

---

# ReasonBENCH: Benchmarking the (In)Stability of LLM Reasoning

---

Anonymous Authors<sup>1</sup>

## Abstract

Large language model (LLM) reasoning is typically evaluated using single runs, masking how much performance can vary across repeated executions. This practice obscures both reliability and cost, and can lead to misleading comparisons between reasoning methods and models. We introduce REASONBENCH, a benchmark suite and open-source library for controlled multi-run evaluation of LLM reasoning. For each model–strategy–task configuration, we perform repeated trials across 6 diverse benchmarks and report variance-aware metrics for both quality and cost, including confidence intervals and run-to-run variability measures. Using standardized implementations, we benchmark 10 widely used reasoning strategies under identical model conditions and evaluate 10 contemporary reasoning-oriented LLMs in a zero-shot setting. Our results show that run-to-run variability is substantial, benchmark-dependent, and often large enough to change model/method rankings relative to single-run averages. Additional analyses reveal that scaling within a model family improves both average quality and stability, while increasing test-time reasoning effort primarily increases cost without yielding statistically significant quality gains. Together, these findings motivate distribution-aware evaluation practices and provide reproducible tooling to support more reliable progress in LLM reasoning research. REASONBENCH is publicly available at <https://anonymous.4open.science/r/ReasonBench-64B3>.

## 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 instability may appear benign when the user wants more insights into Muhammad Ali’s match history (cf. Fig. 1), but is dangerous 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 strategies (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 (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 very few runs. Consequently, we lack statistically reliable estimates of performance with confidence intervals,

<sup>1</sup>Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

055  
056  
057  
058  
059  
060  
061  
062  
063  
064  
065  
066  
067  
068  
069  
070  
071

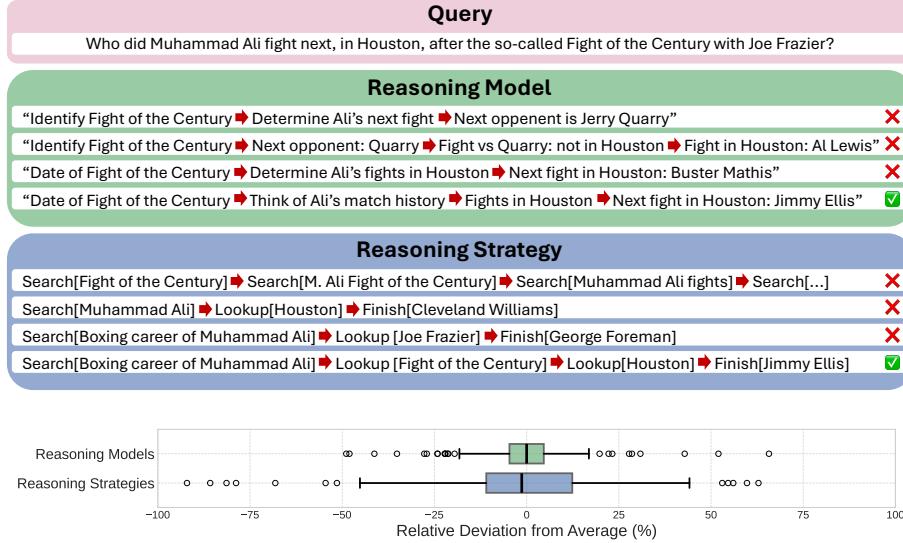


Figure 1. **Instability in LLM Reasoning.** For the same query, different reasoning models (**top**) and reasoning strategies (**middle**) produce distinct chains of thought and frequently contradictory conclusions. Even when working from identical instructions, methods vary widely in their intermediate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

and instead rely on crude measurements that obscure the true instability of LLM reasoning, which, as shown in Fig. 1 (bottom), is substantial across different reasoning applications. 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 10 different LLM reasoning strategies across 4 different models and 6 different tasks with statistically reliable performance numbers (§ 3). Our framework offers

a minimal, yet principled, abstraction layer over common patterns in agentic AI. By building on top of our API, researchers can implement new reasoning methods or tasks, through a guided evaluation framework, with only a few lines of code.

- We perform the **first systematic** multi-run evaluation of LLM reasoning strategies across diverse models and tasks (§ 4). Each model–algorithm–task combination is evaluated with ten independent runs, and we report statistically reliable estimates of quality and cost with confidence intervals.
- We conduct an **insight analysis** of the drivers of reasoning (in)stability (§ 5). Specifically, we study (i) **model scale** and its effect on both mean quality and run-to-run variability, (ii) the **impact of prompting**, (iii) **cost–quality correlations** across reasoning strategies, (iv) the effect of explicit **thinking-effort controls** on performance and stability, and (v) a **causal intervention on evaluation functions** showing improved performance and tighter confidence intervals for search-heavy methods.
- 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. The leaderboard is publicly available at <http://reasonbench.github.io>.

## 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

Table 1. Quality and cost variability across reasoning frameworks under GPT-4.1 Nano. Direct methods show low cost but high instability in quality, while structured and planning-based approaches incur higher cost with mixed consistency. FoA delivers the most stable performance overall, whereas RAP and ToT-BFS exhibits the highest noise across benchmarks and runs respectively. The best performance is shown in **blue** whereas the second best is shown in **orange**.

Strategy	Type	Quality				Cost			
		Average*	Run Deviation*	Noise (Global)	Noise (Run)	Average*	Run Deviation*	Noise (Global)	Noise (Run)
IO	Direct	<b>0.1063 [0.10, 0.12]</b>	13.66% [0.055, 0.229]	<b>0.1957</b>	<b>0.0153</b>	<b>0.0054 [0.01, 0.01]</b>	2.56% [0.008, 0.052]	0.0116	<b>0.0000</b>
CoT (Wei et al., 2022)	Direct	0.2761 [0.25, 0.30]	29.59% [0.152, 0.492]	0.6016	0.0940	0.0130 [0.01, 0.01]	4.73% [0.026, 0.072]	<b>0.0111</b>	0.0000
CoT-SC (Wang et al., 2023)	Direct	<b>0.2281 [0.21, 0.24]</b>	<b>65.54% [0.349, 1.809]</b>	0.3187	0.0427	<b>0.0682 [0.07, 0.07]</b>	<b>0.74% [0.003, 0.012]</b>	0.0649	0.0000
ReAct (Yao et al., 2023b)	Adaptive	0.2956 [0.28, 0.31]	29.14% [0.177, 0.704]	1.1289	0.0219	0.0697 [0.07, 0.07]	<b>6.45% [0.027, 0.125]</b>	0.0235	0.0002
Reflexion (Shin et al., 2023)	Adaptive	<b>0.2815 [0.27, 0.30]</b>	27.75% [0.146, 0.458]	1.3080	0.0413	0.1647 [0.15, 0.17]	4.79% [0.037, 0.061]	0.0315	0.0007
ToT-DFS (Yao et al., 2023a)	Structured	<b>0.1272 [0.10, 0.14]</b>	5.15% [0.012, 0.112]	1.2396	0.0353	0.1033 [0.10, 0.11]	3.55% [0.013, 0.059]	0.3541	0.0059
ToT-BFS (Yao et al., 2023a)	Structured	0.4073 [0.38, 0.44]	14.35% [0.054, 0.232]	0.4816	<b>0.1781</b>	0.4428 [0.43, 0.46]	4.82% [0.023, 0.081]	0.9883	<b>0.0064</b>
GoT (Besta et al., 2024)	Structured	<b>0.3361 [0.31, 0.36]</b>	15.64% [0.068, 0.279]	0.5101	0.1203	0.4971 [0.48, 0.51]	1.81% [0.009, 0.029]	1.2939	0.0025
RAP (Hao et al., 2023)	Planning	0.3669 [0.35, 0.38]	18.54% [0.117, 0.417]	<b>1.5461</b>	0.0273	<b>0.5320 [0.52, 0.54]</b>	4.19% [0.008, 0.096]	<b>1.6642</b>	0.0021
FoA (Klein et al., 2025)	Evolutionary	<b>0.4580 [0.43, 0.48]</b> †	<b>7.83% [0.030, 0.173]</b>	0.4716	0.1522	0.3237 [0.32, 0.33]	3.75% [0.016, 0.061]	0.3221	0.0010

† Indicates statistical significance ( $p < 0.05$ ) between the best and the second-best scores.

\* Reports average value and 95% confidence intervals in brackets.

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@ $k_\tau$  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 strategies, 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

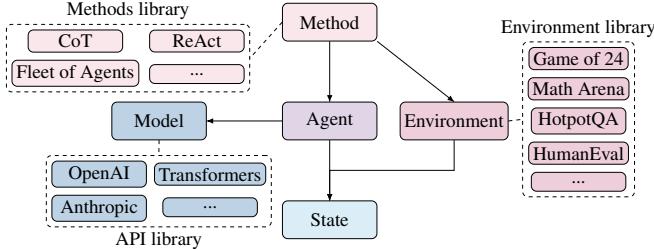


Figure 2. **ReasonBENCH architecture.** Methods orchestrate the three core components: Agents, Environments, and Models. Agents translate states into prompts, query models, and parse responses into actions. Environments are drawn from a large task library and offer functions such as next-step transitions 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.

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.

### 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.

**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

Table 2. Quality and cost variability of contemporary reasoning models across all benchmarks. Gemini-3 Flash achieves the strongest and most stable quality overall, though at the highest cost, while GPT-5 Mini offers competitive performance with minimal cost. GPT-4.1 Nano and DeepSeek-R1 show the worst performance with the latest having the worst overall variability in terms of both quality and cost. The best performance is shown in **blue** whereas the worst best is shown in **orange**.

Reasoning Model	Provider	Quality				Cost			
		Average*	Run Deviation*	Noise (Global)	Noise (Run)	Average*	Run Deviation*	Noise (Global)	Noise (Run)
DeepSeek R1	DeepSeek	0.2217 [0.20, 0.25]	17.81% [0.083, 0.324]	<b>1.1096</b>	0.0381	<b>1.3141</b> [ <b>1.27, 1.36</b> ]	4.69% [0.019, 0.078]	<b>0.7978</b>	<b>0.0168</b>
Llama 4 Maverick	Meta	0.4029 [0.38, 0.43]	8.27% [0.035, 0.162]	0.3797	0.0358	0.0186 [0.02, 0.02]	3.24% [0.015, 0.057]	0.0025	0.0000
GPT-4.1 mini	OpenAI	0.4540 [0.43, 0.48]	10.74% [0.070, 0.151]	0.8653	0.0205	0.0145 [0.01, 0.02]	<b>8.9%</b> [ <b>0.034, 0.188</b> ]	<b>0.0023</b>	0.0001
GPT-4.1 nano	OpenAI	<b>0.1063</b> [ <b>0.10, 0.12</b> ]	13.66% [0.055, 0.229]	0.1957	<b>0.0153</b>	<b>0.0054</b> [ <b>0.01, 0.011</b> ]	<b>2.56%</b> [ <b>0.008, 0.052</b> ]	0.0116	<b>0.0000</b>
Qwen3 235B Thinking	Alibaba	0.4124 [0.39, 0.43]	<b>39.38%</b> [ <b>0.193, 1.599</b> ]	0.6612	0.0301	0.5366 [0.52, 0.56]	4.9% [0.022, 0.082]	0.1082	0.0038
GPT-OSS 120B	OpenAI	0.5025 [0.47, 0.53]	9.84% [0.035, 0.174]	<b>0.1331</b>	0.0479	0.0304 [0.03, 0.03]	5.69% [0.025, 0.097]	0.0038	0.0000
GPT-5 mini	OpenAI	0.5644 [0.53, 0.60]	9.5% [0.046, 0.156]	0.2456	0.0531	0.1674 [0.16, 0.18]	4.76% [0.021, 0.078]	0.0593	0.0004
GPT-5 nano	OpenAI	0.5048 [0.48, 0.52]	10.78% [0.061, 0.169]	0.6089	0.0348	0.0591 [0.06, 0.06]	3.84% [0.017, 0.063]	0.0078	0.0000
Claude Haiku 4.5	Anthropic	0.3777 [0.36, 0.40]	11.7% [0.033, 0.228]	0.2485	<b>0.0537</b>	0.1099 [0.10, 0.12]	3.7% [0.013, 0.074]	0.0052	0.0007
Gemini 3 Flash	Google	<b>0.7810</b> [ <b>0.74, 0.78</b> ] <sup>†</sup>	<b>3.48%</b> [0.015, 0.054]	0.2363	0.0345	1.0451 [0.98, 1.05]	3.38% [0.015, 0.054]	0.3411	0.0124

<sup>†</sup> Indicates statistical significance ( $p < 0.05$ ) between the best and the second-best scores.

\* Reports average value and 95% confidence intervals in brackets.

Note: Models are ordered by release date (2025). Dashed horizontal rules indicate models released in the same quarter.

strategy or base LLMs, if needed, we adapt the original prompts to facilitate the new use cases.

**Tasks and data.** We evaluate on six 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) (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 1 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), and (10) 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) OpenAI: GPT-OSS-120B (Agarwal et al., 2025), GPT-4.1 Mini,

GPT-4.1 Nano, GPT-5 Mini, GPT-5 Nano, (2)DeepSeek: DeepSeek R1 (Guo et al., 2025), (3) Meta: Llama 4 Scout (AI, 2025), (4) Alibaba Cloud Qwen3-235B (Yang et al., 2025a), (5) Google: Gemini-3 Flash (Comanici et al., 2025) and (6) Claude-Haiku 4.5. 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 methods along two dimensions: *quality* and *cost*. For each dimension, we report four complementary metrics. *Average* performance is estimated using a stratified bootstrap over runs, where each benchmark is treated as a stratum and confidence intervals reflect expected performance under reruns of the same benchmark suite. *Run Deviation* measures typical run-to-run deviation from a strategy’s mean on each benchmark, computed as a bootstrapped average of normalized absolute errors. To quantify stochasticity independent of benchmark difficulty, we additionally report two noise metrics based on z-score normalization: *Noise (Global)*, defined as the variance of all z-scores across benchmarks, and *Noise (Run)*, defined as the average within-benchmark z-score variance. Cost metrics (token usage and wall-clock runtime) are reported using the same statistics and expressed in USD based on the provider’s pricing.

## 4. Experiments

Our results are structured around two complementary questions: (i) how do different reasoning strategies compare

Table 3. Impact of prompt and parsing refinements on strategy performance. Enhancing clarity and standardizing output parsing significantly improves accuracy without affecting the stability. Direct prompting methods show the largest gains while the rest showcase similar ones, except RAP which improves the least. The best performance is shown in **blue** whereas the second best is shown in **orange**.

Strategy	Type	Original Prompts*	Improved Prompts*	$\Delta$
IO	Direct	<b>0.106 [0.10, 0.12]</b>	0.313 [0.28, 0.34]	+0.207 <sup>†</sup>
CoT	Direct	0.276 [0.25, 0.30]	0.398 [0.35, 0.43]	+0.122 <sup>†</sup>
CoT-SC	Direct	0.228 [0.21, 0.24]	0.410 [0.40, 0.45]	+0.182 <sup>†</sup>
ReAct	Adaptive	0.295 [0.28, 0.31]	0.391 [0.36, 0.42]	+0.096 <sup>†</sup>
Reflexion	Adaptive	0.282 [0.27, 0.30]	0.411 [0.39, 0.42]	+0.129 <sup>†</sup>
ToT-DFS	Structured	0.127 [0.10, 0.14]	<b>0.177 [0.15, 0.20]</b>	+0.050 <sup>†</sup>
GoT	Structured	0.3361 [0.31, 0.36]	0.420 [0.39, 0.46]	+0.084 <sup>†</sup>
ToT-BFS	Structured	0.407 [0.38, 0.44]	0.506 [0.47, 0.54]	+0.099 <sup>†</sup>
RAP	Planning	0.367 [0.35, 0.38]	0.403 [0.39, 0.41]	+0.036 <sup>†</sup>
FoA	Evolutionary	<b>0.4580 [0.43, 0.48]</b>	<b>0.546 [0.52, 0.58]</b>	+0.088 <sup>†</sup>

<sup>†</sup> Indicates statistical significance ( $p < 0.05$ ) from original.

\* Reports average quality and 95% confidence intervals in brackets.

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 ten representative reasoning strategies across all benchmarks. To address the second, we evaluate ten open- and closed-source reasoning models, from 6 diverse model providers, in a zero-shot setting, measuring their ability to solve tasks without external scaffolding. The resources for reproducing our experiments are available at <https://anonymous.4open.science/r/ReasonBench-64B3>.

#### 4.1. Reasoning strategies

Table 1 reports the same suite-level metrics as Table 2, but for ten reasoning *strategies*. Five results stand out.

**Quality generally increases with cost.** For strategies, quality generally increases with cost. Unlike reasoning models (Table 2), where cost and quality are decoupled, strategy cost reflects deliberate compute investment (branching, sampling, search) that often translates to accuracy gains.

**Top-performing strategies are cost-efficient.** FoA achieves the highest average quality, followed by ToT-BFS. Both sit far above direct prompting baselines in cost, but are also not the most expensive ones, indicating that the quality frontier here is dominated by strategies that do substantially more work per query.

**Within-task stochasticity varies dramatically.** The noise of quality across runs spans more than an order of magnitude. Thus, the *strategy* itself can be a primary driver of repeat-run

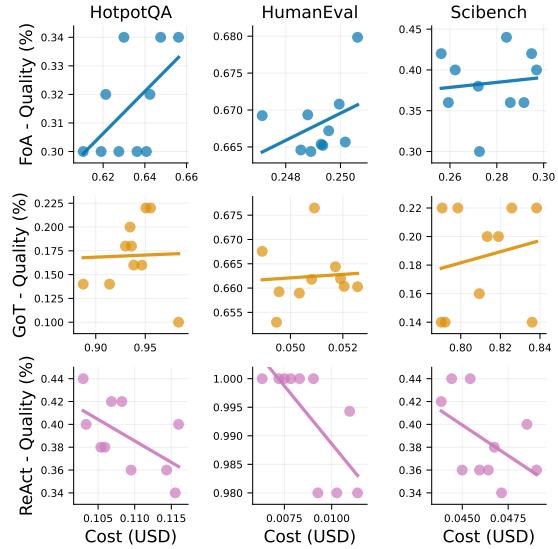


Figure 3. Correlation between Quality and Cost. For FoA, quality scales positively with cost across all benchmarks. ReAct exhibits a consistent negative slope, indicating diminishing returns at higher costs. GoT does not follow a uniform pattern, with its cost-quality relationship varying substantially by task.

variability, especially for high-performing methods such as ToT-BFS.

**Adaptive and planning strategies show strong benchmark sensitivity.** Quality Global Noise spans nearly an order of magnitude. These elevated values indicate uneven standardized performance across benchmarks. As a result, a strategy's overall average can shift substantially as the benchmark mix changes.

**Quality and cost stability decouple.** Cost variability does not reliably mirror quality variability: CoT-SC has the worst quality run deviation while exhibiting the lowest cost run deviation, while ReAct has the highest cost run deviation without being the most quality-unstable. Structured/planning methods also show large cost Global Noise, implying that bills can be highly task-dependent even when the mean cost is stable.

#### 4.2. Reasoning models

Having established that strategies can strongly modulate both accuracy and instability, Table 2 holds the strategy fixed and compares ten contemporary *models* across providers using the same metrics. We highlight five results.

**Cost does not correlate with quality.** Cost and quality are weakly related in Table 2. The two most expensive models illustrate this starkly: DeepSeek R1 has the highest cost but ranks second to last in quality, while Gemini 3 Flash is similarly expensive but best in quality.

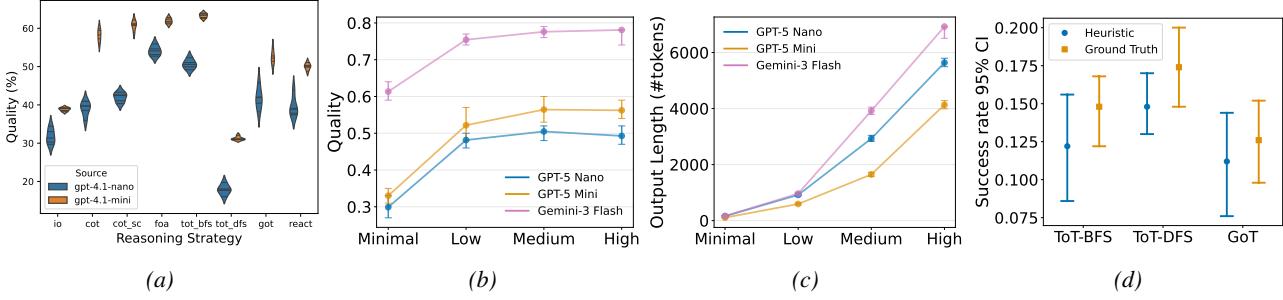


Figure 4. (a) **Model scale and stability:** GPT-4.1 MINI yields higher quality and tighter distributions than GPT-4.1 NANO across strategies. (b–c) **Thinking effort:** increasing the reasoning effort sharply increases output tokens while quality saturates across GPT-5 NANO, GPT-5 MINI, and GEMINI-3 FLASH. (d) **Causal evaluation intervention:** on *Game of 24*, replacing the heuristic evaluator with ground-truth evaluation changes success rates and improving the stability in most cases.

**The highest-quality model is also the most repeatable.** Similar to the reasoning strategies, the best performing model, Gemini 3 Flash achieves the highest quality average and the lowest quality run deviation, contradicting a simple quality–variance trade-off at the top end.

**Instability is multi-dimensional.** The noise quality between runs lies in a relatively narrow band, whereas the run deviation from the average spans an order of magnitude. Thus, large differences in apparent instability are driven more by score-scale and benchmark interaction than by radically different intrinsic stochasticity.

**Global Noise captures benchmark-dependence.** Global Noise varies widely, separating generalists from benchmark-sensitive models. GPT-OSS 120B has the lowest Global Noise (0.133), while DeepSeek R1 (1.110) and GPT-4.1 nano (1.049) are the highest, indicating a strong task sensitivity.

**Quality stability and cost stability decouple.** Quality and cost variability are only weakly related. For example, GPT-4.1 mini has a moderate quality run deviation (10.74%) but the worst cost run deviation (8.9%), while GPT-5 nano has a stable cost (3.84%) despite moderate quality variability (10.78%).

## 5. Analysis

### 5.1. Scaling Effects within a Model Family

In Fig. 4a, 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.

**Model Scaling Improves Accuracy and Stability.** Across all strategies, we observe a consistent scaling effect: GPT-4.1-Mini (gold violins) achieves higher mean quality and exhibits substantially tighter distributions than GPT-4.1-Nano

(blue violins). This indicates that increasing model size within the same family not only improves average performance but also reduces Noise (Run), leading to more stable reasoning behavior overall.

### 5.2. Impact of Prompts on Stability

We investigate whether instability in reasoning performance arises from prompts and parsers, in addition to the reasoning strategies themselves. Prompting artifacts, such as underspecified answer formats, amplify stochastic variation and cause divergent outputs across runs. If such interface-level choices drive instability, then improving prompt clarity and parsing robustness should reduce variability without altering the underlying reasoning logic. The results are reported in Table 3.

**Prompt Refinements Improve Quality but Not Stability.** Across all strategies, clarifying prompts and strengthening the parsing logic leads to statistically significant improvements in average quality, while run-to-run variance remains largely unchanged. This indicates that existing reasoning methods are sensitive to prompt specification artifacts, and that prompt-level refinements can systematically improve outcomes without altering the underlying reasoning strategy.

### 5.3. Correlation between Quality and Cost

Finally, we analyze the relationship between variability in quality and cost. Our approach allows us to examine multiple outcomes at the level of individual samples, recording quality and the exact cost incurred for each run and benchmark. Results are shown in Fig. 3.

**Divergent Cost–Quality Relationships.** Reasoning strategies differ fundamentally in how additional computation translates into quality, with some benefiting from higher cost, others degrading, and some exhibiting task-dependent behavior. In particular, FoA shows a consistently positive association, with higher-cost samples yielding higher-quality

385 output, indicating stable scaling behavior. In contrast, ReAct  
 386 exhibits a negative slope across all tasks, where increased  
 387 computation corresponds to less reliable reasoning. GoT  
 388 shows no consistent trend, reflecting sensitivity to task struc-  
 389 ture.  
 390

#### 391 **5.4. Thinking effort**

392 In Figs. 4b and 4c we study the effect of explicit reasoning  
 393 effort controls on model performance and stability. We  
 394 evaluate three models that can control how many reasoning  
 395 tokens to generate before creating a response to the prompt:  
 396 GPT-5-nano, GPT-5-mini, and Gemini 3 Flash.  
 397

398 **More for less.** Increasing reasoning effort consistently in-  
 399 creases computational cost, while accuracy gains are lim-  
 400 ited and non-monotonic. Performance improves from low  
 401 to medium effort and then saturates or declines, despite  
 402 sharply higher token usage, consistent with the inverted U-  
 403 shaped relationship between the reasoning length and the  
 404 accuracy (Wu et al., 2025). However, across multiple inde-  
 405 pendent runs, a new story unfolds as differences in average  
 406 performance are not statistically significant.  
 407

#### 408 **5.5. Intervening on Evaluation Functions**

409 Many reasoning strategies rely on an evaluation function to  
 410 estimate progress toward a solution. To test the causal role  
 411 of this signal, we perform a controlled intervention on the  
 412 *Game of 24* task, which allows exact ground-truth evaluation  
 413 of intermediate states. We benchmark the same reasoning  
 414 strategy with the heuristic evaluation function but also by  
 415 replacing it with the ground-truth evaluation.  
 416

417 **Causal effect of evaluation functions.** Replacing heuristic  
 418 evaluation with ground-truth evaluation on *Game of 24*  
 419 consistently improves reasoning performance (Fig. 4d) and  
 420 stability across all strategies (Fig. 9). Ground-truth evalua-  
 421 tion yields higher average quality, lower score variance, and  
 422 tighter confidence intervals, with the largest gains observed  
 423 for search-heavy methods such as ToT-BFS.  
 424

## 426 **6. Discussion**

### 427 **6.1. Implications and Broader Impact**

428 **Prefer reasoning models over external strategies.** Reasoning-capable models are often the preferable default:  
 429 they exhibit substantially lower variance than external rea-  
 430 soning strategies and typically provide a better quality–cost  
 431 trade-off, whereas strategy-level gains are frequently accom-  
 432 panied by higher instability and less predictable spend.  
 433

434 **Reasoning effort scales cost, not quality.** Increasing the  
 435 reasoning effort during test-time, consistently and markedly  
 436

437 raises cost, yet yields limited and mostly statistically indis-  
 438 tinguishable improvements in average performance.  
 439

440 **Global Noise reveals benchmark-dependence.** Global  
 441 Noise varies widely across systems, indicating that some  
 442 systems are markedly more task-sensitive even when ag-  
 443 gregate scores appear competitive; benchmark-dependence  
 444 should therefore be treated as a first-class reliability issue.  
 445

### 446 **6.2. Recommendations to the Community**

447 **Adaptive, variance-aware reasoning stacks.** Search-heavy  
 448 strategies can improve mean performance while increasing  
 449 run-to-run variance, and no single method is uniformly best  
 450 across tasks. A practical direction is to build *compositional*  
 451 pipelines with a learned *router* that gates between simple  
 452 decoding and structured search using problem features or  
 453 early signals, trained to minimize variance under constraints.  
 454

455 **Distribution-aware benchmarking (beyond best-of- $k$ ).** Best-of- $k$  reporting is brittle in high-variance regimes and  
 456 can alter the reasoning system’s rankings as  $k$  changes.  
 457 Benchmarks should emphasize distributional reporting and  
 458 avoid best-of- $k$  as a primary leaderboard criterion.  
 459

460 **Evaluators and Stability Diagnostics.** Evaluator quality  
 461 can be causal for both accuracy and stability in search, so  
 462 verifiers should be calibrated and uncertainty-aware, with  
 463 explicit study of how evaluator noise propagates into the  
 464 downstream variance. More broadly, we advocate an analy-  
 465 sis lens for reasoning that probes into branching decisions  
 466 and verifier signals that could localize where stochastic di-  
 467 vergence begins and guide stability-oriented design.  
 468

### 469 **6.3. Limitations and Future Work**

470 **Limitations.** We investigate decoding stochasticity, while  
 471 ignoring other variability sources. Then, our benchmark  
 472 suite, strategy set, and evaluated reasoning models are rep-  
 473 resentative but incomplete. Broader domain coverage across  
 474 all dimensions can strengthen our findings and potentially  
 475 unveil even more elusive findings. Moreover, while we study  
 476 prompt and parsing sensitivity, we do not evaluate more  
 477 advanced prompt interventions, such as automatic prompt  
 478 optimization, and their effects on variability.  
 479

480 **Future work.** Future work should move from measuring  
 481 instability to *optimizing* it: develop stability-aware training  
 482 and selection objectives that penalize variance rather than  
 483 optimizing mean accuracy alone. Additionally, learn adap-  
 484 tive routers that jointly choose the reasoning method and  
 485 compute budget to satisfy reliability constraints under cost  
 486 limits. Finally, extending REASONBENCH to tool-using  
 487 agents would test stability under external nondeterminism.  
 488

## 440 Impact Statement

441 This work argues that *stability is a first-class property of*  
 442 *LLM reasoning*, not a peripheral evaluation detail. REA-  
 443 SONBENCH operationalizes this view by making run-to-  
 444 run variability measurable and comparable for both quality  
 445 and cost, using controlled multi-run protocols and variance-  
 446 aware metrics. The immediate impact is methodological:  
 447 it enables more faithful scientific conclusions by revealing  
 448 when apparent improvements in mean performance are not  
 449 robust, when cost and quality decouple, and when bench-  
 450 mark dependence can dominate aggregate scores. Prac-  
 451 tically, these measurements support safer and more pre-  
 452 dictable deployment by highlighting lower-bound behavior,  
 453 budget-latency uncertainty, and task sensitivity that are in-  
 454 visible to single-run reporting.

455 Beyond measurement, REASONBENCH provides reusable  
 456 tooling that lowers the barrier to adopting variance-aware  
 457 evaluation, helping the community reproduce results, rerun  
 458 experiments as pipelines evolve, and compare methods under  
 459 standardized implementations. We expect this to reduce  
 460 overclaiming and “leaderboard churn” driven by noise, and  
 461 to incentivize research on models and methods that are reli-  
 462 able under realistic constraints. A potential negative impact  
 463 is increased evaluation cost from repeated trials; however,  
 464 the ability to share standardized pipelines, cache interme-  
 465 diate artifacts, and reuse multi-run statistics can amortize  
 466 this overhead and make reliability checks routine rather than  
 467 exceptional.

468 Finally, the broader implications of our findings are not  
 469 merely observational: the Discussion (§ 6) of the main pa-  
 470 per includes dedicated *Implications and Broader Impact*  
 471 (§ 6.1) and *Recommendations to the Community* (§ 6.2)  
 472 sections that translate empirical results into actionable guid-  
 473 ance. We hope REASONBENCH catalyzes a shift toward  
 474 distribution-aware reporting standards and stability-oriented  
 475 design, accelerating progress toward reasoning systems that  
 476 are not only accurate on average but stable, reproducible,  
 477 and dependable in practice.

## 478 References

479 Agarwal, S., Ahmad, L., Ai, J., Altman, S., Applebaum, A.,  
 480 Arbus, E., Arora, R. K., Bai, Y., Baker, B., Bao, H., et al.  
 481 gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint*  
 482 *arXiv:2508.10925*, 2025.

483 Ahn, J., Verma, R., Lou, R., Liu, D., Zhang, R., and Yin, W.  
 484 Large language models for mathematical reasoning: Pro-  
 485 gresses and challenges. *arXiv preprint arXiv:2402.00157*,  
 486 2024.

487 AI, M. Introducing llama 4: Advancing multimodal intelli-  
 488 gence, 2025. Accessed: 2025-09-22.

489 Besta, M., Blach, N., Kubicek, A., Gerstenberger, R., Pod-  
 490 stawski, M., Gianinazzi, L., Gajda, J., Lehmann, T.,  
 491 Niewiadomski, H., Nyczek, P., et al. Graph of thoughts:  
 492 Solving elaborate problems with large language models.  
 493 In *AAAI*, volume 38, pp. 17682–17690, 2024.

494 Blackwell, R. E., Barry, J., and Cohn, A. G. Towards re-  
 495 producible llm evaluation: Quantifying uncertainty in llm  
 496 benchmark scores. *ArXiv*, abs/2410.03492, 2024.

497 Chen et al. Evaluating large language models trained on  
 498 code, 2021. *arXiv eprint 2107.03374, cs.LG*.

499 Comanici, G., Bieber, E., Schaekermann, M., Pasupat, I.,  
 500 Sachdeva, N., Dhillon, I., Blistein, M., Ram, O., Zhang,  
 501 D., Rosen, E., et al. Gemini 2.5: Pushing the frontier  
 502 with advanced reasoning, multimodality, long context,  
 503 and next generation agentic capabilities. *arXiv preprint*  
*arXiv:2507.06261*, 2025.

504 Geman, S., Bienenstock, E., and Doursat, R. Neural net-  
 505 works and the bias/variance dilemma. *Neural computa-*  
*506 tion*, 4(1):1–58, 1992.

507 Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R.,  
 508 Zhu, Q., Ma, S., Wang, P., Bi, X., et al. Deepseek-r1: In-  
 509 centivizing reasoning capability in llms via reinforcement  
 510 learning. *arXiv preprint arXiv:2501.12948*, 2025.

511 Hao, S., Gu, Y., Ma, H., Hong, J. J., Wang, Z., Wang, D. Z.,  
 512 and Hu, Z. Reasoning with language model is planning  
 513 with world model. In *EMNLP*, 2023.

514 Hastie, T., Tibshirani, R., Friedman, J., et al. The elements  
 515 of statistical learning, 2009.

516 Hoppe, F., Ilievski, F., and Kalo, J.-C. Investigating the  
 517 robustness of deductive reasoning with large language  
 518 models. *arXiv preprint arXiv:2502.04352*, 2025.

519 Hou, Y., Xiao, Z., Yu, F., Jiang, Y., Wei, X., Huang, H.,  
 520 Chen, Y., and Chen, G. Automatic robustness stress  
 521 testing of llms as mathematical problem solvers. *arXiv*  
*522 preprint arXiv:2506.05038*, 2025.

523 Huang, S., Yang, L., Song, Y., Chen, S., Cui, L., Wan,  
 524 Z., Zeng, Q., Wen, Y., Shao, K., Zhang, W., et al. Thinkbench:  
 525 Dynamic out-of-distribution evaluation for  
 526 robust llm reasoning. *arXiv preprint arXiv:2502.16268*,  
 527 2025.

528 Jaech, A., Kalai, A., Lerer, A., Richardson, A., El-Kishky,  
 529 A., Low, A., Helyar, A., Madry, A., Beutel, A., Car-  
 530 ney, A., et al. Openai o1 system card. *arXiv preprint*  
*arXiv:2412.16720*, 2024.

531 Ji, K., Guo, Y., Zhang, Z., Zhu, X., Tian, Y., Liu, N., and  
 532 Zhai, G. Medomni-45 { \deg}: A safety-performance

- 495 benchmark for reasoning-oriented llms in medicine.  
 496 *arXiv preprint arXiv:2508.16213*, 2025.
- 497 Jiang, E., Xu, C., Singh, N., and Singh, G. Misaligning  
 498 reasoning with answers—a framework for assessing lilm  
 499 cot robustness. *arXiv preprint arXiv:2505.17406*, 2025.
- 500 Jin, B., Yoon, J., Kargupta, P., Arik, S. O., and Han,  
 501 J. An empirical study on reinforcement learning for  
 502 reasoning-search interleaved lilm agents. *arXiv preprint*  
 503 *arXiv:2505.15117*, 2025.
- 504 Karia, R., Bramblett, D., Dobhal, D., and Srivastava, S.  
 505 Autonomous evaluation of llms for truth maintenance and  
 506 reasoning tasks. *arXiv preprint arXiv:2410.08437*, 2024.
- 507 Klein, L. H., Potamitis, N., Aydin, R., West, R., Gulcehre,  
 508 C., and Arora, A. Fleet of agents: Coordinated problem  
 509 solving with large language models. In *Forty-second*  
 510 *International Conference on Machine Learning*, 2025.
- 511 Klingbeil, A., Grützner, C., and Schreck, P. Trust and re-  
 512 liance on ai—an experimental study on the extent and  
 513 costs of overreliance on ai. *Computers in Human Behavior*,  
 514 160:108352, 2024.
- 515 Li, Y., Zhang, W., Yang, Y., Huang, W.-C., Wu, Y., Luo, J.,  
 516 Bei, Y., Zou, H. P., Luo, X., Zhao, Y., et al. Towards agen-  
 517 tic rag with deep reasoning: A survey of rag-reasoning  
 518 systems in llms. *arXiv preprint arXiv:2507.09477*, 2025.
- 519 Liu, J., wei Liu, H., Xiao, L., Wang, Z., Liu, K., Gao,  
 520 S., Zhang, W., Zhang, S., and Chen, K. Are your llms  
 521 capable of stable reasoning? In *Annual Meeting of the*  
 522 *Association for Computational Linguistics*, 2024.
- 523 Madaan, L., Singh, A. K., Schaeffer, R., Poultion, A.,  
 524 Koyejo, S., Stenetorp, P., Narang, S., and Hupkes, D.  
 525 Quantifying variance in evaluation benchmarks. *arXiv*  
 526 *preprint arXiv:2406.10229*, 2024.
- 527 Miller, E. Adding error bars to evals: A statistical ap-  
 528 proach to language model evaluations. *arXiv preprint*  
 529 *arXiv:2411.00640*, 2024.
- 530 Mizrahi, M., Kaplan, G., Malkin, D., Dror, R., Shahaf, D.,  
 531 and Stanovsky, G. State of what art? a call for multi-  
 532 prompt lilm evaluation. *Transactions of the Association*  
 533 *for Computational Linguistics*, 12:933–949, 2023.
- 534 Mo, S. and Xin, M. Tree of uncertain thoughts reasoning  
 535 for large language models. In *ICASSP*, pp. 12742–12746,  
 536 2024.
- 537 Mondorf, P. and Plank, B. Beyond accuracy: evaluating the  
 538 reasoning behavior of large language models—a survey.  
 539 *arXiv preprint arXiv:2404.01869*, 2024.
- 540 Ni, J., Xue, F., Yue, X., Deng, Y., Shah, M., Jain, K., Neubig,  
 541 G., and You, Y. Mixeval: Deriving wisdom of the crowd  
 542 from lilm benchmark mixtures. In Globerson, A., Mackey,  
 543 L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J., and  
 544 Zhang, C. (eds.), *Advances in Neural Information Pro-  
 545 cessing Systems*, volume 37, pp. 98180–98212. Curran  
 546 Associates, Inc., 2024.
- 547 Phan, L., Gatti, A., Han, Z., Li, N., Hu, J., Zhang, H., Zhang,  
 548 C. B. C., Shaaban, M., Ling, J., Shi, S., et al. Humanity’s  
 549 last exam. *arXiv preprint arXiv:2501.14249*, 2025.
- 550 Potamitis, N., Klein, L. H., Mohammadi, B., Xu, C.,  
 551 Mukherjee, A., Tandon, N., Bindschaedler, L., and  
 552 Arora, A. Cache saver: A modular framework for ef-  
 553 ficient, affordable, and reproducible LLM inference. In  
 554 *EMNLP*, pp. 25703–25724, 2025. doi: 10.18653/v1/2025.  
 555 findings-emnlp.1402.
- 556 Shinn, N., Cassano, F., Gopinath, A., Narasimhan, K., and  
 557 Yao, S. Reflexion: language agents with verbal reinforce-  
 558 ment learning. In *NeurIPS*, pp. 8634–8652, 2023.
- 559 Suzgun, M. and Kalai, A. T. Meta-prompting: Enhancing  
 560 language models with task-agnostic scaffolding. *arXiv*  
 561 *preprint arXiv:2401.12954*, 2024.
- 562 Wang, G., Chen, Z., Li, B., and Xu, H. Cer-eval: Certifiable  
 563 and cost-efficient evaluation framework for llms. *arXiv*  
 564 *preprint arXiv:2505.03814*, 2025.
- 565 Wang, X., Wei, J., Schuurmans, D., Le, Q. V., Chi,  
 566 E. H., Narang, S., Chowdhery, A., and Zhou, D. Self-  
 567 consistency improves chain of thought reasoning in lan-  
 568 guage models. In *ICLR*, 2023.
- 569 Wang, X., Hu, Z., Lu, P., Zhu, Y., Zhang, J., Subramaniam,  
 570 S., Loomba, A. R., Zhang, S., Sun, Y., and Wang, W.  
 571 Scibench: evaluating college-level scientific problem-  
 572 solving abilities of large language models. In *ICML*,  
 573 2024a.
- 574 Wang, Y. and Zhao, Y. Rupbench: Benchmarking reasoning  
 575 under perturbations for robustness evaluation in large lan-  
 576 guage models. *arXiv preprint arXiv:2406.11020*, 2024.
- 577 Wang, Y., Jiang, Z., Chen, Z., Yang, F., Zhou, Y., Cho, E.,  
 578 Fan, X., Lu, Y., Huang, X., and Yang, Y. Recmind: Large  
 579 language model powered agent for recommendation. In  
 580 *NAACL-HLT (Findings)*, pp. 4351–4364, 2024b.
- 581 Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi,  
 582 E., Le, Q. V., Zhou, D., et al. Chain-of-thought prompting  
 583 elicits reasoning in large language models. *Advances in*  
 584 *neural information processing systems*, 35:24824–24837,  
 585 2022.

- 550 Wu, Y., Wang, Y., Ye, Z., Du, T., Jegelka, S., and Wang,  
 551 Y. When more is less: Understanding chain-of-thought  
 552 length in llms. *arXiv preprint arXiv:2502.07266*, 2025.
- 553 Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B.,  
 554 Yu, B., Gao, C., Huang, C., Lv, C., et al. Qwen3 technical  
 555 report. *arXiv preprint arXiv:2505.09388*, 2025a.
- 556 Yang, K., Swope, A., Gu, A., Chalamala, R., Song, P.,  
 557 Yu, S., Godil, S., Prenger, R. J., and Anandkumar, A.  
 558 Leandojo: Theorem proving with retrieval-augmented  
 559 language models. *Advances in Neural Information Pro-  
 560 cessing Systems*, 36:21573–21612, 2023.
- 561 Yang, K., Poesia, G., He, J., Li, W., Lauter, K., Chaudhuri,  
 562 S., and Song, D. Formal mathematical reasoning: A new  
 563 frontier in ai. *arXiv preprint arXiv:2412.16075*, 2024a.
- 564 Yang, L., Yu, Z., Zhang, T., Cao, S., Xu, M., Zhang, W.,  
 565 Gonzalez, J. E., and Cui, B. Buffer of thoughts: Thought-  
 566 augmented reasoning with large language models. In  
 567 *NeurIPS*, 2024b.
- 568 Yang, Y., Yamada, H., and Tokunaga, T. Evaluating ro-  
 569 bustness of llms to numerical variations in mathematical  
 570 reasoning. In *The Sixth Workshop on Insights from Neg-  
 571 ative Results in NLP*, pp. 171–180, 2025b.
- 572 Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T., Cao, Y.,  
 573 and Narasimhan, K. Tree of thoughts: Deliberate problem  
 574 solving with large language models. *Advances in neural  
 575 information processing systems*, 36:11809–11822, 2023a.
- 576 Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan,  
 577 K., and Cao, Y. React: Synergizing reasoning and act-  
 578 ing in language models. In *International Conference on  
 579 Learning Representations (ICLR)*, 2023b.
- 580 Ye, F., Yang, M., Pang, J., Wang, L., Wong, D., Yilmaz,  
 581 E., Shi, S., and Tu, Z. Benchmarking llms via uncer-  
 582 tainty quantification. *Advances in Neural Information  
 583 Processing Systems*, 37:15356–15385, 2024.
- 584 Zhang, D., Zhoubian, S., Hu, Z., Yue, Y., Dong, Y., and  
 585 Tang, J. ReST-MCTS\*: LLM self-training via process  
 586 reward guided tree search. In *NeurIPS*, 2024.
- 587 Zhilin, Y., Peng, Q., Saizheng, Z., Yoshua, B., William,  
 588 C., Ruslan, S., and Manning, C. D. Hotpotqa: A data-  
 589 set for diverse, explainable multi-hop question answer-  
 590 ing. *Proceedings of the 2018 Conference on Empirical  
 591 Methods in Natural Language Processing*, 2018. doi:  
 592 10.18653/v1/d18-1259.
- 593 Zhou, A., Yan, K., Shlapentokh-Rothman, M., Wang, H.,  
 594 and Wang, Y.-X. Language agent tree search unifies  
 595 reasoning, acting, and planning in language models. In  
 596 *ICML*, 2024a.
- 597 Zhou, L., Schellaert, W., Martínez-Plumed, F., Moros-Daval,  
 598 Y., Ferri, C., and Hernández-Orallo, J. Larger and more  
 599 instructable language models become less reliable. *Nature*,  
 600 634(8032):61–68, 2024b.
- 601
- 602
- 603
- 604

## 605 A. Additional Experimental Details

### 606 A.1. Detailed Task Descriptions

#### 608 A.1.1. GAME OF 24

609 The Game of 24 is a math puzzle where players are given four numbers and must use each of them exactly once, along with  
 610 the basic arithmetic operations ( $+, -, \times, \div$ ), to form an expression that evaluates to 24.

612 Our benchmark includes 1,362 such puzzles collected from 4nums.com, organized in ascending order of difficulty. Each  
 613 puzzle provides four input numbers, and the goal is to generate a valid equation that results in 24. Following the approach of  
 614 ToT (Yao et al., 2023a), we designate puzzles numbered 901 to 1000 as our test set.

#### 616 A.1.2. SCIBENCH

618 SciBench (Wang et al., 2024a) is a scientific reasoning benchmark designed to evaluate college-level problem-solving  
 619 abilities across subjects such as mathematics, physics, and chemistry. Each task presents an open-ended problem that  
 620 requires multi-step reasoning, domain-specific knowledge, and advanced computations, including calculus and differential  
 621 equations. Problems are drawn from widely used textbooks and university exams.

622 Following the approach of ReST-MCTS (Zhang et al., 2024), we sampled 109 problems spanning different subjects to form  
 623 the test set. Quality is measured using an *accuracy* metric, defined as the proportion of problems correctly solved according  
 624 to the official solutions (exact matching).

#### 626 A.1.3. HUMAN EVAL

628 HumanEval (Chen et al., 2021) is a code generation benchmark where participants are given natural language docstrings and  
 629 must generate Python functions that correctly implement the described behavior. Each problem includes a hidden test suite  
 630 used to verify functional correctness.

631 Following the setup from Reflexion (Shinn et al., 2023), the benchmark consists of 100 programming tasks in the test set.  
 632 We evaluate performance using the *pass@1* metric, which measures the proportion of problems solved correctly on the first  
 633 attempt.

#### 635 A.1.4. HOTPOTQA

637 HotpotQA (Zhilin et al., 2018) is a large-scale question answering benchmark that tests an agent’s ability to perform  
 638 multi-hop reasoning across multiple documents. Multi-step approaches, such as ToT, are permitted to interact with an API  
 639 that enables document retrieval and targeted information lookup.

640 Following prior work (Zhou et al., 2024a; Shinn et al., 2023), we evaluate on a set of 100 randomly selected questions. The  
 641 quality of a response is judged based on *exact match* (EM) with the oracle answer.

#### 643 A.1.5. SHAKESPEAREAN SONNET WRITTING

645 Shakespearean Sonnet Writing (Suzgun & Kalai, 2024) is a creative generation task where the goal is to compose a 14-line  
 646 sonnet adhering to the classic rhyme scheme “ABAB CDCD EFEF GG”. Each sonnet must include three provided words  
 647 verbatim.

648 Following Suzgun et al. (Suzgun & Kalai, 2024), we randomly sampled 50 datapoints to form the test set. Quality is  
 649 measured using an *accuracy* metric, which reflects the proportion of sonnets that both satisfy the rhyme scheme and include  
 650 all three required words exactly as given.

#### 652 A.1.6. HUMANITY’S LAST EXAM

654 Humanity’s Last Exam (Phan et al., 2025) is a challenging, multidisciplinary benchmark designed to probe the upper limits  
 655 of general reasoning and knowledge in large language models. The benchmark consists of carefully curated questions  
 656 spanning mathematics, natural sciences, humanities, and abstract reasoning, with an emphasis on problems that require deep  
 657 understanding, precise reasoning, and resistance to shallow pattern matching.

660 Each task is presented as a standalone question with a single correct answer, typically requiring multi-step logical inference,  
661 symbolic manipulation, or synthesis of domain knowledge. The benchmark is designed to be closed-book and does not  
662 permit external tool use.

663 Following the official evaluation protocol, we evaluate models on a 50 sample subset and report performance using an  
664 *accuracy* metric, defined as the proportion of questions answered exactly correctly. Correct answers are determined based  
665 on the original author’s LLM as a Judge system with the recommended prompts and models (GPT-o3 Mini).

## 667 668 **A.2. Detailed Descriptions of Reasoning Strategies**

### 669 670 **A.2.1. INPUT-OUTPUT (IO)**

671 A direct prompting strategy where the model maps an input to an output in a single step, without generating or exposing  
672 intermediate reasoning. IO relies entirely on the model’s internal representations and is typically used as a baseline for  
673 comparison with more explicit reasoning methods.

### 674 675 **A.2.2. CHAIN-OF-THOUGHT (CoT)**

676 Encourages the model to generate an explicit sequence of intermediate reasoning steps before producing a final answer. By  
677 verbalizing its reasoning process, CoT is expected to improve performance on multi-step and compositional reasoning tasks  
678 (Wei et al., 2022).

### 680 681 **A.2.3. CHAIN-OF-THOUGHT WITH SELF-CONSISTENCY (CoT-SC)**

682 Extends Chain-of-Thought by sampling multiple independent reasoning chains and aggregating their final answers via  
683 a consistency-based voting mechanism. This approach mitigates errors from individual reasoning paths and improves  
684 robustness and accuracy (Wang et al., 2023).

### 685 686 **A.2.4. REFLEXION**

687 A reasoning framework that enables models to iteratively reflect on and critique their own outputs using feedback from  
688 prior attempts or environment interactions. Reflexion leverages self-evaluation to generate corrective insights, which are  
689 incorporated into subsequent reasoning steps to improve task performance over time (Shinn et al., 2023).

### 691 692 **A.2.5. TREE OF THOUGHTS (ToT)**

693 Decomposes the problem into multiple chains of thoughts, organized in a tree structure. Thought evaluation and search  
694 traversal algorithms are utilized to solve the problem (Yao et al., 2023a).

### 695 696 **A.2.6. FLEET OF AGENTS (FoA)**

697 Decomposes the problem into multiple chains of thoughts. Employs a genetic-type particle filtering approach to navigate  
698 through dynamic tree searches to solve the problem (Klein et al., 2025).

### 700 701 **A.2.7. GRAPH OF THOUGHTS (GoT)**

702 Allows the organization of thoughts in a graph structure (Besta et al., 2024). It introduces arbitrary graph-based thought  
703 transformations such as thought aggregation and thought refinement.

### 704 705 **A.2.8. REACT**

706 A reasoning method that interleaves reasoning (thought generation) and acting (taking environment-interacting actions)  
707 to solve problems interactively. Each action’s output informs subsequent reasoning, enabling adaptive and dynamic  
708 problem-solving (Yao et al., 2023b).

715 A.2.9. REASONING VIA PLANNING (RAP)  
 716

717 is a reasoning framework that equips Large Language Models (LLMs) with an internal world model and employs Monte  
 718 Carlo Tree Search (MCTS) for strategic exploration of reasoning paths. RAP repurposes the LLM to simulate future states  
 719 and evaluate potential actions, enabling deliberate planning and improved problem-solving performance (Hao et al., 2023)

 720 A.3. Detailed Reasoning Models Descriptions  
 721

 722 A.4. Implementation Details  
 723

724 **Platforms.** GPT models were accessed through the [OpenAI API](#), Google models through the [Gemini API](#) while the  
 725 utilization of the rest of the models was facilitated by the [TogetherAI API](#).  
 726

727 **Model checkpoints and prices.** To compute the costs of our experiments we used the current model prices indicated  
 728 by OpenAI, Gemini and Together AI, accordingly to the model. The specific models snapshot we used, along with their  
 729 respective prices are presented in 4.  
 730

731 *Table 4.* Cost of each model of that we have used, at this project's time of execution.  
 732

Model	Provider used	Input (\$/1M)	Output (\$/1M)
DeepSeek R1	Together API	3.00	7.00
Llama 4 Maverick	Together AI	0.27	0.85
GPT-4.1 mini	OpenAI API	0.40	1.60
GPT-4.1 nano	OpenAI API	0.10	0.40
Qwen3 235B Thinking	Together AI	0.65	3.00
GPT-OSS 120B	Together AI	0.15	0.60
GPT-5 mini	OpenAI API	0.25	2.00
GPT-5 nano	OpenAI API	0.05	0.40
Claude Haiku 4.5	Anthropic API	1.00	5.00
Gemini 3 Flash	Google Gemini API	0.50	3.00

 745 A.4.1. MODEL CONFIGURATIONS  
 746

747 Generation parameters specified when making calls to any of the models used throughout this project. These parameters  
 748 were not defined by us, but by the implementation where the respective prompts were introduced. However, as newer models  
 749 were used for this study, we only adjusted the maximum allowed completion tokens as needed to ensure compatibility and  
 750 successful completion of responses.  
 751

752 *Table 5.* Generation parameters specified when making requests to a base LLM.  
 753

	max_tokens	temperature	top_p	stop
<b>Game of 24</b>	200	0.7	1	Null
<b>SciBench</b>	300	0.7	1	Null
<b>Humanity's Last Exam</b>	300	0.7	1	Null
<b>HumanEval</b>	200	0.7	1	Null
<b>HotpotQA</b>	300	0.7	1	Null
<b>Sonnet Writing</b>	800	1.0	1	Null

 763 A.4.2. PROMPTS  
 764

765 Due to the large number of methods and tasks presented in this paper, including all corresponding prompts would  
 766 be impractical within the main text. Therefore, we provide a comprehensive collection of all prompts used in our  
 767 experiments (both original and improved) on our GitHub repository: [https://anonymous.4open.science/r/  
 768 ReasonBench-64B3/prompts.md](https://anonymous.4open.science/r/ReasonBench-64B3/prompts.md).  
 769

## B. Additional Results

### B.0.1. THINKING EFFORT

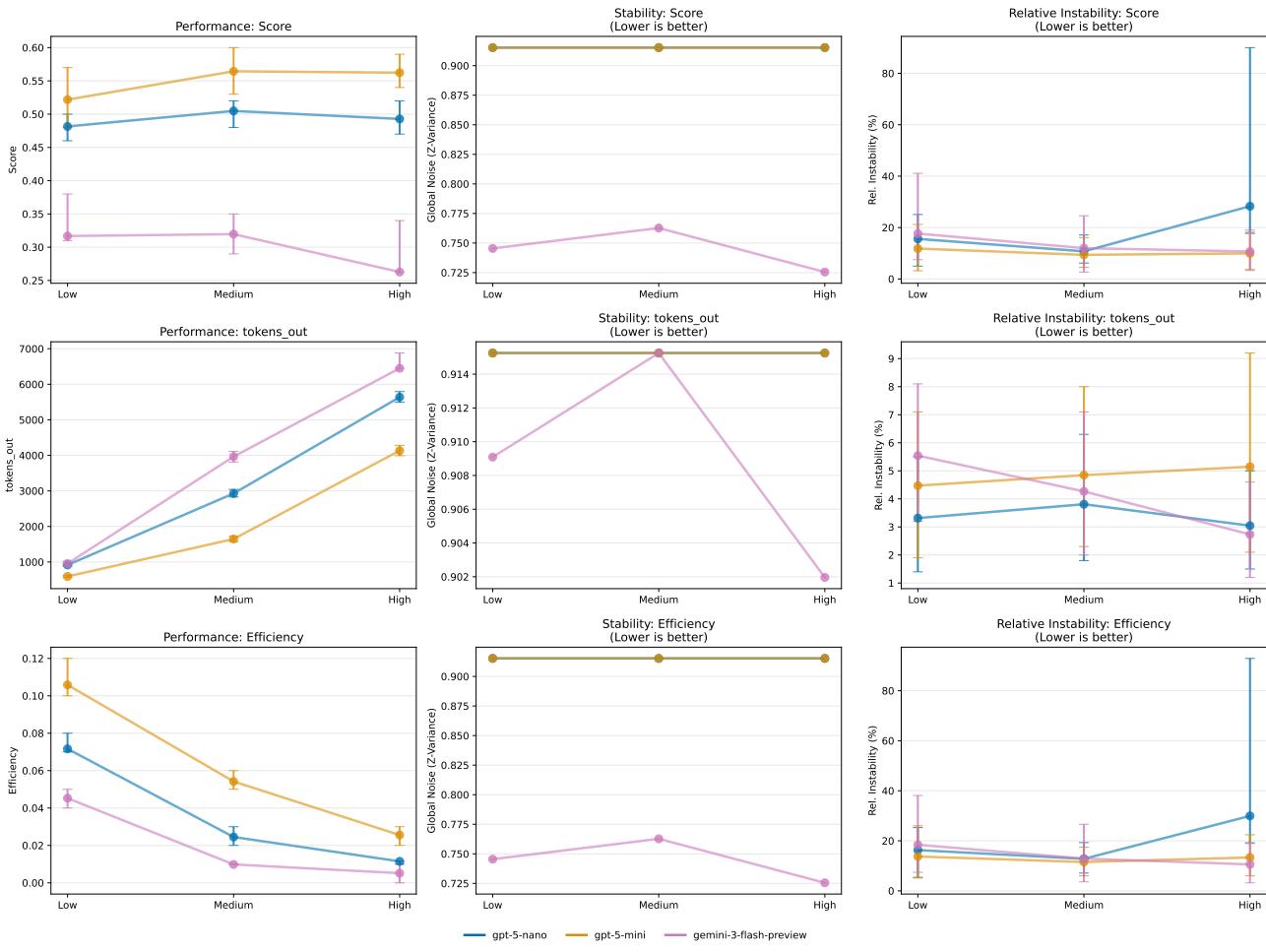


Figure 5. Effect of test-time reasoning effort on mean performance and stability for GPT-5 NANO, GPT-5 MINI, and GEMINI-3 FLASH. Rows report quality (**Score**), cost proxy (**tokens\_out**), and efficiency (**Quality per token**); columns report the mean metric (**Performance**), cross-benchmark variability (**Global Noise**, lower is better), and run-to-run deviation (**Relative Instability**, lower is better). Error bars denote uncertainty estimates from repeated runs.

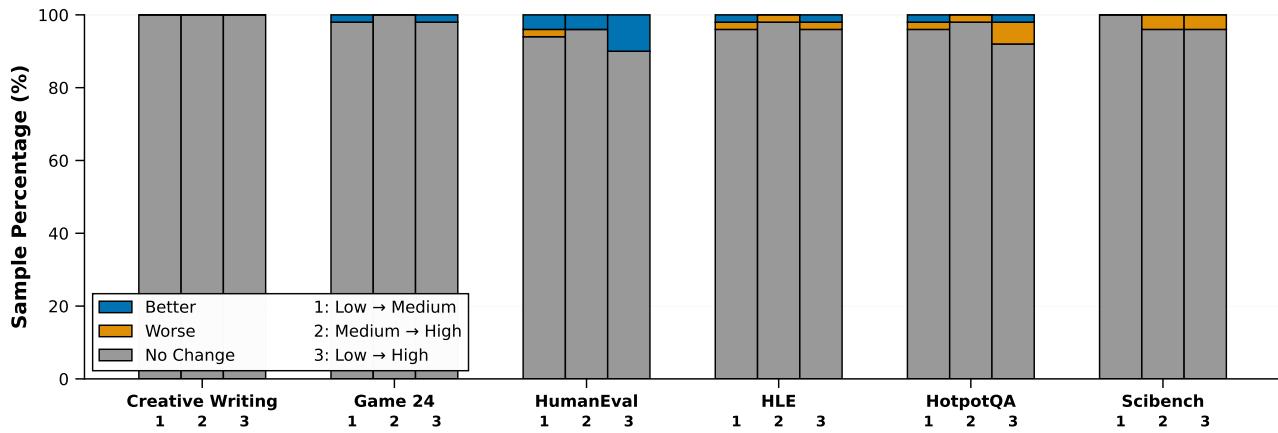
We extend the reasoning effort study by jointly analyzing mean behavior and stability under test-time compute scaling. We evaluate GPT-5-NANO, GPT-5-MINI, and GEMINI-3 FLASH at three effort settings (*low*, *medium*, *high*) on the full ReasonBench suite. For each (model, effort) configuration, we repeat evaluation for 10 independent runs to estimate both expected performance and run-to-run variability.

We measure three quantities: (*i*) **Score** as the primary quality metric, (*ii*) **tokens\_out** as a direct proxy for generation cost, and (*iii*) **Efficiency** as a quality–cost trade-off metric. For each quantity we report: **Average** performance using stratified bootstrap confidence intervals across benchmarks; **Run Deviation** (relative instability) as the typical percent deviation of runs from their benchmark-level mean; and **Global Noise** as the variance of z-scored outcomes across benchmarks, capturing benchmark-dependence after normalizing for task difficulty. The full results are summarized in Fig. 5.

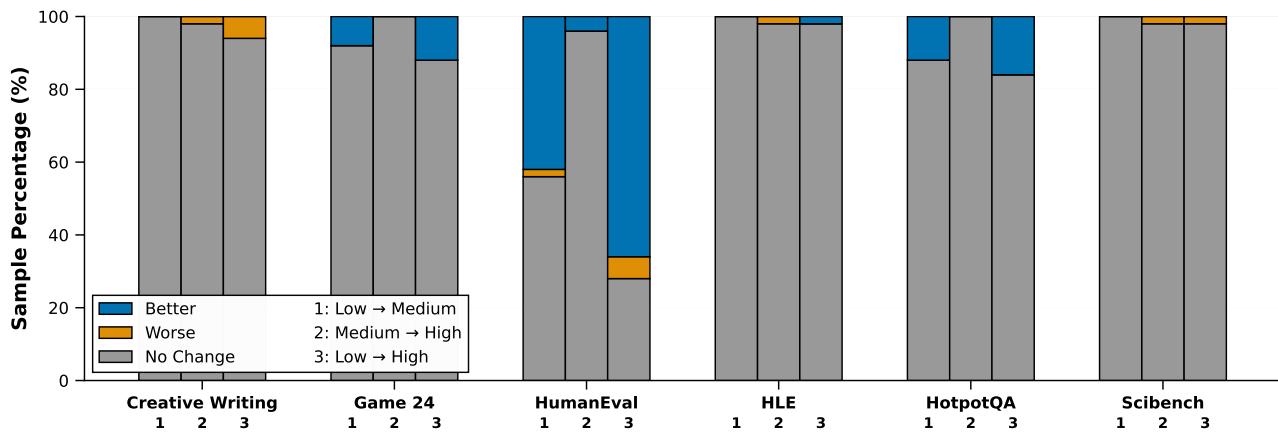
### B.1. Per-sample significance under reasoning-effort scaling

To complement aggregate averages, we analyze how reasoning effort affects outcomes at the level of individual benchmark instances. For each model, benchmark, and effort setting (*low*, *medium*, *high*), we estimate per-sample quality together with uncertainty by aggregating results over 10 independent runs and computing a confidence interval for each sample. We

825 then compare effort levels pairwise (*low*→*medium*, *medium*→*high*, and *low*→*high*) and classify each sample that *improved*,  
 826 *worsened*, or *was unaffected* based on whether the confidence intervals indicate a statistically significant increase, a  
 827 statistically significant decrease, or an overlap. Finally, we report, for each of the three models and for each benchmark, the  
 828 percentage of samples in each category, quantifying how often increased effort yields meaningful gains (or regressions)  
 829 beyond what is explained by run-to-run variability.  
 830



845 *Figure 6.* Per-sample significance analysis for GPT-5 Nano: for each benchmark, stacked bars show the percentage of instances  
 846 whose quality is *better*, *worse*, or shows *no statistically significant change* when increasing reasoning\_effort (low→medium,  
 847 medium→high, and low→high), based on confidence-interval comparisons over 10 independent runs.  
 848  
 849



864 *Figure 7.* Per-sample significance analysis for GPT-5 Mini: for each benchmark, stacked bars show the percentage of instances  
 865 whose quality is *better*, *worse*, or shows *no statistically significant change* when increasing reasoning\_effort (low→medium,  
 866 medium→high, and low→high), based on confidence-interval comparisons over 10 independent runs.  
 867  
 868

### B.1.1. CAUSAL ANALYSIS

869 In the main paper, we report the causal intervention results using mean quality to isolate the effect of replacing heuristic  
 870 evaluation with ground-truth evaluation. In the appendix, we provide a fuller stability characterization in Fig. 9 by  
 871 additionally reporting the variance of quality in and the relative run deviation (with confidence intervals) for the same  
 872 intervention, highlighting how evaluation accuracy affects not only expected performance but also run-to-run variability.  
 873  
 874

### B.1.2. DIAGNOSIS

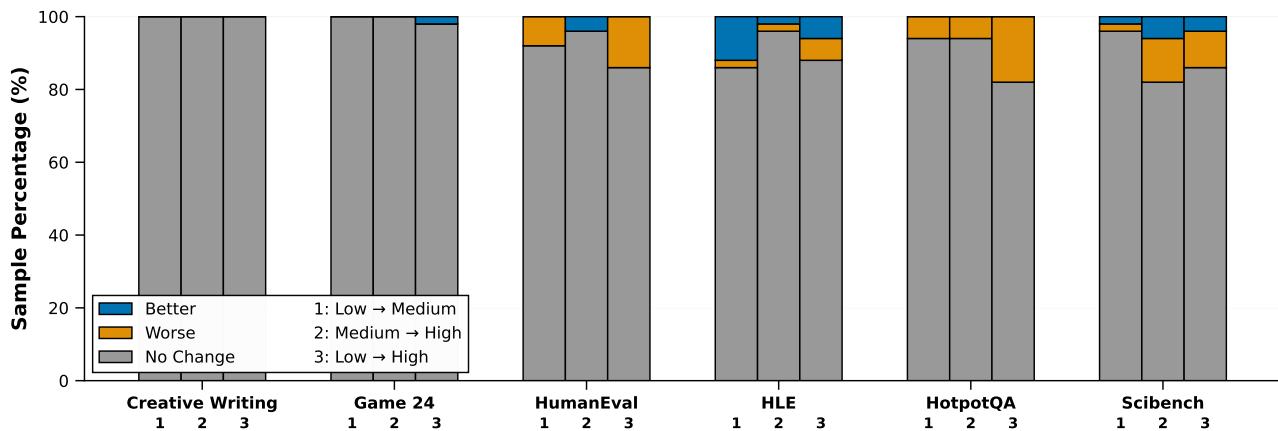


Figure 8. Per-sample significance analysis for Gemini-3 Flash: for each benchmark, stacked bars show the percentage of instances whose quality is *better*, *worse*, or shows *no statistically significant change* when increasing `reasoning_effort` (low→medium, medium→high, and low→high), based on confidence-interval comparisons over 10 independent runs.

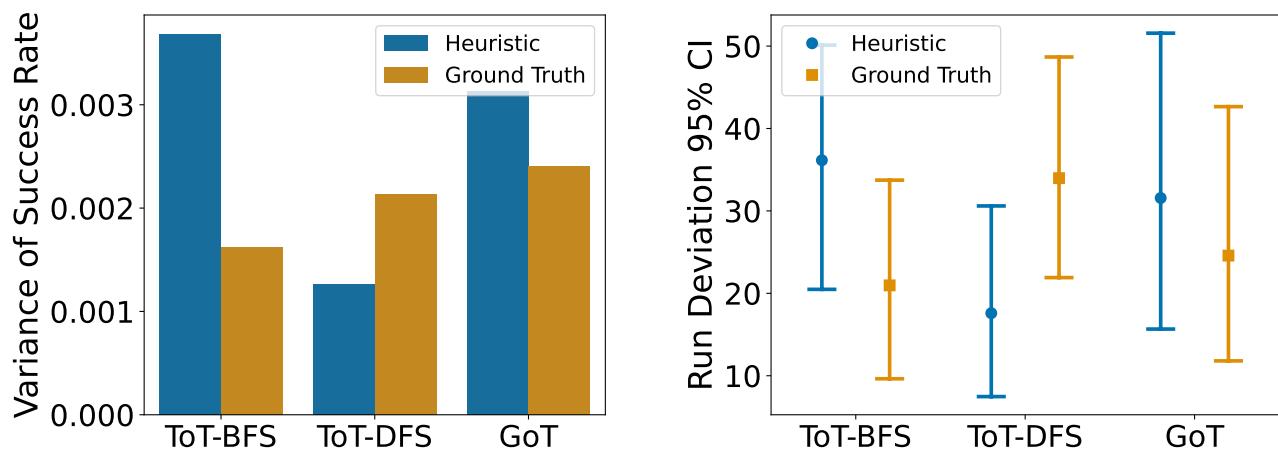


Figure 9. Run-to-run deviation (95% CI) and variance of solution quality under heuristic versus ground-truth evaluation for ToT-BFS, ToT-DFS, and GoT on *Game of 24*.

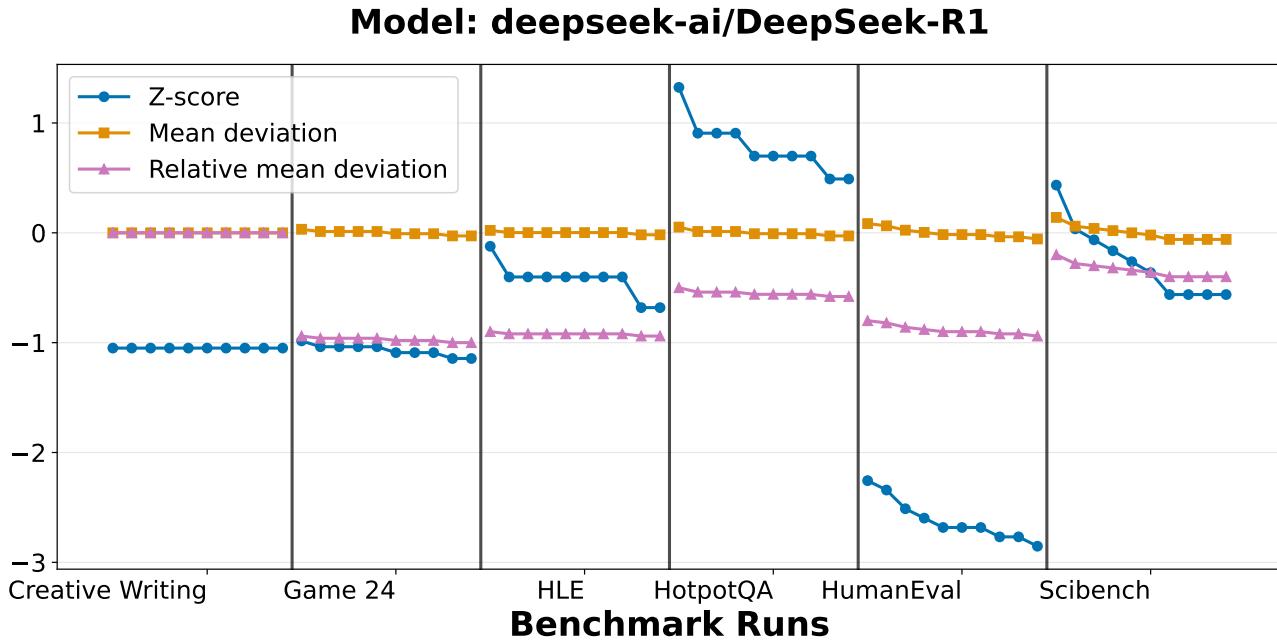


Figure 10. Run-level stability trace: per-run outcomes grouped by benchmark, showing benchmark-normalized z-scores and absolute/relative deviations from the benchmark mean across repeated runs.

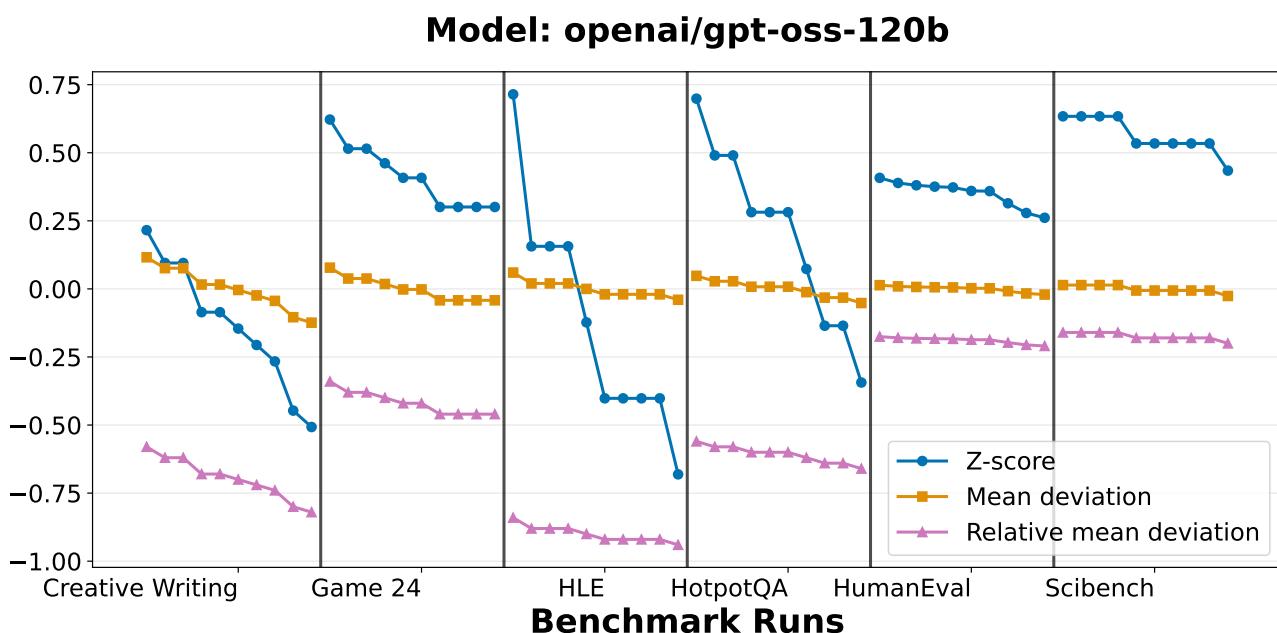


Figure 11. Run-level stability trace for DeepSeek-R1: per-run outcomes grouped by benchmark, showing benchmark-normalized z-scores and absolute/relative deviations from the benchmark mean across repeated runs.

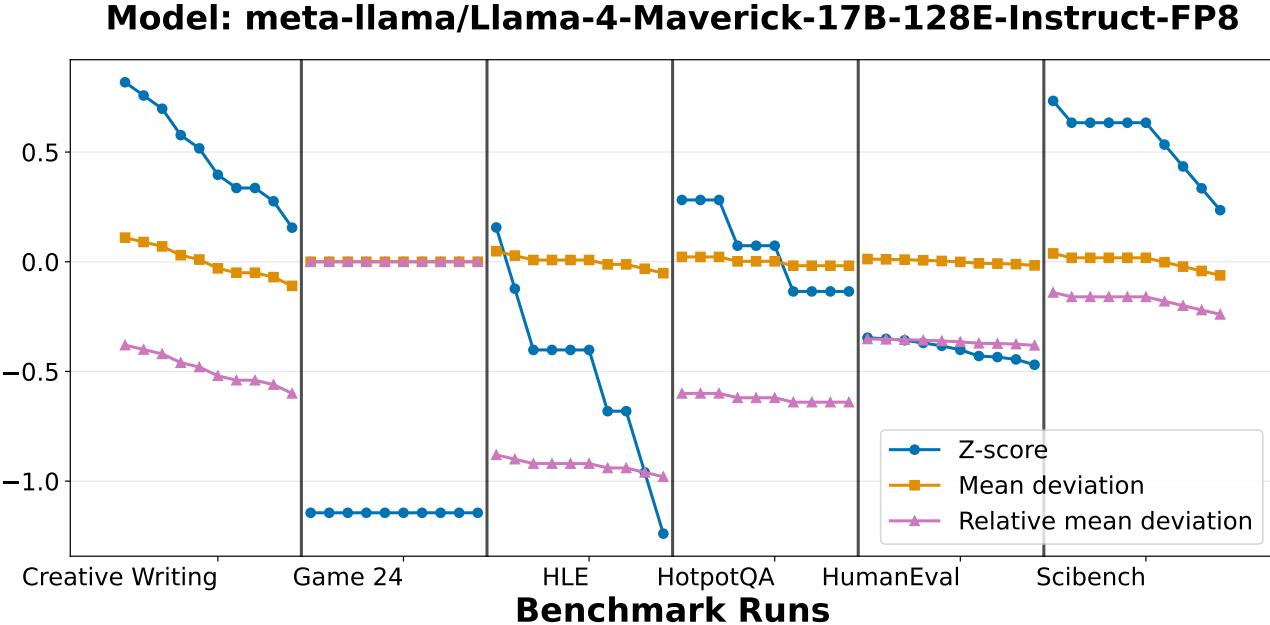


Figure 12. Run-level stability trace for Llama-4-Maverick: per-run outcomes grouped by benchmark, showing benchmark-normalized z-scores and absolute/relative deviations from the benchmark mean across repeated runs.

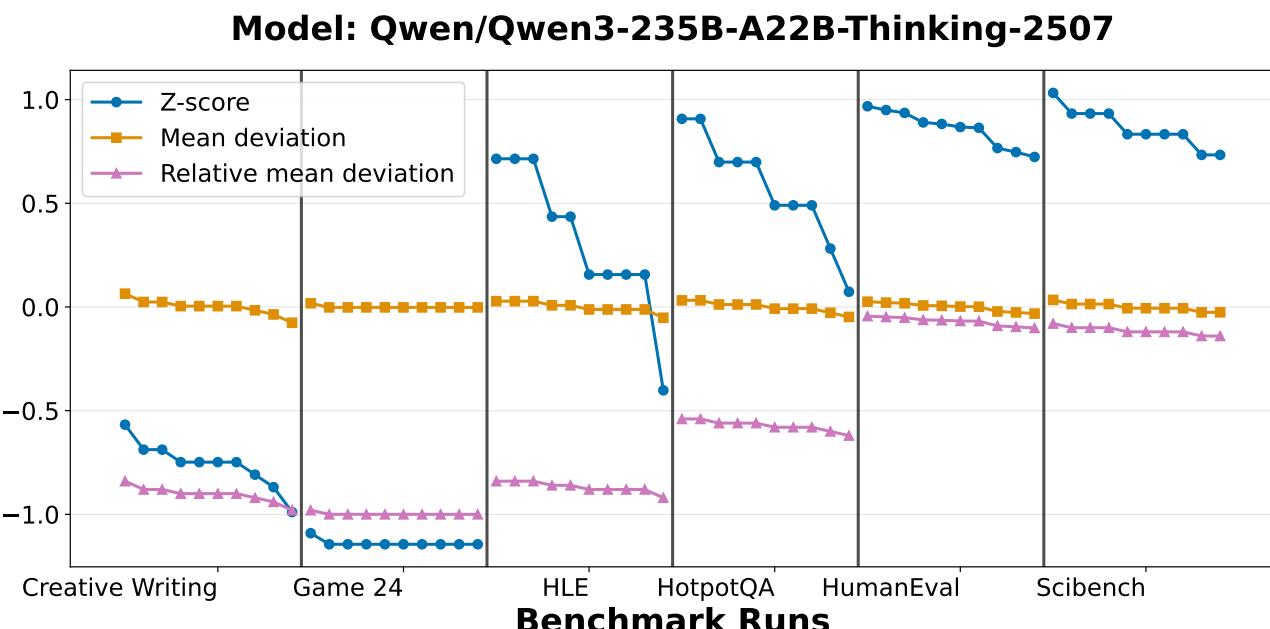


Figure 13. Run-level stability trace for Qwen3-235B Thinking: per-run outcomes grouped by benchmark, showing benchmark-normalized z-scores and absolute/relative deviations from the benchmark mean across repeated runs.

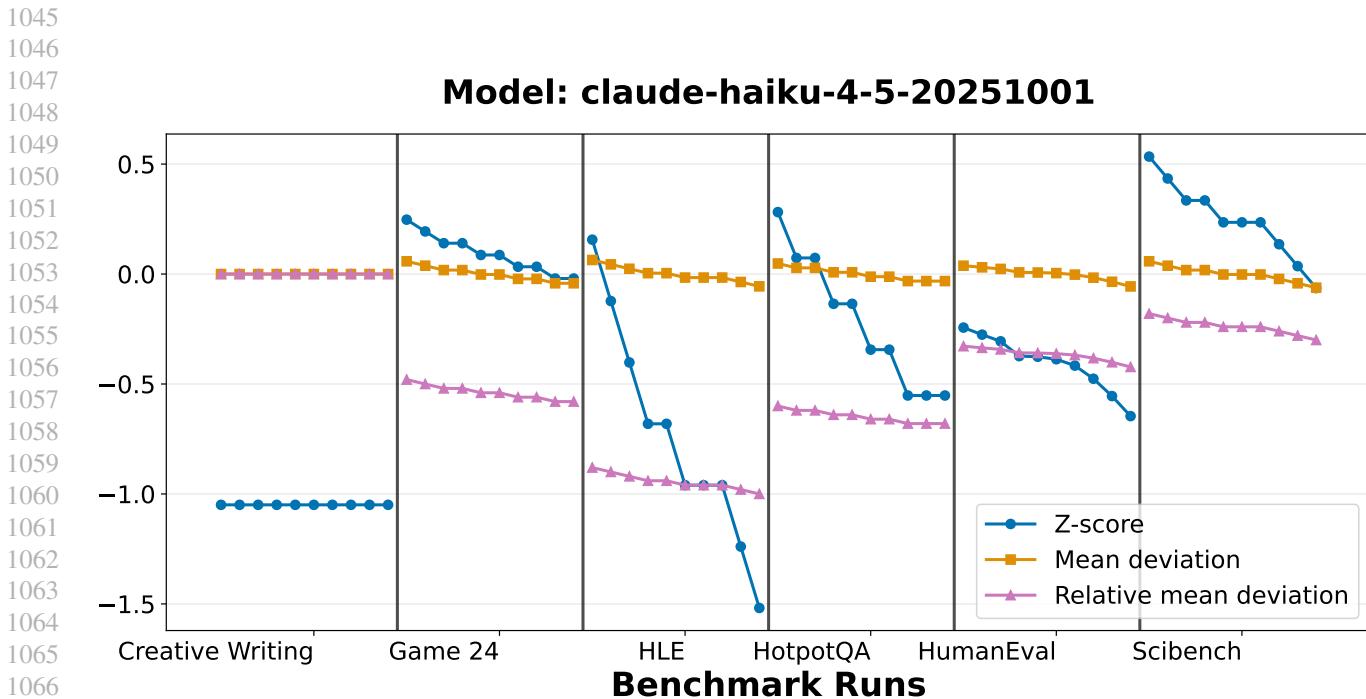


Figure 14. Run-level stability trace for Claud-Haiku 4.5: per-run outcomes grouped by benchmark, showing benchmark-normalized z-scores and absolute/relative deviations from the benchmark mean across repeated runs.

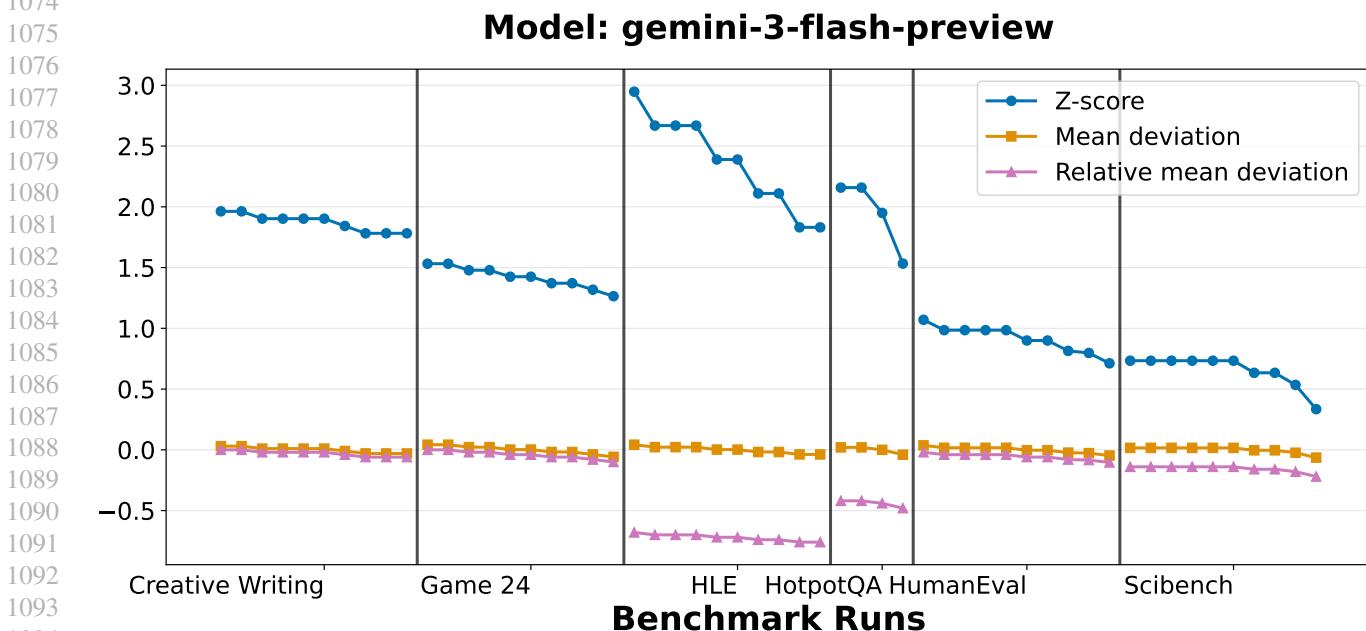


Figure 15. Run-level stability trace for Gemini-3 Flash: per-run outcomes grouped by benchmark, showing benchmark-normalized z-scores and absolute/relative deviations from the benchmark mean across repeated runs.

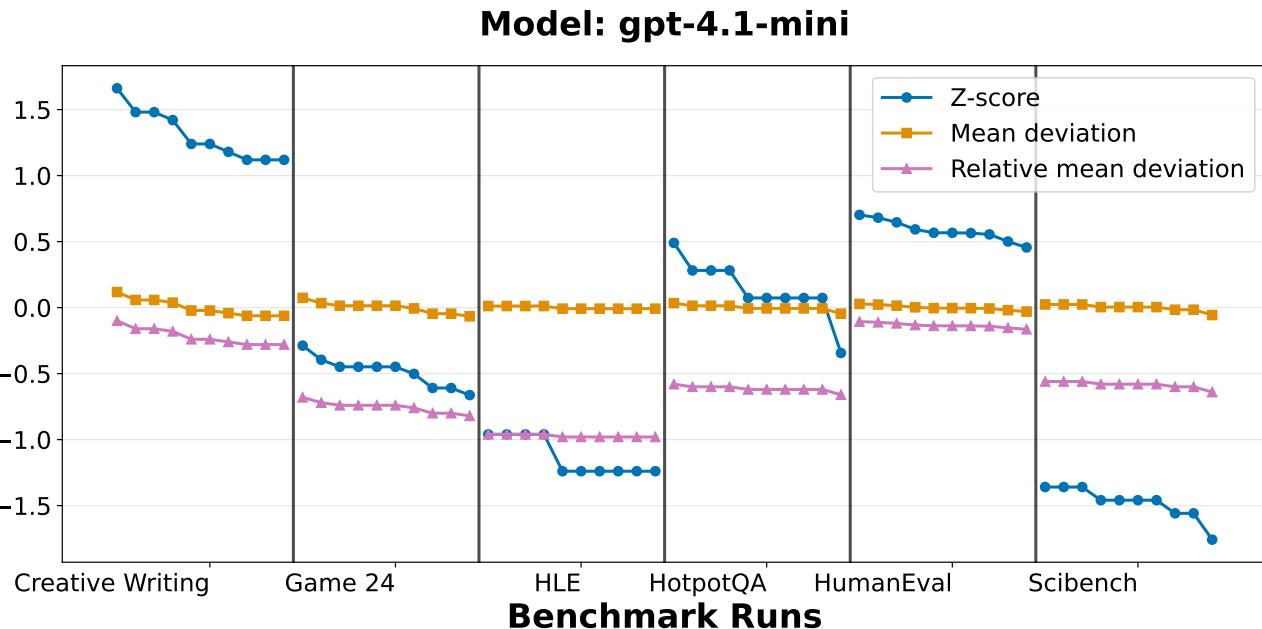


Figure 16. Run-level stability trace for GPT-4.1 Mini: per-run outcomes grouped by benchmark, showing benchmark-normalized z-scores and absolute/relative deviations from the benchmark mean across repeated runs.

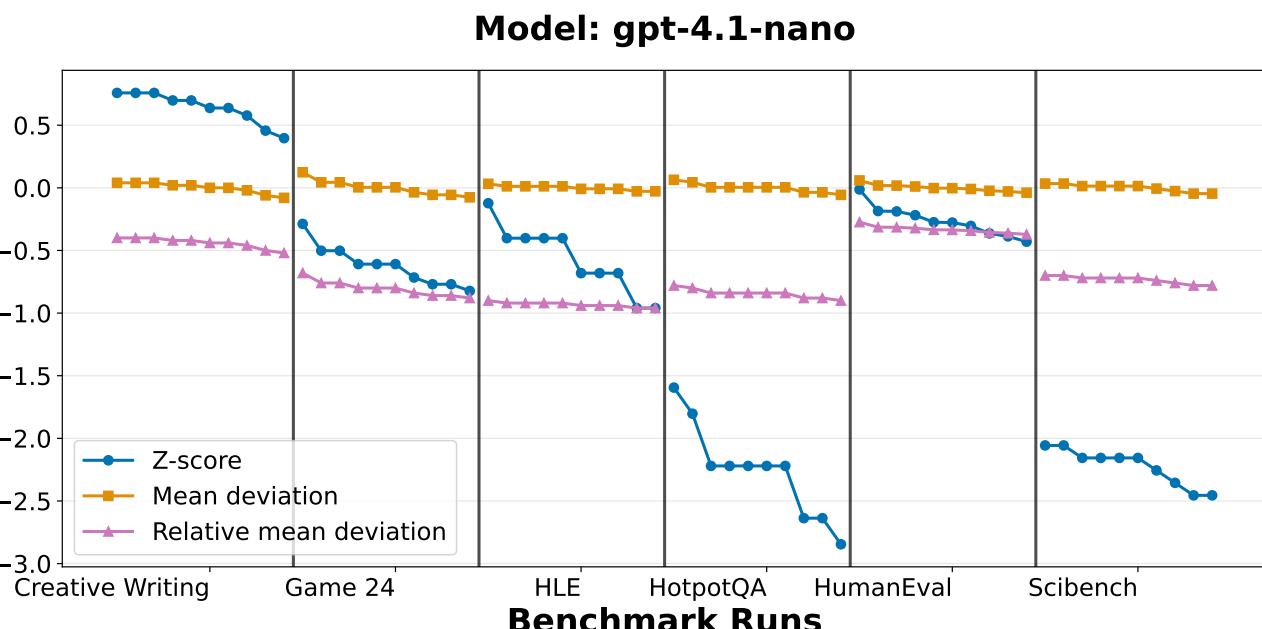


Figure 17. Run-level stability trace for GPT-4.1 Nano: per-run outcomes grouped by benchmark, showing benchmark-normalized z-scores and absolute/relative deviations from the benchmark mean across repeated runs.

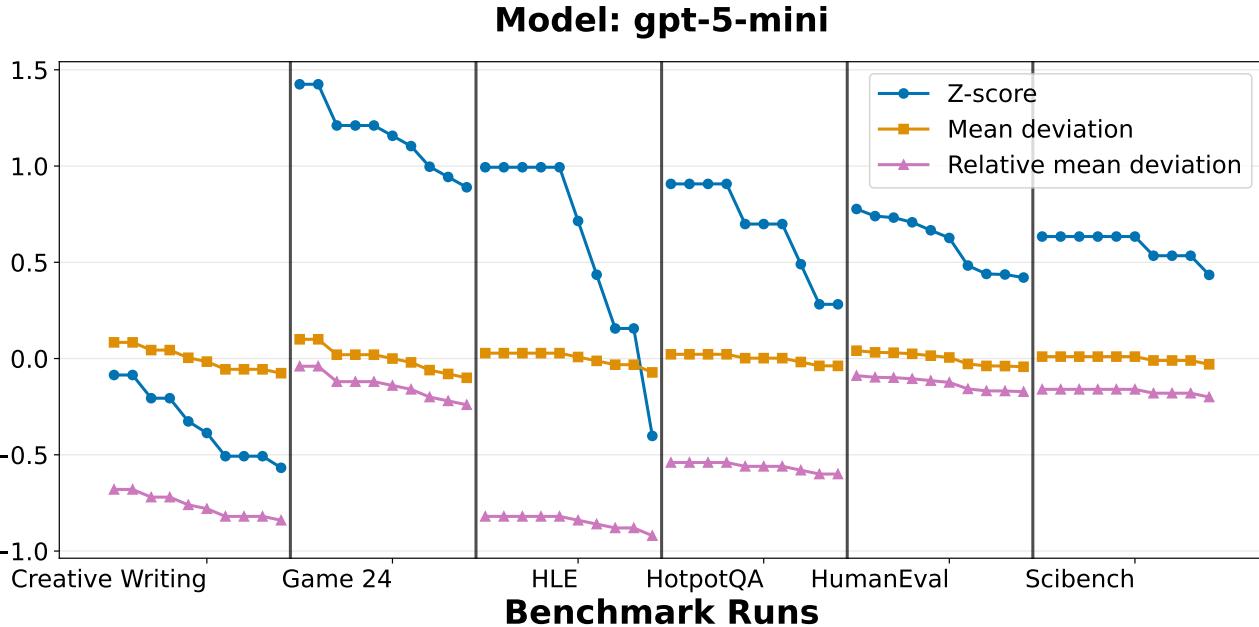


Figure 18. Run-level stability trace for GPT-5 Mini: per-run outcomes grouped by benchmark, showing benchmark-normalized z-scores and absolute/relative deviations from the benchmark mean across repeated runs.

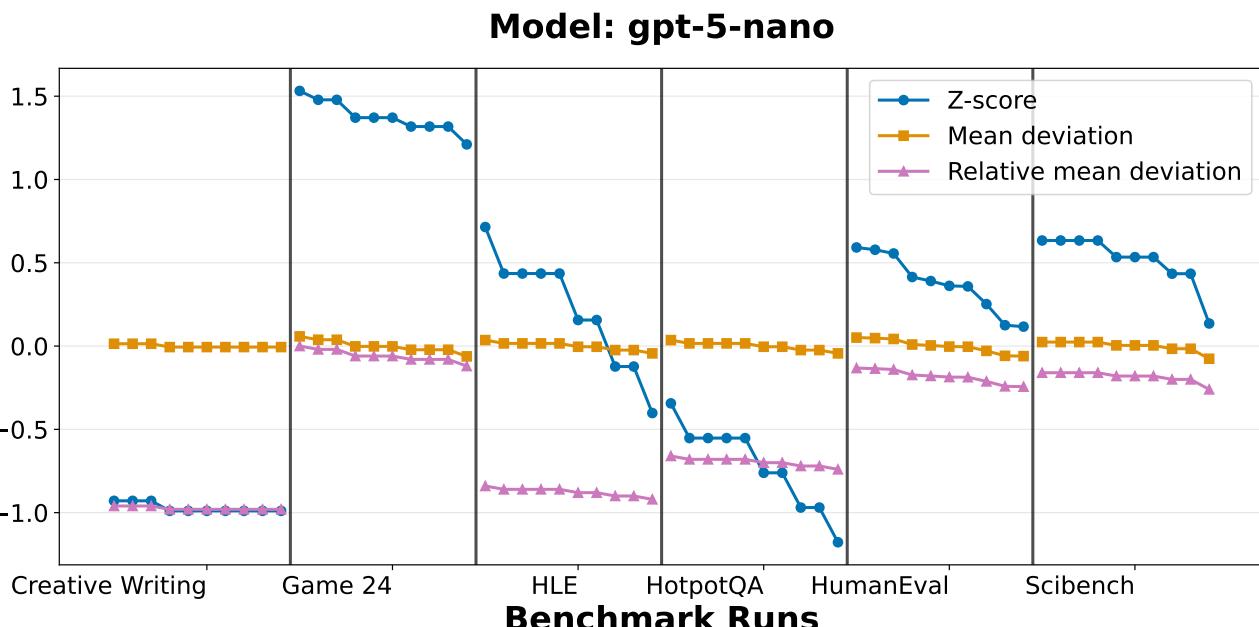


Figure 19. Run-level stability trace for GPT-5 Nano: per-run outcomes grouped by benchmark, showing benchmark-normalized z-scores and absolute/relative deviations from the benchmark mean across repeated runs.