

ReasonBENCH: Benchmarking the (In)Stability of LLM Reasoning

Nearchos Potamitis¹ Lars Klein² Akhil Arora¹

Abstract

Large language models (LLMs) are increasingly deployed in settings where reasoning, such as multi-step problem solving and chain-of-thought, is essential. Yet, current evaluation practices overwhelmingly report single-run accuracy while ignoring the intrinsic uncertainty that naturally arises from stochastic decoding. This omission creates a blind spot because practitioners cannot reliably assess whether a method’s reported performance is stable, reproducible, or cost-consistent. We introduce REASONBENCH, the first benchmark designed to quantify the underlying instability in LLM reasoning. REASONBENCH provides (i) a modular evaluation library that standardizes reasoning frameworks, models, and tasks, (ii) a multi-run protocol that reports statistically reliable metrics for both quality and cost, and (iii) a public leaderboard to encourage variance-aware reporting. Across tasks from different domains, we find that the vast majority of reasoning strategies and models exhibit high instability. Notably, even strategies with similar average performance can display confidence intervals up to four times wider, and the top-performing methods often incur higher and less stable costs. Such instability compromises reproducibility across runs and, consequently, the reliability of reported performance. To better understand these dynamics, we further analyze the impact of prompts, model families, and scale on the trade-off between solve rate and stability. Our results highlight reproducibility as a critical dimension for reliable LLM reasoning and provide a foundation for future reasoning methods and uncertainty quantification techniques. REASONBENCH is publicly available at <https://github.com/au-clan/ReasonBench>.

¹Aarhus University, Aarhus, Denmark ²EPFL, Lausanne, Switzerland. Correspondence to: Nearchos Potamitis <nearchos.potamitis@cs.au.dk>.

1. Introduction

Recent studies highlight a growing tension between the promise of large language models (LLMs) and the risks of their adoption. On the one hand, even the mere knowledge that advice originates from an AI system has been shown to induce over-reliance by users (Klingbeil et al., 2024). On the other hand, evidence demonstrates that larger and more instructable models are becoming less reliable (Zhou et al., 2024b). This combination creates a concerning dynamic: users are predisposed to trust LLM outputs while the models themselves may be increasingly unstable. Such risks are amplified in safety-critical domains such as medical decision-making, legal and financial reasoning, and autonomous systems, where unreliable outputs can carry severe consequences.

At the center of these concerns lies reasoning, which has become a primary frontier in the development of LLMs. Recent advances increasingly revolve around reasoning, whether through specialized frameworks (Wei et al., 2022; Yao et al., 2023a; Klein et al., 2025), reasoning-focused training regimes such as DeepSeek R1 and OpenAI o1 (Guo et al., 2025; Jaech et al., 2024), or tool-augmented reasoning systems like Anthropic’s Model Context Protocol and OpenAI’s Deep Research variants of flagship models. The demand for reliable reasoning is driven by some of the most impactful applications of LLMs: information seeking and search (Jin et al., 2025; Li et al., 2025), mathematical and formal logic reasoning including theorem proving (Yang et al., 2023; 2024a), and many other domains where structured problem solving is essential. While reasoning is not the only use case for LLMs, it has become a key driver of both research progress and practical deployment, making its robustness and reliability central to the field.

Traditionally, the behavior of machine learning algorithms has been framed through the bias–variance paradigm (Geman et al., 1992; Hastie et al., 2009). In this view, bias corresponds to systematic error, typically captured by measures of accuracy, while variance reflects the instability of results between runs and can be interpreted as a form of uncertainty. Although this perspective has long guided the analysis of classical ML algorithms, evaluations of LLMs, especially in reasoning tasks, have focused almost exclusively on bias by reporting average accuracy from single or

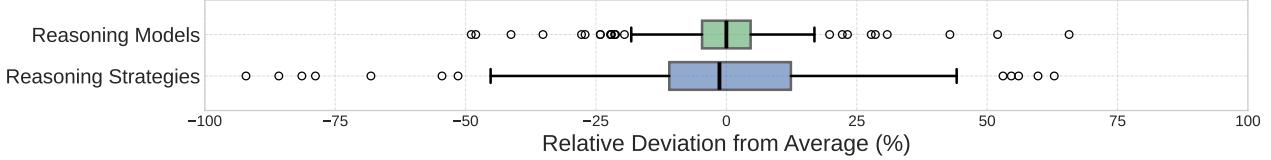
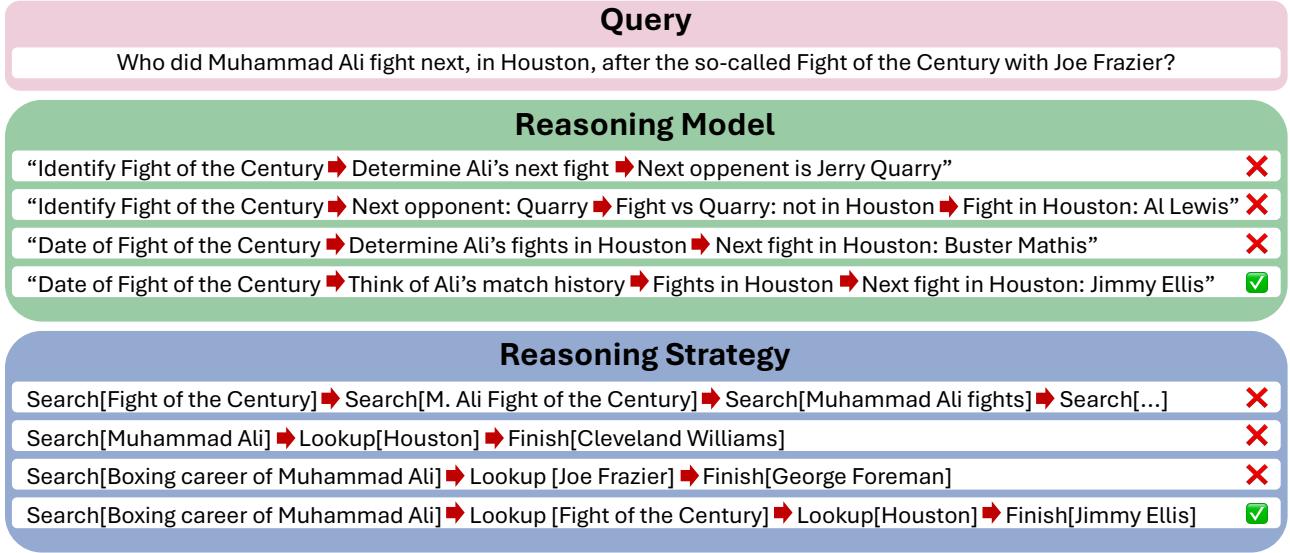


Figure 1. Instability in LLM Reasoning. For the same query, different reasoning models (top) and reasoning strategies (middle) produce distinct chains of thought and frequently contradictory conclusions. Even when working from identical instructions, methods vary widely in their intermediate reasoning steps and the correctness of their final answers. The bottom panel summarizes this variability quantitatively, showing the relative deviation from average performance across reasoning models and strategies.

very few runs. Consequently, we lack statistically reliable estimates of performance with confidence intervals, and instead rely on crude measurements that obscure the true instability of LLM reasoning. For many practical scenarios, and in particular safety-critical applications, it is not only the mean accuracy that matters but also the lower bound of the confidence interval or the worst-case performance, which determines whether a system can be trusted in deployment.

Present Work. In this paper, we revisit the oldest trick in experimental science: repeat the experiment. We conduct an in-depth evaluation of LLM reasoning by running 10 independent trials for each model–algorithm–task combination, and we report not only the mean but also the variance and confidence intervals of key performance metrics. Beyond evaluation, we address the practical challenge of reproducibility by releasing an agentic AI library as an artifact of this work, whose architecture is illustrated in Fig. 2. The library implements ten representative state-of-the-art reasoning algorithms and integrates with CacheSaver (Potamitis et al., 2025), a client-side inference optimization framework that enables reproducible and cost-efficient LLM-based experiments. This combination allows us to establish reproducible baselines, uncover the instability of LLM reasoning

strategies, and provide practitioners with statistically reliable performance estimates.

Contributions.

- We introduce the **ReasonBench AI Library**, the first benchmark of 11 different LLM reasoning methods across 4 different models and 7 different tasks with statistically reliable performance numbers(§ 3). Our framework offers a minimal, yet principled, abstraction layer over common patterns in agentic AI. Its versatility and expressiveness are showcased by our reference implementation of 11 diverse reasoning methods. By building on top of our API, researchers can implement new reasoning methods or tasks, through a guided framework, with only a few lines of code. Consequently, all evaluation routines are handled automatically, so researchers can avoid the complexity of building their own benchmarking scaffolds.
- We perform the first systematic *multi-run evaluation* of LLM reasoning algorithms across diverse models and tasks (§ 4). Each model–algorithm–task combination is evaluated with ten independent runs, and we report statistically reliable estimates of accuracy and cost with confidence intervals.

- We conduct an **insight analysis** of scaling effects and variance sources (§ 5). This includes comparisons across models of the same size but different families (e.g., Qwen-3B vs. Llama-3B), models of the same family but different sizes (e.g., Llama-3B vs. Llama-11B), correlations between cost and performance, as well as the impact of prompting on stability.
- Based on our findings, we release a **leaderboard** that evaluates models through the **lens of stability** and propose **best practices and a call to action** for variance-aware evaluation in LLM reasoning research (§ 6). We recommend reporting variance-aware metrics such as confidence intervals and percentiles, and we argue that reproducible multi-run evaluation should become the standard for reasoning benchmarks.

2. Related works

Instability in LLM Reasoning. A growing body of work highlights that LLM reasoning can be brittle and unstable. Benchmarks such as (Jiang et al., 2025; Wang & Zhao, 2024) show that small lexical or semantic changes to inputs can cause inconsistent reasoning chains and consequently large drops in performance. Similar insights emerge from perturbation studies in deductive logic and mathematics, including (Hoppe et al., 2025) and (Yang et al., 2025b). Beyond perturbations, survey work such as (Ahn et al., 2024) documents that models often arrive at different answers for identical problems via divergent reasoning paths. Stress-test frameworks such as (Hou et al., 2025) and (Huang et al., 2025) generate adversarial or out-of-distribution prompting variants to reveal systematic weaknesses in mathematical and commonsense reasoning. Across studies, the findings point to an endemic problem: LLM reasoning is highly sensitive to perturbations and randomness, making reproducibility an open problem.

Calls for Better Evaluation Practices. Alongside these studies, researchers are emphasizing the need for more rigorous evaluation methodologies. (Miller, 2024) summarizes the best-practice methodology from a statisticians toolbox and provides LLM-focused guidelines on reporting uncertainty, advocating for confidence intervals, clustered standard errors, and statistical tests based on question-level paired differences. Similar calls appear in (Mizrahi et al., 2023), which demonstrates the sensitivity of results to prompt wording, and in (Ni et al., 2024), which argues for aggregating across benchmarks to reduce instability. (Blackwell et al., 2024) argues that, even on simple QA benchmarks, repeated runs are required to reach statistically reliable conclusions. Survey contributions such as (Mondorf & Plank, 2024) echo this perspective, arguing that focusing on shallow accuracy metrics obscures important behavioral properties. Collectively, these works call for reproducibility,

uncertainty quantification, and explicit accounting for variance as essential components of reliable LLM evaluation.

Closely Related Variance-Aware Benchmarks. Only a few recent efforts go beyond calls to action and directly propose frameworks for variance-aware evaluation. (Liu et al., 2024) introduces the G -Pass@ k_τ metric to capture stability in reasoning tasks, though it condenses variability into a single scalar. (Madaan et al., 2024) studies variance from a different angle, analyzing differences across training seeds and checkpoints rather than stochastic decoding. (Ye et al., 2024) integrates uncertainty measures into multi-task benchmarking, showing that accuracy and certainty do not necessarily correlate. (Wang et al., 2025) derives theoretical sample complexity bounds to support statistically sound evaluations at lower cost. Autonomous or domain-specific benchmarks such as (Karia et al., 2024) and (Ji et al., 2025) highlight the growing recognition of reliability in evaluation, though they do not systematically address run-to-run variance.

Our Work. Our work builds on this trajectory by making stability across multiple, independent runs as the central object of our study. We echo the call to action for reliable benchmarking and reproducible science and claim that an important additional analysis is the sampling budget. We find that modern reasoning algorithms may reach state-of-the-art accuracy but only at a disproportionate cost. At the same time, the most sophisticated algorithms also seem to be the most brittle. The question of sample efficiency is closely related to reliable accuracy and reproducible results.

While prior efforts either stress brittleness under perturbations or argue for statistical rigor, REASONBENCH is, to our knowledge, the first benchmark that systematically quantifies stability across reasoning frameworks, models, and tasks through controlled multi-run evaluation. By coupling reproducible implementations of reasoning strategies with a variance-aware analysis, we aim to make stability and reliability first-class metrics in LLM reasoning research.

3. REASONBENCH

In this section, we provide a detailed description of our benchmarking framework, REASONBENCH, which we release as both a benchmark suite and an open-source AI library. REASONBENCH is designed with three goals in mind: (i) principled implementations of diverse reasoning strategies, (ii) reproducible and cost-efficient experimentation, and (iii) extensibility so the community can easily contribute new methods, models, or tasks.

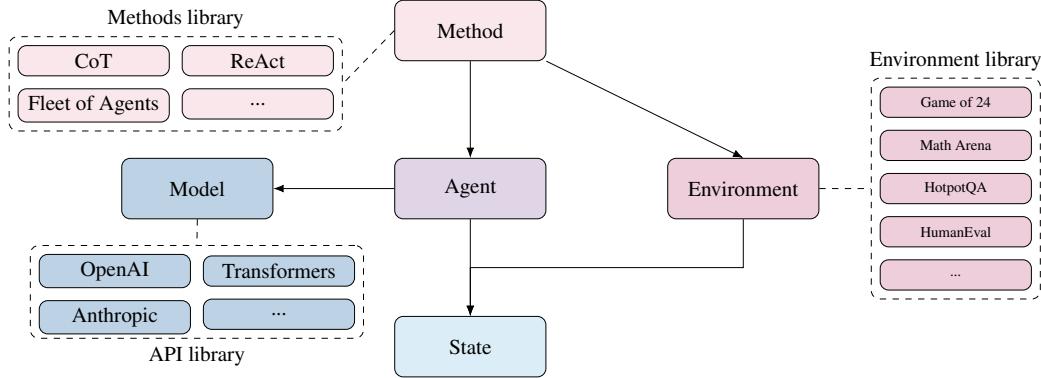


Figure 2. ReasonBench architecture. Methods orchestrate the three core components: Agents, Environments, and Models. Agents translate states into prompts, query models, and parse responses into actions. Environments are drawn from a large task library and offer functions such as next-step transitions and scoring heuristics. Models provide a unified interface to external LLM APIs. States record the intermediate configurations of reasoning, enabling reproducibility and fair comparison across tasks and methods.

3.1. Library design

REASONBENCH is organized around a set of core abstractions that capture the building blocks of reasoning pipelines. The principal components are the Method, Environment, Agent, State, and Model, which together define a modular interface for implementing reasoning algorithms, connecting to LLMs, and interacting with tasks. In designing these components, we followed established principles from software architecture engineering, emphasizing modularity, separation of concerns, and extensibility. Fig. 2 illustrates the relationships between these abstractions.

At a high level, the Model abstraction provides standardized access to language models, while Agents translate states into prompts and parse model outputs into actions. The Environment governs how these actions modify States and how solutions are evaluated. Methods sit on top of these components, orchestrating agents, environments, and models into complete reasoning strategies that can be executed and benchmarked in a uniform way. This layered design allows REASONBENCH to support both simple prompting baselines and complex search-based algorithms, while ensuring fair comparison across tasks, models, and evaluation metrics. We next describe each abstraction in detail.

Method Abstraction. The method abstraction specifies the overall logic of a reasoning strategy independently of the underlying model or task. A method integrates agents, which construct prompts and parse responses; the environment, which maintains and updates the task state; and the model, which produces candidate outputs. Each method exposes a standard interface for solving tasks by generating and updating sequences of states, and a benchmarking routine that runs multiple problem instances in parallel. This makes methods interchangeable and extensible: once the interface is implemented, a new reasoning algorithm can be evaluated

consistently across models, tasks, and metrics within the benchmarking pipeline.

Environment Abstraction. The environment abstraction formalizes the task-specific dynamics of reasoning. It governs how a state evolves in response to an action, how to determine whether an action is valid, when a trajectory has reached a terminal condition, and how to evaluate the final outcome. By encapsulating these rules, the environment decouples domain logic from reasoning algorithms, allowing the same method to be applied consistently across tasks while ensuring that actions and evaluations remain faithful to each benchmark.

Agent Abstraction. The agent abstraction defines the interface between methods, models, and states. Agents specify how prompts are constructed from the current state, how queries are issued to the model, and how responses are parsed into actions that update the environment. This unified interface makes it possible to express a wide spectrum of reasoning strategies: from simple input–output prompting to multi-step reasoning, search procedures, candidate aggregation, and self-evaluation. By isolating prompt construction and response handling, ReasonBench supports diverse reasoning paradigms without altering the abstractions for methods, environments, or models.

State Abstraction. The state abstraction captures the intermediate configuration of a reasoning process. It provides a standardized way to represent progress on a task and to handle states with controlled randomness. Methods interact only with states, while environments define how actions modify them and how final outcomes are assessed. This separation ensures that reasoning trajectories can be reproduced, compared, and analyzed independently of the underlying task domain.

Model Abstraction. The model abstraction provides a uniform interface for interacting with language models, supporting both single and batched queries across diverse providers. Built on top of asynchronous execution (via *asyncio*) and integrated with response caching through *CacheSaver*, it is both extensible and accountable: new models can be added without modifying the framework, and every interaction logs latency, token usage, and generation metadata. This combination enables deterministic reproducibility across repeated experiments while distinguishing between newly generated, reused, and deduplicated outputs.

3.2. Experimental Setup

Number of runs. We repeat all experiments 10 times and report both mean and confidence intervals of the evaluation metrics.

Prompts. To ensure a fair evaluation of the benchmarked reasoning strategies, we reuse the prompts introduced by prior methods. Whenever two strategies can utilize the same prompt, we use a shared version to enable direct comparison. For cases without existing prompts, e.g., novel reasoning strategy or base LLMs, if needed, we adapt the original prompts to facilitate the new use cases.

Tasks and data. We evaluate on five benchmark tasks selected to cover a broad spectrum of reasoning, planning, and general problem-solving abilities. These tasks span diverse domains: (1) mathematical reasoning: Game of 24 (Yao et al., 2023a) and MathArena (Balunović et al., 2025), (2) coding: HumanEval (Chen et al., 2021), (3) question answering and knowledge reasoning: HotpotQA (Zhilin et al., 2018) and Humanity’s Last Exam (Phan et al., 2025), (4) scientific reasoning: SciBench (Wang et al., 2024a), and (5) creative writing: Shakespearean Sonnet Writing (Suzgun & Kalai, 2024). For consistency, we rely on the test sets released with the original benchmarks.

Reasoning strategies. We experiment with 11 representative state-of-the-art reasoning strategies: (1) IO prompting, (2) CoT, (3) CoT-SC, (4) React (Yao et al., 2023b), (5) Reflexion, (6) ToT-DFS (Yao et al., 2023a), (7) TOT-BFS (Yao et al., 2023a), (8) GoT, (9) RAP (Hao et al., 2023), (10) ReST-MCTS* (Zhang et al., 2024), and (11) FoA (Klein et al., 2025). To ensure that comparisons between methods are *fair*, each strategy has been re-implemented within ReasonBench using a standardized interface, which harmonizes prompt handling, state transitions, and evaluation. Our selection criterion requires that methods provide publicly available code for at least one of the tasks considered in this study. Consequently, we exclude TouT (Mo & Xin, 2024), and RecMind (Wang et al., 2024b). We also omit BoT (Yang et al., 2024b), where the code is released but a

key resource (the meta-buffer) is missing, preventing reproducibility. LATS (Zhou et al., 2024a) is excluded due to its prohibitive computational cost.

Reasoning models. We evaluate a diverse set of contemporary reasoning models spanning multiple providers: (1) GPT-OSS-120B (Agarwal et al., 2025), (2) DeepSeek R1 (Guo et al., 2025), (3) Llama 4 Scout (AI, 2025), (4) Qwen3-32B (Yang et al., 2025a), and (5) Gemini 2.5 Pro (Comanici et al., 2025). These models represent the latest generation of systems that aim to perform end-to-end reasoning, without requiring explicit scaffolding through external frameworks. To ensure comparability, all models are evaluated in a zero-shot setting using identical benchmark prompts, with decoding parameters harmonized across providers. Our selection criterion prioritizes flagship reasoning-oriented releases from major labs that are accessible via public APIs at the time of writing.

Evaluation metrics. We evaluate along two dimensions: *quality*, and *cost* (token usage and running time). Cost is reported in USD. For locally hosted LLMs, we compute cost by counting input/output tokens and applying a provider’s pricing for the corresponding model.

4. Experiments

Our results are structured around two complementary questions: (i) how do different reasoning frameworks compare when applied under identical model conditions, and (ii) how do different reasoning models perform when asked to solve benchmarks directly without additional framework support. To answer the first question, we fix GPT-4.1-Nano as the underlying model and evaluate eleven representative reasoning frameworks across all benchmarks. To address the second, we evaluate multiple open- and closed-source reasoning models in a zero-shot setting, measuring their ability to solve tasks without external scaffolding. This separation allows us to disentangle the contribution of explicit reasoning frameworks from that of models designed to reason end-to-end. The resources for reproducing our experiments are available at <https://github.com/au-clan/ReasonBench>.

4.1. Reasoning Strategies

In Table 1, we fix GPT-4.1 as the underlying model and systematically compare the eleven reasoning strategies across all benchmarks. Each framework is executed with ten independent runs per task, and we report both average performance and variance, including confidence intervals and percentile statistics.

Across the evaluated frameworks, we observe that increased methodological sophistication generally corresponds to im-

Table 1. Quality and cost variability across reasoning frameworks under GPT-4.1. Direct methods show low cost but high instability in quality, while structured and planning-based approaches incur higher cost with mixed consistency. FoA and ToT-BFS deliver the most stable performance overall, whereas GoT exhibits the highest variability, highlighting substantial differences in robustness across reasoning paradigms.

Reasoning Strategy	Type	Quality			Cost		
		Mean ± CI	CV	MAD	Mean ± CI	CV	MAD
IO	Direct	3.0 ± 0.8	0.62	2.1	0.01 ± 0.02	0.41	0.01
CoT (Wei et al., 2022)	Direct	8.0 ± 1.6	0.38	3.2	0.02 ± 0.02	0.29	0.01
CoT-SC (Wang et al., 2023)	Direct	15.0 ± 1.2	0.14	2.8	0.15 ± 0.06	0.10	0.02
ReAct (Yao et al., 2023b)	Adaptive	31.0 ± 2.1	0.12	3.6	0.03 ± 0.02	0.03	0.00
Reflexion (Shinn et al., 2023)	Adaptive	25.0 ± 1.1	0.09	3.4	0.06 ± 0.03	0.26	0.02
ToT-DFS (Yao et al., 2023a)	Structured	25.0 ± 4.5	0.39	4.8	1.05 ± 0.09	0.13	0.12
ToT-BFS (Yao et al., 2023a)	Structured	35.0 ± 2.1	0.08	2.8	1.10 ± 0.05	0.04	0.05
GoT (Besta et al., 2024)	Structured	10.0 ± 2.4	0.58	4.9	1.55 ± 0.09	0.02	0.02
RAP (Hao et al., 2023)	Planning	22.0 ± 2.4	0.20	2.6	1.60 ± 0.37	0.15	0.18
MCTS* (Zhang et al., 2024)	Planning	33.0 ± 1.9	0.08	1.9	1.55 ± 0.28	0.13	0.16
FoA (Klein et al., 2025)	Evolutionary	36.0 ± 1.4	0.05	1.3	0.42 ± 0.05	0.05	0.02

Table 2. Quality and cost variability of contemporary reasoning models across all benchmarks. DeepSeek R1 achieves the strongest and most stable quality, though at the highest cost, while Llama 4 Maverick offers competitive performance with minimal cost. GPT-OSS-120B shows moderate stability, whereas Qwen3-235B exhibits the highest variability, underscoring that model price and scale do not reliably correspond to consistency.

Reasoning Model	Provider	Quality			Cost		
		Mean ± CI	CV	MAD	Mean ± CI	CV	MAD
DeepSeek R1	DeepSeek	48.7 ± 4.3	0.293	8.0	1.97 ± 0.16	0.27	0.47
Llama 4 Maverick 17B	Meta	45.7 ± 4.5	0.380	11.0	0.03 ± 0.00	0.31	0.01
GPT-OSS-120B	OpenAI	48.5 ± 5.9	0.473	18.0	0.04 ± 0.01	0.37	0.01
Qwen3-235B A22B	Alibaba	46.2 ± 12.4	0.773	26.3	0.78 ± 0.08	0.31	0.21

proved solution quality, but this relationship is neither monotonic nor uniformly reliable. While several complex approaches, such as FoA and MCTS*, achieve the highest mean performance with comparatively tight confidence intervals, other equally intricate methods like GoT, ToT-BFS, and ToT-DFS exhibit substantial instability, suggesting that complexity alone does not guarantee robustness. Variance emerges as a critical factor affecting both quality and cost, yet these forms of variance behave independently: some methods (e.g., ReAct, FoA) simultaneously deliver high quality and low dispersion across metrics, whereas others (e.g., GoT) show low cost variance but large fluctuations in quality. These results underscore that the benefits of complex reasoning frameworks depend not only on their structural depth but also on the stability of their underlying search or adaptation mechanisms, emphasizing the importance of evaluating performance and cost stability jointly rather than relying solely on average outcomes.

4.2. Reasoning Models

In Table 2, we evaluate a set of contemporary reasoning models by directly asking them to solve the benchmarks without any external framework support. Each model is run independently ten times per task, and we report mean accuracy, confidence intervals, and percentile statistics alongside token-level cost. This experiment captures the intrinsic reasoning ability of the models in a zero-shot setting and enables a variance-aware comparison across providers.

Our results indicate that inference price is not a reliable proxy for consistency across contemporary reasoning models. Although DeepSeek R1 achieves the strongest and most stable performance, its consistency advantages over substantially cheaper systems such as Llama 4 Maverick remain unexpectedly narrow, suggesting diminishing returns at higher cost tiers. Conversely, Qwen3-235B A22B exhibits the biggest variance despite being more than twenty times more expensive than both GPT-OSS-120B from OpenAI

and Llama 4 Maverick from Meta, with variability metrics more than double those of these lower-cost models. This mismatch between price and consistency underscores that current model pricing does not reliably reflect stability, and that some low-cost models offer competitive or superior stability relative to far more expensive alternatives.

5. Analysis

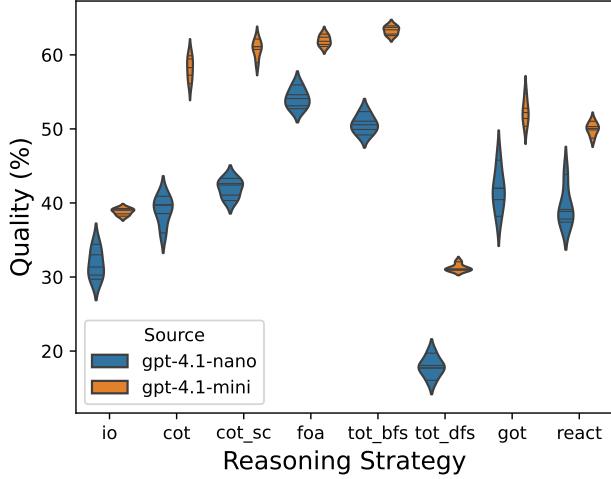


Figure 3. Scaling Effects within a Model Family. Quality distributions for gpt-4.1-nano and gpt-4.1-mini across multiple reasoning strategies. The larger model achieves higher mean performance and exhibits markedly lower variance, suggesting greater stability in its reasoning behavior.

5.1. Scaling Effects within a Model Family

We analyze the stability of reasoning performance within a single model family at different scales. We consider GPT-4.1-Nano and GPT-4.1-Mini, evaluating them on all benchmarks with ten independent runs. This experiment highlights the effect of scaling within one architecture, allowing us to observe whether increased size systematically improves not only average quality but also stability across runs. The results can be found in Fig. 3.

Across all strategies, we observe a consistent scaling effect: GPT-4.1-Mini achieves higher mean quality and exhibits substantially tighter distributions than GPT-4.1-Nano. This indicates that increasing model size within the same family not only improves average performance but also reduces run-to-run variability, leading to more stable reasoning behavior overall.

5.2. Impact of prompts on stability

A nontrivial portion of instability stems not from the reasoning algorithms themselves but from the prompts and parsers that mediate their interaction with LLMs. Prompts often

contain minor ambiguities, loosely specified answer styles, or implicit assumptions about how models structure their reasoning. It is possible that these can magnify stochastic differences and lead to divergent outputs across runs. In REASONBENCH, we make small, fidelity-preserving refinements to these prompts, clarifying instructions and standardizing output expectations without altering the underlying reasoning logic. In tandem, we strengthen the parsing layer to robustly extract answers despite common formatting deviations, reducing failure cases caused by parsing brittleness rather than genuine reasoning errors.

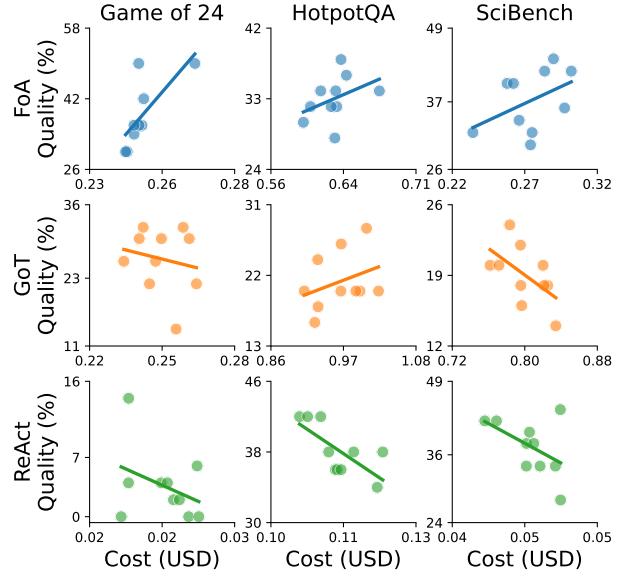


Figure 4. Correlation between Quality and Cost. For FoA, quality scales positively with cost across all benchmarks. ReAct exhibits a consistent negative slope, indicating diminishing returns at higher costs. GoT does not follow a uniform pattern, with its cost-quality relationship varying substantially by task.

Across all frameworks, clarifying prompts and strengthening the parsing logic consistently reduce variance, indicating that a meaningful portion of instability comes from avoidable formatting ambiguities rather than true algorithmic randomness. While every method benefits from these improvements, structured and search-based approaches show the largest reductions, suggesting that multi-step frameworks are especially sensitive to prompt clarity and output handling. The detailed results can be found in Table 3.

These observations highlight a broader challenge in LLM evaluation: benchmarking pipelines themselves are not static artifacts but evolving systems. In practice, evaluation procedures and prompting conventions continually shift, minor prompt edits are rarely recorded and changes to third-party APIs are easily overlooked. Yet benchmarking results are only meaningful when they support reliable comparison, making it essential to rerun evaluations when needed and to maintain up-to-date performance measurements. REA-

Table 3. Impact of prompt and parsing refinements on framework performance. Enhancing clarity and standardizing output parsing consistently improve both accuracy and stability, with structured and search-based methods showing the largest gains.

Framework	Type	Original Prompts	Improved Prompts	Absolute Difference
IO	Direct	3.0 ± 0.8	31.3 ± 0.7	+28.3 ± 2.0
CoT (Wei et al., 2022)	Direct	8.0 ± 1.6	39.8 ± 1.4	+31.8 ± 2.7
CoT-SC (Wang et al., 2023)	Direct	15.0 ± 1.2	41.1 ± 1.1	+26.1 ± 1.8
ReAct (Yao et al., 2023b)	Adaptive	31.0 ± 2.1	39.1 ± 1.9	+8.1 ± 2.6
Reflexion (Shinn et al., 2023)	Adaptive	25.0 ± 1.1	41.1 ± 1.0	+16.1 ± 3.0
ToT-BFS (Yao et al., 2023a)	Structured	35.0 ± 2.1	50.6 ± 1.8	+15.6 ± 4.1
GoT (Besta et al., 2024)	Structured	10.0 ± 2.4	42.0 ± 2.2	+32.0 ± 3.5
RAP (Hao et al., 2023)	Planning	22.0 ± 2.4	40.3 ± 2.2	+18.3 ± 2.9
MCTS* (Zhang et al., 2024)	Planning	33.0 ± 2.5	51.2 ± 1.7	+18.2 ± 2.7
FoA (Klein et al., 2025)	Evolutionary	36.0 ± 1.4	54.6 ± 1.3	+18.6 ± 2.1

SONBENCH provides a practical remedy. Algorithms expressed through its simple but highly modular API can be rerun seamlessly, allowing results to be regenerated along with the updates.

5.3. Correlation between Quality and Cost

Finally, we investigate the relationship between the stability in quality and cost. Using all reasoning strategies, we take a more intrinsic look at variability by examining outcomes at the level of individual samples. For each run of each benchmark, we record whether the model’s answer was correct and measure the exact cost incurred for that attempt. This analysis tests whether methods that are unstable in terms of accuracy also tend to be unstable in cost, thereby probing a potential correlation between two critical dimensions of reproducibility. The results can be found in Fig. 4.

Across benchmarks, we observe distinct patterns linking cost and quality variability. FoA (Klein et al., 2025) exhibits a consistently positive relationship, with higher-cost samples tending to yield higher-quality outputs, indicating stable scaling behavior. In contrast, ReAct (Yao et al., 2023b) shows a negative slope on all tasks, suggesting that increased computational effort often corresponds to less reliable reasoning trajectories. GoT displays no uniform trend, with the cost–quality relationship flipping direction across benchmarks, reflecting the method’s sensitivity to task structure.

6. Discussion

6.1. Summary of Findings

Our study reveals that the underlying instability is a pervasive and underexamined property of LLM reasoning. Across frameworks, tasks, and models, we find that single-run accuracy can systematically overestimate the stability of reasoning performance, obscuring wide differences in both quality and cost consistency. More sophisticated reasoning algo-

rithms often achieve higher mean accuracy, but this does not guarantee robustness: several structured and search-based approaches exhibit substantial instability, while simpler or more adaptive methods can outperform them by being more stable. At the model level, inference price is not a reliable proxy for stability as some of the most expensive models show higher variability than significantly cheaper alternatives. Finally, even small refinements to prompts and parsing logic meaningfully reduce extrinsic variance, indicating that a nontrivial share of previously reported instability stemmed from preventable ambiguities rather than true model and framework behavior. Together, these findings underscore that reproducibility is a critical dimension of LLM reasoning and should be treated as a first-class metric alongside average performance.

6.2. Limitations and Future Work

While REASONBENCH provides the first systematic multi-run evaluation of reasoning frameworks and models, several limitations remain. First, our analysis focuses on decoding stochasticity; additional sources of variability such as API instability, and model updates, are important directions for deeper investigation. Second, our benchmark covers a representative but still limited set of frameworks, tasks, and proprietary reasoning-oriented models; expanding ReasonBench to more domains will enable broader conclusions. Third, our multi-run protocol uses ten repetitions, which we found sufficient for stable confidence intervals, but future work may explore adaptive or task-aware sampling budgets that balance statistical reliability with cost efficiency. Finally, the impact of prompt clarity suggests opportunities for systematic prompt optimization, controllable reasoning formats, and parser-aware training objectives designed to reduce variability at the source.

References

- Agarwal, S., Ahmad, L., Ai, J., Altman, S., Applebaum, A., Arbus, E., Arora, R. K., Bai, Y., Baker, B., Bao, H., et al. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*, 2025.
- Ahn, J., Verma, R., Lou, R., Liu, D., Zhang, R., and Yin, W. Large language models for mathematical reasoning: Progresses and challenges. *arXiv preprint arXiv:2402.00157*, 2024.
- AI, M. Introducing llama 4: Advancing multimodal intelligence, 2025. Accessed: 2025-09-22.
- Balunović, M., Dekoninck, J., Petrov, I., Jovanović, N., and Vechev, M. Matharena: Evaluating llms on uncontaminated math competitions. *arXiv preprint arXiv:2505.23281*, 2025.
- Besta, M., Blach, N., Kubicek, A., Gerstenberger, R., Podstawska, M., Gianinazzi, L., Gajda, J., Lehmann, T., Niewiadomski, H., Nyczek, P., et al. Graph of thoughts: Solving elaborate problems with large language models. In *AAAI*, volume 38, pp. 17682–17690, 2024.
- Blackwell, R. E., Barry, J., and Cohn, A. G. Towards reproducible lilm evaluation: Quantifying uncertainty in lilm benchmark scores. *ArXiv*, abs/2410.03492, 2024.
- Chen et al. Evaluating large language models trained on code, 2021. *arXiv eprint 2107.03374, cs.LG*.
- Comanici, G., Bieber, E., Schaeckermann, M., Pasupat, I., Sachdeva, N., Dhillon, I., Blstein, M., Ram, O., Zhang, D., Rosen, E., et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.
- Geman, S., Bienenstock, E., and Doursat, R. Neural networks and the bias/variance dilemma. *Neural computation*, 4(1):1–58, 1992.
- Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.
- Hao, S., Gu, Y., Ma, H., Hong, J. J., Wang, Z., Wang, D. Z., and Hu, Z. Reasoning with language model is planning with world model. In *EMNLP*, 2023.
- Hastie, T., Tibshirani, R., Friedman, J., et al. The elements of statistical learning, 2009.
- Hoppe, F., Ilievski, F., and Kalo, J.-C. Investigating the robustness of deductive reasoning with large language models. *arXiv preprint arXiv:2502.04352*, 2025.
- Hou, Y., Xiao, Z., Yu, F., Jiang, Y., Wei, X., Huang, H., Chen, Y., and Chen, G. Automatic robustness stress testing of llms as mathematical problem solvers. *arXiv preprint arXiv:2506.05038*, 2025.
- Huang, S., Yang, L., Song, Y., Chen, S., Cui, L., Wan, Z., Zeng, Q., Wen, Y., Shao, K., Zhang, W., et al. Thinkbench: Dynamic out-of-distribution evaluation for robust lilm reasoning. *arXiv preprint arXiv:2502.16268*, 2025.
- Jaech, A., Kalai, A., Lerer, A., Richardson, A., El-Kishky, A., Low, A., Helyar, A., Madry, A., Beutel, A., Carnay, A., et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.
- Ji, K., Guo, Y., Zhang, Z., Zhu, X., Tian, Y., Liu, N., and Zhai, G. Medomni-45 { \deg }: A safety-performance benchmark for reasoning-oriented llms in medicine. *arXiv preprint arXiv:2508.16213*, 2025.
- Jiang, E., Xu, C., Singh, N., and Singh, G. Misaligning reasoning with answers—a framework for assessing lilm cot robustness. *arXiv preprint arXiv:2505.17406*, 2025.
- Jin, B., Yoon, J., Kargupta, P., Arik, S. O., and Han, J. An empirical study on reinforcement learning for reasoning-search interleaved lilm agents. *arXiv preprint arXiv:2505.15117*, 2025.
- Karia, R., Bramblett, D., Dobhal, D., and Srivastava, S. Autonomous evaluation of llms for truth maintenance and reasoning tasks. *arXiv preprint arXiv:2410.08437*, 2024.
- Klein, L. H., Potamitis, N., Aydin, R., West, R., Gulcehre, C., and Arora, A. Fleet of agents: Coordinated problem solving with large language models. In *Forty-second International Conference on Machine Learning*, 2025.
- Klingbeil, A., Grützner, C., and Schreck, P. Trust and reliance on ai—an experimental study on the extent and costs of overreliance on ai. *Computers in Human Behavior*, 160:108352, 2024.
- Li, Y., Zhang, W., Yang, Y., Huang, W.-C., Wu, Y., Luo, J., Bei, Y., Zou, H. P., Luo, X., Zhao, Y., et al. Towards agentic rag with deep reasoning: A survey of rag-reasoning systems in llms. *arXiv preprint arXiv:2507.09477*, 2025.
- Liu, J., wei Liu, H., Xiao, L., Wang, Z., Liu, K., Gao, S., Zhang, W., Zhang, S., and Chen, K. Are your llms capable of stable reasoning? In *Annual Meeting of the Association for Computational Linguistics*, 2024.
- Madaan, L., Singh, A. K., Schaeffer, R., Poulton, A., Koyejo, S., Stenetorp, P., Narang, S., and Hupkes, D. Quantifying variance in evaluation benchmarks. *arXiv preprint arXiv:2406.10229*, 2024.

- Miller, E. Adding error bars to evals: A statistical approach to language model evaluations. *arXiv preprint arXiv:2411.00640*, 2024.
- Mizrahi, M., Kaplan, G., Malkin, D., Dror, R., Shahaf, D., and Stanovsky, G. State of what art? a call for multi-prompt llm evaluation. *Transactions of the Association for Computational Linguistics*, 12:933–949, 2023.
- Mo, S. and Xin, M. Tree of uncertain thoughts reasoning for large language models. In *ICASSP*, pp. 12742–12746, 2024.
- Mondorf, P. and Plank, B. Beyond accuracy: evaluating the reasoning behavior of large language models—a survey. *arXiv preprint arXiv:2404.01869*, 2024.
- Ni, J., Xue, F., Yue, X., Deng, Y., Shah, M., Jain, K., Neubig, G., and You, Y. Mixeval: Deriving wisdom of the crowd from llm benchmark mixtures. In Globerson, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J., and Zhang, C. (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 98180–98212. Curran Associates, Inc., 2024.
- Phan, L., Gatti, A., Han, Z., Li, N., Hu, J., Zhang, H., Zhang, C. B. C., Shaaban, M., Ling, J., Shi, S., et al. Humanity’s last exam. *arXiv preprint arXiv:2501.14249*, 2025.
- Potamitis, N., Klein, L. H., Mohammadi, B., Xu, C., Mukherjee, A., Tandon, N., Bindschaedler, L., and Arora, A. Cache saver: A modular framework for efficient, affordable, and reproducible LLM inference. In *EMNLP*, pp. 25703–25724, 2025. doi: 10.18653/v1/2025.findings-emnlp.1402.
- Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K., and Yao, S. Reflexion: language agents with verbal reinforcement learning. In *NeurIPS*, pp. 8634–8652, 2023.
- Suzgun, M. and Kalai, A. T. Meta-prompts: Enhancing language models with task-agnostic scaffolding. *arXiv preprint arXiv:2401.12954*, 2024.
- Wang, G., Chen, Z., Li, B., and Xu, H. Cer-eval: Certifiable and cost-efficient evaluation framework for llms. *arXiv preprint arXiv:2505.03814*, 2025.
- Wang, X., Wei, J., Schuurmans, D., Le, Q. V., Chi, E. H., Narang, S., Chowdhery, A., and Zhou, D. Self-consistency improves chain of thought reasoning in language models. In *ICLR*, 2023.
- Wang, X., Hu, Z., Lu, P., Zhu, Y., Zhang, J., Subramanian, S., Loomba, A. R., Zhang, S., Sun, Y., and Wang, W. Scibench: evaluating college-level scientific problem-solving abilities of large language models. In *ICML*, 2024a.
- Wang, Y. and Zhao, Y. Rupbench: Benchmarking reasoning under perturbations for robustness evaluation in large language models. *arXiv preprint arXiv:2406.11020*, 2024.
- Wang, Y., Jiang, Z., Chen, Z., Yang, F., Zhou, Y., Cho, E., Fan, X., Lu, Y., Huang, X., and Yang, Y. Recmind: Large language model powered agent for recommendation. In *NAACL-HLT (Findings)*, pp. 4351–4364, 2024b.
- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q. V., Zhou, D., et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.
- Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025a.
- Yang, K., Swope, A., Gu, A., Chalamala, R., Song, P., Yu, S., Godil, S., Prenger, R. J., and Anandkumar, A. Leandojo: Theorem proving with retrieval-augmented language models. *Advances in Neural Information Processing Systems*, 36:21573–21612, 2023.
- Yang, K., Poesia, G., He, J., Li, W., Lauter, K., Chaudhuri, S., and Song, D. Formal mathematical reasoning: A new frontier in ai. *arXiv preprint arXiv:2412.16075*, 2024a.
- Yang, L., Yu, Z., Zhang, T., Cao, S., Xu, M., Zhang, W., Gonzalez, J. E., and Cui, B. Buffer of thoughts: Thought-augmented reasoning with large language models. In *NeurIPS*, 2024b.
- Yang, Y., Yamada, H., and Tokunaga, T. Evaluating robustness of llms to numerical variations in mathematical reasoning. In *The Sixth Workshop on Insights from Negative Results in NLP*, pp. 171–180, 2025b.
- Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T., Cao, Y., and Narasimhan, K. Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems*, 36:11809–11822, 2023a.
- Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., and Cao, Y. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023b.
- Ye, F., Yang, M., Pang, J., Wang, L., Wong, D., Yilmaz, E., Shi, S., and Tu, Z. Benchmarking llms via uncertainty quantification. *Advances in Neural Information Processing Systems*, 37:15356–15385, 2024.
- Zhang, D., Zhoubian, S., Hu, Z., Yue, Y., Dong, Y., and Tang, J. ReST-MCTS*: LLM self-training via process reward guided tree search. In *NeurIPS*, 2024.

Zhilin, Y., Peng, Q., Saizheng, Z., Yoshua, B., William, C., Ruslan, S., and Manning, C. D. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018. doi: 10.18653/v1/d18-1259.

Zhou, A., Yan, K., Shlapentokh-Rothman, M., Wang, H., and Wang, Y.-X. Language agent tree search unifies reasoning, acting, and planning in language models. In *ICML*, 2024a.

Zhou, L., Schellaert, W., Martínez-Plumed, F., Moros-Daval, Y., Ferri, C., and Hernández-Orallo, J. Larger and more instructable language models become less reliable. *Nature*, 634(8032):61–68, 2024b.