TRACING UNCERTAINTY IN LANGUAGE MODEL REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Language model (LM) *reasoning*, commonly described as Chain-of-Thought or test-time scaling, often improves benchmark performance, but the dynamics underlying this process remain poorly understood. We study these dynamics through the lens of uncertainty quantification by treating the reasoning traces, the intermediate token sequences generated by LMs, as evolving model states. We summarize each trace by an *uncertainty trace profile*: a small set of features describing the shape of the uncertainty signal over its trace, such as its slope and linearity. We find that across eight LMs from three different families evaluated on eight datasets on mathematical and logical reasoning, these profiles predict whether a trace yields a correct final answer with AUROC up to 0.984, improving markedly on recent related work. We reach AUROC 0.801 using only the first few hundred tokens of full traces, suggesting that errors can be detected early in the generation. A detailed comparison of correct and incorrect traces further reveals qualitatively distinct uncertainty profiles, with correct traces showing a steeper and less linear decline in uncertainty. Together, the results suggest that our method, grounded in decision-making under uncertainty, provides a principled lens for studying the generative process underlying LM reasoning.

1 INTRODUCTION

The notion of *reasoning* is increasingly being used to describe the output generated by language models (LMs). Whether referred to as a "scratchpad" (Nye et al., 2021), "Chain-of-Thought" (CoT) (Wei et al., 2022; Kojima et al., 2022), or test-time scaling (Muennighoff et al., 2025), researchers generally refer to the same process: an LM generating a long sequence of intermediate tokens that—ideally—results in a correct final answer. It is currently being debated whether "reasoning" is the right term to describe this process (Stechly et al., 2024; 2025; Kambhampati et al., 2025; Højer, 2025), but irrespective of that debate, open-ended long-form generation has been shown to improve performance on a variety of benchmarks with verifiable labels such as mathematical and logical reasoning (Yang et al., 2025; Guo et al., 2025). The connection between the length of the generated sequence and task performance is mainly correlational (Liu et al., 2025), although some work suggests that intervening on the "thinking" budget can directly affect performance (Muennighoff et al., 2025).

A definition of reasoning still eludes NLP and AI, and labeling the generative process of LMs as such arguably reflects misguided anthropomorphization. Broadly speaking, definitions of reasoning in NLP and AI describe *some* process that is "*logical and systematic*" (Wei et al., 2022; Kojima et al., 2022; Huang & Chang, 2023), but even this vague "definition" is questionable. Generally, LMs do not enforce the logical validity of generated outputs, and the reinforcement learning (RL) optimization explicitly rewards only the generation of a correct answer, not the validity of the reasoning trace (Guo et al., 2025; Yang et al., 2025).

In this paper, we treat the generated trace of an LM as an evolving state of the model; a state which can be more or less certain. Figure 1 illustrates this perspective: on the left we show the normalized average of the *distributional uncertainty* (entropy) computed at every step of generation. While correct and incorrect traces overlap and show a similar decrease in uncertainty as a trace progresses, the gap plot (right) indicates sections of the averaged traces where correct and incorrect traces clearly deviate; it is this deviation in the uncertainty dynamics we utilize. We apply uncertainty quantification (UQ) to compute three measures of uncertainty to study how the model state develops as a time

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

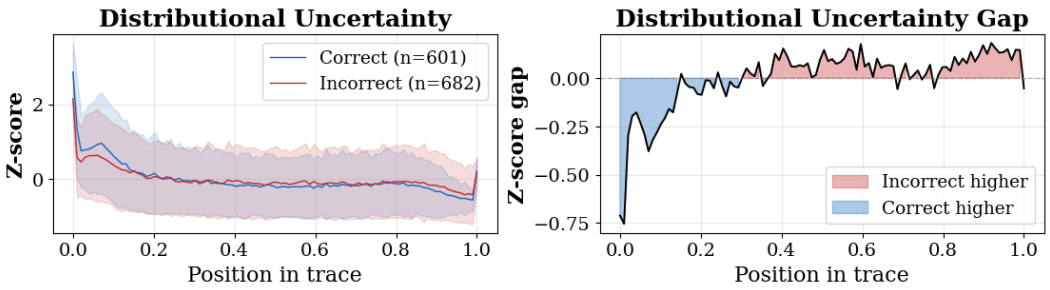

Figure 1: The average uncertainty trace for Qwen 2.5 on GSM8K. Left illustrates the average distributional uncertainty (entropy) computed over all generated sequences. Right shows the average uncertainty gap between correct and incorrect sequences.

series, and show that the resulting **uncertainty trace profile** is strongly predictive of final answer correctness for mathematical and logical reasoning tasks.

Related work. A growing body of work has explored methods for improving LM performance at inference time via various interventions and uses of test-time compute, often by sampling multiple outputs and selecting among them. Wang et al. (2023) propose *Self-Consistency* which samples CoT reasoning paths, selecting the most frequent answer by majority voting. *Universal self-consistency* (Chen et al., 2023) extends this approach to open-ended tasks by prompting the model itself to identify the most consistent response. To address limitations related to scale, Kang et al. (2025) propose *Self-Certainty*, a metric based on the Kullback-Leibler divergence of each token's predicted distribution from a uniform distribution, enabling best-of-N selection yielding better scaling as well as generalization to open-ended generation. Similarly, Fu et al. (2025) introduce *DeepConf*, a sampling method that uses model-internal confidence signals to filter low-quality reasoning traces during or after generation. Other approaches rely on hidden state representations to improve performance on verifiable tasks (Højer et al., 2025; Zhang et al., 2025) and Zhao et al. (2026) propose *Circuit-Based Reasoning Verification* (*CRV*) using transducers and the computation graph of Llama-3.1-8B to distinguish correct from incorrect generations.

Novelty and contribution. The aggregation inherent to other methods (*CRV*, *Self-Certainty*, *DeepConf*) discards most of the temporal structure of the generative process; we treat per-token uncertainty as a time series, enabling richer characterization of how and where reasoning quality changes over the course of generation. Our work is thus complementary to said approaches: like *Self-Certainty* and *DeepConf*, we leverage uncertainty signals from model outputs and internals, but we are the first to utilize the trace as an evolving state as illustrated in Figure 1. Our primary research questions are thus: **RQ1)** are uncertainty trace profiles predictive of correctness; and **RQ2)** what are the signals of uncertainty that distinguish incorrect from correct generations?

2 Background

2.1 Reasoning in Language Models

To formalize the mechanics of LM generation, we consider an LM as a parameterized function $f_\theta : \mathcal{V}^{\leq K} \to \Delta(\mathcal{V})$, where $\mathcal{V}$ is the vocabulary, $\mathcal{V}^{\leq K}$ denotes a sequence of up to K tokens, and $\Delta(\mathcal{V})$ denotes the probability simplex over the vocabulary. The model parameters θ define a conditional distribution $p(v_t \mid v_{<t}, \theta)$ over the next token $v_t \in \mathcal{V}$ given the prefix $v_{<t} = (v_1, \ldots, v_{t-1})$, from which outputs are sampled during generation.

Generating from this model autoregressively is the foundation for producing long sequences of intermediate tokens that simulate reasoning. The finding that such intermediate generations improve performance on some verifiable tasks has resulted in the development of *reasoning models* (RMs) (Yang et al., 2025; Guo et al., 2025). In this paradigm a model is trained with RL to generate an extensive reasoning trace before generating the final answer. Recent discussions emphasize that RMs are post-trained with both a different style of data as well as learning objective, such as RL with

verifiable feedback (RLVF). Some argue that models with this additional complexity are therefore necessarily not *just* "next-token predictors" (Downes et al., 2025), while others question how different RLVF is from standard optimization techniques (Shanahan, 2023; Samineni et al., 2025). Irrespective of this debate, the resulting models generate thousands of tokens and achieve improved performance on some verifiable tasks.

At surface-level, both LMs and RMs generate sequences of human-like reasoning. But while longer generated traces are correlated with improved performance on some benchmarks, the logical validity, reliability, and faithfulness of reasoning traces have been contested (Lanham et al., 2023; Turpin et al., 2023; Mirzadeh et al., 2025; Mondorf & Plank, 2024; Arcuschin et al., 2025; Chen et al., 2025). This calls into question approaches that analyze RM outputs through their semantic content alone (Ling et al., 2023), especially given evidence that training on semantically invalid traces can still improve performance (Stechly et al., 2025).

2.2 Uncertainty in Machine Learning

Most machine learning is inherently probabilistic; even models that produce deterministic point predictions are implicitly embedded in a probabilistic framework (MacKay, 1992; Neal, 2012; Blundell et al., 2015). Within this framework, two kinds of uncertainty are commonly differentiated: aleatoric and epistemic (Hora, 1996; Der Kiureghian & Ditlevsen, 2009; Hüllermeier & Waegeman, 2021).

1. *Aleatoric* uncertainty reflects the inherent, irreducible difficulty of the task itself: variability that no amount of further training could remove.
2. *Epistemic* uncertainty reflects the degree to which a model is informed by its training data about a particular prediction; a decrease over the course of generation can thus be interpreted as "being closer" to regions of the output space encountered during training.

Uncertainty quantification (UQ). Quantifying these uncertainties is commonly approached using information-theoretic measures, specifically through an entropy decomposition of the predictive distribution $H(y \mid x) = H(y \mid x, \theta) + I(y; \theta)$, where x denotes the input, y the prediction, θ the model parameters, and the two components on the right represent aleatoric and epistemic uncertainty, respectively (Smith & Gal, 2018).

Recently, this decomposition has been subject to criticism (Wimmer et al., 2023; Tomov et al., 2025) and a label-wise variance-based alternative has been proposed (Sale et al., 2023; 2024). This alternative leverages the law of total variance to decompose the total uncertainty for a given label k, defined as $\mathrm{Var}[Y_k]$, into an epistemic component $\mathrm{Var}_\theta[p(y_k \mid x, \theta)]$, capturing the variability of the predictive probability across different parameterizations, and an aleatoric component $\mathbb{E}_\theta[p(y_k \mid x, \theta)(1 - p(y_k \mid x, \theta))]$, capturing the expected inherent randomness of the prediction under a given parameterization. For efficient estimation of epistemic uncertainty in LMs, we adopt a recently proposed isotropic approximation (Grünefeld et al., 2026), which to our knowledge is the only computationally tractable approach for variance-based uncertainty quantification at this scale, particularly for the epistemic component.

In this work, we primarily use the variance-based framework for the epistemic component, since its information-theoretic analogue—the mutual information $I(y; \theta)$—has no tractable estimator for models on the scale of LMs. On the aleatoric side, we use both frameworks to further distinguish between two sub-types, committal and distributional:

1a. *Committal* aleatoric uncertainty, $\mathbb{E}_\theta[p(\hat{y}_k \mid x, \theta)(1 - p(\hat{y}_k \mid x, \theta))]$, the expected Bernoulli variance over θ of the predicted class $\hat{y}_k$, taken from the variance-based decomposition.

1b. *Distributional* aleatoric uncertainty, $\mathbb{E}_\theta[H(y \mid x, \theta)]$, the expected conditional entropy of the predictive distribution over θ, taken from the information-theoretic decomposition.

Committal uncertainty depends only on the probability p assigned to the selected token, and distinguishes two qualitatively different ways in which a token can be generated. When p is near 1, the model is committed to the specific token, in what we call a process of **concentration**; when p is near 0.5, committal uncertainty is at its maximum and the top token is selected not by being likely, but by being marginally less unlikely than its alternatives, in what we call a process of **elimination**.

Distributional uncertainty, by contrast, is sensitive to the shape of the entire predictive distribution rather than just the probability of its mode. The two measures can therefore differ: a model may be firmly in the concentration regime, yielding low committal uncertainty, while still exhibiting high distributional uncertainty if the remaining mass is spread diffusely across many alternatives. We provide intuition for these uncertainty types and illustrative examples in Appendix A.

3 METHODS

We analyze the development of uncertainty over the course of a trace as a novel approach to studying the generative process, in contrast to prior work which discards most of the temporal structure of autoregressive generation.

3.1 UNCERTAINTY ESTIMATION

Following the framework of subsection 2.2, we estimate the three uncertainty types at every token position. As described in subsection 2.1, the autoregressive generation is conditioned on the trace prefix $v_{<t}$ and we let y be the target. The choice of target distinguishes two settings: for **trace uncertainty** the target is the next token v_t, while for **answer uncertainty** it is the eventual final answer generated by the model, $\hat{y}$.

The epistemic component is estimated using the squared L2 norm of the gradient of the predicted probability of the target with respect to all model parameters, derived via a first-order delta-method expansion of $\mathrm{Var}_\theta[p(y \mid v_{<t}, \theta)]$ around a point estimate $\hat{\theta}$ of the parameters (Grünefeld et al., 2026):

$$\mathbb{U}_{\mathrm{E}} \coloneqq \mathrm{Var}_\theta[p(y \mid v_{<t}, \theta)] \approx \left\| \nabla_\theta \, p(y \mid v_{<t}, \theta) \, \big|_{\theta=\hat{\theta}} \right\|^2 . \tag{1}$$

Committal aleatoric uncertainty is estimated as the Bernoulli variance of the target's probability,

$$\mathbb{U}_{\mathrm{C}} \coloneqq \mathbb{E}_\theta[p(y \mid v_{<t}, \theta)(1 - p(y \mid v_{<t}, \theta))] \approx p(y \mid v_{<t}, \hat{\theta}) \cdot (1 - p(y \mid v_{<t}, \hat{\theta})). \tag{2}$$

Distributional aleatoric uncertainty is computed as the predictive entropy over the full vocabulary, conditioned on the prefix and evaluated at the point estimate $\hat{\theta}$:

$$\mathbb{U}_{\mathrm{D}} \coloneqq \mathbb{E}_\theta[H(\cdot \mid v_{<t}, \theta)] \approx H(\cdot \mid v_{<t}, \hat{\theta}) = -\sum_{v \in \mathcal{V}} p(v \mid v_{<t}, \hat{\theta}) \log p(v \mid v_{<t}, \hat{\theta}). \tag{3}$$

Trace and answer uncertainty. At each step t, trace uncertainty, $\mathbb{U}^{\mathrm{Tr}}$, instantiates the three measures above with $y = v_t$, characterizing the state of a model with respect to the token it generates next. Answer uncertainty, $\mathbb{U}^{\mathrm{A}}$, instantiates them with $y = \hat{y}$, characterizing instead the state of a model with respect to the eventual answer; this requires a second forward and backward pass at each step, conditioned on the trace prefix $v_{<t}$ and applied to $\hat{y}$ rather than v_t (see Appendix B). Since $\hat{y}$ is in general a multi-token span, we use the sequence-level extensions of Grünefeld et al. (2026) for epistemic and committal uncertainties, and analogously the average entropy for distributional uncertainty.

We refer to $\mathbb{U}_{\mathrm{E}}$, $\mathbb{U}_{\mathrm{C}}$, and $\mathbb{U}_{\mathrm{D}}$ as the three **types** of uncertainty, and to $\mathbb{U}^{\mathrm{Tr}}$ and $\mathbb{U}^{\mathrm{A}}$ as the two uncertainty **channels**. The two are orthogonal: each type has a trace variant and an answer variant, giving six time series per generation that together describe how local and answer-directed uncertainty co-evolve.

The answer-level channel by definition requires the generated final answer, and thus cannot be used as a means of intervention. Additionally, epistemic uncertainty requires substantially more computational resources to estimate (Appendix E) and is more difficult to utilize in an online setting. Therefore, we further investigate the utility of only the aleatoric trace-level features for final answer correctness, specifically in larger models.

3.2 EXPERIMENTAL PROCEDURE

Models & datasets. We evaluate Llama-3.1 (*8B*), Llama-3.2 (*1B*), Qwen-2.5 (*0.5B*), DeepSeek R1 Distill Qwen (*1.5B*), and Qwen-3 (*0.6B*) using the full uncertainty trace profile. Additionally, we

evaluate Qwen-3 (*8B and 14B*) and Qwen-3.5 (*9B*) using only the aleatoric trace-level features. These models are chosen for comparability with related work (Zhao et al., 2026) and to represent different training techniques. Llama 3 models are trained primarily using supervised fine-tuning and limited RL with human feedback and thus without explicit reasoning optimization (Grattafiori et al., 2024). Qwen-2.5 was trained similarly, but with a focus on reasoning data (Yang et al., 2024), whereas DeepSeek R1, Qwen-3 and Qwen-3.5 were trained with the explicit objective of generating reasoning traces to improve correctness (Guo et al., 2025; Yang et al., 2025). These models range from standard LMs (the non-reasoning type) (Llama) to RMs (the reasoning-type) (R1, Qwen-3, Qwen-3.5), with Qwen-2.5 being in-between. This enables an assessment of how uncertainty dynamics differ on this spectrum.

We evaluate on eight datasets in total: GSM8K (Cobbe et al., 2021), chosen for comparability with related work (Zhao et al., 2026), as well as ProntoQA (Saparov & He, 2023), FOLIO (Han et al.), BHH-Tracking (Kazemi et al.), CRV Arithmetic and Boolean (Zhao et al., 2026). Zebralogic (Lin et al.) and Omni-Math (Gao et al.) are further chosen as more difficult tasks for the larger models. We further study examples from GSM-Symbolic (Mirzadeh et al., 2025); a variant of GSM8K with controlled perturbations that preserve the underlying logic of questions.

Generating traces. We evaluate all models using greedy decoding. While it is common practice to use different sampling strategies for reasoning models, we employ greedy decoding to not introduce the choice of sampling strategy and sampling variation as a confounding variable.

The generated outputs are split into a reasoning trace and a final answer using model-specific splitting strategies that are documented in Appendix B. We initially set a max sequence length of 2048 for the trace and filter out responses that do not produce a final answer. For the reasoning models we append the `</think>` token to any trace that did not generate a final answer within the initial generation limit, restart the generation with an additional 512-token budget, and again filter out the traces that did not generate a final answer (details in Appendix D); the filtering removed approximately 10% of samples for DeepSeek R1 and Qwen 3.

3.3 Predictive Modeling & Reasoning Dynamics

Feature engineering & correctness prediction. We extract a small set of features from all type-channel combinations ($\mathbb{U}_{\mathrm{D}}^{\mathrm{Tr}}, \mathbb{U}_{\mathrm{C}}^{\mathrm{Tr}}, \mathbb{U}_{\mathrm{E}}^{\mathrm{Tr}}, \mathbb{U}_{\mathrm{D}}^{\mathrm{A}}, \mathbb{U}_{\mathrm{C}}^{\mathrm{A}}, \mathbb{U}_{\mathrm{E}}^{\mathrm{A}}$) to summarize the dynamics of a trace into the uncertainty trace profile. We compute the early mean (μ_{early}), the middle mean (μ_{mid}), the late mean (μ_{late}), the linear slope of the trace ($\mathbf{m}$), and the fit of the slope (r^2), resulting in 30 features in total (see Appendix C; Table 3). Using these features, we train two binary classifiers for each model-dataset combination to predict the correctness of the final answer of every given trace: a logistic regression and a gradient boosted model (reporting the best performance). Full experimental details are provided in Appendix D. We use the area under the receiver operating characteristic curve (AUROC) to measure the predictive performance of each classifier in a five-fold cross-validation setup.

Early correctness detection. To assess how early we can predict correctness, we bin traces by number of tokens generated, and extract features based only on the tokens of the trace before the bin limit. We train the same classifiers based on the constrained traces, to analyze how informative early trace dynamics are for predicting correctness.

4 Analysis

We report AUROC scores across models and datasets, and perform an analysis of the uncertainty trace profile for correct and incorrect traces (i.e. traces that result or don't result in a correct answer).

4.1 Correctness Prediction

Table 1 displays the correctness prediction AUROC scores using our full uncertainty trace profiles on GSM8K and ProntoQA, with Table 7 reporting the performance on the other four datasets. Table 2 further presents our results using the aleatoric trace-level uncertainties on larger models. We report

the scores achieved with our uncertainty trace profiles as well as the *Self-Certainty*, *DeepConf*, the mean generation entropy, and the number of generated tokens as baseline comparisons.

Our uncertainty-based approach outperforms all baselines on 18 out of 29 model-dataset combinations using the full uncertainty trace profiles, and on 3 out of 5 combinations using the aleatoric trace-level uncertainties. *Self-Certainty* and *DeepConf*, in contrast, achieve the best performance in 4 and 3 cases, respectively, in total. The number of generated tokens and the mean token entropy perform similarly, at 3 combinations each in total. Notably, even for the full uncertainty trace profiles, an ablation by uncertainty type shows that most of the predictive power is carried by $\mathbb{U}^{\mathrm{Tr}}$ (see Appendix F).

	GSM 8K					ProntoQA				
	UTP	*SC*	*DC*	*#Tok.*	*Ent.*	UTP	*SC*	*DC*	*#Tok.*	*Ent.*
Llama 3.1	**0.783**	0.491	0.532	0.684	0.571	**0.804**	0.566	0.700	0.609	0.737
Llama 3.2	**0.767**	0.566	0.609	0.623	0.627	0.550	0.549	**0.557**	0.543	0.538
Qwen 2.5	**0.807**	0.689	0.704	0.698	0.725	0.521	0.565	0.564	0.522	**0.567**
DeepSeek R1	**0.786**	0.703	0.678	0.478	0.606	0.672	0.615	0.671	**0.753**	0.657
Qwen 3	0.727	**0.728**	0.719	0.677	0.723	**0.657**	0.611	0.605	0.640	0.540

Table 1: Mean AUROC scores (5-fold CV) across five models for GSM8K and ProntoQA. UTP = Best classifier based on *uncertainty trace profile*, *SC = Self-Certainty*, *DC = DeepConf*, *#Tok. = Generated Tokens*, *Ent. = Average Token Entropy*.

	ZebraLogic					Omni-Math				
	UTP	*SC*	*DC*	*#Tok.*	*Ent.*	UTP	*SC*	*DC*	*#Tok*	*Ent.*
Qwen 3 8B	—	—	—	—	—	**0.891**	0.889	0.882	0.858	0.692
Qwen 3 14B	**0.948**	0.932	0.930	0.935	0.537	0.882	**0.891**	0.886	0.857	0.683
Qwen 3.5 9B	0.928	0.918	**0.936**	0.847	0.593	**0.899**	0.787	0.799	0.645	0.518

Table 2: Mean AUROC scores (5-fold CV) across models for ZebraLogic and Omni-Math. UTP = Best classifier based on *uncertainty trace profile*, *SC = Self-Certainty*, *DC = DeepConf*, *#Tok. = Generated Tokens*, *Ent. = Average Token Entropy*.

Early correctness detection. Our analysis also indicates that the uncertainty trace profiles are informative of correctness early in the generation for trace-level uncertainty, $\mathbb{U}^{\mathrm{Tr}}$. We observe AUROC up to 0.801 at the 300th token for GSM8K, and a similar trend across all five models tested with both uncertainty channels, with AUROC rising steadily as larger parts of the full trace are included (Appendix F; Figure 4). A similar trend is visible for ProntoQA, although results are more noisy. While we primarily focus on the dynamics of uncertainty traces over the course of generation, this result indicates that our approach could be used for early intervention during generation.

Feature importance. We measure feature importance via the coefficient share of the logistic regression, which is computed as the absolute coefficient of each feature as a fraction of the sum of all absolute coefficients. We aggregate static ($\mu_{\mathrm{early}}, \mu_{\mathrm{mid}}, \mu_{\mathrm{late}}$) and dynamic (slope, r^2) features to assess the group-level importance, which is computed by summing the coefficient shares in each group. The coefficient shares change over the course of a trace: at the very beginning of a generated trace the static uncertainty markers have the largest coefficient share, but as the generated sequences become longer, the slope and linearity coefficients of the uncertainty trace profile become more important in predicting correctness (see App., Figure 5).

4.2 Feature Analysis

We examine the per-model uncertainty trace profiles shown in Figure 2, which displays features for incorrect and correct traces standardized by their z-score to focus on relative differences between correct and incorrect traces. We standardize within model because the absolute magnitudes of two of our measures are not informative for cross-model comparison. Gradient-based epistemic uncertainty, $\mathbb{U}_{\mathrm{E}}$, is not directly comparable across models and entropy, $\mathbb{U}_{\mathrm{D}}$, is comparable in principle but dominated in scale by the long tail of effectively irrelevant tokens.

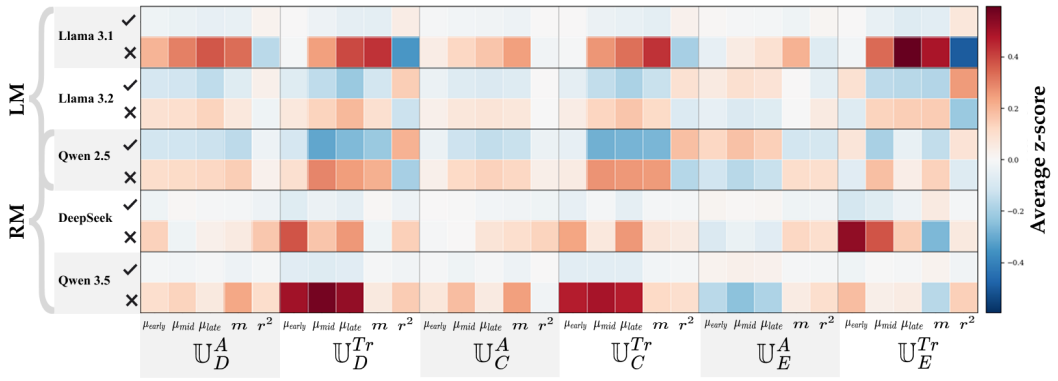

Figure 2: Heatmap of uncertainty features across models and datasets. The heatmap is organized into six feature blocks—$\mathbb{U}_{\mathrm{D}}$, $\mathbb{U}_{\mathrm{C}}$, and $\mathbb{U}_{\mathrm{E}}$ for $\mathbb{U}^{\mathrm{A}}$ and $\mathbb{U}^{\mathrm{Tr}}$ respectively—each containing five features: μ_{early}, μ_{mid}, μ_{late}, slope ($\mathbf{m}$), and r^2. Colors indicate relative differences within model between correct and incorrect traces.

Levels. Across all five models, μ_{early}, μ_{mid}, and μ_{late} of most uncertainty types are higher for incorrect traces than for correct ones: the static level of uncertainty points to an incorrect answer as can be seen by comparing the rows of Figure 2. The gap is more pronounced for trace-level uncertainty, $\mathbb{U}^{\mathrm{Tr}}$, than for answer-level uncertainty, $\mathbb{U}^{\mathrm{A}}$, consistent with the finding that $\mathbb{U}^{\mathrm{Tr}}$ features carry most of the predictive power (Appendix F; Table 5). The clear exception is answer-level epistemic uncertainty, $\mathbb{U}_{\mathrm{E}}^{\mathrm{A}}$, where we observe the opposite, with incorrect traces showing lower uncertainty. The two epistemic measures therefore point in opposite directions on failure. $\mathbb{U}_{\mathrm{E}}^{\mathrm{Tr}}$ is elevated on incorrect traces, indicating that the trace-level steps of an incorrect trace are less informed by training data than those of a correct trace; $\mathbb{U}_{\mathrm{E}}^{\mathrm{A}}$ is depressed on incorrect traces, indicating that the eventual wrong answer is more informed by training data than correct answers tend to be. Our results can be interpreted as follows: when a model generates an incorrect answer, it generates less-supported steps to arrive at a wrong answer that is nonetheless closer to regions encountered during training. This pattern is complementary to the elevated aleatoric uncertainty measures observed on the same incorrect traces.

Temporal dynamics. Slope ($\mathbf{m}$) and r^2 characterize how the uncertainty trajectory develops over the course of generation as illustrated by the columns of Figure 2. $\mathbf{m}$ distinguishes incorrect from correct traces consistently across models: in almost all cases, the slope of incorrect traces is higher relative to correct traces, indicating that uncertainty decreases less in incorrect traces as the generation progresses. The exception is $\mathbb{U}_{\mathrm{E}}^{\mathrm{Tr}}$: in Llama 3.1 the slope is still higher for incorrect traces, but the gap shrinks as we move from LMs and inverts in RMs, so that for DeepSeek-R1 and Qwen 3 the $\mathbb{U}_{\mathrm{E}}^{\mathrm{Tr}}$ slope of incorrect traces is again lower.

The fit quality, r^2, follows a similar trend. In Llama 3.1, almost every uncertainty type has higher r^2 for correct traces than for incorrect ones, with correct trajectories following tighter linear trends and incorrect trajectories appearing more irregular. Moving from LMs to RMs this pattern flips such that incorrect traces appear more linear than correct ones.

In Llama 3.1, the local steps of incorrect traces are persistently less informed by training data than those of correct traces, with $\mathbb{U}_{\mathrm{E}}^{\mathrm{Tr}}$ elevated throughout and not decreasing as the trace progresses. The eventual wrong answer is nonetheless more strongly supported by training data than correct answers tend to be. The trace and the answer it produces thus pull in opposite directions. In the RMs the picture changes: $\mathbb{U}_{\mathrm{E}}^{\mathrm{Tr}}$ decreases more steeply on incorrect traces than on correct ones, so the local steps move progressively closer to better-supported territory as the model converges on the wrong answer that $\mathbb{U}_{\mathrm{E}}^{\mathrm{A}}$ already flagged as well-supported. The smoother, more linear r^2 profiles of incorrect RM traces are consistent with this convergence. Together, the level and temporal dynamics describe a qualitative shift in how incorrect traces fail across model classes.

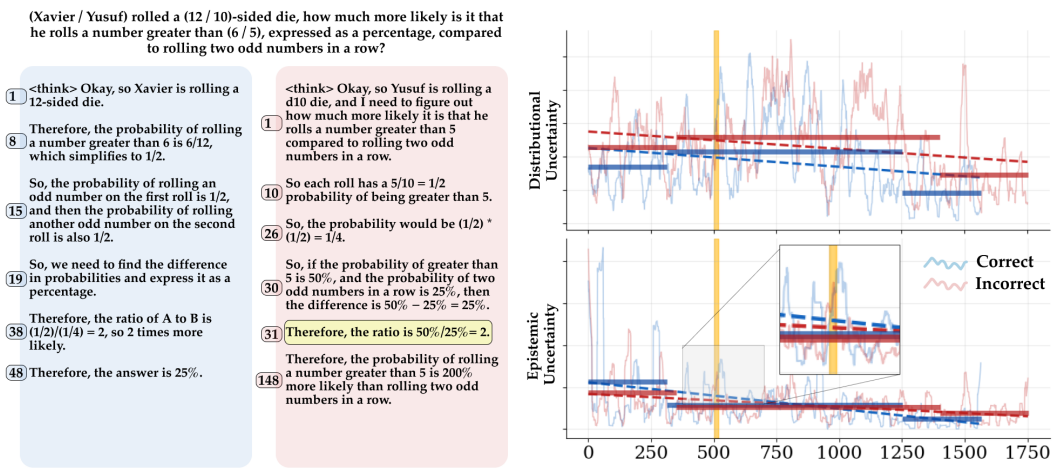

Figure 3: Sample from GSM-Symbolic. Numbers indicate sentence indices. μ_{early}, μ_{mid}, and μ_{late} are shown as horizontal lines and slope as a dashed linear-fit line, with the corresponding r^2 reflected in the spread of the data around that line. Each plot shows the uncertainty type with the token index on the x-axis, and the location of the first error in the incorrect trace is highlighted in yellow.

4.3 Qualitative Analysis

To illustrate qualitatively how the analysis of uncertainty can be applied to study LM reasoning, we select a pair of sequences generated by Qwen 3 when evaluated on a GSM-Symbolic example (Mirzadeh et al., 2025). Recall that GSM-Symbolic examples are perturbed GSM8K questions that preserve the underlying logic. Figure 3 shows an example where the name and the size of the die differ between questions. We show the generated sequences (left) alongside the smoothed trace-level trajectories of distributional uncertainty $\mathbb{U}_{\text{D}}^{\text{Tr}}$ and epistemic uncertainty $\mathbb{U}_{\text{E}}^{\text{Tr}}$, with the uncertainty trace profile features overlaid on the raw trace values (right): horizontal lines show the three means, dashed lines show the linear trend, and r^2 captures the variation around the linear slope. $\mathbb{U}_{\text{D}}^{\text{Tr}}$ is elevated on the incorrect trace across μ_{early}, μ_{mid}, and μ_{late} and the incorrect slopes fail to decrease as cleanly as the correct ones, consistent with the analysis of subsection 4.2.

The model correctly computes the relevant probabilities (50% for the first event, 25% for two consecutive odd rolls) for both sequences and correctly considers the same two interpretations of *"how much more likely"*: the *difference* and the *ratio*. In the incorrect sequence, the model commits to using the ratio, shown in the highlighted sentence 31 of Figure 3 which corresponds to the highlighted token number 500; this results in the model incorrectly outputting 200% instead of 25% as the final answer. The correct trace peaks at the multiplicative-probability composition, while the incorrect trace shows a local elevation when committing to the ratio interpretation, with the signal reaching roughly $2.4\times$ the average of the surrounding 200-token window. For the correct trace, there is no peak at the mention of the ratio, but at the point of committing to the difference. Importantly, focusing on the first semantically observable error in the generated reasoning aligns only with a single peak of the trace, at token 500. Many of the earlier peaks could be relevant for error-detection.

Both trajectories additionally show their largest relative peaks at structural boundaries—the trace opening and the post-think *"Final Answer"* section header—which we read as tokenization-driven gradient elevation rather than reasoning content; against this baseline the smaller reasoning-content peaks are the ones that distinguish the two traces and that flag the specific reasoning steps at which the incorrect trace shows errors.

5 Discussion

The UQ lens reveals a stark difference between generative processes resulting in correct and incorrect final answers, answering **RQ1** in the affirmative: uncertainty trace profiles predict correctness with

AUROC up to 0.984, outperforming *Self-Certainty* (Kang et al., 2025) and *DeepConf* (Fu et al., 2025), and one can predict correctness with AUROC up to 0.801 using only the first 300 tokens.

UQ provides an interpretative lens through which to understand the process of language model reasoning. Correct and incorrect traces are distinguishable across the studied models, and the analysis shows that the demarcation is made based on a relative difference in the slope and linearity of the entropy, $\mathbb{U}_{\mathrm{D}}$, and Bernoulli variance, $\mathbb{U}_{\mathrm{C}}$, with a steeper and more linear slope being indicative of a trace being incorrect. Notably, the epistemic uncertainty presents a complementary behavior, being smaller for incorrect traces than for correct traces. Together, the aleatoric trajectory and the epistemic dissociation answer **RQ2**: the discriminating signals lie in the joint behavior of the two channels, not in either taken in isolation. Under the classical interpretation of aleatoric and epistemic uncertainty discussed in subsection 2.2, the most intuitive failure mode couples the two measures: an out-of-distribution input should yield elevated epistemic uncertainty, and a difficult input should yield elevated aleatoric uncertainty. Our findings deviate from this interpretation: incorrect traces show *elevated* aleatoric and *depressed* epistemic uncertainty. The model is uncertain during generation, yet the wrong answer it produces is seemingly well-supported by training data; aleatoric and epistemic thus carry separate diagnostic signals: neither alone reveals this failure mode.

Implications. First, the predictive power of uncertainty trace profiles offers a practical signal for early correctness estimation, in both LMs and RMs. This has potential applications in selective generation and test-time scaling that aim to intervene on generation prior to completion. Second, the observed dissociation between aleatoric and epistemic measures of uncertainty points to a characterization of failure modes that simpler measures miss. Third, our case study illustrates failures that uncertainty-based analysis can identify but semantic analysis of the trace cannot, establishing the uncertainty lens as an analytical tool that yields complementary insights to semantic inspection. Finally, our results indicate that modeling the temporal aspects of the uncertainty trace profile utilizes information that was not modeled by prior approaches such as *Self-Certainty* and *DeepConf*.

Limitations and future work. We propose a method for analyzing reasoning based on temporal dynamics of the generative process, which have for the most part been ignored by prior work. Our method and analysis rely on interpretations of aleatoric and epistemic uncertainty estimates that are commonly used and well-founded, but nevertheless rest on approximations that are defensible in the overparameterized regime, yet not strictly verifiable. Further, the long sequences generated by RMs sometimes exceed the maximum length of 2048 tokens set for the experiments, potentially distorting results; we mitigate this through the filtering procedure described in subsection 3.2. Finally, we generate output using greedy decoding across all of our experiments, while LMs are usually deployed with various sampling strategies. We emphasize that this paper presents a proof-of-concept for the informativeness of uncertainty measures across reasoning traces, which we show on eight models (0.5B-14B parameters) spanning the LM to RM spectrum and eight datasets for mathematical and logical reasoning. Future work should expand the evaluation to larger models and particularly study the effect of sampling strategies on the uncertainty trace profile to further generalize our results.

6 Conclusion

We have proposed treating language model reasoning traces as time series of uncertainty estimates, summarizing each trace by an uncertainty trace profile that captures the level and dynamics of uncertainty over the course of generation. The uncertainty trace profile is strongly predictive of correctness across eight models and eight datasets, with AUROC up to 0.984 on full traces and 0.801 using only the first few hundred tokens, a significant improvement over prior approaches. We further show that correct and incorrect traces differ systematically under this lens, and that the epistemic and aleatoric uncertainties carry diagnostic signal in opposite directions, indicating that the two cannot be summarized by a single measure. Taken together, these findings indicate that uncertainty quantification across reasoning traces captures substantial information about the dynamics of the process, and offers a lens for studying it that semantic inspection alone does not provide.

REFERENCES

Iván Arcuschin, Jett Janiak, Robert Krzyzanowski, Senthooran Rajamanoharan, Neel Nanda, and Arthur Conmy. Chain-of-Thought Reasoning In The Wild Is Not Always Faithful, March 2025.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight Uncertainty in Neural Networks. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1613–1622. PMLR, June 2015.

Xinyun Chen, Renat Aksitov, Uri Alon, Jie Ren, Kefan Xiao, Pengcheng Yin, Sushant Prakash, Charles Sutton, Xuezhi Wang, and Denny Zhou. Universal Self-Consistency for Large Language Model Generation, November 2023.

Yanda Chen, Joe Benton, Ansh Radhakrishnan, Jonathan Uesato, Carson Denison, John Schulman, Arushi Somani, Peter Hase, Misha Wagner, Fabien Roger, Vlad Mikulik, Sam Bowman, Jan Leike, Jared Kaplan, and Ethan Perez. Reasoning Models Don't Always Say What They Think, May 2025.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training Verifiers to Solve Math Word Problems, November 2021.

Armen Der Kiureghian and Ove Ditlevsen. Aleatory or epistemic? Does it matter? *Structural Safety*, 31(2):105–112, 2009. ISSN 0167-4730. doi: 10.1016/j.strusafe.2008.06.020. Risk Acceptance and Risk Communication.

Stephen M. Downes, Patrick Forber, and Alex Grzankowski. LLMs are not just next token predictors. *Inquiry: An Interdisciplinary Journal of Philosophy*, January 2025. doi: 10.1080/0020174X.2024.2446240. Published online 12 January 2025.

Yichao Fu, Xuewei Wang, Yuandong Tian, and Jiawei Zhao. Deep Think with Confidence, August 2025.

Bofei Gao, Feifan Song, Zhe Yang, Zefan Cai, Yibo Miao, Qingxiu Dong, Lei Li, Chenghao Ma, Liang Chen, Runxin Xu, Zhengyang Tang, Benyou Wang, Daoguang Zan, Shanghaoran Quan, Ge Zhang, Lei Sha, Yichang Zhang, Xuancheng Ren, Tianyu Liu, and Baobao Chang. Omni-MATH: A Universal Olympiad Level Mathematic Benchmark for Large Language Models. URL https://openreview.net/forum?id=yaqPf0KAlN.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The Llama 3 Herd of Models, November 2024. Author list truncated; full author list available at the arXiv page.

Nils Grünefeld, Jes Frellsen, and Christian Hardmeier. An Isotropic Approach to Efficient Uncertainty Quantification with Gradient Norms. In *Proceedings of The 1st Symposium on Probabilistic Machine Learning*, pp. 86–118. PMLR, September 2026.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, September 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. Author list truncated for brevity; see Nature publication for full list.

Simeng Han, Hailey Schoelkopf, Yilun Zhao, Zhenting Qi, Martin Riddell, Wenfei Zhou, James Coady, David Peng, Yujie Qiao, Luke Benson, Lucy Sun, Alexander Wardle-Solano, Hannah Szabó, Ekaterina Zubova, Matthew Burtell, Jonathan Fan, Yixin Liu, Brian Wong, Malcolm Sailor, Ansong Ni, Linyong Nan, Jungo Kasai, Tao Yu, Rui Zhang, Alexander Fabbri, Wojciech Maciej Kryscinski, Semih Yavuz, Ye Liu, Xi Victoria Lin, Shafiq Joty, Yingbo Zhou, Caiming Xiong, Rex Ying, Arman Cohan, and Dragomir Radev. FOLIO: Natural Language Reasoning with First-Order Logic. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 22017–22031. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.1229. URL https://aclanthology.org/2024.emnlp-main.1229/.

Bertram Højer. On the Notion that Language Models Reason. In *1st Workshop on Epistemic Intelligence in Machine Learning (EIML), EurIPS 2025*, 2025. doi: 10.48550/arXiv.2511.11810.

Bertram Højer, Oliver Jarvis, and Stefan Heinrich. Improving Reasoning Performance in Large Language Models via Representation Engineering. In *The Thirteenth International Conference on Learning Representations*. OpenReview, 2025. doi: 10.48550/arXiv.2504.19483.

Stephen C. Hora. Aleatory and epistemic uncertainty in probability elicitation with an example from hazardous waste management. *Reliability Engineering & System Safety*, 54(2):217–223, 1996. ISSN 0951-8320. doi: 10.1016/S0951-8320(96)00077-4. Treatment of Aleatory and Epistemic Uncertainty.

Jie Huang and Kevin Chen-Chuan Chang. Towards Reasoning in Large Language Models: A Survey. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 1049–1065, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.67.

Eyke Hüllermeier and Willem Waegeman. Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine Learning*, 110(3):457–506, March 2021. ISSN 1573-0565. doi: 10.1007/s10994-021-05946-3.

Subbarao Kambhampati, Karthik Valmeekam, Siddhant Bhambri, Vardhan Palod, Lucas Saldyt, Kaya Stechly, Soumya Rani Samineni, Durgesh Kalwar, and Upasana Biswas. Position: Stop Anthropomorphizing Intermediate Tokens as Reasoning/Thinking Traces!, April 2025.

Zhewei Kang, Xuandong Zhao, and Dawn Song. Scalable Best-of-N Selection for Large Language Models via Self-Certainty. In *The Thirty-Ninth Annual Conference on Neural Information Processing Systems*, 2025. doi: 10.48550/arXiv.2502.18581.

Mehran Kazemi, Bahare Fatemi, Hritik Bansal, John Palowitch, Chrysovalantis Anastasiou, Sanket Vaibhav Mehta, Lalit K Jain, Virginia Aglietti, Disha Jindal, Peter Chen, Nishanth Dikkala, Gladys Tyen, Xin Liu, Uri Shalit, Silvia Chiappa, Kate Olszewska, Yi Tay, Vinh Q. Tran, Quoc V Le, and Orhan Firat. BIG-Bench Extra Hard. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 26473–26501. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.1285. URL https://aclanthology.org/2025.acl-long.1285/.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large Language Models are Zero-Shot Reasoners. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 22199–22213. Curran Associates, Inc., 2022.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, Kamilė Lukošiūtė, Karina Nguyen, Newton Cheng, Nicholas Joseph, Nicholas Schiefer, Oliver Rausch, Robin Larson, Sam McCandlish, Sandipan Kundu, Saurav Kadavath, Shannon Yang, Thomas Henighan, Timothy Maxwell, Timothy Telleen-Lawton, Tristan Hume, Zac Hatfield-Dodds, Jared Kaplan, Jan Brauner, Samuel R. Bowman, and Ethan Perez. Measuring Faithfulness in Chain-of-Thought Reasoning, July 2023.

Bill Yuchen Lin, Ronan Le Bras, Kyle Richardson, Ashish Sabharwal, Radha Poovendran, Peter Clark, and Yejin Choi. ZebraLogic: On the scaling limits of LLMs for logical reasoning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *ICML'25*, pp. 37889–37905. JMLR.org.

Zhan Ling, Yunhao Fang, Xuanlin Li, Zhiao Huang, Mingu Lee, Roland Memisevic, and Hao Su. Deductive Verification of Chain-of-Thought Reasoning. In *Advances in Neural Information Processing Systems*, volume 36. Curran Associates, Inc., 2023. doi: 10.48550/arXiv.2306.03872.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding R1-Zero-Like Training: A Critical Perspective. In *Conference on Language Modeling (COLM)*, 2025. doi: 10.48550/arXiv.2503.20783.

David J. C. MacKay. A Practical Bayesian Framework for Backpropagation Networks. *Neural Computation*, 4(3):448–472, May 1992. ISSN 0899-7667. doi: 10.1162/neco.1992.4.3.448.

Iman Mirzadeh, Keivan Alizadeh, Hooman Shahrokhi, Oncel Tuzel, Samy Bengio, and Mehrdad Farajtabar. GSM-Symbolic: Understanding the Limitations of Mathematical Reasoning in Large Language Models. In *The Thirteenth International Conference on Learning Representations*, 2025. doi: 10.48550/arXiv.2410.05229.

Philipp Mondorf and Barbara Plank. Comparing Inferential Strategies of Humans and Large Language Models in Deductive Reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9370–9402, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.508.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. S1: Simple test-time scaling. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 20275–20321, Suzhou, China, 2025. Association for Computational Linguistics.

Radford M. Neal. *Bayesian Learning for Neural Networks*, volume 118. Springer Science & Business Media, 2012.

Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, Charles Sutton, and Augustus Odena. Show Your Work: Scratchpads for Intermediate Computation with Language Models, November 2021.

Yusuf Sale, Paul Hofman, Lisa Wimmer, Eyke Hüllermeier, and Thomas Nagler. Second-Order Uncertainty Quantification: Variance-Based Measures, December 2023.

Yusuf Sale, Paul Hofman, Timo Löhr, Lisa Wimmer, Thomas Nagler, and Eyke Hüllermeier. Label-wise Aleatoric and Epistemic Uncertainty Quantification. In *Proceedings of the Fortieth Conference on Uncertainty in Artificial Intelligence*, volume 244 of *Proceedings of Machine Learning Research*, pp. 3159–3179. PMLR, July 2024.

Soumya Rani Samineni, Durgesh Kalwar, Karthik Valmeekam, Kaya Stechly, and Subbarao Kambhampati. RL in Name Only? Analyzing the Structural Assumptions in RL post-training for LLMs. In *NeurIPS 2025 Workshop on Bridging Language, Agent, and World Models for Reasoning and Planning (LAW)*, 2025. doi: 10.48550/arXiv.2505.13697. Also accepted at NeurIPS 2025 Workshop on Foundations of Reasoning in Language Models (ForLM).

Abulhair Saparov and He He. Language Models Are Greedy Reasoners: A Systematic Formal Analysis of Chain-of-Thought, 2023. URL `https://arxiv.org/abs/2210.01240`. Published at ICLR 2023.

Murray Shanahan. Talking About Large Language Models, February 2023.

Lewis Smith and Yarin Gal. Understanding Measures of Uncertainty for Adversarial Example Detection. In Amir Globerson and Ricardo Silva (eds.), *Proceedings of the Thirty-Fourth Conference on Uncertainty in Artificial Intelligence, UAI 2018, Monterey, California, USA, August 6-10, 2018*, pp. 560–569. AUAI Press, 2018.

Kaya Stechly, Karthik Valmeekam, and Subbarao Kambhampati. Chain of Thoughtlessness? An Analysis of CoT in Planning. In *Advances in Neural Information Processing Systems*, volume 37. Curran Associates, Inc., 2024. doi: 10.48550/arXiv.2405.04776.

Kaya Stechly, Karthik Valmeekam, Atharva Gundawar, Vardhan Palod, and Subbarao Kambhampati. Beyond Semantics: The Unreasonable Effectiveness of Reasonless Intermediate Tokens. In *The Thirty-Ninth Annual Conference on Neural Information Processing Systems*, 2025. doi: 10.48550/arXiv.2505.13775.

Tim Tomov, Dominik Fuchsgruber, Tom Wollschläger, and Stephan Günnemann. Entropy Is Not Enough: Uncertainty Quantification for LLMs fails under Aleatoric Uncertainty. In *NeurIPS 2025 Workshop on Structured Probabilistic Inference & Generative Modeling*, 2025.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. Language Models Don't Always Say What They Think: Unfaithful Explanations in Chain-of-Thought Prompting. In *Advances in Neural Information Processing Systems*, volume 36. Curran Associates, Inc., 2023.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In *The Eleventh International Conference on Learning Representations*, 2023. doi: 10.48550/arXiv.2203.11171.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 24824–24837. Curran Associates, Inc., 2022.

Lisa Wimmer, Yusuf Sale, Paul Hofman, Bernd Bischl, and Eyke Hüllermeier. Quantifying Aleatoric and Epistemic Uncertainty in Machine Learning: Are Conditional Entropy and Mutual Information Appropriate Measures? In Robin J. Evans and Ilya Shpitser (eds.), *Proceedings of the Thirty-Ninth Conference on Uncertainty in Artificial Intelligence*, volume 216 of *Proceedings of Machine Learning Research*, pp. 2282–2292. PMLR, July 2023. doi: 10.48550/arXiv.2209.03302.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. Qwen2 Technical Report, September 2024. Author list truncated; full author list available at the arXiv page.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 Technical Report, May 2025. Author list truncated; full author list available at the arXiv page.

Anqi Zhang, Yulin Chen, Jane Pan, Chen Zhao, Aurojit Panda, Jinyang Li, and He He. Reasoning Models Know When They're Right: Probing Hidden States for Self-Verification, April 2025.

Zheng Zhao, Yeskendir Koishekenov, Xianjun Yang, Naila Murray, and Nicola Cancedda. Verifying Chain-of-Thought Reasoning via Its Computational Graph. In *The Fourteenth International Conference on Learning Representations*, 2026. doi: 10.48550/arXiv.2510.09312. Accepted as Oral.

A Intuitions for the Uncertainty Types

The three uncertainty types defined in subsection 2.2 can be unfamiliar on first reading. This appendix offers worked intuitions for each, grouped by the aleatoric–epistemic distinction (subsection A.1) and the committal–distributional distinction within aleatoric uncertainty (subsection A.2).

A.1 Aleatoric and Epistemic Uncertainty

Consider trying to predict the outcome of a coin flip for a coin you have just been handed. You have flipped it a few times to estimate whether it is fair, but you remain unsure. Two distinct sources contribute to your uncertainty about the next flip.

Aleatoric (irreducible) uncertainty. A coin flip is, in principle, a random event. Even if you knew with perfect certainty that the coin was exactly fair, you still could not predict the outcome of any particular flip with confidence; the randomness is intrinsic to the task. This is aleatoric uncertainty. Flipping the coin more times to learn about its bias does not reduce it. In the language of subsection 2.2, it is the noise in y given x that no parameterization θ could remove.

Epistemic (knowledge-based) uncertainty. You have flipped the coin only a handful of times, and you are not yet sure whether it is fair, slightly biased toward heads, or slightly biased toward tails. This second source of uncertainty is not about the coin flip itself but about your belief regarding the coin's underlying bias. Someone who had flipped the coin a thousand times would be far more certain about its bias and therefore better calibrated about the next flip, even though the flip itself remains as random as ever. This difference is epistemic uncertainty: it is reducible in principle, by gathering more evidence. In the language of subsection 2.2, it is the variability in the predictive distribution across plausible parameterizations θ given the observed data.

A.2 Committal and Distributional Aleatoric Uncertainty

Within the aleatoric component, our work distinguishes two sub-types that capture different aspects of how a model commits to a prediction. Consider a language model deciding which token to generate next, with a vocabulary of $50{,}000$ tokens. Three illustrative cases bring out the distinction:

Case 1: One token clearly dominates. The model assigns probability 0.95 to the token `the`, with the remaining 0.05 spread roughly evenly across the other $49{,}999$ tokens. *Committal* uncertainty is low: the Bernoulli variance $0.95 \cdot 0.05 = 0.0475$ is close to zero, indicating that the model is firmly committed to `the`. *Distributional* uncertainty is also low: the entropy is small because almost all the mass sits on a single token. This is the regime we call **concentration**: the top token wins by being likely.

Case 2: One token nominally wins, but mass is broadly distributed. The model assigns probability 0.15 to `the`, with the remaining 0.85 spread roughly evenly across $1{,}000$ other tokens that each receive roughly 0.00085. *Committal* uncertainty is high: the Bernoulli variance $0.15 \cdot 0.85 = 0.13$ is substantial relative to its maximum of 0.25, indicating that the model's top choice carries little of the total probability mass. *Distributional* uncertainty is also high: the entropy is large because mass is spread across many alternatives. This is the regime we call **elimination**: the token `the` wins not by being likely in absolute terms, but by being marginally more probable than each of many alternatives that are individually even less probable.

Case 3: Two tokens roughly tied. The model assigns probability 0.5 to `the` and 0.5 to `a`, with negligible mass elsewhere. *Committal* uncertainty is at its maximum: the Bernoulli variance $0.5 \cdot 0.5 = 0.25$ reflects that the top token wins by an arbitrarily small margin. *Distributional* uncertainty is low: the entropy is just $\log_2 2 = 1$ bit, because only two tokens carry mass. This case sits at the boundary of the elimination regime, with the elimination occurring among only two alternatives rather than many; with so little of the mass distributed, distributional uncertainty stays low even though committal uncertainty is maximal.

The cases above show that the two measures can move independently: the elimination regime (Case 2) and a near-tied two-way contest (Case 3) both produce high committal uncertainty, but only Case

2 also produces high entropy. The same low-entropy reading can therefore correspond to confident commitment (Case 1) or to a near-tied two-way contest (Case 3), and only committal uncertainty distinguishes them.

B Extracting the Final Answer, $\hat{y}$

Computing answer uncertainty $\mathbb{U}^{\mathrm{A}}$ requires identifying, for each generation, the span $\hat{y}$ corresponding to the model's final answer. We split each model's output into a trace section and an answer section using model-specific rules, since different model families structure their outputs differently. In all cases the split is a position in the generated string, with the concatenation of the two sections reproducing the original output exactly. The strategies were developed iteratively by inspecting splits on samples of correct and incorrect generations from each model-dataset combination, with the additional check that trace and answer sections concatenate back to the original output character-for-character.

Reasoning models (DeepSeek R1, Qwen 3, Qwen 3.5). All three models wrap their internal reasoning in explicit `<think>...</think>` tags. We first locate the closing `</think>` tag and then search the post-think content for the latest occurrence of an answer-declaration pattern, such as "Final Answer", "## Final Answer", "Answer:", or "Therefore, the answer is". The split is placed immediately after this declaration, so the answer section contains only the answer itself and not the surrounding markup. By construction (see subsection 3.2), every retained reasoning model trace contains a closing `</think>` tag and a recognizable answer declaration: traces missing these were either regenerated with an additional 512-token budget or filtered out if the extended budget still did not yield an answer.

Non-reasoning models (Qwen 2.5, Llama 3.1, Llama 3.2). These models lack explicit reasoning sections and instead structure their outputs as plain text in which the final answer is conventionally stated in a concluding paragraph. We split at the last paragraph break (an empty line, matched as `\n\s*\n`), taking everything after this break as the answer. If no paragraph break is found, we fall back to the last single newline; if neither is present and cannot reliably split the section we exclude the sample from the $\mathbb{U}^{\mathrm{A}}$ computation.

C Trace Features

Based on the generated traces and the uncertainty measure we compute the following metrics for each type of uncertainty. Recall that the uncertainty types are the entropy of the predictive distribution, an approximation to the aleatoric uncertainty and finally an approximation to the epistemic uncertainty as described in subsection 3.1.

Feature	Symbol	Description
Static Features		
Early mean	μ_{early}	Arithmetic mean of the first 25% trace.
Middle mean	μ_{mid}	Arithmetic mean of the middle 50% trace.
Late Mean	μ_{late}	Arithmetic mean of the last 25% trace.
Dynamic Features		
Slope	$\mathbf{m}$	Slope of a linear trend fitted on a trace.
Linear fit	r^2	Goodness-of-fit of the linear trend.

Table 3: Overview of features extracted from every uncertainty trace.

D EXPERIMENTAL SETUP

This appendix lists implementation details that are needed for reproducibility but kept out of the main text. The model lineup, datasets, generation protocol, uncertainty measures, and feature definitions themselves are described in subsection 3.2 and the trace/answer splitting in the section above.

D.1 SAMPLE RETENTION

Two filters are applied to each generated sample at load time, before any analysis:

1. **`contains_answer` check.** A regex-based extractor attempts to pull a final answer from the answer portion of each generation. Generations for which extraction fails are dropped.
2. **LLM correctness re-evaluation.** For samples whose extracted answer differs from the reference under the regex matcher, we query an LLM auditor (`gpt-5-2025-08-07`) to re-evaluate correctness under a more lenient semantic match (e.g., 0.5 vs $1/2$, mathematically equivalent expressions). When the auditor confirms correctness, we flip `is_correct` to `True`; otherwise we keep the original label.

For Qwen 3 and DeepSeek R1, traces that exhaust the 2048-token budget without emitting a closing `</think>` tag are regenerated with a forced `</think>` injection and an additional 512-token continuation; traces still missing a final answer after the regeneration are dropped.

D.2 PREDICTIVE MODELING

The two classifiers in subsection 4.1 are instantiated as

- `LogisticRegression(max_iter=1000, random_state=42)` with default L2 regularization ($C = 1$);
- `GradientBoostingClassifier(n_estimators=100, max_depth=3, random_state=42)`,

both with `StandardScaler` fit on the training fold inside the CV loop. Cross-validation uses `StratifiedKFold(n_splits=5, shuffle=True, random_state=42)`, constructed independently per (model, dataset) combination.

E COMPUTE AND REPRODUCIBILITY

All classifiers and CV splits are seeded with `random_state=42`. Trace generation is greedy decoding, deterministic given the model weights and tokenizer. All experiments are run on H100 GPUs. Source code is released alongside the paper.

Stage	Time / sample (s)	Relative to inference	Incr. peak memory (GB)
Standard inference (baseline)	3.34	1.0×	—
Trace-level aleatoric (U_D^{Tr}, U_C^{Tr})	0.06	0.02×	+0.13
Answer-level aleatoric (U^A)	4.11	1.2×	+0.13
Trace-level epistemic (U_E^{Tr})	21.2	6.3×	+5.69
Answer-level epistemic (U_E^A)	24.4	7.3×	+4.03
Full profile	49.8	14.9×	+5.69 (max)

Table 4: Computational cost per pipeline stage for Llama 3.1 8B on 100 GSM8K problems on a single H100. Times are mean per sample. Memory is incremental peak GPU memory on top of the 30.0 GB base load of standard inference. The full profile is the sum of the four uncertainty stages, excluding the inference pass itself.

F Additional Analysis

F.1 AUROC by Uncertainty Type

We distinguish between type-channel combinations, and observe that the trace-level uncertainties, $\mathbb{U}^{\mathrm{Tr}}$, are generally more predictive than answer-level uncertainties, $\mathbb{U}^{\mathrm{A}}$. Additionally, both aleatoric and epistemic uncertainty carry relevant predictive power as seen in columns $\mathbb{U}_{\mathrm{D}}$ and $\mathbb{U}_{\mathrm{E}}$.

		GSM8K		ProntoQA	
		LR	GB	LR	GB
$\mathbb{U}^{\mathrm{Tr}}$	Llama 3.1	0.781	0.741	0.777	0.712
	Llama 3.2	0.754	0.747	0.538	0.553
	Qwen 2.5	0.787	0.777	0.521	0.501
	DeepSeek R1	0.765	0.753	0.686	0.647
	Qwen 3	0.722	0.692	0.674	0.578
$\mathbb{U}^{\mathrm{A}}$	Llama 3.1	0.682	0.670	0.669	0.714
	Llama 3.2	0.675	0.660	0.544	0.529
	Qwen 2.5	0.755	0.716	0.566	0.481
	DeepSeek R1	0.642	0.629	0.660	0.638
	Qwen 3	0.685	0.616	0.619	0.502

Table 5: AUROC scores (5-fold CV, curated features) across models and classifiers for GSM8K and ProntoQA. Values shown as mean ± std. LR = Logistic Regression, GB = Gradient Boosting.

		GSM8K			ProntoQA		
		$\mathbb{U}_{\mathrm{D}}$	$\mathbb{U}_{\mathrm{C}}$	$\mathbb{U}_{\mathrm{E}}$	$\mathbb{U}_{\mathrm{D}}$	$\mathbb{U}_{\mathrm{C}}$	$\mathbb{U}_{\mathrm{E}}$
$\mathbb{U}^{\mathrm{A}}$	Llama 3.1	0.682	0.630	0.572	0.629	0.603	0.523
	Llama 3.2	0.634	0.582	0.587	0.537	0.516	0.529
	Qwen 2.5	0.718	0.655	0.678	0.571	0.539	0.496
	DeepSeek R1	0.613	0.549	0.568	0.693	0.477	0.658
	Qwen 3	0.657	0.659	0.636	0.525	0.679	0.555
$\mathbb{U}^{\mathrm{Tr}}$	Llama 3.1	0.717	0.705	0.778	0.811	0.776	0.712
	Llama 3.2	0.684	0.679	0.757	0.532	0.520	0.512
	Qwen 2.5	0.788	0.786	0.738	0.544	0.562	0.517
	DeepSeek R1	0.703	0.615	0.736	0.664	0.679	0.504
	Qwen 3	0.723	0.717	0.549	0.600	0.558	0.528

Table 6: AUROC scores (Logistic regression with 5-fold CV) by uncertainty type across models for GSM8K and ProntoQA. Values shown as mean ± std.

F.2 ADDITIONAL BENCHMARKS

	FOLIO					BHH				
	UTP	*SC*	*DC*	*#Tok.*	*Ent.*	UTP	*SC*	*DC*	*#Tok.*	*Ent.*
Llama 3.1	0.619	0.456	0.449	**0.625**	0.487	**0.678**	0.542	0.592	0.527	0.560
Llama 3.2	0.499	0.509	0.511	0.487	**0.529**	**0.609**	0.526	0.557	0.590	0.596
Qwen 2.5	0.539	**0.541**	0.538	0.532	0.483	0.624	0.520	0.590	**0.628**	0.613
DeepSeek R1	**0.650**	0.626	0.618	0.563	0.460	**0.703**	0.621	0.665	0.638	0.631
Qwen 3	**0.609**	0.569	0.566	0.561	0.567	**0.640**	0.484	0.492	0.529	0.446
	CRV-Bool					CRV-Arit.				
	UTP	*SC*	*DC*	*#Tok.*	*Ent.*	UTP	*SC*	*DC*	*#Tok.*	*Ent.*
Llama 3.1	—	—	—	—	—	**0.984**	0.637	0.725	0.689	0.729
Llama 3.2	0.630	0.616	**0.649**	0.482	0.572	**0.861**	0.720	0.710	0.543	0.713
Qwen 2.5	0.567	0.533	0.553	0.438	**0.588**	**0.864**	0.618	0.660	0.658	0.456
DeepSeek R1	0.621	**0.627**	0.605	0.571	0.614	**0.901**	0.711	0.653	0.513	0.535
Qwen 3	**0.647**	0.535	0.509	0.526	0.463	**0.912**	0.555	0.544	0.564	0.461

Table 7: Mean AUROC scores (5-fold CV) across five models for FOLIO, BHH, CRV-Bool and CRV-Arit. UTP = best of Logistic Regression and Gradient Boosting, *SC* = *Self-Certainty*, *DC* = *DeepConf*, *#Tok.* = *Generated Tokens*, *Ent.* = *Average Token Entropy*. Llama 3.1 on CRV-Bool is omitted because it produced only four incorrect answers.

F.3 EARLY CORRECTNESS DETECTION

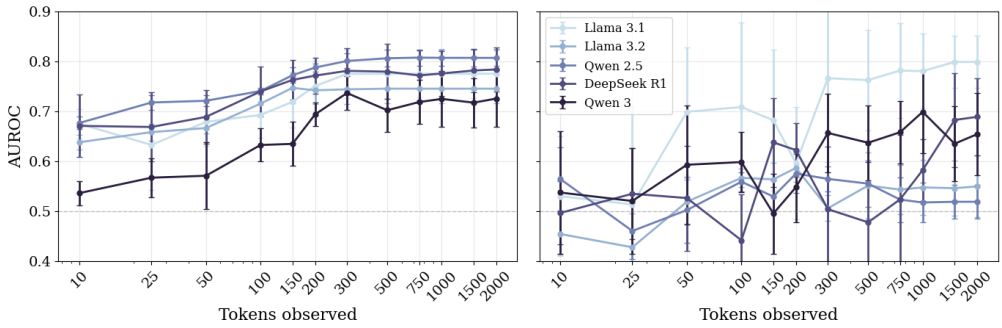

Figure 4: AUROC scores on models trained on features extracted based on increasing shares of the full traces (left: GSM8K, right: ProntoQA).

F.4 COEFFICIENT SHARE

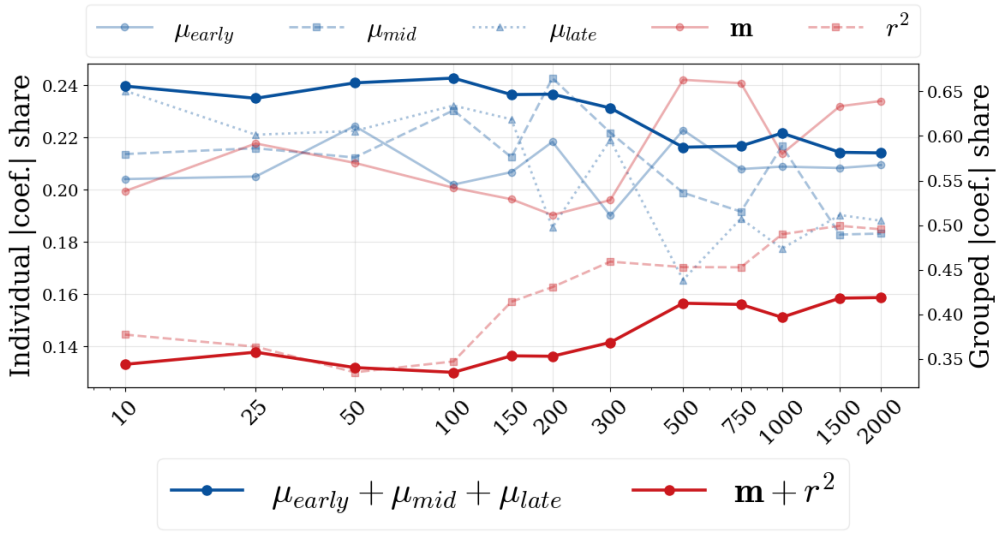

Figure 5: Change in coefficient share over the course of generation.