# Identifying and Transferring Reasoning-Critical Neurons: Improving LLM Inference Reliability via Activation Steering

Anonymous ACL submission


## Abstract

Despite the strong reasoning capabilities of recent large language models (LLMs), achieving reliable performance on challenging tasks often requires post-training or computationally expensive sampling strategies, limiting their practical efficiency. In this work, we first show that a small subset of neurons in LLMs exhibits strong predictive correlations with reasoning correctness. Based on this observation, we propose **AdaRAS** (**Ada**ptive **R**easoning **A**ctivation **S**teering), a lightweight test-time framework that improves reasoning reliability by selectively intervening on neuron activations. AdaRAS identifies Reasoning-Critical Neurons (RCNs) via a polarity-aware mean-difference criterion and adaptively steers their activations during inference, enhancing incorrect reasoning traces while avoiding degradation on already-correct cases. Experiments on 10 mathematics and coding benchmarks demonstrate consistent improvements, including over 13% gains on AIME-24 and AIME-25. Moreover, AdaRAS exhibits strong transferability across datasets and scalability to stronger models, outperforming post-training methods with only minimal additional computation cost[1].


## 1 Introduction

Recent large language models (LLMs) (Jaech et al., 2024; Yang et al., 2025a; DeepSeek-AI, 2025) have demonstrated strong capability across a wide range of natural language processing tasks. In particular, test-time scaling (Wei et al., 2022; Brown et al., 2024) has improved LLM performance by allocating additional computation at inference, enabling applications in mathematical problem solving (Guan et al., 2025; Muennighoff et al., 2025), code generation (Shi et al., 2024; Jain et al., 2025), and spatial reasoning (Fu et al., 2024). Despite these advances, reliable reasoning remains challenging even for state-of-the-art models, and most

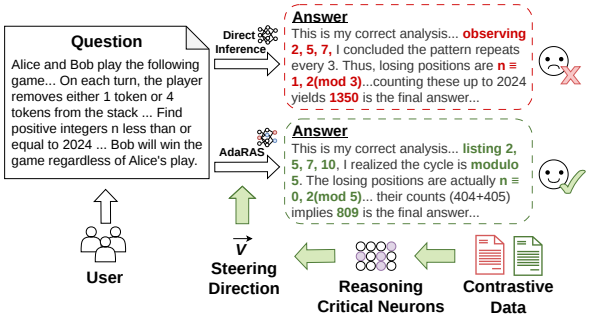



Figure 1: An example of AdaRAS activation steering correcting an erroneous reasoning trajectory.

improvements rely on post-training methods (Shao et al., 2024; Cui et al., 2025; Yu et al., 2025) or costly test-time strategies such as prompt engineering (Tian et al., 2024; Saha et al., 2025), self-consistency (Wang et al., 2023; Qiu et al., 2024), or multi-step calibration (Cobbe et al., 2021; Li et al., 2025; Snell et al., 2024). While effective, these approaches treat LLMs as black boxes, incur substantial inference overhead, and offer limited insight into the sources of reasoning errors.

Recent research works in activation engineering (Turner et al., 2023) provides an internal alternative, showing that selectively manipulating activations in attention heads (Li et al., 2023b) or MLP neurons (Rimsky et al., 2024) can control model behaviors, such as truthfulness (Marks and Tegmark, 2023), refusal (Lee et al., 2025), or reducing bias (Lu and Rimsky, 2024), without modifying model parameters or increasing inference cost. This suggests that internal activations encode functionally specialized signals that can be directly leveraged for controllable behavior. However, reasoning reliability is a fundamentally different problem: it is a trajectory-level property requiring temporally coherent, multi-step inference with long-range dependencies, rather than adjustment of a single output attribute. It therefore remains unclear whether existing activation engineering techniques

[1]Code and data will be publicly released after reviewing.

can identify and enhance the internal mechanisms underlying reasoning correctness. This raises a central question: *Can activation-level interventions improve LLM reasoning reliability?*

To address this question, we first examine whether reasoning correctness can be inferred via internal activations. Inspired by probing studies of LLMs (Gurnee et al., 2023), a pilot analysis reveals that a small subset of MLP neurons exhibits polarized activations between correct and incorrect reasoning trajectories, and that these activations are predictive of reasoning outcomes. Building by this observation, we propose AdaRAS, an adaptive activation steering framework for test-time reasoning enhancement. AdaRAS identifies RCNs by contrasting activations from correct and incorrect samples and applies polarity-based filtering to obtain a sparse, functionally consistent neuron set. During inference, AdaRAS selectively intervenes on these RCNs only when a trajectory is likely incorrect, enabling targeted correction without degrading already-correct one. Experiments on ten mathematic and coding benchmarks show that AdaRAS consistently improves accuracy across models, generalizes across tasks and datasets, and requires no additional training or sampling.

The contributions of this paper are threefold: 1) To our knowledge, we present the first systematic evidence that reasoning correctness can be predicted and improved through neuron interventions, establishing activation steering as a viable tool for enhancing LLM reasoning. 2) We introduce AdaRAS, a parameter-free, test-time activation steering framework that consistently enhances reasoning performance and transfers across tasks and datasets. 3) We provide mechanistic insights showing that AdaRAS stabilizes latent reasoning trajectories while preserving semantic representations, enabling plug-and-play deployment.

## 2 Preliminaries

This section presents preliminary results that specific neuron activations predict reasoning correctness, thereby motivating our proposed AdaRAS.

### 2.1 Task Formulation

We first define *Reasoning-Critical Neurons (RCNs)* as neurons who positively contribute to correct reasoning outcomes. Given a dataset $D = \{(x_i, y_i)\}$ and a reasoning language model $\mathcal{M}$, we denote by $r_i$ the model-generated reasoning trace for input $x_i$, and by $a_i$ the final answer extracted from $r_i$. We define a binary correctness label $c_i = \mathbb{I}[a_i = y_i]$, which serves as the target signal throughout this work. Architecturally, the model $\mathcal{M}$ consists of $L$ Transformer decoder layers. Following prior work (Geva et al., 2021; Meng et al., 2022), we focus on the MLP blocks, which are widely believed to encode high-level semantic patterns. Specifically, we take the intermediate activation for each MLP block as the internal representation.

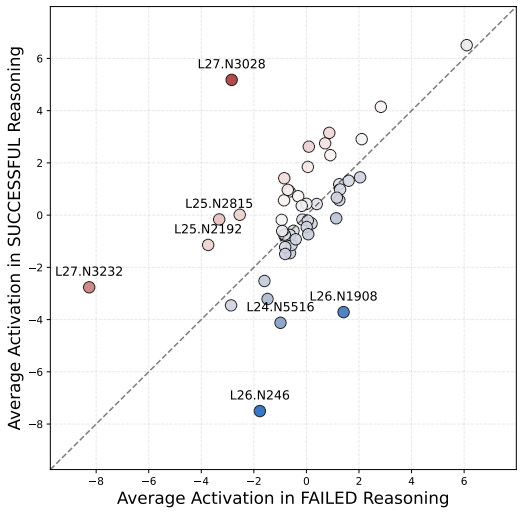



Figure 2: Comparison of activations of key neurons under successful and failed reasoning on AIME.

### 2.2 LLM Reasoning Signatures

Motivated by recent advances in LLM interpretability (Belinkov, 2022; Gurnee et al., 2023), we ask whether reasoning correctness can be inferred directly from intermediate neuron activations at test time. As a preliminary study, we probe last-token activations of Qwen3 series models (Yang et al., 2025a) on mathematic datasets (i.e., AIME and AMC-12) to assess their predictive power for reasoning reliability. In contrast to prior work on stylistic control or knowledge editing, we focus on identifying neurons that are specifically predictive of reasoning correctness. Detailed experimental setups are provided in the Appendix A.

**Finding 1. LLM reasoning traces leading to correct versus incorrect answers exhibit distinct activation patterns.** As shown in Figure 2, we compare the mean activations of highly contrastive neurons under successful and failed reasoning. Notably, the 3028-th neuron in layer 27 shows substantially higher activation during successful reasoning, whereas the 246-th and 1908-th neurons in layer 26

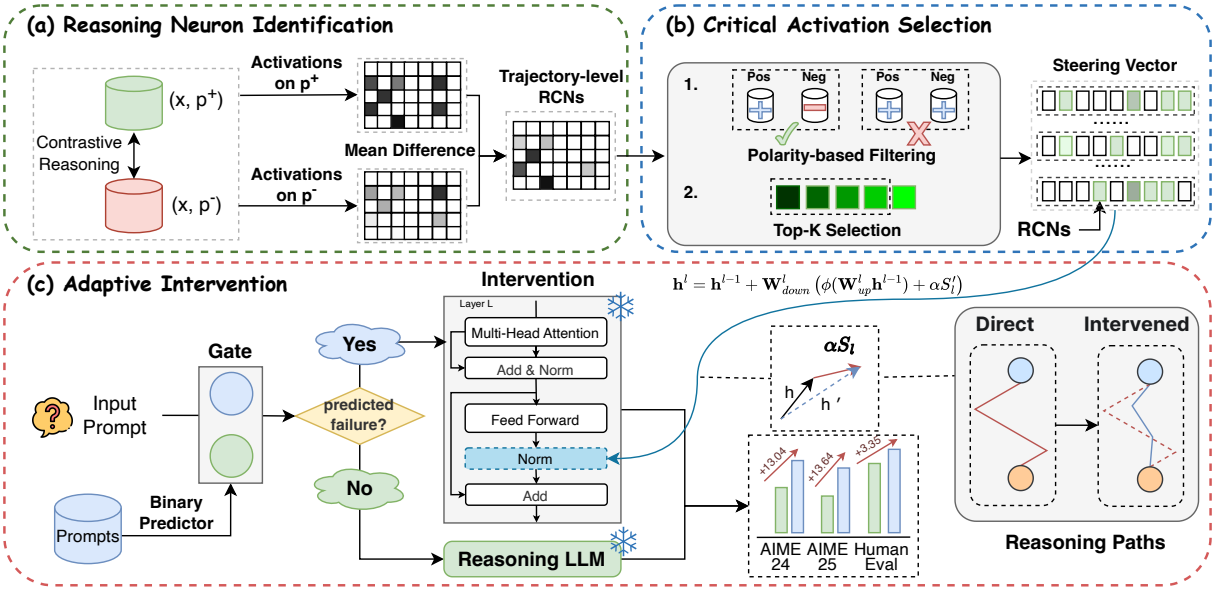



Figure 3: Overview of AdaRAS: (1) reasoning neuron identification (§3.1), which identifies critical neurons by measuring global activation differences between contrastive reasoning trajectories; (2) critical activation selection (§3.2), which further refines RCNs based on activation polarity variations; (3) adaptive intervention (§3.3), which enhances the reliability of steering by predicting reasoning failures and performing adaptive interventions.

Table 1: AUROC of probing classifiers trained on last-token activations for predicting reasoning correctness.

| Dataset | Model | AUROC |
|---|---|---|
| AIME (24+25) | Qwen3-1.7B | 0.7639 |
| | Qwen3-4B | 0.7153 |
| AMC-12 | Qwen3-1.7B | 0.7091 |
| | Qwen3-4B | 0.6727 |

exhibit the opposite trend. This indicates that specific neurons capture informative signals associated with reasoning success at test time.

**Finding 2. Token-level neuron activations are predictive of the final correctness of LLM reasoning.** As reported in Table 1, probing classifiers trained solely on last-token activations achieve AUROC scores around 0.7 across two datasets and two models, reaching up to 0.76 on AIME with the Qwen3-1.7B model. These observations suggest that neuron activation states are systematically correlated with reasoning outcomes.

## 3 Methodology

Figure 3 overviews our AdaRAS framework. First, we identify RCNs by contrasting activation patterns from correct and incorrect reasoning samples (§3.1). Then, we apply sparse activation steering to selectively intervene on these neurons (§3.2). To avoid perturbing already-correct reasoning, we introduce an adaptive intervention strategy that conditionally modulates activations at test time (§3.3). Finally, we describe the construction of parallel reasoning traces used in our experiments (§3.4).

### 3.1 Reasoning Neuron Identification

We identify reasoning-critical neurons (RCNs) by directly contrasting activation patterns between correct and incorrect reasoning trajectories. While probing-based methods can assign neuron importance via classifier weights, we find them unreliable in practice due to their sensitivity to feature selection and dependence on probe generalization. Following prior works (Rimsky et al., 2024), we therefore adopt a probing-free, activation-level criterion based on mean activation differences.

Let $a_i^l(t)$ denote the activation value of neuron $i$ at layer $l$ for token $t$. Given paired correct and incorrect reasoning traces $\mathcal{P} = \{(p_k^+, p_k^-)\}_{k=1}^N$, we define the reasoning-critical score of neuron $(l, i)$ as the difference in mean activations:

$$S(l,i) = \mathop{\mathbb{E}}_{k\sim[N]}\left[\mu(a_i^l, p_k^+) - \mu(a_i^l, p_k^-)\right], \quad \mu(a,p) \triangleq \frac{1}{|p|}\sum_{t\in p} a(t), \tag{1}$$

where $|p|$ denotes the length of the reasoning trace. Unlike prior works (Rimsky et al., 2024) which adopt token-level MD applied to the final answer,

this global formulation captures sustained activation patterns accumulated throughout the reasoning process. Specifically, for each layer $l$, the scores of all neurons form a layer-wise vector $\mathbf{S}_l = [S(l,1), \ldots, S(l,d)]^\top$, which serves as the steering direction for test-time intervention.

### 3.2 Critical Activation Selection

From preliminary results, our key observation is that correct reasoning is supported by structured activation patterns formed by a small subset of neurons, rather than uniformly distributed across entire layers. Therefore, instead of intervening at the layer level, we perform neuron-level selection and steering. Specifically, most modern LLMs, such as LLaMA and Qwen, employ SwiGLU activations (Shazeer, 2020), whose signed output is necessary for representations learning (Huang, 2024). Our empirical analysis also reveals that neurons whose average activation polarity differs between correct and incorrect reasoning trajectories are particularly discriminative (see 4.4 for detailed results). Intuitively, such polarity reversals indicate neurons that actively support or suppress valid reasoning, depending on the reasoning outcome.

Formally, for each neuron $(l,i)$, we retain it for steering only if

$$\mathop{\mathbb{E}}_{k}\Big[\mu(a_i^l, p_k^+)\Big] \cdot \mathop{\mathbb{E}}_{k}\Big[\mu(a_i^l, p_k^-)\Big] < 0, \qquad (2)$$

i.e., its mean activation exhibits a sign flip between correct and incorrect traces. This polarity-based filtering yields a sparse candidate set of RCNs. We then rank the retained neurons by the magnitude of their importance scores $|S(l,i)|$ and select the top-$K$ neurons to construct a sparse steering vector $\boldsymbol{S}_l'$ for each layer. The steering strength is controlled by a positive scalar $\alpha \in [0, \infty]$. Activation steering is applied within the MLP block of the LLM:

$$\mathbf{h}^l = \mathbf{h}^{l-1} + \mathbf{W}_{down}^l \left(\phi(\mathbf{W}_{up}^l \mathbf{h}^{l-1}) + \alpha \boldsymbol{S}_l'\right), \qquad (3)$$

where $\boldsymbol{S}_l'$ is constructed once from a reference dataset and applied uniformly at each decoding step during test-time inference.

### 3.3 Adaptive Intervention Strategy

Our early experiments shown a trade-off in activation steering: while it effectively rectifies incorrect reasoning, it inadvertently degrade performance on some originally correct samples. To address this, we introduce a lightweight failure prediction module that adaptively gates steering at test time. Specifically, we formulate a prediction task to estimate whether the base model can correctly answer a given input. The predictor operates on early neuron activations induced by the input prompt, which capture the model's initial reasoning state. To reduce dimensionality, we select a small set of activations using $F$-statistics method as the input features. A attention-based classifier is then trained on these features and achieves an AUROC of 0.8347 on AIME dataset, which further supports our insights that there exists a strong correlation between activation patterns and reasoning outcomes. During inference, the predictor serves as a gate: AdaRAS is applied only when a reasoning failure is predicted. This adaptive intervention improves overall reasoning reliability while preserving performance on inputs that are already correctly handled.

### 3.4 Contrastive Data Construction

As mentioned in Section 3.1, identifying RCNs requires contrastive reasoning traces that share the same input but differ in outcome correctness. To this end, we construct paired positive and negative reasoning trajectories via self-sampling. For each input prompt, we sample multiple reasoning trajectories using high-temperature decoding and partition them into positive and negative sets based on answer correctness. Contrastive pairs are then formed by matching traces with identical inputs but opposite outcomes. Aggregating all pairs yields a contrastive reasoning set $\mathcal{P} = \{(p^+, p^-)\}$, which is used to identify and transfer RCN activations.

## 4 Experiments

### 4.1 Experimental Setup

**Dataset.** We evaluate AdaRAS on 10 widely used mathematics and coding benchmarks that require coherent reasoning. These include standard competition-level benchmarks such as AIME-24/25 and AMC-12, as well as large-scale datasets such as GSM8K (Hendrycks et al., 2021) and MBPP+ (Liu et al., 2024). For datasets lacking an official training split, we partition the data into a small probing set and a remaining set reserved for test-time steering evaluation. For each probing sample, we generate 4 pairs of positive and negative reasoning traces to identify RCNs. Dataset statistics are summarized in Table 8.

**Baseline.** We compare AdaRAS with the following baselines: (1) **Chain-of-Thought (CoT)**. We

Table 2: Experimental results on mathematics and coding benchmarks. We compare AdaRAS with probing-based steering and post-training baselines, evaluated using the Accuracy metric. **Bold** numbers denote the best performance on each dataset. **Red subscripts** indicate the improvement of AdaRAS over the CoT prompting.

| **Task** | **Dataset** | | **Post-training** | | | **Steering** | |
|---|---|---|---|---|---|---|---|
| | | CoT | R1-Distill | Nemotron | OpenThinker | Probing | AdaRAS |
| Math | AIME-24 | 47.83 | 34.78 | 39.13 | 26.08 | 43.48 | **60.87** $_{+13.04}$ |
| | AIME-25 | 40.91 | 22.73 | 40.91 | 50.00 | 50.00 | **54.55** $_{+13.64}$ |
| | AIME-Extend | 47.33 | 21.33 | 47.33 | 42.67 | 48.00 | **52.67** $_{+5.34}$ |
| | MATH-500 | 84.80 | 66.20 | 85.40 | 82.00 | 85.40 | **86.40** $_{+1.60}$ |
| | GSM8K | 88.32 | 72.93 | 87.57 | 67.55 | 88.32 | **89.08** $_{+0.76}$ |
| | AMC-12 | 65.93 | 41.76 | 65.93 | 53.85 | **73.63** | 70.33 $_{+4.40}$ |
| Code | HumanEval | 77.18 | 45.64 | 57.72 | 59.06 | 77.85 | **79.19** $_{+2.01}$ |
| | HumanEval+ | 69.80 | 42.95 | 51.68 | 53.69 | 72.48 | **73.15** $_{+3.35}$ |
| | MBPP | 68.78 | 42.59 | 53.97 | 35.71 | 70.37 | **72.22** $_{+3.44}$ |
| | MBPP+ | 58.20 | 37.30 | 43.92 | 30.95 | 59.52 | **60.58** $_{+2.38}$ |

include the vanilla CoT prompting performance of Qwen3-1.7B as a baseline, on which AdaRAS is applied. (2) **Post-trained reasoning models**. Post-training methods (e.g., SFT, PPO, GRPO) are known to enhance model's reasoning performance. We therefore consider three publicly available post-trained models with comparable scales: DeepSeek-R1-Distill-Qwen-1.5B (DeepSeek-AI, 2025), OpenThinker-3-1.5B (Guha et al., 2025), and OpenReasoning-Nemotron-1.5B (Ahmad et al., 2025). (3) **Probing-based steering**. We use the weights of the probing classifier trained in §2.2 as neuron importance scores, select the same number of top-ranked neurons as in AdaRAS, and perform activation addition steering (Turner et al., 2023) accordingly. This baseline is denoted as *Probing*.

**Evaluation.** All methods are evaluated under the same greedy decoding setup for fair comparison; see Appendix C.2 for details. Following prior work, we report Accuracy as the metric across all datasets.

### 4.2 Main Results

Table 2 compares AdaRAS against the CoT and post-training methods across ten benchmarks.

**AdaRAS consistently improves reasoning correctness across all datasets, even surpassing post-trained LLMs.** On average, AdaRAS achieves an improvement of ∼5% over CoT-based inference. Notably, on the challenging AIME-25 benchmark, AdaRAS yields a substantial gain of 13.64%, surpassing all existing open-source post-trained models of comparable scale. We further observe that the performance gains introduced by AdaRAS are more pronounced on hard reasoning benchmarks, such as AIME and AMC. On these datasets, AdaRAS achieves an average improvement of 9.11%, compared to a more modest gain of 2.26% on relatively easier benchmarks. These results suggest that steering RCN activations toward favorable states can substantially enhance the reliability of the generation process, particularly for complex and cognitively demanding reasoning tasks.

**Probing-based activation steering is inherently unstable.** As shown in Table 2, vanilla probing-based steering yields mixed results across datasets. While it achieves notable gains on AMC-12, even surpassing AdaRAS, it fails to generalize and can be detrimental on more challenging benchmarks. In particular, on AIME-24, probing-based steering degrades accuracy by 4.35% compared to the unsteered CoT baseline. These results indicate that naive probing-based approaches are highly sensitive to neuron selection, and may harm reasoning performance without adaptive intervention mechanisms, which are explicitly addressed in AdaRAS.

### 4.3 Generalizability

Tables 3 and 4 show the transferability and generalization of AdaRAS, covering cross-dataset setting and larger model scale.

**RCNs exhibit strong cross-dataset and cross-task transferability.** As shown in Table 3, com-

Table 3: Transferability results of AdaRAS across datasets. We compare in-domain steering using RCNs identified on each dataset (*ID. RCNs*) with cross-dataset steering using RCNs identified on AIME (*AIME-RCNs*).

| Dataset | ID. RCNs | AIME-RCNs |
|---|---|---|
| MATH-500 | 86.40 | **87.40** |
| GSM8K | 89.08 | **89.39** |
| AMC-12 | 70.33 | **71.43** |
| HumanEval | 79.19 | **80.54** |
| HumanEval+ | 73.15 | 73.15 |
| MBPP | 72.22 | **72.75** |
| MBPP+ | 60.58 | **61.11** |

Table 4: Scalability results of AdaRAS on Qwen3-4B. We evaluate AdaRAS on a larger base model using cross-dataset steering with RCNs identified on AIME.

| Dataset | Qwen3-4B | +AdaRAS |
|---|---|---|
| AIME-24 | 56.52 | **60.87** |
| AIME-25 | 59.09 | **72.73** |
| AIME-Extend | 68.67 | **76.67** |
| MATH-500 | 90.00 | **91.20** |
| GSM8K | 92.72 | **93.93** |
| AMC-12 | 76.92 | **80.22** |
| HumanEval | 91.28 | **92.62** |
| HumanEval+ | 81.88 | **84.56** |
| MBPP | 79.37 | **82.28** |
| MBPP+ | 68.78 | **69.31** |

pared to using dataset-specific RCNs, we observe that cross-dataset activation steering with RCNs identified on AIME consistently yields better performance, resulting in small but stable gains, within 1%, across all evaluated datasets. Notably, these RCNs also transfer across task domains: when applied to coding benchmarks, they continue to improve performance. These results support our motivation for identifying RCNs and suggest that such neurons capture task-agnostic reasoning mechanisms. In particular, RCNs identified from more challenging tasks appear to generalize broadly to diverse reasoning settings.

**AdaRAS scales effectively to stronger reasoning models.** Table 4 shows that AdaRAS continues to yield performance improvements when applied to the larger and more capable reasoning model. Even when the base model already achieves high accuracy, whereas Qwen3-4B surpasses 90% accuracy on relatively easier benchmarks such as MATH-500, GSM8K, and HumanEval, AdaRAS still delivers consistent accuracy gains of around 1%. Notably, on more challenging datasets such as AIME, the improvements are more pronounced, mirroring the trends observed in previous experiments. Overall, these results indicate that AdaRAS acts as a complementary test-time intervention, providing consistent gains even as the base model's reasoning capability improves.

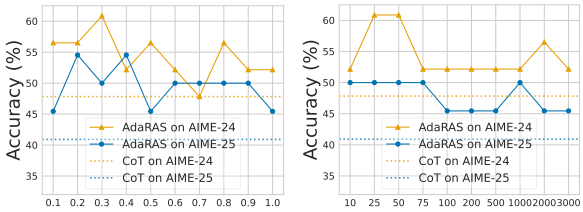
Moreover, additional results in the Appendix E show that AdaRAS remains effective against stronger test-time baselines, scales to larger LLMs, exhibits stability across prompt variants, and incurs minimal inference and offline training overhead.

Table 5: Ablation results of key components in AdaRAS. *Random Steering* applies steering to randomly sampled neurons. *AdaRAS w/o MD* reduces the method to probing-based neuron importance estimation, instead of Mean Difference designed in §3.1. *AS* and *AI* denote Activation Selection and Adaptive Intervention, corresponding to the designs in §3.2 and §3.3, respectively.

| Method | AIME-24 | AIME-25 |
|---|---|---|
| Qwen3-1.7B | 47.83 | 40.91 |
| Random Steering | 34.78 | 27.27 |
| AdaRAS w/o MD | 43.48 | 50.00 |
| AdaRAS w/o AS | 52.17 | 45.45 |
| AdaRAS w/o AI | 56.52 | 45.45 |
| AdaRAS | **60.87** | **54.55** |

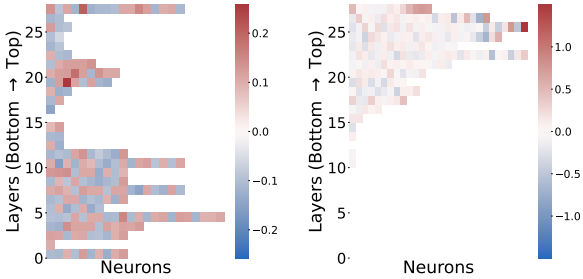
### 4.4 Ablation Studies

Table 5 reports the ablation results for AdaRAS. (1) Random Steering consistently degrades performance, indicating that indiscriminate neuron intervention is harmful. (2) Removing the MD-based estimation results in the sharpest decline on AIME-24 (i.e., 60.87% $\rightarrow$ 43.48%), dropping even below the unsteered baseline. This underscores the necessity of MD for accurately identifying RCNs. (3) Disabling polarity-based activation selection causes an approximate 10% accuracy drop on both AIME-24 and AIME-25, highlighting the importance of discriminative activation polarity. (4) Omitting adaptive intervention degrades overall performance by interfering with originally correct reasoning trajectories. Overall, all components are essential to the effectiveness of AdaRAS; limited by space, further analysis is in Appendix E.5.




(a) Intervention strength $\alpha$ (b) Top-$K$ neurons

Figure 4: Effect of hyperparameter.




(a) Probing-based steering (b) AdaRAS

Figure 5: Visualization of activation shifts induced by steering. Neurons are sorted by their indices in layer, and unsteered neurons are omitted for clarity.

# 5 Analysis

## 5.1 Effect of Hyperparameter

In this section, we vary the intervention strength $\alpha \in [0, 1]$ in Eq.3 to control the activation shift magnitude. We further vary the number of selected neurons to assess the impact of RCN sparsity.

**Effect of intervention strength.** As shown in Figure 4a, AdaRAS achieves peak performance at $\alpha = 0.3$ on AIME-24 and $\alpha = 0.4$ on AIME-25. Increasing $\alpha$ beyond these values leads to performance decline, suggesting that excessive intervention interferes with the model's reasoning process. Notably, performance drops to the unsteered baseline only when $\alpha = 0.7$ on AIME-24. This suggests that AdaRAS is relatively robust to suboptimal hyperparameter choices and does not significantly harm the original reasoning capability under most settings.

**Effect of top-$K$ RCNs selection.** In the ablation study, we have shown that removing AdaRAS's polarity-aware filtering of RCNs leads to degraded steering performance. Figure 4b further reveals the sparse nature of RCNs. Specifically, steering performance improves as more RCNs are intervened on, peaks at approximately top-50 neurons (about 0.03% of all neurons), and degrades thereafter. In particular, when intervening on top-2000 neurons (about 1.20%), performance on AIME-24 drops by around 6% compared to performance at top-50. These results provide strong empirical evidence supporting our hypothesis in §3.2 that RCNs are sparse and highly selective, and that indiscriminate intervention on a large set of neurons can be detrimental to reasoning performance.

## 5.2 Visualization of Steering

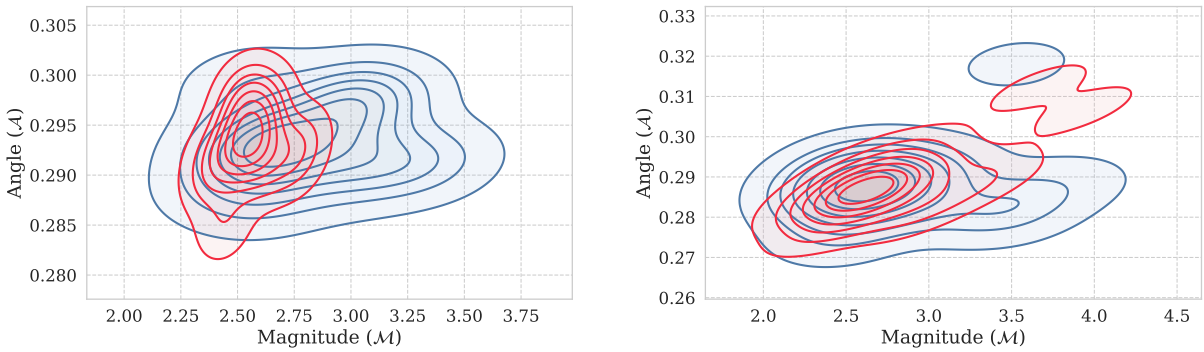
We visualize the effects of AdaRAS from neuron-level activation shifts to trajectory-level dynamics.

**AdaRAS mainly intervenes on later layer neurons.** Figure 5 compares activation shifts of RCNs induced by vanilla probing-based steering and AdaRAS on AIME-24 and AIME-25, where shifts are computed as the difference between mean neuron activations before and after steering. Compared to probing-based methods, which introduce widespread changes in early layers, AdaRAS concentrates its interventions on later layers and induces substantially smaller activation shifts, with most values tightly centered around zero. This observation aligns with prior findings (Biran et al., 2024) that later layer neurons are more important for multi-hop reasoning.

**AdaRAS stabilizes latent reasoning trajectories without altering semantic modeling.** To further analyze the effect of AdaRAS on the reasoning process, we examine the stability of latent reasoning paths. Following Wang et al. (2025b), we compute two trajectory-level features from layer-wise activations: **magnitude** and **angle**, which capture reasoning path curvature and semantic modeling level, respectively. Formal definitions are provided in the Appendix D due to space limitations. As shown in Figure 6, under both in-domain and cross-dataset steering, steered samples exhibit significantly lower and more concentrated magnitude values than unsteered samples, indicating that AdaRAS reduces fluctuations in latent reasoning trajectories. In contrast, the angle metric remains largely unchanged, suggesting that AdaRAS stabilizes reasoning dynamics without substantially altering semantic modeling behavior.

# 6 Related Works

**Reasoning Reliability.** Improving reasoning reliability in LLMs has been widely studied via test-time scaling. Prior work archives performance gains via post-training methods (Ouyang et al.,




(a) AIME (In-domain steering)

(b) MBPP (Cross-task steering)

Figure 6: Feature distribution discrepancy of activations before and after intervention. The x-axis denotes the magnitude feature, and the y-axis denotes the angle feature of activations. Larger values indicate greater deviations in activation space. Detailed definitions are provided in Appendix D.

2022; Lightman et al., 2024; Shao et al., 2024) or costly test-time techniques, like prompt engineering (Wei et al., 2022; Tian et al., 2024), self-consistency (Wang et al., 2023; Saunders et al., 2022), and multi-step calibration (Li et al., 2025; Snell et al., 2024). Other approaches focus on the token or latent level, including redundancy pruning (Xia et al., 2025), optimization of high-entropy tokens (Wang et al., 2025a; Qian et al., 2025), and latent-space reasoning (Chen et al., 2025b; Wang et al., 2025b). Recent studies have also explored reasoning mechanisms from internal activations. Yang et al. (2025b) improves reasoning via token-level entropy guidance during inference. Tang et al. (2025) identifies reasoning neurons from activation differences for steering, but achieves limited performance gains. Concurrent work (Galichin et al., 2026) leverages Sparse Autoencoders (SAEs) and reasoning-token heuristics to identify interpretable reasoning features for activation steering. In contrast, AdaRAS directly identifies RCNs without auxiliary modules or token heuristics, and introduces neuron filtering and adaptive intervention to achieve more stable and effective steering.

**Mechanistic Interpretability.** Recent studies have advanced the understanding of transformer-based language models through techniques such as vocabulary projection (Belrose et al., 2023), probing classifiers (Belinkov, 2022; Gurnee et al., 2023), and sparse autoencoders (Bricken et al., 2023; Xin et al., 2025). These approaches show that specific neurons could encode semantical features, including truthfulness (Burns et al., 2023; Marks and Tegmark, 2023), sentiment (Tigges et al., 2023), refusal (Arditi et al., 2024), and stylistic attributes (Li et al., 2023a; Stolfo et al., 2024; Zou et al., 2023; Rimsky et al., 2024). Building on these insights, a growing body of work explores modifying activations during inference to steer model behavior, achieving regulation of personality traits (Chen et al., 2025a), improved instruction following (Stolfo et al., 2025; Liu et al., 2024), and editing of factual knowledge (Subramani et al., 2022; Hernandez et al., 2023). While existing steering methods focus on output-level behaviors, we target trajectory-level reasoning using intervention signals derived from contrastive reasoning traces. Crucially, our approach identifies these signals offline, applying them at inference time without task-specific retraining.

## 7 Conclusion

In this work, we propose AdaRAS, an inference-time activation steering framework for improving reasoning reliability in LLMs. We first demonstrate that reasoning correctness is strongly associated with a small subset of neurons. AdaRAS identifies these reasoning-critical neurons via mean activation differences between correct and incorrect reasoning trajectories, and further refines them with polarity-based activation filtering. AdaRAS then applies an adaptive intervention to selectively steer neuron activations during inference. Extensive experiments across ten reasoning benchmarks show that AdaRAS consistently improves accuracy with only minimal additional computation overhead. Finally, trajectory-level analyses using latent reasoning features provide mechanistic evidence that AdaRAS stabilizes reasoning paths without affecting semantic modeling level of LLMs.

## Limitations

Although effective, our work has several limitations. First, our analysis focuses on the Qwen3 series models and STEM benchmarks. Extending the evaluation to a broader range of model architectures and complex reasoning settings (e.g., spatial or multi-hop reasoning) is necessary to assess the generalizability of our findings for future work. Second, as with many interpretability studies, our approach relies on constructing contrastive data pairs. This requirement may limit applicability to models at capability extremes, where obtaining paired correct and incorrect reasoning trajectories is non-trivial. Exploring alternative data construction strategies that relax this requirement is an important direction for future work. Third, to identify RCNs, we focus on neuron-level activations with the mean difference metric. Future work could incorporate other advanced techniques, such as sparse autoencoders or vocabulary projection analyses, to further elucidate the mechanistic underpinnings of RCNs.

## Ethics Statement

This work investigates inference-time activation steering methods for improving the reasoning reliability of large language models. All experiments are conducted using publicly available models and open-source benchmarks. The study does not involve the collection of personal data, human subjects, or real-world deployment, etc.

## References


Wasi Uddin Ahmad, Sean Narenthiran, Somshubra Majumdar, Aleksander Ficek, Siddhartha Jain, Jocelyn Huang, Vahid Noroozi, and Boris Ginsburg. 2025. Opencodereasoning: Advancing data distillation for competitive coding. *CoRR*, abs/2504.01943.

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. 2024. Refusal in language models is mediated by a single direction. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and 1 others. 2021. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*.

Yonatan Belinkov. 2022. Probing classifiers: Promises, shortcomings, and advances. *Comput. Linguistics*, 48(1):207–219.

Nora Belrose, Zach Furman, Logan Smith, Danny Halawi, Igor Ostrovsky, Lev McKinney, Stella Biderman, and Jacob Steinhardt. 2023. Eliciting latent predictions from transformers with the tuned lens. *CoRR*, abs/2303.08112.

Eden Biran, Daniela Gottesman, Sohee Yang, Mor Geva, and Amir Globerson. 2024. Hopping too late: Exploring the limitations of large language models on multi-hop queries. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 14113–14130. Association for Computational Linguistics.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, and 6 others. 2023. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*. Https://transformer-circuits.pub/2023/monosemantic-features/index.html.

Bradley C. A. Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V. Le, Christopher Ré, and Azalia Mirhoseini. 2024. Large language monkeys: Scaling inference compute with repeated sampling. *CoRR*, abs/2407.21787.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. 2023. Discovering latent knowledge in language models without supervision. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, and 39 others. 2021. Evaluating large language models trained on code. *Preprint*, arXiv:2107.03374.

Runjin Chen, Andy Arditi, Henry Sleight, Owain Evans, and Jack Lindsey. 2025a. Persona vectors: Monitoring and controlling character traits in language models. *CoRR*, abs/2507.21509.

Xinghao Chen, Anhao Zhao, Heming Xia, Xuan Lu, Hanlin Wang, Yanjun Chen, Wei Zhang, Jian Wang, Wenjie Li, and Xiaoyu Shen. 2025b. Reasoning beyond language: A comprehensive survey on latent chain-of-thought reasoning. *CoRR*, abs/2505.16782.

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. 2021. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168.

Ganqu Cui, Lifan Yuan, Zefan Wang, Hanbin Wang, Wendi Li, Bingxiang He, Yuchen Fan, Tianyu Yu, Qixin Xu, Weize Chen, Jiarui Yuan, Huayu Chen, Kaiyan Zhang, Xingtai Lv, Shuo Wang, Yuan Yao, Xu Han, Hao Peng, Yu Cheng, and 4 others. 2025. Process reinforcement through implicit rewards. *CoRR*, abs/2502.01456.

DeepSeek-AI. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *CoRR*, abs/2501.12948.

Rao Fu, Jingyu Liu, Xilun Chen, Yixin Nie, and Wenhan Xiong. 2024. Scene-llm: Extending language model for 3d visual understanding and reasoning. *CoRR*, abs/2403.11401.

Andrey V. Galichin, Alexey Dontsov, Polina Druzhinina, Anton Razzhigaev, Oleg Rogov, Elena Tutubalina, and Ivan V. Oseledets. 2026. I have covered all the bases here: Interpreting reasoning features in large language models via sparse autoencoders. In *Fortieth AAAI Conference on Artificial Intelligence, Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence, Sixteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2026, Singapore, January 20-27, 2026*, pages 30771–30779. AAAI Press.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 5484–5495. Association for Computational Linguistics.

Xinyu Guan, Li Lyna Zhang, Yifei Liu, Ning Shang, Youran Sun, Yi Zhu, Fan Yang, and Mao Yang. 2025. rstar-math: Small llms can master math reasoning with self-evolved deep thinking. In *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*. OpenReview.net.

Etash Kumar Guha, Ryan Marten, Sedrick Keh, Negin Raoof, Georgios Smyrnis, Hritik Bansal, Marianna Nezhurina, Jean Mercat, Trung Vu, Zayne Sprague, Ashima Suvarna, Benjamin Feuer, Liangyu Chen, Zaid Khan, Eric Frankel, Sachin Grover, Caroline Choi, Niklas Muennighoff, Shiye Su, and 31 others. 2025. Openthoughts: Data recipes for reasoning models. *CoRR*, abs/2506.04178.

Wes Gurnee, Neel Nanda, Matthew Pauly, Katherine Harvey, Dmitrii Troitskii, and Dimitris Bertsimas. 2023. Finding neurons in a haystack: Case studies with sparse probing. *Trans. Mach. Learn. Res.*, 2023.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Evan Hernandez, Belinda Z Li, and Jacob Andreas. 2023. Inspecting and editing knowledge representations in language models. *arXiv preprint arXiv:2304.00740*.

Allen Hao Huang. 2024. Expanded gating ranges improve activation functions. *CoRR*, abs/2405.20768.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, Alex Iftimie, Alex Karpenko, Alex Tachard Passos, Alexander Neitz, Alexander Prokofiev, Alexander Wei, Allison Tam, Ally Bennett, Ananya Kumar, and 80 others. 2024. Openai o1 system card. *CoRR*, abs/2412.16720.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. 2025. Livecodebench: Holistic and contamination free evaluation of large language models for code. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net.

Bruce W. Lee, Inkit Padhi, Karthikeyan Natesan Ramamurthy, Erik Miehling, Pierre L. Dognin, Manish Nagireddy, and Amit Dhurandhar. 2025. Programming refusal with conditional activation steering. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net.

Chengpeng Li, Mingfeng Xue, Zhenru Zhang, Jiaxi Yang, Beichen Zhang, Xiang Wang, Bowen Yu, Binyuan Hui, Junyang Lin, and Dayiheng Liu. 2025. START: self-taught reasoner with tools. *CoRR*, abs/2503.04625.

Kenneth Li, Aspen K. Hopkins, David Bau, Fernanda B. Viégas, Hanspeter Pfister, and Martin Wattenberg. 2023a. Emergent world representations: Exploring a sequence model trained on a synthetic task. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Kenneth Li, Oam Patel, Fernanda B. Viégas, Hanspeter Pfister, and Martin Wattenberg. 2023b. Inference-time intervention: Eliciting truthful answers from a language model. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe.

2024. Let's verify step by step. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. 2023. Is your code generated by chatGPT really correct? rigorous evaluation of large language models for code generation. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Sheng Liu, Haotian Ye, Lei Xing, and James Y. Zou. 2024. In-context vectors: Making in context learning more effective and controllable through latent space steering. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Dawn Lu and Nina Rimsky. 2024. Investigating bias representations in llama 2 chat via activation steering. *CoRR*, abs/2402.00402.

Samuel Marks and Max Tegmark. 2023. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. *CoRR*, abs/2310.06824.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel J. Candès, and Tatsunori Hashimoto. 2025. s1: Simple test-time scaling. *CoRR*, abs/2501.19393.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Chen Qian, Dongrui Liu, Haochen Wen, Zhen Bai, Yong Liu, and Jing Shao. 2025. Demystifying reasoning dynamics with mutual information: Thinking tokens are information peaks in LLM reasoning. *CoRR*, abs/2506.02867.

Jiahao Qiu, Yifu Lu, Yifan Zeng, Jiacheng Guo, Jiayi Geng, Huazheng Wang, Kaixuan Huang, Yue Wu, and Mengdi Wang. 2024. Treebon: Enhancing inference-time alignment with speculative tree-search and best-of-n sampling. *CoRR*, abs/2410.16033.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Matt Turner. 2024. Steering llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 15504–15522. Association for Computational Linguistics.

Swarnadeep Saha, Xian Li, Marjan Ghazvininejad, Jason E. Weston, and Tianlu Wang. 2025. Learning to plan & reason for evaluation with thinking-llm-as-a-judge. In *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*. OpenReview.net.

William Saunders, Catherine Yeh, Jeff Wu, Steven Bills, Long Ouyang, Jonathan Ward, and Jan Leike. 2022. Self-critiquing models for assisting human evaluators. *CoRR*, abs/2206.05802.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *CoRR*, abs/2402.03300.

Noam Shazeer. 2020. GLU variants improve transformer. *CoRR*, abs/2002.05202.

Quan Shi, Michael Tang, Karthik Narasimhan, and Shunyu Yao. 2024. Can language models solve olympiad programming? *CoRR*, abs/2404.10952.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. 2024. Scaling LLM test-time compute optimally can be more effective than scaling model parameters. *CoRR*, abs/2408.03314.

Alessandro Stolfo, Vidhisha Balachandran, Safoora Yousefi, Eric Horvitz, and Besmira Nushi. 2025. Improving instruction-following in language models through activation steering. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net.

Alessandro Stolfo, Ben Wu, Wes Gurnee, Yonatan Belinkov, Xingyi Song, Mrinmaya Sachan, and Neel Nanda. 2024. Confidence regulation neurons in language models. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Nishant Subramani, Nivedita Suresh, and Matthew E. Peters. 2022. Extracting latent steering vectors from pretrained language models. In *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 566–581. Association for Computational Linguistics.

Yiru Tang, Kun Zhou, Yingqian Min, Wayne Xin Zhao, Jing Sha, Zhichao Sheng, and Shijin Wang. 2025.

Enhancing chain-of-thought reasoning via neuron activation differential analysis. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 16151–16159. Association for Computational Linguistics.

Ye Tian, Baolin Peng, Linfeng Song, Lifeng Jin, Dian Yu, Lei Han, Haitao Mi, and Dong Yu. 2024. Toward self-improvement of llms via imagination, searching, and criticizing. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Curt Tigges, Oskar John Hollinsworth, Atticus Geiger, and Neel Nanda. 2023. Linear representations of sentiment in large language models. *CoRR*, abs/2310.15154.

Alexander Matt Turner, Lisa Thiergart, David Udell, Gavin Leech, Ulisse Mini, and Monte MacDiarmid. 2023. Activation addition: Steering language models without optimization. *CoRR*, abs/2308.10248.

Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xionghui Chen, Jianxin Yang, Zhenru Zhang, Yuqiong Liu, An Yang, Andrew Zhao, Yang Yue, Shiji Song, Bowen Yu, Gao Huang, and Junyang Lin. 2025a. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for LLM reasoning. *CoRR*, abs/2506.01939.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Yiming Wang, Pei Zhang, Baosong Yang, Derek F. Wong, and Rui Wang. 2025b. Latent space chain-of-embedding enables output-free LLM self-evaluation. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Heming Xia, Yongqi Li, Chak Tou Leong, Wenjie Wang, and Wenjie Li. 2025. Tokenskip: Controllable chain-of-thought compression in llms. *CoRR*, abs/2502.12067.

Chunlei Xin, Shuheng Zhou, Huijia Zhu, Weiqiang Wang, Xuanang Chen, Xinyan Guan, Yaojie Lu, Hongyu Lin, Xianpei Han, and Le Sun. 2025. Sparse latents steer retrieval-augmented generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 4547–4562. Association for Computational Linguistics.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 40 others. 2025a. Qwen3 technical report. *CoRR*, abs/2505.09388.

Zhen Yang, Mingyang Zhang, Feng Chen, Ganggui Ding, Liang Hou, Xin Tao, Pengfei Wan, and Ying-Cong Chen. 2025b. Less is more: Improving LLM reasoning with minimal test-time intervention. *CoRR*, abs/2510.13940.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, and 16 others. 2025. DAPO: an open-source LLM reinforcement learning system at scale. *CoRR*, abs/2503.14476.

Andy Zou, Long Phan, Sarah Li Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, and 2 others. 2023. Representation engineering: A top-down approach to AI transparency. *CoRR*, abs/2310.01405.

# A Preliminary Study of Probing

We conduct a preliminary probing study on mathematical reasoning benchmarks, including AIME and AMC-12. First, we construct contrastive datasets consisting of paired correct and incorrect reasoning trajectories for each input as described in §3.4. Then, using Qwen3 series models (1.7B and 4B), we extract last-token activation values across all layers as input features for probing classifiers. All samples are randomly partitioned into training and testing sets with a 4:1 ratio. Subsequently, we train a binary classifier to predict reasoning correctness from last-token activations. To mitigate the high dimensionality of the activation space, we apply $F$-statistic-based feature selection to identify the most discriminative neurons.

## A.1 Data Construction

We utilize Qwen3-32B to generate contrastive reasoning trajectories. Specifically, starting from all AIME (24+25) and AMC-12 problems, we sample 8 reasoning traces per question using a generation temperature of 1.0. We retain only those questions that yield a balanced outcome of exactly 4 correct and 4 incorrect trajectories. Under this setting, we obtain 15 AIME problems (from an initial 60) and 13 AMC-12 problems (from an initial 104), resulting in 120 and 104 samples, respectively, for probing classifier training.

## A.2 Preprocessing

For each reasoning trace, we extract last-token activations across all neurons as input features for the probing classifier. Specifically:

- **Source:** Hidden states of the last token in the reasoning path, immediately before final answer generation.
- **Position:** Post-activation outputs of the MLP blocks (i.e., after the SwiGLU activation for Qwen3 series model).
- **Dimension:** Activations from all transformer layers are concatenated into a single feature vector of dimension $L \times d_{\text{mlp}}$, where $L$ denotes the number of layers.
- **Normalization:** Feature values are scaled by dividing by $10 \times \sigma$, where $\sigma$ is the standard deviation for each neuron computed over the training set.

## A.3 Probing Classifier

Given the high dimensionality of the activation space relative to the training sample size, feature selection is critical to mitigate overfitting. We rank neurons using the ANOVA $F$-statistic (`sklearn.f_classif`) based on their linear association with the correctness label, and select the top-$K$ neurons with $K = 131{,}072$. For the probing classifier, we employed a Logistic Regression model with L1 regularization. The model was trained using 5-fold cross-validation with a stratified 4:1 train-test split. The detailed settings are summarized in Table 6.

Table 6: Implementation details of probing classifier.

| **Setting** | **Configuration / Value** |
|---|---|
| Feature | |
| Source | Last token of reasoning path |
| Position | MLP post-activation output |
| Scope | Global (all layers) |
| Normalization | Scaled by $1/(10 \times \text{std}_{\text{train}})$ |
| Feature Selection | ANOVA $F$-statistic |
| Dimension of Input | 131,072 |
| Training | |
| Architecture | Logistic Regression (`sklearn`) |
| Regularization | L1 (Lasso) |
| Solver | SAGA |
| Class Weight | Balanced |
| Penalty Strength | Grid search $\in [10^{-4}, 10]$ |
| Max Iterations | 50 |
| Random Seed | 42 |
| Validation | 5-fold cross-validation |
| Metric | AUROC |

# B Detailed Data Statistics

## B.1 Data for Adaptive Intervention Module

The adaptive intervention module is a binary classifier that predicts whether a given input is likely to fail under the base model and therefore requires intervention. Training data are constructed by sampling the base model on the training split for each benchmarks. Samples answered correctly are labeled as negative (i.e., no need for intervention), while incorrect samples are labeled as positive (i.e., intervention required). Dataset statistics are summarized in Table 7. Notably, for AMC-12 and HumanEval, we evaluate them only in a transfer setting using estimators trained on AIME and MBPP, due to lacking of training split; consequently, they are omitted from Table 7.

Table 7: Statistics of the training and validation data for the adaptive intervention module.

| Dataset | Total Samples | Number of each label (Train / Val) | |
|---|---|---|---|
| | | No Intervention | Need Intervention |
| AIME | 828 | 392 / 99 | 270 / 67 |
| MATH | 1,000 | 622 / 270 | 78 / 30 |
| GSM8K | 1,700 | 1,097 / 459 | 103 / 41 |
| MBPP | 163 | 83 / 29 | 37 / 14 |

### B.2 Data for Evaluation

We evaluate AdaRAS on a diverse set of benchmarks, where the dataset statistics are summarized in Table 8. All datasets are publicly available. Specifically, GSM8K, MATH, MBPP, and HumanEval are distributed under open licenses (i.e., MIT or Apache 2.0). The AIME and AMC datasets are sourced from public academic benchmarks widely accepted for evaluating mathematical reasoning. We split the datasets as follows:

- **AIME.** A unified set of samples is applied across all AIME datasets (2024, 2025, and Extend). This set contains 15 held-out samples (7 from AIME-24 and 8 from AIME-25), which are strictly excluded from all evaluation sets.
- **AMC-12, HumanEval.** Since these datasets do not provide official training splits, we reserve a small subset from the test split for performing AdaRAS. For example, we hold out 15 HumanEval samples for RCNs identification and evaluate on the remaining 149 samples.
- **Other datasets.** For benchmarks with official training splits (GSM8K, MATH, MBPP), patterns are extracted exclusively from the training set.

## C Details of Implementation

We use the following hyperparameters for AdaRAS in main experiments:

- **Number of top-$K$ neurons.** The number of intervened neurons is fixed to $K = 50$. As shown in Figure 5, these neurons are distributed across layers, with a higher concentration in middle-to-late layers.
- **Steering strength ($\alpha$).** The best steering strength coefficient $\alpha$ is searched ranging $[0.1, 0.3]$.

All experiments are conducted on 4 NVIDIA A100 GPUs and 8 NVIDIA RTX 3090 GPUs.

### C.1 Adaptive Intervention Module

We adopt a lightweight attention-based classifier for adaptive intervention module. Instead of using only the last token, this module take all token activations via an attention pooling mechanism to capture the global reasoning state. Detailed specifications are provided in Table 9.

### C.2 Evaluation

To ensure fair comparison and reproducibility, we use greedy decoding during inference across all benchmarks.

**Mathematic Benchmarks.** For the AIME, MATH-500, GSM8K, and AMC-12 datasets, we use the following prompt:

```
User: Question: {question}
Thinking process:
Please provide a step-by-step thinking process
and put your final answer in \boxed{}.
```

**Coding Benchmarks.** For the HumanEval and MBPP datasets, we use the following prompt:

```
User: {question}
```

We employed the EvalPlus (Liu et al., 2023) library to safely extract and execute code blocks.

## D Details of Reasoning Features

In Section 5.2, we use two trajectory-level metrics, Magnitude ($\bar{\mathcal{M}}$) and Angle ($\bar{\mathcal{A}}$), to analyze the geometric properties of reasoning trajectories in latent space. As in Figure 6, each point in the kernel density estimation (KDE) plot corresponds to the aggregated $(\bar{\mathcal{M}}, \bar{\mathcal{A}})$ values of a single complete reasoning trace. These metrics characterize the curvature and directional consistency of representation evolution across layers. Hidden states $\mathbf{h} \in \mathbb{R}^d$ are extracted from the model for each input, and the analysis is performed exclusively on generated reasoning tokens, excluding the input prompt. Following Wang et al. (2025b), the formal definitions are as follows:

**Sequence-Averaged Magnitude ($\bar{\mathcal{M}}$).** This metric measures the "tortuosity" of the reasoning trajectory in the activation space. It is defined as the ratio of the accumulated layer-wise changes to the

Table 8: Statistics of datasets. #Probe indicates the number of samples used for identifying RCNs by AdaRAS.

| Dataset | #Probe | #Test | Test split |
|---|---|---|---|
| Mathematic | | | |
| AIME-24[1] | 7 | 23 | Self-split |
| AIME-25[2] | 8 | 22 | Self-split |
| AIME-Extend[3] | 15 | 150 | Official |
| MATH-500[4] (Lightman et al., 2024) | 15 | 500 | Official |
| GSM8K[5] (Hendrycks et al., 2021) | 15 | 1,319 | Official |
| AMC-12[6] | 13 | 91 | Self-split |
| Coding | | | |
| HumanEval[7] (Chen et al., 2021) | 15 | 149 | Self-split |
| HumanEval+[8] (Liu et al., 2023) | 15 | 149 | Self-split |
| MBPP[9] (Austin et al., 2021) | 15 | 378 | Official |
| MBPP+[10](Liu et al., 2023) | 15 | 378 | Official |

[1] https://huggingface.co/datasets/Maxwell-Jia/AIME_2024
[2] https://huggingface.co/datasets/opencompass/AIME2025
[3] https://www.kaggle.com/datasets/hemishveeraboina/aime-problem-set-1983-2024
[4] https://huggingface.co/datasets/HuggingFaceH4/MATH-500
[5] https://huggingface.co/datasets/openai/gsm8k
[6] https://huggingface.co/datasets/rulins/amc12_22-24
[7] https://huggingface.co/datasets/openai/openai_humaneval
[8] https://huggingface.co/datasets/evalplus/humanevalplus
[9] https://huggingface.co/datasets/google-research-datasets/mbpp
[10] https://huggingface.co/datasets/evalplus/mbppplus

Table 9: Architecture details for the adaptive intervention module.

| Setting | Configuration |
|---|---|
| Architecture | |
| Structure | Linear(256, 256)<br>→ ReLU<br>→ Dropout(0.3)<br>→ Linear(256, 1) |
| Training | |
| Feature Selection | ANOVA $F$-statistic |
| Dimension of Input | 256 |
| Optimizer | Adam (LR=$10^{-4}$, Weight Decay=$10^{-5}$) |
| Loss Function | BCEWithLogitsLoss |
| Epochs | 100 (Early Stopping Patience=10) |

net change from the first to the last layer. For a single token $t$, the magnitude score $\mathcal{M}_t$ is computed as the $L_2$-norm of the difference vector between adjacent layers, normalized by the global displacement:

$$\mathcal{M}_t = \frac{1}{L} \cdot \frac{\sum_{l=0}^{L-1} \|\mathbf{h}_t^{l+1} - \mathbf{h}_t^l\|_2}{\|\mathbf{h}_t^L - \mathbf{h}_t^0\|_2 + \epsilon}, \tag{4}$$

where $\epsilon = 10^{-6}$ is a small constant for numerical stability. The final sequence-averaged metric is obtained by averaging over all generated tokens:

$$\bar{\mathcal{M}} = \frac{1}{T} \sum_{t=1}^{T} \mathcal{M}_t. \tag{5}$$

A lower $\bar{\mathcal{M}}$ indicates a straighter, more direct transformation of representations through the network depth, which we associate with more stable reasoning.

**Sequence-Averaged Angle ($\bar{\mathcal{A}}$).** This metric captures the directional stability of the semantic evolution. It calculates the angular deviation between adjacent layers relative to the global angular shift. We first define the cosine similarity $\text{sim}(\mathbf{u}, \mathbf{v}) = \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\|\|\mathbf{v}\|}$. The angle score $\mathcal{A}_t$ for token $t$ is defined as:

$$\mathcal{A}_t = \frac{1}{L} \cdot \frac{\sum_{l=0}^{L-1} \arccos\left(\text{sim}(\mathbf{h}_t^l, \mathbf{h}_t^{l+1})\right)}{\arccos\left(\text{sim}(\mathbf{h}_t^0, \mathbf{h}_t^L)\right) + \epsilon}. \tag{6}$$

Similarly, the sequence-averaged angle is computed as:

$$\bar{\mathcal{A}} = \frac{1}{T} \sum_{t=1}^{T} \mathcal{A}_t. \tag{7}$$

Table 10: Comparison with strong test-time baselines, including $k$-sampling, verifier-based reranking, and few-shot CoT prompting.

| Dataset | $k$-Sampling | Verifier | Few-shot | AdaRAS |
|---|---|---|---|---|
| AIME-24 | 47.83 (maj@4) | 30.43 (Best-of-4) | 39.13 (3-shot) | **60.87** (Greedy) |
| AIME-25 | 50.00 (maj@4) | 40.91 (Best-of-4) | 31.82 (3-shot) | **54.55** (Greedy) |
| HumanEval | 93.96 (pass@4) | 79.19 (Best-of-4) | – | **96.64** (pass@4) |
| HumanEval+ | 87.92 (pass@4) | 71.82 (Best-of-4) | – | **89.26** (pass@4) |

Table 11: Comparison with post-trained reasoning models under sampling-based metrics. AdaRAS uses greedy decoding on AIME and pass@4 on coding tasks.

| Dataset | CoT | R1-Distill | Nemotron | OpenThinker | AdaRAS |
|---|---|---|---|---|---|
| AIME-24 (maj@4) | 47.83 | 26.09 | 52.17 | 52.17 | **60.87** |
| AIME-25 (maj@4) | 50.00 | 36.36 | 45.45 | 50.00 | **54.55** |
| HumanEval (pass@4) | 93.96 | 75.17 | 78.52 | 83.89 | **96.64** |
| HumanEval+ (pass@4) | 87.92 | 69.80 | 73.83 | 78.52 | **89.26** |

Larger values of $\bar{\mathcal{M}}$ or $\bar{\mathcal{A}}$ imply that the reasoning process undergoes significant fluctuations or detours in the latent space.

# E Additional Experiments

## E.1 Stronger Test-Time Baselines

To further examine whether the gains of AdaRAS can be matched by stronger test-time baselines, we compare AdaRAS with three representative alternatives: (1) sampling-based methods, including majority voting on mathematical reasoning tasks and pass@4 on coding tasks; (2) verifier-based reranking, where a trained verifier selects the best response from multiple generations; and (3) few-shot chain-of-thought prompting, where demonstration examples are provided in the prompt. These baselines increase the inference budget or provide additional test-time supervision, and therefore serve as strong controls for evaluating whether AdaRAS improves reasoning beyond simple statistical scaling or prompt augmentation.

As shown in Table 10, AdaRAS consistently outperforms these alternatives. On AIME, AdaRAS uses only single-pass greedy decoding, yet surpasses majority voting with four sampled generations. This indicates that its improvement cannot be explained by simple statistical scaling. AdaRAS also outperforms verifier reranking, suggesting that directly steering reasoning-critical activations is more effective than selecting among completed trajectories. In addition, 3-shot CoT prompting fails to close the gap, showing that the identified reasoning-critical neurons provide transferable activation-level guidance rather than merely encoding prompt demonstrations.

Table 12: Scalability analysis on Qwen3-4B-Thinking-2507 and Qwen3-14B. All results are evaluated under our Greedy-CoT setting.

| Dataset | 4B-2507 | +Ours | 14B | +Ours |
|---|---|---|---|---|
| AIME-24 | 56.52 | **65.22** | **60.87** | **60.87** |
| AIME-25 | 63.64 | **77.27** | 72.73 | **77.27** |
| HumanEval | 97.99 | **99.33** | – | – |
| HumanEval+ | 92.62 | **94.63** | – | – |

We also compare AdaRAS with post-trained reasoning models under sampling-based metrics. As shown in Table 11, AdaRAS achieves the best performance across all mathematical reasoning and coding benchmarks. Notably, on AIME benchmarks, AdaRAS uses single-pass greedy decoding, while the compared baselines are evaluated with majority voting over four sampled generations. These results show that AdaRAS provides a compute-efficient and effective test-time enhancement compared with repeated sampling, verifier reranking, few-shot prompting, and post-trained reasoning variants.

## E.2 Model Scalability

We further evaluate AdaRAS on Qwen3-4B-Thinking-2507 and Qwen3-14B to examine whether the identified reasoning-critical neurons re-

Table 13: Linguistic feature analysis of generated responses. AdaRAS does not consistently increase response length or structural complexity.

| Dataset | #Char | | #Words | | #Sentences | | Dep. Depth | |
|---|---|---|---|---|---|---|---|---|
| | CoT | AdaRAS | CoT | AdaRAS | CoT | AdaRAS | CoT | AdaRAS |
| AIME-Extend | 36616.7 | 35966.5 | 7555.6 | 7379.4 | 557.3 | 573.5 | 4.67 | 4.62 |
| Math-500 | 11664.4 | 11581.7 | 2407.4 | 2380.6 | 224.6 | 219.5 | 4.38 | 4.36 |
| GSM8K | 4765.3 | 4585.2 | 939.3 | 906.1 | 94.4 | 90.9 | 4.34 | 4.34 |
| AMC-12 | 24386.8 | 22399.2 | 5059.1 | 4656.6 | 462.8 | 413.7 | 4.36 | 4.37 |
| HumanEval | 10091.0 | 11192.5 | 1881.7 | 2104.0 | 203.3 | 222.4 | 4.01 | 4.06 |
| MBPP | 8456.7 | 8118.2 | 1546.7 | 1492.3 | 149.4 | 144.5 | 4.25 | 4.23 |

| Dataset | Qwen3-4B Sampling | +AdaRAS Sampling | +AdaRAS Greedy |
|---|---|---|---|
| AIME-24 | $60.87 \pm 0.00$ | **61.96** $\pm 2.18$ | **65.22** |
| AIME-25 | $75.00 \pm 2.62$ | **76.14** $\pm 1.97$ | **77.27** |

Table 14: Reproduction under the official Qwen3 sampling setting. Sampling results are averaged over four independent runs with different random seeds.

main effective across model variants and scales. As shown in Table 12, AdaRAS brings consistent improvements on the reasoning-tuned Qwen3-4B-Thinking-2507 model and further improves Qwen3-14B on AIME-25. These results suggest that AdaRAS is not limited to a single base model, and can transfer to both reasoning-tuned and larger models. For reproducibility, we rerun Qwen3-4B-Thinking-2507 under the same Greedy-CoT evaluation protocol used in this paper. The obtained scores may differ from previously reported results due to differences in decoding and prompting setups.

To account for sampling randomness, we further evaluate AdaRAS under the official Qwen3 sampling setting over four independent runs with different random seeds. As shown in Table 14, AdaRAS is evaluated with the same decoding configuration and consistently improves over the sampling baseline. Notably, AdaRAS achieves even stronger performance under single-pass greedy decoding, suggesting that its gains do not rely on sampling randomness.

### E.3 Efficiency Analysis

**Reasoning Length.** We analyze whether AdaRAS improves reasoning by simply changing response format or increasing generation length. Table 13 reports several linguistic features of the generated responses. Across different reasoning and coding benchmarks, AdaRAS shows no consistent increase in response length, number of words, number of sentences, or dependency tree depth. This suggests that the performance gains do not come from superficial formatting changes or longer reasoning traces.

Table 15: Computational cost analysis. Training time is the offline cost for fitting the gating predictor, while inference overhead is measured per sample.

| Dataset | Train (s) | Infer (ms) |
|---|---|---|
| AIME-Extend | 1103.36 | 89.1 |
| Math-500 | 1047.86 | 90.7 |
| GSM8K | 1674.78 | 108.8 |
| AMC-12 | 35.68 | 91.1 |
| HumanEval | – | 88.5 |
| MBPP | 81.84 | 88.0 |

**Computational Cost.** We also report the computational cost of our AdaRAS in Table 15. The gating predictor is trained offline and requires only a small one-time cost. During inference, AdaRAS introduces only about 0.09–0.11 seconds of overhead per sample, which is substantially cheaper than sampling-based methods requiring multiple generations. For HumanEval, we use the predictor trained on MBPP because HumanEval does not provide a training split. Overall, AdaRAS improves reasoning reliability without substantially increasing output length or inference latency.

### E.4 Prompt Stability

We evaluate the robustness of AdaRAS to prompt variations by comparing performance with and without the "Let's think step by step" trigger phrase.

Table 16: Impact of prompt variations. AdaRAS consistently improves over CoT with and without the "Let's think step by step" trigger phrase.

| Setting | AIME-24 | | AIME-25 | | HumanEval | | HumanEval+ | |
|---|---|---|---|---|---|---|---|---|
| | CoT | AdaRAS | CoT | AdaRAS | CoT | AdaRAS | CoT | AdaRAS |
| w/ "let's think" | 47.83 | **60.87** | 40.91 | **54.55** | 74.50 | **79.19** | 67.79 | **71.14** |
| w/o "let's think" | 30.43 | **43.48** | 50.00 | **50.00** | 77.18 | **79.19** | 69.80 | **73.15** |

Table 17: Ablation of the adaptive gating component on the probing baseline. Probing+ adds the same adaptive gating component to Probing, while AdaRAS further uses RCN selection.

| Dataset | Probing | Probing+ | AdaRAS |
|---|---|---|---|
| AIME-24 | 43.48 | 52.17 | **60.87** |
| AIME-25 | 50.00 | 40.91 | **54.55** |
| AIME-Extend | 48.00 | 49.33 | **52.67** |
| Math-500 | 85.40 | 84.40 | **86.40** |
| GSM8K | 88.32 | 89.01 | **89.08** |
| AMC-12 | **73.63** | 72.53 | 70.33 |
| HumanEval | 77.85 | **79.19** | **79.19** |
| HumanEval+ | 72.48 | 71.81 | **73.15** |
| MBPP | 70.37 | 70.90 | **72.22** |
| MBPP+ | 59.52 | 59.79 | **60.58** |

As shown in Table 16, AdaRAS consistently outperforms CoT under both prompting settings. On AIME-24, CoT degrades substantially without the reasoning trigger, while AdaRAS maintains a clear advantage. On coding tasks, where verbose reasoning prompt may sometimes hurts performance, AdaRAS still improves over the corresponding baseline. These results indicate that AdaRAS is less sensitive to prompt selection and provides more stable test-time reasoning enhancement.

## E.5 Additional Ablations

### E.5.1 Adaptive Intervention on Probing Baseline

We further isolate the effect of the adaptive gating component by adding it to the vanilla probing baseline, denoted as Probe+Predict. As shown in Table 17, adaptive gating brings limited improvements over Probe on some datasets, but it does not consistently improve the baseline and remains clearly weaker than AdaRAS. This indicates that the adaptive intervention module alone is not the main source of the gains. The proposed Reasoning-Critical Neuron (RCN) selection strategy is necessary for effective and stable activation steering.

Table 18: Effectiveness of polarity-based filtering. "Magnitude Only" selects top-50 neurons solely based on importance scores without polarity filtering.

| **Dataset** | **Accuracy (%)** | | |
|---|---|---|---|
| | **Base** | **Magnitude Only** | **Polarity-Aware (Ours)** |
| AIME-24 | 47.83 | 56.52 | **60.87** |
| AIME-25 | 40.91 | 45.45 | **54.55** |
| AIME-Extended | 47.33 | 50.67 | **52.67** |

### E.5.2 Effect of Activation Polarity

In Section 3.2, we proposed a polarity-based filtering method: constructing the steering vector by retaining only neurons that exhibit discriminative sign flips between correct and incorrect traces. To verify the necessity of this strategy, we conducted an additional ablation study on the AIME benchmarks. We compare our method against a baseline selection strategy ("Magnitude Only") that selects the Top-$K$ neurons solely based on the absolute importance score $|S(l, i)|$, without enforcing the polarity constraint.

Table 18 presents the results. While selecting neurons based on magnitude alone yields performance gains over the base model (e.g., improving AIME-24 from 47.83% to 56.52%), introducing the polarity-aware filter significantly amplifies these gains. Specifically, our method achieves 60.87% on AIME-24 and 54.55% on AIME-25, consistently outperforming the magnitude-based selection. These results empirically confirm that neurons exhibiting contradictory activation patterns (sign flips) are important for reasoning validity, justifying our construction of $\boldsymbol{S}'_l$.

## E.6 Reasoning Trajectories in Latent Space

To further validate the motivation that correct and incorrect reasoning exhibit distinct activation patterns, we visualize the evolution of latent representations. We use the weights of the linear probing

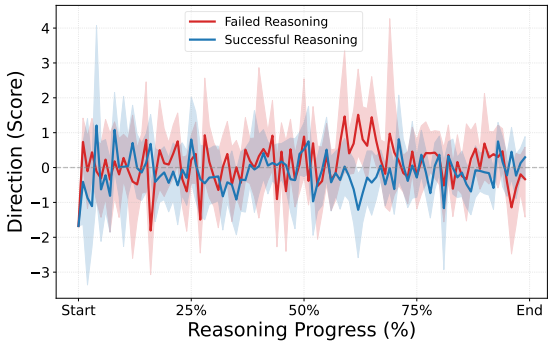



Figure 7: Projection of reasoning states onto the probing classifier learned space. Successful (Blue) and failed (Red) reasoning processes occupy distinguishable regions in the latent activation space, particularly in the middle and later stages of reasoning.

classifier trained in §2.2 to define a reference direction. Specifically, we project the layer-wise hidden states of reasoning traces onto the direction defined by the top-$K$ most discriminative neurons identified by the probe.

Figure 7 illustrates the aggregated trajectories for successful and failed reasoning samples on the AIME dataset. Although the two curves may overlap locally at certain reasoning stages, they exhibit noticeably different overall trends, particularly in the middle and later stages of reasoning. Successful traces generally maintain higher projection scores and more stable trajectories, whereas failed traces show larger fluctuations and progressive downward drifts. These observations suggest that successful and failed reasoning processes occupy distinguishable regions in the latent activation space, supporting the feasibility of extracting reasoning-critical signals from activation patterns.

## F Case Study

To intuitively understand how AdaRAS corrects reasoning errors, we present three randomly selected examples from the AIME dataset. In each case, we compare the reasoning trace generated by the base model (which fails) against the trace generated with our sparse activation steering. Blue text highlights the correct logical steps induced by AdaRAS, while red text indicates the errors or hallucinations in the base model's reasoning.

### Case 1: Correcting Domain Constraints in Geometry

**Problem:** Let $A, B, C, D$ be points on the hyperbola $\frac{x^2}{20} - \frac{y^2}{24} = 1$ such that $ABCD$ is a rhombus whose diagonals intersect at the origin. Find the greatest real number that is less than $BD^2$ for all such rhombi.

**Comparison of Reasoning Traces**

**Base Model (Failed):**
... We derive the expression for the diagonal squared as $BD^2 = 80 + \frac{22}{3}d^2$. Since $d^2 \geq 0$, the expression is always greater than 80. As $d \to 0$, $BD^2$ approaches 80. Therefore, the greatest real number less than $BD^2$ is $\boxed{80}$.

**AdaRAS (Success):**
... We derive the expression $BD^2 = 80 + \frac{22}{3}d^2$. However, the vertices must lie on the hyperbola. Solving the system for the rhombus vertices $(c, 0)$ and $(0, d)$ yields the constraint that $d^2$ cannot be zero. Specifically, the minimum value is found to be $d^2 = \frac{600}{11}$. Substituting this into the expression: $BD^2 = 80 + \frac{22}{3} \cdot \frac{600}{11} = 480$. Thus, the greatest real number is $\boxed{480}$.

**Analysis:** In this geometry problem, the Base model correctly derives the algebraic formula but fails to check the *geometric domain constraints* (it assumes $d$ can be arbitrarily small). AdaRAS steers the model to verify the existence condition of the rhombus, identifying the lower bound of $d^2$ implied by the hyperbola's equation. This illustrates that our steering vector encodes signals related to "constraint satisfaction."

### Case 2: Rectifying Symbolic Manipulation

**Comparison of Reasoning Traces**

**Base Model (Failed):**
... We simplify the general term. The rational part telescopes to $\frac{31}{13}$. For the logarithmic part, we consider $\prod \log_{k+1} k$. This product telescopes to $\frac{1}{\log_4 64} = \frac{1}{3}$. Multiplying the results yields $\frac{31}{13} \cdot \frac{1}{3} = \frac{31}{39}$. Thus $m + n = \boxed{70}$.

**AdaRAS (Success):**
... We simplify the general term using logarithmic identities: $\frac{\log_k(5^{k^2-1})}{\log_{k+1}(5^{k^2-4})} \to \cdots \to \log_k(k+1)$. We split the product. The rational part is $\frac{31}{13}$. The logarithmic part is $\prod_{k=4}^{63} \log_k(k+1)$. Using the change of base formula $\log_k(k+1) = \frac{\ln(k+1)}{\ln k}$, this telescopes to $\frac{\ln 64}{\ln 4} = \log_4 64 = 3$. Multiplying the results gives $\frac{31}{13} \cdot 3 = \frac{93}{13}$. Summing them yields $\boxed{106}$.

**Problem:** Evaluate the product:

$$\prod_{k=4}^{63} \frac{\log_k(5^{k^2-1})}{\log_{k+1}(5^{k^2-4})}.$$

Find $m + n$ where the result is $m/n$.

**Analysis:** Here, the Base model performs a valid rational simplification but hallucinates the result of the logarithmic telescoping product (calculating it as $1/3$ instead of 3, effectively inverting the base and argument). AdaRAS corrects this symbolic manipulation error, guiding the model to apply the change-of-base formula correctly.

### Case 3: Deriving Correct Geometric Relationships

**Problem:** Circle $\omega_1$ (radius 6) is internally tangent to $\omega_2$ (radius 15). A rectangle $EFGH$ is inscribed in $\omega_1$ such that specific area conditions are met. Find the area of $EFGH$.

**Comparison of Reasoning Traces**

**Base Model (Failed):**
... Let the vertices be $(a, b)$. Equating the areas of $\triangle DGF$ and $\triangle CHG$ yields the condition $|a(12 - b)| = |b(24 + a)|$. Simplifying this leads to the relationship $b = \frac{6a}{a+12}$. Substituting this into the circle equation and solving... yields an area of 48. Final Answer: $\boxed{49}$.

---

**AdaRAS (Success):**
... Let the rectangle dimensions be determined by coordinate $y$ and width $m$. Equating the areas of $\triangle DGF$ and $\triangle CHG$ leads to the equation $12m - my = 48y - my$. The term $-my$ cancels out, simplifying to $12m = 48y$, which implies $m = 4y$. Substituting this into the circle constraint $(m/2)^2 + y^2 = 36 \implies 5y^2 = 36$. The area is $m \cdot 2y = 8y^2 = \frac{288}{5}$. Final Answer: $\boxed{293}$.

**Analysis:** The Base model sets up the coordinate geometry correctly but fails at the algebraic simplification step, hallucinating a complex rational relationship ($b = \frac{6a}{a+12}$) that does not follow from the area equality. AdaRAS successfully recovers the linear relationship ($m = 4y$) by correctly canceling terms during the derivation, ensuring the subsequent area calculation is based on valid premises.