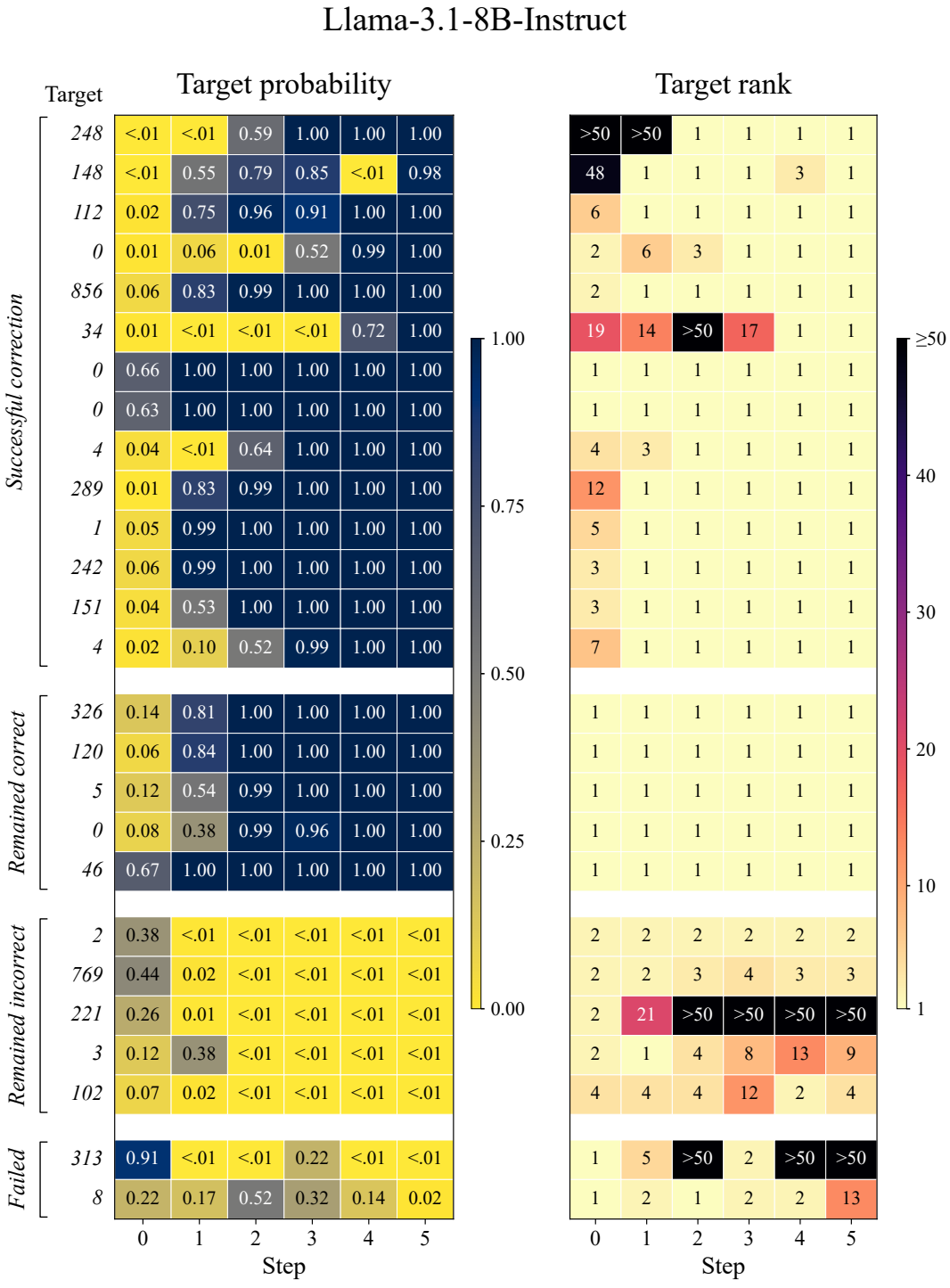


Figure 30: **Target probability and rank on Llama-3.1-8B-Instruct.** Here and in the following two figures, probabilities use $q = \mathrm{softmax}(0.5z)$ over numerical tokens, with ranks within that subset. Grouping and rank clipping follow the preceding figure; the probability protocols are interpreted separately.

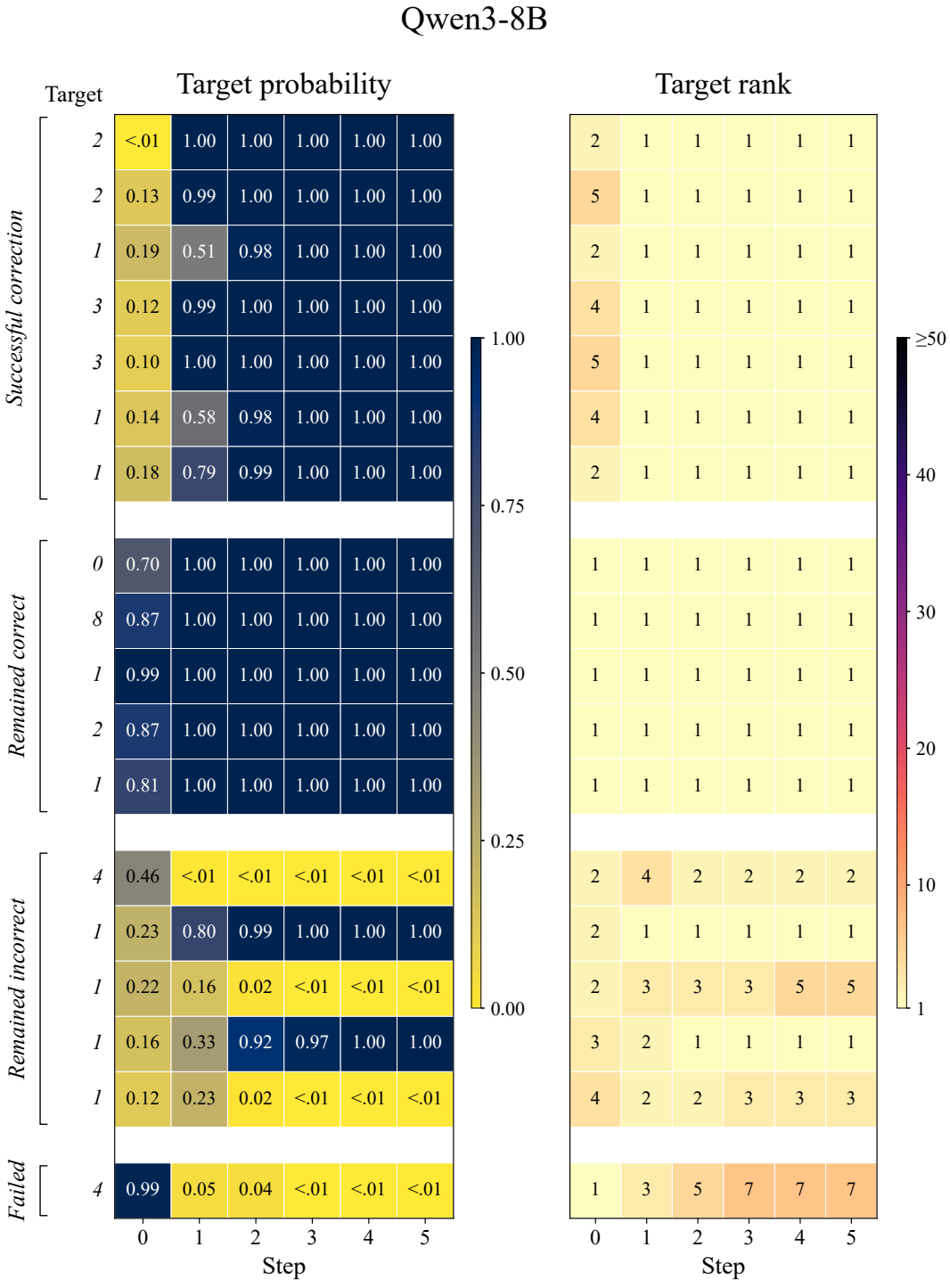


Figure 31: **Target probability and rank on Qwen3-8B.** Selected positions under the same numerical-subset protocol. Groups follow initial and final full-vocabulary correctness.

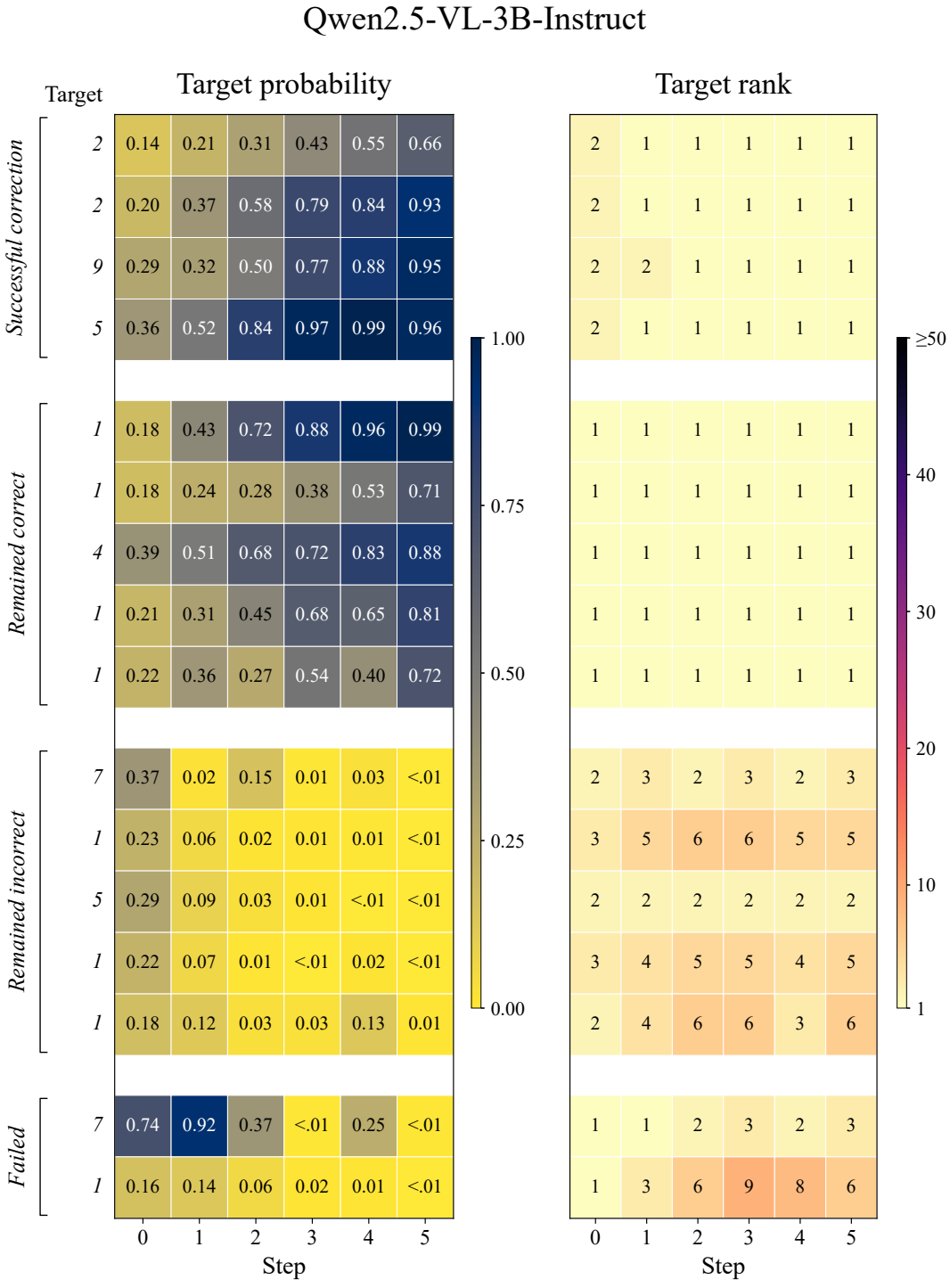


Figure 32: **Target probability and rank on Qwen2.5-VL-3B-Instruct.** Selected visual-task positions under the same numerical-subset protocol, grouped by initial and final full-vocabulary correctness.

P GENERATION AND ADAPTATION COSTS

Measurements. We measure generation on V100-SXM2 GPUs with FP16, SDPA, and batch size one, using ten fixed questions per dataset. Llama-3.2-3B-Instruct is used for Addition, MATH401, and GSM8K, and Qwen2.5-VL-3B-Instruct for ChartQA. Each configuration is timed three times on the same questions and workers. We first average measurements within each repeat, then report the mean and sample standard deviation of the three repeat means. The standard deviation describes timing variation, not variation across questions or random seeds. Selection and step costs are averaged over update events; total latency is averaged over questions and workers.

Avg. weights the four datasets equally and is omitted when any dataset is missing. For latency, we average across datasets within each timing repeat before computing the mean and sample standard deviation across repeats. Tables 27 and 28 report generation and adaptation costs.

Generation and timing. Direct inference uses greedy decoding with a KV cache. Program-aided inference generates Python code, executes it, and receives its output before producing the final answer. Chain-of-thought prompting uses a step-by-step prompt. Full MLP and Sparse use the same `NuRA` objective with different fractions of updated MLP parameters. Adaptation performs five updates at each triggered numerical position; sparse variants reuse the parameters selected at the first update of that position. Select measures parameter selection and update-state construction, excluding the first forward and backward passes. Step averages updates two through five. Total includes complete generation, all updates, selection, and weight restoration; program-aided timing includes code execution and result feedback. Its timer also includes prompt construction and tokenization, whereas the other methods use prepared inputs. Generated length counts all model-generated tokens, including code, intermediate reasoning, and the final answer.

Computational trade-off. Program-aided inference and chain-of-thought prompting generate more intermediate text, while adaptation preserves the short-answer format. Its latency is lower than these baselines on three of the four datasets and similar on Addition. The sparse implementation computes weight gradients for selected coordinates or rows after the initial selection, but still requires dense forward computation and backward propagation through hidden states. This helps explain why fewer updated parameters do not yield proportional latency savings. The timings characterize these implementations and prompts; accuracy is not matched across them.

Table 27: **Complete generation costs.** Total latency (mean ± std) and mean generated length (Len., tokens). Avg. uses all four datasets. Llama-3.2-3B-Instruct is used for Addition, MATH401, and GSM8K; Qwen2.5-VL-3B-Instruct is used for ChartQA.

Task	**Calculation**				**Reasoning**		**Visual**			
Method	**Addition**		**MATH401**		**GSM8K**		**ChartQA**		**Avg.**	
	Total (s)	Len.	Total (s)	Len.	Total (s)	Len.	Total (s)	Len.	Total (s)	Len.
Direct inference	0.138 ± 0.001	3.5	0.124 ± 0.003	3.1	0.106 ± 0.000	2.7	0.282 ± 0.000	3.5	0.162 ± 0.001	3.2
Program-aided	2.333 ± 0.007	60.5	2.638 ± 0.039	65.2	–	–	–	–	–	–
Chain-of-Thought	–	–	–	–	6.714 ± 0.043	169.6	8.220 ± 0.071	133.8	–	–
Full MLP (100%)	2.407 ± 0.031	3.5	1.403 ± 0.023	3.1	1.185 ± 0.004	2.7	4.227 ± 0.019	3.5	2.306 ± 0.017	3.2
Sparse (10%)	2.656 ± 0.033	3.5	1.531 ± 0.024	3.1	1.294 ± 0.006	2.7	4.532 ± 0.014	3.5	2.503 ± 0.015	3.2
Sparse (1%)	2.323 ± 0.034	3.5	1.451 ± 0.204	3.1	1.132 ± 0.009	2.7	4.237 ± 0.024	3.5	2.286 ± 0.061	3.2
Sparse (0.1%)	2.229 ± 0.031	3.5	1.301 ± 0.049	3.1	1.080 ± 0.009	2.7	4.101 ± 0.035	3.5	2.178 ± 0.025	3.2
Sparse (0.01%)	2.298 ± 0.135	3.5	1.344 ± 0.126	3.1	1.128 ± 0.082	2.7	4.131 ± 0.092	3.5	2.225 ± 0.108	3.2
Sparse (0.001%)	2.205 ± 0.078	3.5	1.371 ± 0.202	3.1	1.089 ± 0.050	2.7	4.100 ± 0.092	3.5	2.191 ± 0.105	3.2
Sparse (0.0001%)	2.107 ± 0.216	3.5	1.172 ± 0.054	3.1	1.022 ± 0.089	2.7	4.066 ± 0.227	3.5	2.092 ± 0.146	3.2

Table 28: **Adaptation costs.** Parameter selection and subsequent update latency (mean ± std). Avg. uses all four datasets. Llama-3.2-3B-Instruct is used for Addition, MATH401, and GSM8K; Qwen2.5-VL-3B-Instruct is used for ChartQA.

Task	**Calculation**				**Reasoning**		**Visual**			
Method	**Addition**		**MATH401**		**GSM8K**		**ChartQA**		**Avg.**	
	Select (ms)	Step (ms)	Select (ms)	Step (ms)	Select (ms)	Step (ms)	Select (ms)	Step (ms)	Select (ms)	Step (ms)
Full MLP (100%)	–	166 ± 2.4	–	163 ± 2.7	–	160 ± 0.6	–	299 ± 1.6	–	197 ± 1.5
Sparse (10%)	191 ± 0.2	153 ± 2.6	190 ± 0.2	151 ± 2.9	191 ± 0.2	148 ± 0.7	225 ± 1.2	287 ± 1.3	199 ± 0.3	185 ± 1.5
Sparse (1%)	88 ± 0.3	148 ± 2.6	164 ± 131.3	147 ± 2.9	88 ± 0.6	143 ± 1.0	113 ± 3.3	284 ± 1.5	113 ± 33.5	180 ± 1.5
Sparse (0.1%)	54 ± 0.1	147 ± 2.8	54 ± 0.1	148 ± 7.6	53 ± 0.2	141 ± 1.3	65 ± 0.2	281 ± 3.5	56 ± 0.0	179 ± 3.0
Sparse (0.01%)	54 ± 0.2	154 ± 13.1	54 ± 0.0	155 ± 21.6	95 ± 71.6	141 ± 0.7	65 ± 0.2	285 ± 10.0	67 ± 17.9	183 ± 11.0
Sparse (0.001%)	52 ± 0.2	145 ± 7.1	53 ± 0.2	161 ± 34.8	52 ± 0.2	144 ± 10.7	64 ± 0.1	282 ± 10.1	55 ± 0.1	183 ± 15.7
Sparse (0.0001%)	48 ± 0.0	139 ± 20.6	48 ± 0.1	129 ± 8.4	92 ± 76.3	122 ± 0.5	59 ± 0.2	280 ± 25.3	62 ± 19.1	167 ± 13.5

Q HYPERPARAMETER SENSITIVITY

Figure 33 reports supplementary sweeps on Llama-3.2-3B for Addition-50, MATH401, and GSM8K, and Qwen2.5-VL-3B for ChartQA. Numerical tokens are evaluated under reference answer prefixes, with answer accuracy computed from reconstructed answers. The Llama sweeps use normalized input embeddings and update all MLP and normalization parameters. They assess angular-dispersion adaptation under this configuration, separately from the output-head geometry and sparse updates of the main method. Here Top-k selects candidates from the full vocabulary; All uses the numerical vocabulary without a shortlist.

Smaller γ softens the candidate weights, while larger γ sharpens them. GSM8K favors softer weighting than Addition-50 and MATH401, but the smallest value reduces accuracy markedly. Since γ also affects gradient magnitude, this difference reflects both posterior weighting and optimization. The learning-rate curves likewise favor different values across tasks. Candidate-support changes have more modest effects, and additional steps beyond the first few provide limited improvement in these sweeps.

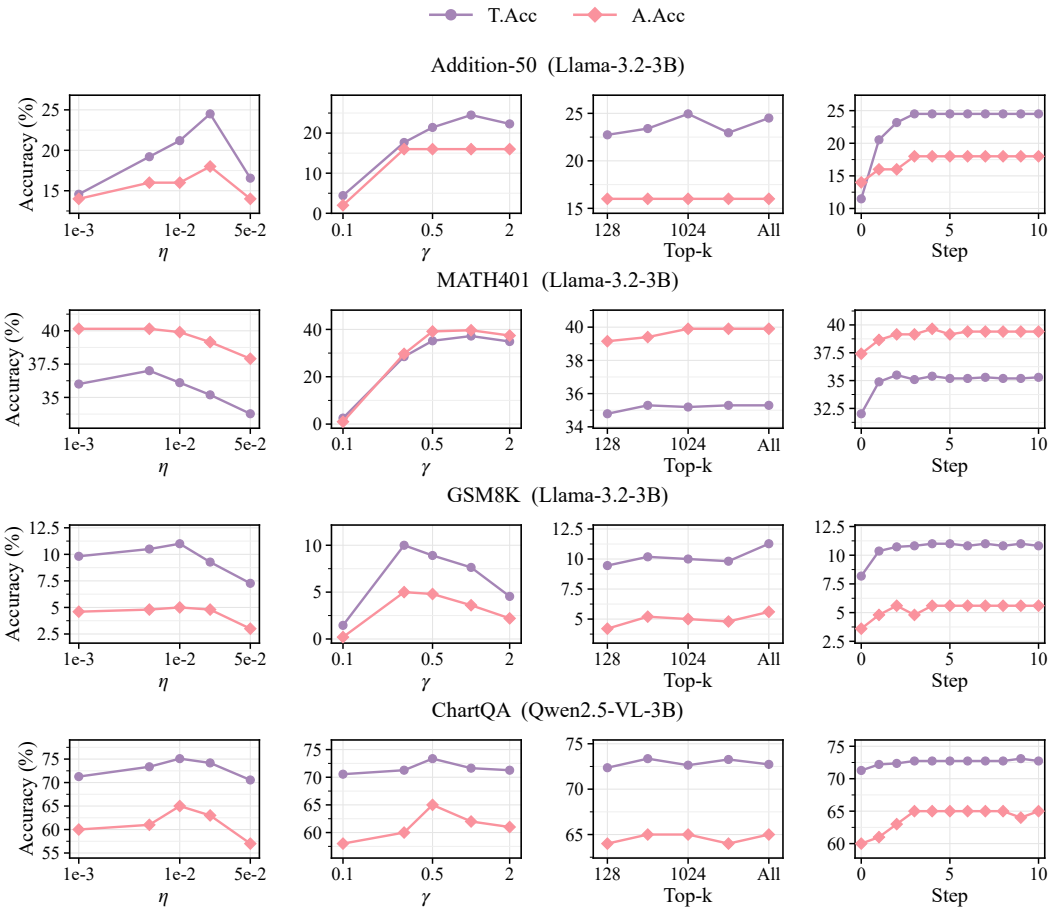


Figure 33: **Sensitivity of angular-dispersion adaptation.** Token and answer reconstruction accuracy (%) across learning rates, inverse temperatures, candidate supports, and update steps. Each row uses one model and dataset; vertical scales vary by panel.

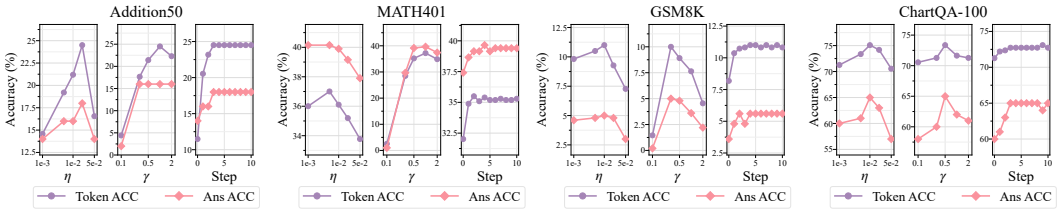


Figure 34: **Hyperparameter sensitivity.** Token and answer reconstruction accuracy across learning rates, inverse temperatures, and update steps. Appendix Q provides the experimental details, enlarged plots, and the Top-k sweep.