

From Complexity to Clarity: A Systematic Study of Geometric Dynamics in Hidden Representations of LLMs

Xiang Li

Stony Brook University
xiang.li.13@stonybrook.edu

Luzhe Sun

Toyota Technological Institute at Chicago
luzhesun@ttic.edu

Matthew R. Walter

Toyota Technological Institute at Chicago
mwalter@ttic.edu

Jiawei Zhou

Stony Brook University
jiawei.zhou.1@stonybrook.edu

Abstract

Large language models (LLMs) perform inference through hidden representations that evolve across layers and generated tokens, forming complex trajectories in hidden representation space. We study the geometric dynamics of these trajectories for label-free model self-evaluation problem: does the hidden-state tensor already contain model response correctness signals, and where do those signals reside? Our analysis is organized along three intrinsic dimensions of inference-time computation: layer depth, token span, and representation space. Across seven benchmarks spanning mathematical reasoning, commonsense reasoning, and multilingual understanding, and two primary instruction-tuned LLMs, we find that correctness separability is highly localized, consistently concentrating in late-layer representations and limited token regions. Motivated by this, we propose Norm Drop Rate (NDR), a minimal diagnostic metric based on final-layer norm contraction. NDR achieves competitive AUROC across tested models and the best AUPR on Qwen2-7B-Instruct, while remaining the simplest in design and most robust when controlled for response length or changing representation bases. These results suggest that hidden-state trajectory analysis should focus less on increasingly complex diagnoses but more on identifying simple, localized signals.¹

1 Introduction

Large language models (LLMs) are increasingly deployed in settings where users rely on a single generated response to be correct (Achiam et al., 2023; Touvron et al., 2023). Yet fluent generations can be unsupported or wrong, making failures hard to detect. A practical remedy is single-pass model correctness evaluation (Kadavath et al., 2022; Manakul et al., 2023; Kuhn et al., 2023): estimating

correctness using only inference-time signals, without training a separate verifier or sampling many alternative responses.

Geometric signal in hidden representation. For each token position and each layer, the autoregressive transformer updates a hidden representation that supports the next-token prediction (Vaswani et al., 2017). For a fixed generation, the hidden states indexed by layer depth and generated tokens form a structured tensor that can be viewed as trajectories in hidden representation space. This suggests a training-free approach to self-evaluation: correctness may be reflected not only in what the model outputs, but also in how internal representations move through space during inference.

From composite heuristics to systematic diagnosis. Recent work has shown that intermediate representations expose meaningful stage-wise computation (Belrose et al., 2023; Halawi et al., 2024; Shai et al., 2024), and Chain-of-Embeddings (CoE) aggregates geometric statistics over token-averaged layer trajectories for output-free self-evaluation (Wang et al., 2024). However, geometry-based scores are often composite heuristics: it remains unclear which quantities carry signal, where that signal localizes, and how robust it is to common confounds (e.g., generation length). We address this by treating the hidden-state tensor as the central object of analysis and decomposing it along three intrinsic dimensions of inference-time computation. The **layer dimension** asks where in layer depth a label-free scalar score separates correct from incorrect generations. The **token dimension** asks which response regions drive the signal and whether it persists after controlling for response length. The **representation dimension** asks whether geometric signals remain stable when computed under different representation bases. Throughout, labels are used only for evaluation; every score is computed from a single forward generation.

¹Our code is available at <https://github.com/Zesearch/llm-trajectory-geometry-analysis>.

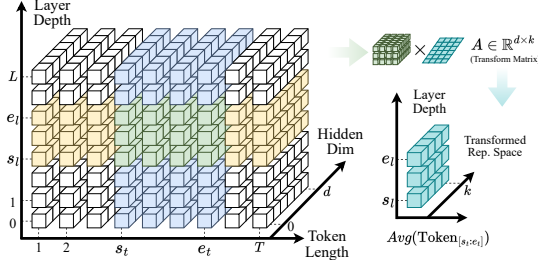


Figure 1: **Three-dimensional diagnostic framework.** From one prompt and one generated response, we log hidden states $\mathcal{H}$ over generated tokens and layers. A training-free score is computed from a selected sub-tensor defined by three components: Token Span (Token $[s_t, e_t]$), Layer Window (Layer $[s_l, e_l]$), and Representation Space.

Complex trajectories, localized signals. Across tasks and models, we find that discriminative power in hidden-state trajectories is highly localized, concentrating in late layer representations and partial token regions. Motivated by this, we propose **Norm Drop Rate (NDR)**, an extremely simple unsupervised metric based on final-layer norm contraction that matches or exceeds more complex trajectory summaries across benchmarks while requiring substantially lower computation costs.

Contributions. Our contributions are as follows: (i) a systematic diagnostic framework that decomposes hidden-state correctness signals along three dimensions: layer depth, token span, and representation space; (ii) an empirical localization result showing that trajectory-based methods concentrate their useful signal in late layers and hallucinated token spans; (iii) NDR, a minimal label-free score distilled from layerwise norm dynamics that is robust under length matching and less dependent on fragile subspace selection than other metrics.

2 Related Work

2.1 Hidden-State Geometry

A growing body of work studies LLM hidden states through a geometric lens, treating activations as points, directions, subspaces, or trajectories in hidden representation space. One line identifies linear directions that encode truthfulness, latent knowledge, or hallucination signals (Marks and Tegmark, 2023; Burns et al., 2022; Park et al., 2025; O’Neill et al., 2025) and uses such directions for steering and intervention (Zou et al., 2023). Belief-state analyses further show that residual streams can encode the geometry of latent predictive states (Shai

et al., 2024), and Li et al. (2026) unveils the model internal geometric picture collectively through the lens of hidden representation dispersion. Closest to our setting, trajectory-based methods treat the evolution of hidden states across layers as a discrete curve and derive training-free correctness scores from angle, displacement, or curvature statistics (Wang et al., 2024; Manson, 2025; Chang et al., 2026; Miao et al., 2026). These works demonstrate that hidden-state geometry captures meaningful signals about semantics, belief, and reliability, but most anchor their analysis to a fixed layer range, token span, or basis. Our work systematically decomposes where such signals arise across three dimensions of the hidden-state tensor.

2.2 Layer-Wise Signals in Hidden Representations

Different layers of a Transformer have been found to serve different computational roles. Logit-lens and tuned-lens analyses show that answer content becomes progressively decodable from intermediate residual states (nostalgabraist, 2020; Belrose et al., 2023), and activation patching localizes factual recall to specific middle layers (Meng et al., 2022; Zhang and Nanda, 2024). Recent studies further suggest that different depth ranges support qualitatively different computations: middle layers can act as semantic workspaces where answer selection occurs, while late layers amplify the selected answer in vocabulary space (Wiegrefe et al., 2025); layer-intervention work identifies distinct inference stages from early contextualization through late-stage residual sharpening (Lad et al., 2026); and analysis under misleading demonstrations reveals overthinking phenomena in late layers (Halawi et al., 2024). While these results establish that different depths carry different types of information, geometric analyses have not yet systematically studied how correctness-related signals distribute across layers.

2.3 Label-Free Self-Evaluation

Self-evaluation estimates whether a generated response is correct using only inference-time information (e.g., hidden states). Output-based methods elicit explicit confidence scores, aggregate multiple samples, or estimate semantic uncertainty (Kadavath et al., 2022; Wang et al., 2022; Manakul et al., 2023; Kuhn et al., 2023; Farquhar et al., 2024). Activation-based methods train supervised probes on hidden states to detect truthfulness or hallucina-

tion (Azaria and Mitchell, 2023; Liu et al., 2024; Orgad et al., 2025), but require labeled data and offer limited interpretability. Training-free activation-based methods are closest to our setting: CoE derives correctness scores directly from hidden-state trajectories without fitting a detector (Wang et al., 2024), and hidden representation dispersion measurements reflect the likelihood of model correctness on a dataset without labels (Li et al., 2026). We adopt the same label-free self-evaluation setting but use it as a diagnostic testbed: given one response and its hidden-state tensor, we ask which simple geometric signals, and which regions of the tensor, carry the most discriminative power.

3 Systematic Analysis Framework for Representation Geometric Dynamics

3.1 Preliminaries

Transformer LM Computation Let f_θ be a decoder-only Transformer with L layers and hidden dimension d . Given a prompt x , the model autoregressively generates a response of T tokens $y_{1:T}$. For each generated token position $t \in \{1, \dots, T\}$ and layer $l \in \{0, \dots, L\}$, we record a vector $h_t^l \in \mathbb{R}^d$, where $l = 0$ denotes the token embeddings input to the first Transformer layer. For $l = 1, \dots, L - 1$, h_t^l denotes the residual-stream hidden state after the full l -th Transformer block, including attention, MLP, and residual updates. The final-layer state h_t^L is the representation consumed by the unembedding, producing the next-token distribution $p_\theta(\cdot | x, y_{1:t}) = \text{softmax}(W_U h_t^L)$, where $W_U \in \mathbb{R}^{|\mathcal{V}| \times d}$ is the output embedding matrix over vocabulary $\mathcal{V}$. The central object of study is the hidden-state tensor

$$\mathcal{H} \in \mathbb{R}^{(L+1) \times T \times d}, \quad \mathcal{H}_{l,t,:} = h_t^l. \quad (1)$$

We specify how h_t^L is extracted in Appendix J.1, and the decoding protocol that produces $y_{1:T}$ in Appendix D.

Token-mean layer trajectory. Our scores operate on a layer-indexed trajectory derived from $\mathcal{H}$. Given a token span $\mathcal{T} \subseteq \{1, \dots, T\}$, we average hidden states over tokens to obtain

$$\bar{h}_{\mathcal{T}}^l = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} h_t^l, \quad l = 0, \dots, L. \quad (2)$$

The sequence $(\bar{h}_{\mathcal{T}}^0, \dots, \bar{h}_{\mathcal{T}}^L)$ is the token-mean layer trajectory. Unless otherwise stated, $\mathcal{T}$ contains all generated response tokens and the trajectory is evaluated in the native hidden space.

3.2 Label-Free Model Correctness Evaluation

We study label-free model correctness evaluation: given a prompt and one autoregressive generation, estimate response correctness using only inference-time signals from the hidden-state tensor $\mathcal{H}$, without ground-truth labels. This setting requires no trained verifier, no external tool, and no labeled supervision (Wang et al., 2024; Kadavath et al., 2022; Farquhar et al., 2024).

In this paper, every score $s(\mathcal{H}) \in \mathbb{R}$ is computed from the hidden-state tensor defined above. A good score should separate correct from incorrect generations. Correctness labels are used only after scoring, to measure this separability.

3.3 Three Diagnostic Dimensions

The hidden-state tensor $\mathcal{H}$ naturally suggests three diagnostic dimensions, corresponding to its layer, token, and representation structure (Figure 1). Each dimension isolates a different factor that may affect the correctness score.

Layer dimension. Where along depth does correctness separability concentrate? We fix the token span and sweep layer windows $[s_l, e_l]$ to localize which depth regions carry signal.

Token dimension. Which response regions drive the signal, and does the signal remain after removing the effect of response length? We vary the token span $\mathcal{T}$ across different response regions.

Representation dimension. Do geometric signals remain stable under different representation bases? We recompute scores under alternative linear maps such as PCA subspaces and logit space; the default is native hidden representation space.

All later sections modify one of these three choices while keeping the same scoring setup.

4 Experimental Setup

We evaluate hidden-state scoring on seven benchmarks spanning mathematical reasoning, broad knowledge, commonsense reasoning, and multi-lingual understanding. For every method, we follow the same procedure: generate one response per prompt with greedy decoding, log the hidden-state tensor $\mathcal{H}$, compute a scalar score without using labels, and evaluate the score against binary correctness labels.

(a) Qwen2-7B-Instruct (AUROC↑ / FPR95↓ / AUPR↑)								
Method	GSM8K	MATH	MMLU	TheoremQA	CSQA	Belebele	MGSM	Avg.
MaxProb [◦]	65.3 / 77.1 / 88.7	68.7 / 69.5 / 61.9	54.3 / 91.4 / 64.8	61.1 / 88.9 / 33.4	56.8 / 94.5 / 79.8	58.6 / 89.7 / 87.9	73.4 / 69.8 / 91.0	62.6 / 83.0 / 72.5
Entropy [◦]	67.3 / <u>72.7</u> / 89.5	70.0 / 67.4 / 63.3	54.9 / <u>90.6</u> / 65.2	61.9 / 87.9 / 33.9	<u>58.3</u> / 93.9 / 80.8	60.3 / 88.2 / 88.6	75.8 / <u>55.8</u> / 92.0	64.1 / <u>79.5</u> / 73.3
Perplexity [◦]	65.5 / 76.7 / 88.8	69.6 / <u>67.4</u> / 62.7	54.4 / 91.3 / 64.9	61.8 / 88.7 / 34.0	57.3 / 94.9 / 80.3	58.7 / 89.0 / 87.9	73.8 / 67.4 / 91.1	63.0 / 82.2 / 72.8
CoE-R [◦]	67.0 / 83.9 / 90.1	70.9 / 84.9 / <u>71.2</u>	<u>56.7</u> / 93.5 / <u>69.7</u>	<u>67.3</u> / 90.2 / <u>47.0</u>	52.2 / 96.1 / 75.9	58.4 / 91.9 / 88.4	75.1 / 86.1 / 93.5	63.9 / 89.5 / 76.5
CoE-C [◦]	<u>69.9</u> / 78.0 / <u>90.8</u>	<u>73.6</u> / 71.7 / 70.9	55.2 / 91.1 / 67.2	<u>66.5</u> / 93.4 / 43.9	56.5 / <u>92.9</u> / 78.0	<u>65.7</u> / <u>87.5</u> / <u>90.8</u>	<u>82.1</u> / 72.1 / <u>95.7</u>	<u>67.1</u> / 83.8 / <u>76.8</u>
Curvature [*]	69.0 / 73.1 / 90.6	75.0 / 67.6 / 73.0	55.4 / 90.5 / 68.4	66.8 / 91.4 / 44.2	53.4 / 91.6 / 75.9	69.9 / 82.4 / 92.2	80.2 / 58.1 / 95.2	67.1 / 79.2 / 77.1
Lateral Ratio [*]	70.7 / 78.9 / 90.9	74.1 / 71.2 / 70.3	56.7 / 91.5 / 69.5	68.9 / 93.7 / 47.7	56.8 / 91.9 / 77.6	67.1 / 87.5 / 91.4	83.1 / 65.1 / 95.9	68.2 / 82.8 / 77.6
NDR [*]	70.5 / 78.9 / 91.0	75.2 / 74.8 / 74.2	58.6 / 90.5 / 71.1	68.2 / 94.0 / 47.3	55.2 / 95.2 / 77.6	67.1 / 87.5 / 91.1	81.5 / 62.8 / 95.5	68.0 / 83.4 / 78.3
(b) Llama-3-8B-Instruct (AUROC↑ / FPR95↓ / AUPR↑)								
Method	GSM8K	MATH	MMLU	TheoremQA	CSQA	Belebele	MGSM	Avg.
MaxProb [◦]	70.9 / 81.4 / 90.6	51.2 / 87.6 / 21.8	52.4 / 93.1 / 66.3	<u>62.6</u> / <u>92.5</u> / <u>31.1</u>	53.9 / 90.4 / 73.9	57.1 / 95.6 / 91.8	73.7 / <u>72.0</u> / 90.0	60.3 / 87.5 / 66.5
Entropy [◦]	<u>72.2</u> / <u>76.5</u> / <u>91.2</u>	52.5 / 86.5 / 22.3	52.8 / 92.4 / 66.6	61.9 / 93.0 / 31.0	<u>54.4</u> / 89.2 / 74.2	<u>57.8</u> / 93.3 / <u>92.1</u>	<u>75.8</u> / <u>72.0</u> / 91.0	<u>61.1</u> / <u>86.1</u> / 66.9
Perplexity [◦]	70.9 / 80.6 / 90.7	51.6 / 88.3 / 21.9	52.8 / 92.9 / 66.6	61.9 / 93.9 / 30.3	54.1 / 90.4 / 73.9	57.3 / 93.3 / 91.9	73.2 / 76.0 / 89.8	60.3 / 87.9 / 66.4
CoE-R [◦]	59.8 / 89.1 / 85.2	62.9 / <u>84.5</u> / 32.7	<u>56.0</u> / 91.8 / <u>69.8</u>	53.3 / 92.7 / 19.1	51.3 / 94.9 / 71.0	54.3 / <u>90.0</u> / 90.3	60.3 / 88.0 / 84.7	56.8 / 90.1 / 64.7
CoE-C [◦]	70.5 / 81.0 / 90.5	<u>66.6</u> / 86.2 / <u>37.7</u>	55.3 / 90.9 / 67.9	52.7 / 95.2 / 21.4	54.2 / 86.8 / <u>74.7</u>	54.8 / 95.6 / 90.2	70.2 / 80.0 / 90.4	60.6 / 87.9 / <u>67.5</u>
Curvature [*]	73.0 / 76.1 / 91.2	73.7 / 72.9 / 42.9	58.0 / 88.7 / 70.8	52.7 / 90.4 / 20.1	53.7 / 85.3 / 73.5	54.5 / 93.3 / 90.6	75.4 / 76.0 / 91.7	63.0 / 83.3 / 68.7
Lateral Ratio [*]	71.3 / 82.6 / 90.7	70.4 / 80.1 / 39.8	53.9 / 90.4 / 67.5	54.2 / 93.1 / 21.6	53.0 / 86.2 / 74.1	56.0 / 94.4 / 90.0	71.0 / 82.0 / 90.8	61.4 / 87.0 / 67.8
NDR [*]	72.2 / 75.7 / 90.9	69.2 / 77.8 / 39.0	54.9 / 90.0 / 68.6	57.6 / 85.0 / 21.9	52.7 / 86.8 / 73.7	56.1 / 93.3 / 90.2	73.2 / 80.0 / 91.1	62.3 / 84.1 / 67.9

Table 1: **Correctness separability across seven benchmarks.** Each cell reports AUROC↑ / FPR95↓ / AUPR↑ (%); **Avg.** is the macro-average over the seven benchmarks. **Bold** marks the best score among our proposed methods (Curvature, Lateral Ratio, NDR); underline marks the best score among the baselines. Gray shading indicates our proposed methods. Method categories: ◦ = output-level, ◊ = trajectory baseline, * = ours. Per-dataset sample counts and accuracies are given in Appendix B (Table 9); paired-bootstrap tests for the macro-average gaps are in Appendix K.2 (Table 13).

4.1 Benchmarks, Models, and Evaluation

Datasets. GSM8K (Cobbe et al., 2021) contains grade-school arithmetic word problems. MATH (Hendrycks et al., 2021) contains competition-level mathematical problems. TheoremQA (Chen et al., 2023) contains theorem-centric STEM questions. MMLU (Hendrycks et al., 2020) contains multiple-choice questions across 57 subjects. CommonsenseQA (Talmor et al., 2019) contains multiple-choice commonsense questions. Belebele (Bandarkar et al., 2024) is a multilingual reading comprehension benchmark. MGSM (Shi et al., 2022) contains multilingual grade-school math word problems. Splits and evaluated sizes are listed in Appendix A.2 (Table 8), per-dataset accuracies and response lengths in Appendix B (Table 9), and artifact licenses in Appendix A.1.

Models. Our primary evaluation uses two frozen instruction-tuned decoder-only LLMs: Qwen2-7B-Instruct (Yang et al., 2024) and Llama-3-8B-Instruct (AI@Meta, 2024). For the model-scale analysis, we additionally evaluate Qwen2.5-Instruct at 3B, 14B, and 32B on GSM8K and MMLU (Section 5.3; Appendix K.1). Unless otherwise stated, visualizations use Qwen2-7B-Instruct. Both models use the same generation, logging, and scoring setup. We use zero-shot chain-of-thought prompting (Kojima et al., 2022): each prompt asks

the model to reason step by step and provide a final answer in a specified format. Prompting templates, answer extraction, and decoding details are in Appendices C–D, and the hardware and software environment in Appendix B.1.

Correctness labels and evaluation. Each generated response receives a binary correctness label $c \in \{0, 1\}$ through deterministic, dataset-specific answer verification: exact option matching for multiple-choice datasets and answer-type-specific normalization for open-answer datasets (Appendix C.2). Labels are used only after scoring. We report AUROC (↑) (Boyd et al., 2013), AUPR (↑) (Manning and Schutze, 1999), and FPR@95 (↓) (Hendrycks and Gimpel, 2016), together with macro-averages across benchmarks. The precise metric definitions and the orientation convention are given in Appendix E.

4.2 Baselines

Output-derived baselines. These methods use the next-token probability distribution without accessing hidden states. For each step t , let $p_t(\cdot)$ be the vocabulary distribution and y_t the generated token. We average over generated tokens to obtain $\mathbf{MaxProb} = \frac{1}{T} \sum_{t=1}^T \max_v p_t(v)$, $\mathbf{Entropy} = -\frac{1}{T} \sum_{t=1}^T \sum_v p_t(v) \log p_t(v)$, and $\mathbf{Perplexity} = \exp(-\frac{1}{T} \sum_{t=1}^T \log p_t(y_t))$.

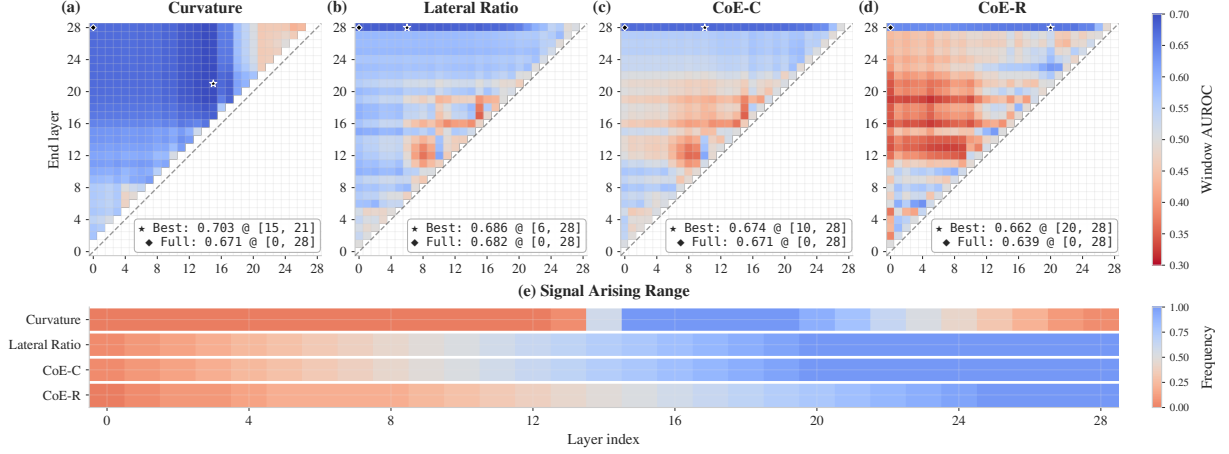


Figure 2: **Layer-window localization of trajectory scores (Qwen2-7B-Instruct).** For each contiguous layer window $[s_l, e_l]$, we recompute (a) Curvature, (b) Lateral Ratio, (c) CoE-C, and (d) CoE-R. $\star$ marks the best window; $\diamond$ marks the full-layer window $[0, L]$. (e) Per-layer frequency of appearing in the top 5% AUROC windows.

Trajectory-based baseline. Trajectory-based methods estimate correctness by summarizing hidden-state trajectories’ geometric properties, without training. The closest prior work is Chain-of-Embedding (CoE) (Wang et al., 2024), which combines magnitude and angle statistics of the token-mean trajectory (Eq. 2) into two composite scores. We define and analyze it alongside our proposed scores in Section 5.

5 Layer-wise Dynamics: From Trajectories to Norm Drop Rate

Takeaways. (i) Single-feature geometric scores can beat composite baselines. (ii) Correctness separability is localized in late layers. (iii) NDR captures this late-layer signal only using norm dynamics.

This section first shows that composite trajectory baselines can be matched by single-feature scores, then localizes where along layers these signals concentrate, and finally motivates our simplest score, NDR, from this localized pattern.

5.1 Trajectory as Diagnostic Scores

A layer trajectory is a high-dimensional vector sequence tracing a hidden representation from input to prediction. Since hidden states encode correctness-related information (Section 2), correct and incorrect generations leave different signatures along this trajectory, and each trajectory-based score captures one geometric facet of that difference. We use these scores as diagnostic probes

for which geometric properties carry correctness separability, and where along layers they concentrate. All scores are computed from the token-mean layer trajectory (Eq. 2).

CoE as a composite baseline. Let $M_l = \|\bar{h}^{l+1} - \bar{h}^l\|$, $A_l = \angle(\bar{h}^l, \bar{h}^{l+1})$, $Z_M = \|\bar{h}^L - \bar{h}^0\|$, and $Z_A = \angle(\bar{h}^0, \bar{h}^L)$. CoE (Wang et al., 2024) combines normalized magnitude and angle changes into two composite scores:

$$\text{CoE-R} = \frac{1}{L} \sum_{l=0}^{L-1} \left(\frac{M_l}{Z_M} - \frac{A_l}{Z_A} \right). \quad (3)$$

CoE-C maps each layer step to the complex plane:

$$\text{CoE-C} = \frac{1}{L} \left[\left(\sum_{l=0}^{L-1} \frac{M_l}{Z_M} \cos A_l \right)^2 + \left(\sum_{l=0}^{L-1} \frac{M_l}{Z_M} \sin A_l \right)^2 \right]^{1/2}. \quad (4)$$

CoE shows that geometric features can summarize a trajectory into an effective correctness signal: correct generations receive higher values than incorrect ones (Wang et al., 2024). However, by fusing magnitude and angle into a hand-crafted composite, it cannot reveal which property drives the signal. We therefore propose single-feature metrics, each isolating one facet of the dynamics.

Curvature. Curvature offers an alternative, single-feature characterization of the hidden-state dynamics, capturing how sharply the path bends locally. Following the standard definition of discrete curvature, and using central differences $v^{(l)} =$

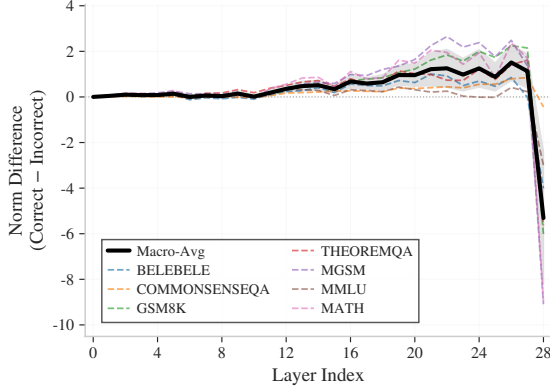


Figure 3: **Layer-wise norm difference.** $\Delta\|\bar{h}^l\|_2 = \mathbb{E}[\|\bar{h}^l\|_2 \mid c = 1] - \mathbb{E}[\|\bar{h}^l\|_2 \mid c = 0]$ across layers. Correct generations maintain larger norms through most layers, then show an abrupt final-layer contraction.

$(\bar{h}^{l+1} - \bar{h}^{l-1})/2$ and $a^{(l)} = \bar{h}^{l+1} - 2\bar{h}^l + \bar{h}^{l-1}$. Let $a_{\perp}^{(l)} = a^{(l)} - \frac{\langle a^{(l)}, v^{(l)} \rangle}{\|v^{(l)}\|^2} v^{(l)}$ be the acceleration component orthogonal to the velocity. We define

$$\text{Curv.} = \frac{1}{L-1} \sum_{l=1}^{L-1} \frac{\|a_{\perp}^{(l)}\|}{\|v^{(l)}\|^2 + \epsilon}. \quad (5)$$

Lower curvature indicates a smoother layerwise trajectory; consistent with this, correct generations tend to have lower curvature.

Lateral Ratio. Lateral Ratio captures a global property: how much the trajectory detours from the direct input-to-output path. Let $\hat{Z} = (\bar{h}^L - \bar{h}^0)/\|\bar{h}^L - \bar{h}^0\|$ be the net displacement direction, $\Delta^l = \bar{h}^{l+1} - \bar{h}^l$, and $\Delta_{\perp}^l = \Delta^l - \langle \Delta^l, \hat{Z} \rangle \hat{Z}$ the step component orthogonal to it. We define

$$\text{LateralRatio} = \frac{\sum_{l=0}^{L-1} \|\Delta_{\perp}^l\|}{\sum_{l=0}^{L-1} \|\Delta^l\| + \epsilon}. \quad (6)$$

A higher ratio indicates a less direct path; correct generations tend to have lower ratios.

Results. Table 1 shows that single-feature scores already match or exceed the composite CoE scores: on Qwen2-7B-Instruct, Lateral Ratio attains the best macro AUROC (68.2%) and Curvature the lowest macro FPR@95 (79.2%); on Llama-3-8B-Instruct, Curvature is strongest (63.0% AUROC). That a single feature score can match a hand-crafted composite suggests the current composite baselines carry redundancy.

5.2 Layer Localization

We next ask where along layers the signal concentrates. For each contiguous window $[s_l, e_l]$, we

Method	AUROC (%)		Drop (%)
	Raw	Length-Matched	
Token Length	67.1	56.7	15.5
CoE-R [◊]	63.9	61.5	3.8
CoE-C [◊]	67.1	64.2	4.3
Curvature [*]	67.1	60.6	9.7
Lateral Ratio [*]	68.2	64.9	4.8
NDR [*]	68.0	66.0	2.9

Table 2: **Length confound and robustness under length matching.** Macro AUROC on the original test set (Raw) and on a strictly length-matched subset. The length baseline collapses toward chance after matching, whereas NDR retains the highest matched AUROC (66.0) and the smallest relative degradation among trajectory scores. Results on Qwen2-7B-Instruct.

recompute each score on $(\bar{h}^l)_{l=s_l}^{e_l}$, with endpoint quantities recomputed inside the window.

Figure 2 shows that correctness signals are far from uniform. Endpoint-referenced scores (Lateral Ratio, CoE) are last-layer anchored: their best windows end near L , and removing late layers degrades performance sharply. Curvature peaks mid-to-late (best at $[15, 21]$). Some early windows yield AUROC below 0.5, meaning separability can flip across layers.

5.3 Norm Drop Rate

Final-layer norm contraction. Motivated by this last-layer concentration, we compare the layerwise norm $\|\bar{h}^l\|_2$ between correct and incorrect generations. Figure 3 reveals a simple yet consistent pattern: correct generations maintain larger token-mean norms through most layers, followed by a sharp contraction at the final layer that flips the sign of the difference.

Definition. We define the **Norm Drop Rate** as

$$\text{NDR} = \frac{\frac{1}{L+1} \sum_{l=0}^L \|\bar{h}^l\|_2}{\|\bar{h}^L\|_2}. \quad (7)$$

A larger NDR indicates stronger final-layer norm contraction relative to the mean norm across layer trajectory.

Results. On Qwen2-7B-Instruct, NDR attains the best macro AUPR (78.3%) and 68.0% macro AUROC, above CoE-C (76.8% AUPR) and all output-level baselines (Table 1a). On Llama-3-8B-Instruct, NDR reaches 62.3% macro AUROC, competitive with the strongest score, Curvature (63.0%). Across both models, NDR ranks among

Token region	Qwen2-7B	Llama-3-8B
Full response	68.0	62.3
Reasoning only	67.1	61.2
Answer only	60.3	55.9
Exclude final token	67.5	61.9

Table 3: **NDR across token regions.** Macro AUROC (%) over all seven benchmarks. On both models, reasoning-only NDR stays within about one point of the full response, whereas answer-only NDR drops by 6–8 points; excluding the final generated token has little effect. The answer boundary and token masks are defined in Appendix J.2.

the top two methods in macro AUROC while being the simplest trajectory-based score. Sections 6 and 7 further test its robustness.

Consistency across scale. To test whether the pattern is specific to 7–8B models, we evaluate Qwen2.5-Instruct at 3B, 14B, and 32B on GSM8K and MMLU. NDR remains competitive at all three scales. Full results are in Appendix K.1 (Table 11).

Relation to output confidence. Since logits are a linear projection of the final hidden state, NDR overlaps partly with output uncertainty (Spearman $\rho = -0.50$ with entropy on Qwen2). After controlling for output-level confidence, a complementary residual signal remains, indicating that NDR is not reducible to output uncertainty; residual-control and feature-complementarity analyses are reported in Appendix J.3.

6 Token-wise Dynamics: Length Confounds and Failure Spans

Takeaways. (i) Length is a strong but insufficient baseline. (ii) NDR is the most length-stable score after strict matching. (iii) Correctness signals mainly localize to hallucination token spans and reasoning token spans.

The layer analysis averages over all generated tokens, which can hide whether the score is confounded by output length and which response token spans drive it. We first control for these confounds, then localize the signal within the response.

6.1 Signal beyond Response Length

Response length alone can serve as a scalar score: models tend to hesitate on problems they get wrong, producing longer answers. Ranking by response

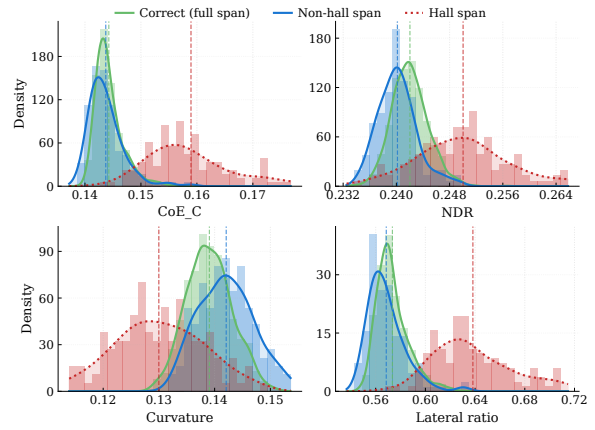


Figure 4: **Geometric signal localization to hallucinated spans (GSM8K).** Distributions of trajectory scores computed on span-mean trajectories. **Green:** correct responses (full span; 1099 samples). **Red:** hallucinated spans $\mathcal{T}_{\text{hall}}$ within incorrect responses (137 samples). **Blue:** non-hallucinated spans $\mathcal{T}_{\neg\text{hall}}$ from the same 137 responses. Hallucinated spans are separable from the correct regime; non-hallucinated spans largely overlap with correct responses.

length alone reaches 67.1% macro AUROC (Table 2), above every output-level method and on par with the trajectory baselines. Recent work finds such length statistics can rival far more elaborate hallucination detectors (Janiak et al., 2025), making length a necessary confound to control. We therefore restrict evaluation on samples with similar length and ask how much of each trajectory score’s separability survives.

For each incorrect sample i with token length t_i , we select a correct sample with nearest length t_j (minimizing $|t_i - t_j|$), form matched pairs per dataset, and recompute AUROC on this subset. Matching is tight: the median length difference is zero across datasets (e.g., GSM8K yields $k = 220$ pairs).

After matching, the length baseline collapses toward chance ($67.1 \rightarrow 56.7$). NDR retains 97.1% of its raw AUROC ($68.0 \rightarrow 66.0$), the smallest relative degradation in Table 2. CoE-R and CoE-C drop 3.8% and 4.3%, Lateral Ratio 4.8%, and Curvature 9.7%. This rules out the simplest explanation that NDR merely tracks response length; jointly regressing out entropy, perplexity, and response length further leaves a residual macro AUROC of 57.2 (Qwen2) and 60.3 (Llama-3), so the signal is not reducible to output-level confidence either (Appendix J.3, Table 14).

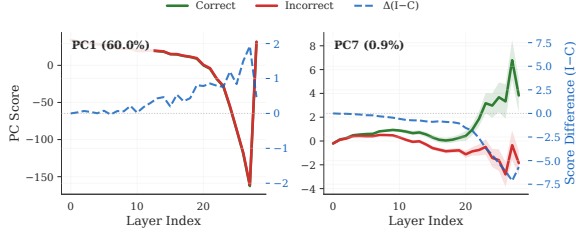


Figure 5: **PC trajectories can separate correctness off the top-variance axes (TheoremQA).** PC1 (60.0% variance) shows overlapping trajectories for correct and incorrect generations; PC7 (0.9% variance) diverges in late layers with large $\Delta(I - C)$.

6.2 Reasoning Tokens Drive the Signal

Length matching removes length as a confound and confirms that the trajectory signal still retains predictive information. This leaves a further question: where along the token span does this signal reside? Since all our metrics are computed through a token mean, the signal could either be distributed across the model’s reasoning or emerge only at the final answer. We therefore recompute NDR by averaging over different token spans (Table 3). On both models, using only the reasoning tokens preserves almost all of the full-response signal (within one AUROC point), while using only the answer tokens is much weaker (6–8 points lower), and dropping the last generated token barely changes the score. The correctness signal therefore builds up across the reasoning rather than appearing only at the final answer.

6.3 Signal Concentrates in Hallucinated Spans

The previous section showed the signal is mostly carried by the reasoning span as a whole; we now ask which tokens drive it. One hypothesis is that the signal depends on token type, specifically, whether a token belongs to an erroneous span. We therefore localize the signal to specific error regions within each response.

For each incorrect response on GSM8K, we use a stronger LLM as a judge² to annotate hallucination spans $\mathcal{T}_{\text{hall}}$; the remaining tokens form the non-hallucinated span $\mathcal{T}_{\text{hall}}$. These annotations are used for analysis only and are never used to compute any score. We then define three groups: (i) correct responses using the full token span, (ii)

²We use GPT-4o to annotate token spans containing factual errors or unsupported logical steps; details are described in Appendix G.

Selection Method		Macro AUROC
Best PC	Best Layer	72.4
Best PC	Layer Mean	67.2
Best PC	Last Layer	66.8
Naive Sum PC Last Layer		59.6

Table 4: **The PCA selection gap.** Macro AUROC (%) over seven benchmarks, Qwen2-7B-Instruct. Label-selected PCs achieve high separability, but naive unsupervised aggregation misses much of the signal.

hallucinated spans from incorrect responses, and (iii) non-hallucinated spans from the same incorrect responses. For each group, we recompute the span-mean trajectory $\bar{h}_{\mathcal{T}}^l$ and apply the same geometric scores.

Figure 4 compares the three groups. Hallucinated spans deviate strongly from the correct distribution, while non-hallucinated spans from the same incorrect responses largely overlap the correct regime. This explains why full-response NDR remains effective despite token averaging: a short failure span with abnormal norm dynamics can shift the token-mean trajectory enough to affect the final-layer contraction ratio.

7 Representation Dynamics: Basis Stability

Takeaways. (i) PCA can capture correctness signals, but the useful component is dataset-dependent and requires labels to select. (ii) NDR is the most stable score under basis changes (2.9 pp spread across four bases). (iii) NDR is also the most efficient, requiring only layerwise norms.

Sections 5 and 6 analyze correctness signals in the native hidden space. This section tests whether those signals depend on the choice of coordinate system.

7.1 PCA Reveals a Selection Bottleneck

Hidden representations are organized differently in the latent space for correct and incorrect responses, so the directions along which they vary should carry information about correctness. PCA is a direct way to expose the principal directions of variation in the hidden space, and we test whether these directions align with the correctness signal.

To test whether a latent subspace exposes additional structure, we fit PCA per dataset using token-mean states $\{\bar{h}^l\}$ as samples across examples

Metric	Hidden	LM Head	Logit	PCA	Spread
CoE-R [◊]	63.9	63.6	64.5	58.8	5.7
CoE-C [◊]	67.1	63.3	55.4	67.6	12.2
Curvature*	67.1	65.9	62.4	68.3	5.9
Lateral Ratio*	68.2	62.4	55.5	68.3	12.8
NDR*	68.0	65.1	66.9	66.1	2.9

Table 5: **Trajectory scores across representation bases.** Macro AUROC over seven datasets (%). Spread = max – min AUROC across bases (smaller = more basis-stable). **Bold** marks the smallest spread. PCA uses $k = 512$.

and layers, then project onto the top- k components: $\tilde{h}^l = U_k^\top (\bar{h}^l - \mu)$, $U_k \in \mathbb{R}^{d \times k}$. Each coordinate $\tilde{h}_j^l$ forms a one-dimensional curve over depth.

A single PC can separate correct from incorrect generations, but the discriminative PC is not reliably the high-variance one and does not appear at consistent ranks across datasets. Figure 5 illustrates: on TheoremQA, PC1 explains 60.0% of the variance but shows overlapping trajectories, while PC7 explains only 0.9% yet exhibits strong late-layer separation.

Table 4 quantifies this selection bottleneck. With label-selected PCs and layers, PCA reaches 72.4% macro AUROC, well above any fully unsupervised score. But naive aggregation (summing all PCs at the last layer) drops to 59.6%, below NDR’s 68.0% on the same Qwen2-7B model. The bottleneck is not signal strength but unsupervised selection: without labels, there is no reliable way to identify which PC carries the signal. NDR sidesteps this entirely by summarizing the trajectory through norms rather than basis-sensitive directions, while maintaining competitive performance.

7.2 NDR Is Stable Across Representation Bases

We next test whether the same trajectory scores remain stable under alternative representation bases. We compare four bases: **Hidden space** (native coordinates, the default); **PCA** (dataset-specific top- k components, $k = 512$); **LM-head SVD** (top- k right-singular directions of W_U); and **Logit space** ($z = W_U h$).

Table 5 shows that NDR is the most basis-stable score: its AUROC ranges from 65.1 to 68.0, a 2.9 percentage-point spread, compared with 12.8 for Lateral Ratio and 12.2 for CoE-C. Shape-sensitive scores degrade sharply under output-aligned projections: Lateral Ratio drops from 68.2 in hidden space to 55.5 in logit space, while NDR remains

Metric	Time (ms)	Mem. (MB)
NDR*	0.076	0.028
CoE/Curvature/Lateral Ratio	0.349–0.378	0.8–1.6

Table 6: **Score-computation cost.** Per-response time and peak temporary memory on cached hidden states; the shared forward pass and hidden-state storage are excluded. The second row gives the range over the three compared trajectory scores.

at 66.9. NDR depends only on L2 norms, which degrade more gracefully than angle-based shape statistics under non-orthogonal projections. Native hidden space is the strongest default overall, but NDR is the score least affected by departing from it.

Computation Efficiency. Beyond basis stability, NDR is also the cheapest trajectory-based score to compute, requiring only $L+1$ layerwise ℓ_2 norms and a single scalar average. This makes it roughly $5\times$ faster and $29\text{--}57\times$ lighter in peak temporary memory than the other trajectory scores; see Table 6 for the measurements and Appendix K.3 for the computation details.

8 Conclusion

We studied geometric dynamics in LLM hidden representations for training-free correctness self-evaluation, organizing the analysis along three dimensions: layer depth, token span, and representation space. Across benchmarks and models, correctness signals concentrate in late-stage computation, and many complex trajectory scores reduce to simple statistics on restricted layer regions. This localization leads to NDR, a minimal score based on final-layer norm contraction that matches or exceeds more complex methods while being the most length-robust, basis-stable, and computationally lightweight score in our comparison. Token-span analysis further shows that geometric deviations localize to hallucinated regions, and representation analysis confirms that NDR is the least sensitive to coordinate changes. Extending this localization-then-distillation approach to broader reliability targets such as faithfulness and safety, to additional model families and architectures, and to earlier detection during generation are natural next steps. These results suggest that hidden-state trajectory analysis should prioritize localizing simple, targeted signals over designing increasingly complex summaries.

Limitations

Model coverage. Our primary evaluation covers Qwen2-7B-Instruct and Llama-3-8B-Instruct across seven benchmarks. The additional scale analysis evaluates Qwen2.5-Instruct at 3B, 14B, and 32B on GSM8K and MMLU. This provides within-family evidence across model scales, but does not establish cross-family generality at larger scales, coverage of other architectures, or robustness to fine-tuning regimes beyond instruction tuning.

Inference protocol. All experiments use greedy decoding with one response per prompt. Trajectory geometry may change under temperature sampling, beam search, or tool-augmented decoding. Extending the analysis to stochastic decoding regimes is a natural next step.

Hallucination span annotations. Token-span localization relies on post-hoc annotations from a stronger LLM. Such annotations may be noisy for subtle logical errors or partially correct reasoning. The qualitative separation between hallucinated and non-hallucinated spans is robust in our setting, but the exact degree of localization may vary with alternative annotation protocols.

Ethical Considerations

This work studies offline correctness-evaluation signals from hidden states of public LLM checkpoints on public benchmarks. We do not collect private user data, train new models, or conduct human-subject experiments. The main risk is over-reliance: a geometric score such as NDR may make an incorrect answer appear more reliable than it is. We therefore present the method as a diagnostic signal rather than a calibrated guarantee of correctness, and recommend additional task-specific validation for high-stakes use cases.

Acknowledgment

Jiawei Zhou is partially supported by a Coefficient Giving Technical AI Safety Research grant.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

AI@Meta. 2024. [Llama 3 model card](#).

Amos Azaria and Tom Mitchell. 2023. [The internal state of an LLM knows when it’s lying](#). In *Findings of the Association for Computational Linguistics: EMNLP*, pages 967–976, Singapore.

Lucas Bandarkar, Davis Liang, Benjamin Muller, Mikel Artetxe, Satya Narayan Shukla, Donald Husa, Naman Goyal, Abhinandan Krishnan, Luke Zettlemoyer, and Madian Khabsa. 2024. The belebele benchmark: a parallel reading comprehension dataset in 122 language variants. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 749–775.

Nora Belrose, Igor Ostrovsky, Lev McKinney, Zach Furman, Logan Smith, Danny Halawi, Stella Biderman, and Jacob Steinhardt. 2023. [Eliciting latent predictions from transformers with the tuned lens](#). *arXiv preprint arXiv:2303.08112*.

Kendrick Boyd, Kevin H Eng, and C David Page. 2013. Area under the precision-recall curve: point estimates and confidence intervals. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 451–466. Springer.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. 2022. [Discovering latent knowledge in language models without supervision](#). *arXiv preprint arXiv:2212.03827*.

Ruidi Chang, Jiawei Zhou, and Hanjie Chen. 2026. Prism: A dual view of llm reasoning through semantic flow and latent computation. *arXiv preprint arXiv:2603.22754*.

Wenhu Chen, Ming Yin, Max Ku, Pan Lu, Yixin Wan, Xueguang Ma, Jianyu Xu, Xinyi Wang, and Tony Xia. 2023. Theoremqa: A theorem-driven question answering dataset. *arXiv preprint arXiv:2305.12524*.

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. *arXiv preprint arXiv:2110.14168*.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630.

Danny Halawi, Jean-Stanislas Denain, and Jacob Steinhardt. 2024. [Overthinking the truth: Understanding how language models process false demonstrations](#). In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

- 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*.
- Dan Hendrycks and Kevin Gimpel. 2016. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*.
- Denis Janiak, Jakub Binkowski, Albert Sawczyn, Bogdan Gabrys, Ravid Shwartz-Ziv, and Tomasz Jan Kajdanowicz. 2025. The illusion of progress: Re-evaluating hallucination detection in llms. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 34716–34733.
- Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, and 1 others. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.
- Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.
- Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *arXiv preprint arXiv:2302.09664*.
- Vedang Lad, Jin Hwa Lee, Wes Gurnee, and Max Tegmark. 2026. Remarkable robustness of llms: Stages of inference? *Advances in Neural Information Processing Systems*, 38:130050–130083.
- Yanhong Li, Ming Li, Karen Livescu, and Jiawei Zhou. 2026. On the predictive power of representation dispersion in language models. In *International Conference on Learning Representations*, volume 2026, pages 57796–57836.
- Junteng Liu, Shiqi Chen, Yu Cheng, and Junxian He. 2024. [On the universal truthfulness hyperplane inside LLMs](#). In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Miami, Florida, USA.
- Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 9004–9017.
- Christopher Manning and Hinrich Schutze. 1999. *Foundations of statistical natural language processing*. MIT press.
- Rob Manson. 2025. [Curved inference: Concern-sensitive geometry in large language model residual streams](#). *arXiv preprint arXiv:2507.21107*.
- Samuel Marks and Max Tegmark. 2023. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. *arXiv preprint arXiv: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 (NeurIPS)*, pages 17359–17372.
- Albert Miao, Chenliang Zhou, Jiawei Zhou, and Cengiz Oztireli. 2026. Features emerge as discrete states: The first application of saes to 3d representations. In *International Conference on Learning Representations*, volume 2026, pages 131229–131248.
- nostalgebraist. 2020. Interpreting GPT: The logit lens. <https://www.lesswrong.com/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens>. Accessed: 2024-01-01.
- Charles O’Neill, Slava Chalnev, Chi Chi Zhao, Max Kirkby, and Mudith Jayasekara. 2025. [A single direction of truth: An observer model’s linear residual probe exposes and steers contextual hallucinations](#). *arXiv preprint arXiv:2507.23221*.
- Hadas Orgad, Michael Toker, Zorik Gekhman, Roi Reichart, Idan Szpektor, Hadas Kotek, and Yonatan Belinkov. 2025. [LLMs know more than they show: On the intrinsic representation of LLM hallucinations](#). In *Proceedings of the International Conference on Learning Representations (ICLR)*.
- Seongheon Park, Xuefeng Du, Min-Hsuan Yeh, Haobo Wang, and Yixuan Li. 2025. [Steer LLM latents for hallucination detection](#). In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 47971–47990.
- Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2025. [Qwen2.5 technical report](#). *Preprint*, arXiv:2412.15115.
- Adam S. Shai, Sarah E. Marzen, Lucas Teixeira, Alexander Gietelink Oldenziel, and Paul M. Riechers. 2024. [Transformers represent belief state geometry in their residual stream](#). *arXiv preprint arXiv:2405.15943*.
- Freda Shi, Mirac Suzgun, Markus Freitag, Xuezhi Wang, Suraj Srivats, Soroush Vosoughi, Hyung Won Chung, Yi Tay, Sebastian Ruder, Denny Zhou, and 1 others. 2022. Language models are multilingual chain-of-thought reasoners. *arXiv preprint arXiv:2210.03057*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. [CommonsenseQA: A question answering challenge targeting commonsense knowledge](#). In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics – Human Language Technologies (NAACL HLT)*, pages 4149–4158, Minneapolis, Minnesota.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. [Attention is all you need](#). In *Advances in Neural Information Processing Systems (NeurIPS)*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Yiming Wang, Pei Zhang, Baosong Yang, Derek F Wong, and Rui Wang. 2024. Latent space chain-of-embedding enables output-free llm self-evaluation. *arXiv preprint arXiv:2410.13640*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed 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 (NeurIPS)*, pages 24824–24837.

Sarah Wiegrefe, Oyvind Tafjord, Yonatan Belinkov, Hanna Hajishirzi, and Ashish Sabharwal. 2025. Answer, assemble, ace: Understanding how llms answer multiple choice questions. In *International Conference on Learning Representations*, volume 2025, pages 78293–78318.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, and 43 others. 2024. [Qwen2 technical report](#). *Preprint*, arXiv:2407.10671.

Fred Zhang and Neel Nanda. 2024. [Towards best practices of activation patching in language models: Metrics and methods](#). In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xu Wang 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](#). *arXiv preprint arXiv:2310.01405*.

A Responsible NLP and Reproducibility Details

This appendix documents the artifacts, execution environment, and protocol details needed to reproduce the reported analyses. The study uses public benchmark datasets and public instruction-tuned model checkpoints. We do not train new models, collect new human-subject data, or use private user data. The only additional annotations are post-hoc hallucination-span annotations produced for analysis, not for model training or score selection.

A.1 Artifact summary, licenses, and intended use

Table 7 summarizes the datasets, models, and derived artifacts used in the paper. Datasets and model checkpoints are obtained from their official releases and remain subject to the access and redistribution terms of their original providers. We release only derived predictions, scalar metrics, and scripts when allowed by the corresponding artifact terms; raw benchmark data and model weights should be obtained from the original providers.

Intended use. The benchmark datasets are used only for offline evaluation of correctness-detection signals. The model checkpoints are used only for deterministic inference and hidden-state extraction. NDR and the other geometric scores are diagnostic signals for model self-evaluation; they are not intended as stand-alone safety filters for high-stakes domains. A high score should not be interpreted as a guarantee of factual correctness.

Privacy and offensive content. We do not collect new personally identifiable information. All inputs come from public benchmarks. We do not perform additional filtering for offensive content beyond the filtering and documentation provided by the original benchmark creators. Generated model outputs may contain erroneous, biased, or unsafe content; the experiments analyze such outputs offline and do not expose them to users.

Responsible use risk. The main risk is over-reliance: a training-free hidden-state score can make an incorrect response appear more reliable than it is. We therefore report AUROC, AUPR, and FPR@95 rather than presenting the method as a calibrated correctness guarantee. In deployment,

Artifact	Type	Creator / citation	Use in this work	License
GSM8K	Dataset	Cobbe et al. (2021)	Grade-school math reasoning; correctness labels from final numerical answers.	MIT
MATH	Dataset	Hendrycks et al. (2021)	Competition mathematics; final-answer extraction with numerical/symbolic normalization.	MIT
TheoremQA	Dataset	Chen et al. (2023)	Theorem-grounded QA; answer format depends on answer type.	MIT
MMLU	Dataset	Hendrycks et al. (2020)	Multi-domain multiple-choice knowledge and reasoning.	MIT
CommonsenseQA	Dataset	Talmor et al. (2019)	Multiple-choice commonsense reasoning.	MIT
Belebele	Dataset	Bandarkar et al. (2024)	Multilingual reading comprehension.	CC BY-SA 4.0
MGSM	Dataset	Shi et al. (2022)	Multilingual grade-school math reasoning.	CC BY-SA 4.0
Qwen2-7B-Instruct	Model	Yang et al. (2024)	Generation and hidden-state logging.	Apache 2.0
Llama-3-8B-Instruct	Model	AI@Meta (2024)	Generation and hidden-state logging.	Llama 3 Community
Qwen2.5-Instruct (3B/14B/32B)	Model	Qwen et al. (2025)	Within-family scale analysis (Appendix K.1).	Apache 2.0
Hidden-state tensors	Derived	This work	Cached activations used to compute all geometric scores.	Not released
Hallucination spans	Derived	This work	Analysis-only span localization; never used to compute any score.	Released with code
Evaluation code	Derived	This work	Generation, logging, and scoring pipeline.	MIT

Table 7: Artifacts used in this work. All datasets and checkpoints are used within their original license terms for non-commercial research evaluation.

these scores should be combined with task-specific validation, uncertainty calibration, and human review where the cost of an error is high.

A.2 Dataset coverage and splits

Table 8 gives the benchmark coverage used for evaluation. For each dataset–model pair, correctness is defined after answer extraction against the benchmark-provided gold answer. Dataset-specific parsers are described in Appendix C.

Multilingual coverage. MGSM is evaluated in the standard language set: English plus Bengali, Chinese, French, German, Japanese, Russian, Spanish, Swahili, Telugu, and Thai. Belebele is evaluated using the language variants included in the benchmark release; when all variants are used, this corresponds to 122 language variants with 900 questions per variant. We do not translate prompts or answers ourselves.

B Dataset Statistics

Table 9 reports per-dataset statistics for Qwen2-7B-Instruct, including sample counts, accuracy, and

mean response lengths for correct and incorrect generations.

Incorrect responses are consistently longer than correct ones across all seven datasets, with the gap most pronounced on MATH (374 vs. 713 tokens). This systematic length difference motivates the length-matched evaluation in Section 6 and confirms that response length is a genuine confound that must be controlled for.

B.1 Models, hardware, and software

Table 10 summarizes the model and compute setup. All experiments are inference-only. No gradient updates, supervised probes, LoRA adapters, or verifier models are trained.

Software environment. The experiments require PyTorch, Hugging Face transformers, datasets, accelerate, NumPy, SciPy, scikit-learn, and a tokenizer implementation matching each model checkpoint. The accompanying source release records the software environment used for the experiments. the metrics reported in the paper are deterministic once model checkpoint, tokenizer version, prompts, and generation settings are fixed.

Dataset	Language coverage	Split / subset	Public size	benchmark	Correctness criterion
GSM8K	English	Test	1,319 examples		Exact final numerical answer after normalization.
MATH	English	Test	5,000 test examples		Final boxed or extracted mathematical answer after normalization.
TheoremQA	English	Evaluation set	800 QA pairs		Match to answer type: boolean, option, number, expression, list, or text.
MMLU	English	Test / evaluation subjects	57 subjects; roughly 14k test questions		Predicted multiple-choice option equals gold option.
CommonsenseQA	English	Development / validation	1,221 labeled validation examples		Predicted multiple-choice option equals gold option.
Belebele	Multilingual	Evaluated language variants	900 questions per language variant; 122 variants in the full release		Predicted multiple-choice option equals gold option.
MGSM	Multilingual	Test	250 problems per language; 11 languages including English		Exact final numerical answer after normalization.

Table 8: Benchmark coverage.

Dataset	N	Correct	Acc.%	Avg. Len	
				C	I
Belebele	900	797	88.6	158	223
CSQA	1221	947	77.6	171	188
GSM8K	1319	1099	83.3	217	285
MATH	5000	2605	52.1	374	713
MGSM	250	215	86.0	214	288
MMLU	14042	9697	69.1	137	167
TheoremQA	221	61	27.6	347	485

Table 9: **Dataset statistics for Qwen2-7B-Instruct.** N = total samples, C/I = correct/incorrect count, Avg. Len = mean response length in tokens. Incorrect responses are systematically longer across all datasets, motivating the length-matched evaluation in Section 6.

C Prompting and Answer Extraction

We use dataset-specific prompts that enforce a standardized final-answer format while allowing the model to generate free-form reasoning, consistent with chain-of-thought style prompting (Wei et al., 2022). Each dataset specifies a required final-line marker such as Answer:, Answer: A, or boxed{ }. The parser only uses the final answer string; intermediate reasoning is not used to decide correctness except in the post-hoc hallucination-span analysis of Appendix G.

C.1 Prompt templates

For mathematical and open-answer datasets, we use the following template:

Question: <question>
Instruction: Solve the problem. Put the final answer on the last line in the format Answer: <answer>.

For multiple-choice datasets, we use:

Question: <question>
Options: (A) <option A> ...
Instruction: Choose the best answer. Put only the final option letter on the last line in the format Answer: <A/B/C/D/...>.

For TheoremQA, the instruction additionally includes the answer type when provided by the dataset. For multilingual datasets, the question and options are kept in the benchmark language; the final-answer marker is kept fixed so that answer extraction is uniform.

C.2 Extraction and normalization

We parse the final answer using deterministic dataset-specific rules:

- **Numerical answers.** We remove commas, trailing punctuation, currency symbols, and surrounding explanatory text. Fractions and decimals are normalized when possible. A prediction is correct if the normalized value matches the gold answer under the dataset-specific tolerance.
- **Mathematical expressions.** We first extract a boxed answer if present, then fall back to the final Answer: line. We normalize whitespace, punctuation, and common LaTeX wrappers before comparison.
- **Multiple choice.** We extract the final option letter from the required final line. If multiple option letters appear, the last explicitly marked final answer is used.

Component	Setting	Hardware	Notes
Qwen2-7B-Instruct inference	bfloat16, no quantization	NVIDIA A100	Hidden states are logged during greedy generation.
Llama-3-8B-Instruct inference	bfloat16, no quantization	NVIDIA A100	Same decoding and logging pipeline as Qwen2.
Metric computation	NumPy / PyTorch / scikit-learn	CPU	AUROC, AUPR, and FPR@95 are computed from cached scalar scores.
PCA / SVD projections	Centered activations; truncated bases where specified	CPU or A100	Projection bases are fit without correctness labels unless explicitly marked as oracle analysis.
Hallucination-span pre-processing	JSON parsing and tokenizer offset alignment	CPU	Stronger-LLM annotations are analysis-only.

Table 10: Model and compute setup. All experiments are inference-only; no gradient updates or learned probes are involved.

- **Boolean and list answers.** Boolean strings are lowercased and mapped to canonical true/false labels. Lists are normalized by delimiter, whitespace, and ordering only when the dataset defines the answer as unordered.

Examples that do not yield a valid final answer are counted as incorrect. The same extraction rules are applied to both models and all compared scores.

D Decoding and Hidden-State Logging

We generate exactly one response per prompt using greedy decoding: temperature = 0, sampling disabled, one returned sequence, and a maximum of 2048 new tokens. Generation stops at the model’s end-of-sequence token or the maximum length. This single-pass protocol is important because the compared scores are intended for label-free self-evaluation from one generated response, not from multiple sampled responses.

For each generated token, we store the residual hidden state at every transformer layer. Let $h_t^l \in \mathbb{R}^d$ denote the hidden state for generated token $t \in \{1, \dots, T\}$ at layer $l \in \{0, \dots, L\}$. Unless otherwise stated, prompt-token states are excluded and scores are computed only over generated tokens. The default token-mean representation at layer l is

$$\bar{h}^l = \frac{1}{T} \sum_{t=1}^T h_t^l. \quad (8)$$

All geometric scores in the main paper are computed from these stored hidden states and never from training labels.

E Metrics and Score Orientation

All scalar scores are oriented so that larger values indicate a higher predicted probability of correctness. AUROC and AUPR are computed with

correctness as the positive class. FPR@95 is the false-positive rate among incorrect responses at the threshold where the true-positive rate on correct responses is 95%. For scores whose natural direction is reversed, such as perplexity or entropy, we multiply by -1 before computing threshold-based metrics.

For output-level baselines, we use the final generated answer tokens and the model’s own token probabilities. For hidden-state scores, we use the tensor $\{h_t^l\}_{t,l}$ logged during the same generation. No score uses gold labels during computation. Labels are used only for evaluation metrics and for defining diagnostic subsets such as length-matched pairs or hallucination-span groups.

F Length-Matching Protocol

Response length is a strong confound for correctness detection, so we evaluate whether hidden-state geometry remains informative after controlling for length. For each dataset and model, let T_i be the number of generated tokens for example i . For each incorrect response, we select a correct response from the same dataset whose length is closest in absolute difference:

$$j^*(i) = \arg \min_{j:c_j=1} |T_i - T_j|, \quad (9)$$

where $c_j = 1$ denotes a correct response. Ties are broken deterministically by example index. Evaluation is then restricted to the matched correct/incorrect subset. This procedure uses correctness labels only to construct the diagnostic evaluation subset; it does not alter the hidden-state scores.

Matching statistics are computed from the same evaluated responses used for the corresponding robustness analysis. The GSM8K analysis in the main text follows this same protocol.

Model	MaxProb	Entropy	PPL	CoE-R	CoE-C	Curvature	Lat. Ratio	NDR
Qwen2.5-3B-Instruct	56.8	58.1	56.6	55.3	59.7	62.8	63.0	63.0
Qwen2.5-14B-Instruct	61.9	63.8	62.4	64.4	66.7	68.2	67.8	69.3
Qwen2.5-32B-Instruct	61.7	64.5	60.3	64.8	66.5	69.1	65.7	68.4

Table 11: Macro AUROC across GSM8K and MMLU for the Qwen2.5-Instruct scale analysis. The table tests within-family scale consistency rather than cross-family generalization.

G Hallucination-Span Annotation Protocol

The token-span analysis in Section 6 uses post-hoc hallucination-span annotations only for diagnostic visualization and group comparison. These annotations are not used to compute NDR, CoE, curvature, output baselines, or any other self-evaluation score.

G.1 Judge input and output

For each incorrect response selected for span analysis, we provide a stronger LLM judge with the benchmark question, the gold answer, the generated response, and the extracted final answer. The judge is asked to identify minimal contiguous spans in the generated response that contain unsupported factual claims, invalid mathematical steps, or logically decisive errors. The judge returns JSON with the following fields:

```
{
  "is_incorrect": true/false,
  "spans": [{
    "start_text": "...",
    "end_text": "...",
    "error_type": "unsupported_claim
                  | invalid_step | arithmetic |
                  contradiction | other",
    "rationale": "brief explanation"
  }]
}
```

The judge temperature is set to zero. If the judge output is not valid JSON or cannot be aligned back to the generated text, the example is excluded from the span-level analysis but remains part of the main correctness-evaluation experiments.

G.2 Span definition and token alignment

A hallucination span is the smallest contiguous text region that makes the response unsupported or wrong relative to the question and gold answer. For mathematical reasoning, the span can be an invalid equation, an arithmetic error, a false inference, or a step that irreversibly leads to the wrong final answer. For factual or commonsense reasoning, the span can be an unsupported claim, contradiction,

or answer option justification that is inconsistent with the gold label.

We map character spans to model tokens with tokenizer offset alignment. Tokens whose character offsets overlap a hallucination span are assigned to $\mathcal{T}_{\text{hall}}$; all remaining generated tokens in the same incorrect response are assigned to $\mathcal{T}_{\neg\text{hall}}$. For each span group, we recompute the layer trajectory by replacing Eq. 8 with

$$\bar{h}_{\mathcal{T}}^l = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} h_t^l. \quad (10)$$

The same geometric scores are then applied to the span-mean trajectories.

G.3 Bias and failure modes

The stronger-LLM judge can miss subtle reasoning errors, mark too broad a span, or confuse a downstream consequence with the first erroneous step. The span analysis should therefore be interpreted qualitatively: it tests whether geometric deviations concentrate in tokens identified as erroneous by a consistent protocol, not whether the judge provides a definitive theory of hallucination. The main paper keeps all primary correctness-detection results independent of these annotations.

H Norm Drop Rate: Final-Layer Contraction Analysis

This section expands the final-layer explanation referenced in Section 5.3. Let h^l denote a token-mean residual state at layer l , and let $f^l(h^l)$ be the update applied by block l . The residual recurrence is

$$h^{l+1} = h^l + f^l(h^l). \quad (11)$$

The squared norm after the update is

$$\|h^{l+1}\|_2^2 = \|h^l\|_2^2 + \|f^l(h^l)\|_2^2 + 2\langle h^l, f^l(h^l) \rangle. \quad (12)$$

Thus the norm decreases when the update is sufficiently anti-aligned with the current state:

$$\langle h^l, f^l(h^l) \rangle < -\frac{1}{2}\|f^l(h^l)\|_2^2. \quad (13)$$

Dataset	Correct	Incorr.	Δ	p
Belebele	-0.768	-0.759	-0.010	4e-5
CSQA	-0.740	-0.739	-0.001	0.28
GSM8K	-0.813	-0.795	-0.019	9e-18
MATH	-0.819	-0.796	-0.023	<1e-99
MGSM	-0.812	-0.789	-0.023	7e-6
MMLU	-0.739	-0.729	-0.009	1e-42
TheoremQA	-0.789	-0.781	-0.008	0.046
Macro	-0.783	-0.770	-0.013	—

Table 12: **Final-layer cosine alignment (Qwen2-7B-Instruct).** $\cos \theta_L = \cos(f^L(h^{L-1}), h^{L-1})$ averaged over correct and incorrect generations; more negative indicates stronger anti-alignment. Correct generations are more anti-aligned in 6/7 datasets ($p < 0.05$).

Empirical verification. We compute the cosine similarity $\cos \theta_L = \langle h^{L-1}, f^L(h^{L-1}) \rangle / (\|h^{L-1}\| \|f^L(h^{L-1})\|)$ for each sample on Qwen2-7B-Instruct and compare correct versus incorrect generations. Table 12 reports the results per dataset.

Across six of seven benchmarks, the final-layer update is significantly more anti-aligned for correct generations. The effect size is small in absolute terms ($|\Delta| \approx 0.01$ – 0.02), which is consistent with the fact that NDR captures a subtle but systematic signal. CommonsenseQA is the only dataset without a significant difference ($p = 0.28$), likely because its short multiple-choice format produces less variation in trajectory dynamics. The consistency of the direction across the remaining datasets supports the interpretation that final-layer norm contraction is a reliable correlate of correctness.

Why not use the cosine directly? The final-update cosine is useful for interpretation, but it requires explicitly comparing consecutive residual updates and can be more sensitive to implementation choices about where hidden states are read from the transformer block. NDR is a simpler observable that remains defined whenever layerwise hidden states are available.

I Representation-Space Analyses

Section 7 tests whether geometric correctness signals are stable under changes of representation basis. We evaluate three spaces: the native hidden-state space, PCA-projected hidden space, and vocabulary/logit-related spaces.

I.1 PCA basis

For each model and dataset, we collect token-mean hidden states from the evaluated examples and center them by subtracting the empirical mean. PCA is fit without correctness labels. Unless otherwise stated, we project to the top $k = 512$ principal components:

$$z^l = U_k^\top (\bar{h}^l - \mu), \quad (14)$$

where U_k contains the top principal directions. Geometric scores are recomputed in the projected coordinates z^l . When the paper reports best-PC or PC-ranking diagnostics, labels are used only after PCA fitting to analyze which directions are most discriminative; those oracle diagnostics are not used as deployable label-free scores.

I.2 LM-head singular-vector basis

Let $W_U \in \mathbb{R}^{|V| \times d}$ be the unembedding matrix. We compute a truncated singular-vector basis from W_U and project hidden states onto the corresponding right singular vectors:

$$z^l = V_k^\top \bar{h}^l. \quad (15)$$

This tests whether correctness signals align with directions that are directly relevant to vocabulary prediction.

I.3 Logit space

For logit-space diagnostics, we map the final hidden representation to vocabulary logits:

$$\ell^l = W_U \bar{h}^l. \quad (16)$$

We then recompute norms, angles, and separability scores in this transformed space. These diagnostics are more expensive than NDR because they require vocabulary-scale projections, so they are used to analyze basis dependence rather than to define the simplest deployable score.

J Additional Protocol Details

J.1 Hidden-state extraction

For every generated response, we retain the hidden-state sequence returned by the decoder at each logged depth and generated token position. The index $l = 0$ denotes the embedding/input residual state, and the remaining indices follow the native order returned by the model implementation. The terminal state h_t^L is the representation supplied to the language-model head in the reported

runs. No PCA, whitening, learned projection, or dataset-dependent normalization is applied before computing the native-space trajectory scores.

Unless a token-region ablation states otherwise, the layer representation used by a response-level score is

$$\bar{h}^l = \frac{1}{T} \sum_{t=1}^T h_t^l, \quad (17)$$

where T is the number of generated response tokens. Prompt tokens are not included in this average. The layer-average representation used in the representation-axis analysis is

$$\bar{h}_{\text{avg}} = \frac{1}{L+1} \sum_{l=0}^L \bar{h}^l, \quad (18)$$

which averages the token-mean trajectory across depth; it is distinct from the token average in Equation 17.

J.2 Token-region definitions

The token-axis ablations use deterministic masks over generated response tokens. *Full response* includes every generated token retained by the generation pipeline. *Reasoning only* includes tokens preceding the dataset-specific final-answer boundary. *Answer only* includes the boundary and the extracted final-answer span. *Exclude final token* uses the full response except for its last generated token; that token is not assumed to be an EOS token. The same layerwise token mean and the same NDR definition are then applied to each mask. These ablations change only the token set and do not retrain, tune, or reorient the score.

J.3 Output-confidence controls

The correlation between NDR and output entropy is expected because the language-model head is applied to the terminal hidden representation. We therefore use two complementary controls. First, for each model–dataset pair, an ordinary least-squares model predicts NDR from output-level covariates; the residual is evaluated as a correctness score. This residualization uses no correctness labels. Second, a supervised diagnostic compares a logistic classifier using entropy, perplexity, and response length with the same classifier after adding NDR. This second experiment is a feature-complementarity diagnostic rather than a label-free deployment setting.

Under the length-controlled feature-complementarity evaluation, adding NDR to

entropy, perplexity, and response length improves macro AUROC by 5.5 points on Qwen2-7B-Instruct and 7.4 points on Llama-3-8B-Instruct. The residual results show that output confidence explains part, but not all, of the NDR signal. We therefore interpret NDR as partially overlapping with output uncertainty rather than as an entirely independent correctness variable.

K Additional Robustness and Scale Results

K.1 Within-family model-scale analysis

We evaluate Qwen2.5-Instruct at 3B, 14B, and 32B on GSM8K and MMLU. Each entry in Table 11 is the macro AUROC across these two benchmarks under the same greedy-generation and score-computation protocol used in the primary experiments.

NDR is tied for the highest value at 3B, has the highest value at 14B, and is second to Curvature at 32B.

K.2 Paired bootstrap comparison

We use 2,000 paired bootstrap resamples. Within each benchmark, examples are sampled with replacement and both methods are evaluated on exactly the same resampled examples. AUROC is recomputed for each benchmark and then macro-averaged. The reported difference is NDR minus the comparator; the confidence interval is the percentile interval over bootstrap differences, and the p -value is two-sided. This analysis quantifies uncertainty in the small macro-AUROC gaps and is not used for model or method selection.

K.3 Score-computation time and temporary memory

The timing comparison isolates score computation after hidden states have been cached; it excludes the shared model forward pass and generation cost. All trajectory scores are measured under the same software and hardware environment on the same cached examples. NDR requires only layerwise norms, whereas the other trajectory scores retain vector-valued states or intermediate segment statistics.

The resulting speed difference is approximately fivefold for score-only computation. NDR still requires white-box hidden-state access and is therefore not more accessible than output-only confidence scores for hosted or black-box models.

Model	Comparator	Δ AUROC	95% CI	p	Result
Qwen2-7B-Instruct	Entropy	+3.8	[+2.7, +5.0]	< 0.001	NDR sig. better
	CoE-R	+4.2	[+3.1, +5.4]	< 0.001	NDR sig. better
	Curvature	+0.9	[+0.2, +1.5]	0.022	NDR sig. better
	Lateral Ratio	−0.2	[−0.8, +0.5]	0.51	Tied
Llama-3-8B-Instruct	Entropy	+1.3	[+0.4, +2.3]	0.017	NDR sig. better
	CoE-R	+5.4	[+4.2, +6.5]	< 0.001	NDR sig. better
	Curvature	−0.6	[−1.6, +0.5]	0.28	Tied
	Lateral Ratio	+1.0	[+0.3, +1.8]	0.014	NDR sig. better

Table 13: **Paired bootstrap comparison of NDR against representative baselines.** Δ is the mean AUROC difference NDR – comparator over 2,000 paired resamples (macro over seven benchmarks); positive favors NDR. CI is the percentile interval and p is two-sided, from the same run; $p < 0.001$ is the resolution limit at 2,000 resamples. NDR is significantly better than Entropy and CoE-R on both models; against our own single-feature scores the pattern is model-dependent.

Model	Raw	Resid. _{ent}	Resid. _{ent+ppl+len}
Qwen2-7B-Instruct	68.0	60.1	57.2
Llama-3-8B-Instruct	62.3	61.2	60.3

Table 14: **Macro AUROC after residualizing NDR against output-level covariates.** “Raw” is the full-sample NDR AUROC; “Resid._{ent}” regresses out entropy; “Resid._{ent+ppl+len}” regresses out entropy, perplexity, and response length. These diagnostic values do not replace the full-sample results in Table 1.

L Interpreting the Geometric Diagnostics

Curvature, Lateral Ratio, and NDR summarize different properties of the same layerwise trajectory. Curvature measures local changes in direction, Lateral Ratio measures how much the path deviates from its endpoint displacement, and NDR isolates the relative norm decrease of the terminal state against the preceding trajectory. Their close aggregate performance does not make the first two redundant. The layer-window analysis uses them as diagnostic probes: their nonuniform depth profiles reveal where correctness separation concentrates, and the recurrent late-layer pattern motivates the more economical NDR summary.

The comparisons should therefore be read as a performance–robustness–cost trade-off rather than a claim that NDR uniformly dominates. Curvature can be stronger for particular models or datasets, including the 32B scale condition, while NDR has the smallest score-computation footprint and the most stable behavior under the controls reported in the paper.