# In LLM Reasoning, there is Irrationality on top of Value Misalignment

**Kejiang Qian**
University of Edinburgh
K.Qian-8@sms.ed.ac.uk

**Fengxiang He**
University of Edinburgh
fhe@ed.ac.uk

## Abstract

Significant progress has been made in aligning LLMs with target value functions. We argue that, even when an LLM has been well aligned in (post-)training, it may still fail to maximise the aligned value in reasoning. We mathematically formalise this gap as rational value risk: the utility discrepancy between a model's deployed reasoning strategy and its rational counterpart, which is defined to be the responses that maximise expected utility in the steepest direction. The estimation error of rational value risk is further decomposed into three components from finite candidates, finite prompts, and imperfect verifiers. Extensive experiments are conducted, covering models Llama-3.1, Qwen-2.5, Tülu-3 families (7B-72B), GPT-5.2, GPT-5.5, and DeepSeek-V4, and benchmarks UltraFeedback, AlpacaEval, GSM8K, MATH, HumanEval, and MathArena. The results validate that (1) rational value risk is widespread; (2) value alignment can reduce, but cannot eliminate, it; (3) the risk is highly sensitive to inference-time reasoning strategy; and (4) longer reasoning improves rationality with diminishing returns. The code is at https://github.com/EVIEHub/LLM-Rationality.

## 1 Introduction

Significant progress has been made in aligning large language models (LLMs) with target value functions. Through value alignment methods such as supervised fine-tuning (SFT) (Ouyang et al., 2022), reinforcement learning from human feedback (RLHF) (Christiano et al., 2017; Stiennon et al., 2020; Ouyang et al., 2022), direct preference optimisation (DPO) (Rafailov et al., 2023), constitutional training (Bai et al., 2022), and reinforcement learning with verifiable rewards (RLVR) (Shao et al., 2024), modern LLMs have become increasingly capable of producing outputs that better reflect human preferences, task-specific objectives, and safety constraints. These advances have substantially improved the helpfulness, reliability, and task performance of LLMs across open-ended dialogue (Thoppilan et al., 2022), mathematical reasoning (Shao et al., 2024), code generation (Chen et al., 2021), amongst others.

This paper argues that value alignment alone does not guarantee ideal reasoning at inference:

> *Even if an LLM has been well aligned in training, it may still fail to maximise the aligned value in reasoning.*

We mathematically formalise this phenomenon by *rational value risk*, defined as the utility discrepancy between a deployed reasoning strategy and its rational counterpart. Here, we define *rational reasoning* as the strategy that maximises expected utility in the steepest direction under the given value function. This definition separates two sources of failure that are often conflated in LLM evaluation: (1) value misalignment, where the learned value function itself is misaligned, and (2) irrational reasoning, where the model fails to realise the best available value under that function.

Since perfectly rational reasoning is generally infeasible to compute, given finite resources, we further study rationality under finite inference-time compute. We introduce a compute-bounded notion of the rational reasoning, defined as the best candidate amongst a finite set of sampled responses. This leads to an empirical estimator of rational value risk that can be applied across both stochastic preference-based tasks and deterministic verifiable reasoning tasks. The estimation error is decomposed into three components: (1) candidate approximation error, which reflects whether the finite candidate set contains a high-utility answer, (2) prompt sampling error, which captures statistical error from evaluating on finitely many prompts, and (3) verification error, which arises when the evaluation signal is stochastic or imperfect.

We conduct extensive experiments across both open-ended conversational tasks and verifiable reasoning tasks. The evaluation covers the Llama-3.1 (Grattafiori et al., 2024), Qwen-2.5(Qwen et al., 2025), and Tülu-3 model (Lambert et al., 2024) families from 7B to 70B parameters, as well as proprietary models including GPT-5.2 (OpenAI, 2025), GPT-5.5 (OpenAI, 2026) and DeepSeek-V4 (DeepSeek-AI, 2026). The benchmarks include UltraFeedback (Cui et al., 2024) and AlpacaEval (Dubois et al., 2024) for conversational preference evaluation, GSM8K (Cobbe et al., 2021) and MATH (Hendrycks et al., 2021) for mathematical reasoning, HumanEval (Chen et al., 2021) for code generation, and MathArena (Balunović et al., 2025) for challenging deployment-style mathematical reasoning.

The empirical results validate four hypotheses:

**H1:** Rational value risk is widespread. Across models and benchmarks, LLMs constantly generate high-utility candidates but fail to consistently deploy them.

**H2:** Value alignment methods can reduce, but cannot eliminate, rational value risk. Rationality is not fully solved by value alignment alone.

**H3:** Rational value risk is highly sensitive to the inference-time reasoning strategy, including sampling temperature and self-consistency.

**H4:** Longer reasoning can improve rationality, but its benefits diminish beyond a certain reasoning budget.

Together, these findings are the first in the literature to formally define and empirically measure rational value risk in LLM reasoning, separating inference-time irrationality from value misalignment, to the best of our knowledge.

## 2 Related works

**Rationality in machine learning.** Valiant (1995) offers a philosophical account of rationality under a PAC-style criterion. Abel (2019) defines bounded rationality in reinforcement learning and also shows that rational decision-making involves a trade-off between representational simplicity and predictive accuracy. From behavioural data, Evans et al. (2025) model bounded rationality through a Wasserstein constraint between the learned policy and a prior. Sunehag and Hutter (2015) derive decision-theoretic axioms for rational reinforcement-learning agents, although these axioms exclude many standard algorithms. Besides these conceptual and empirical advances, Qian et al. (2026) design rationality measures and develop rationality theory for reinforcement learning agents.

However, these results do not apply directly to LLM reasoning, especially post-alignment inference, which is in a significantly different setting. Our work addresses this gap through a full suite of rationality measures, theory, and extensive experiments.

**Rationality in LLMs.** Cognitive-science evaluations test whether models exhibit human-like bounded rationality, heuristics, content effects, and cognitive biases (Binz and Schulz, 2023; Hagendorff et al., 2023; Macmillan-Scott and Musolesi, 2024; Yax et al., 2024; Lampinen et al., 2024; Coda-Forno et al., 2024; Malberg et al., 2025; Brady et al., 2025). Decision-theoretic and economic approaches instead ask whether model choices are consistent with utility maximisation, risk attitudes, revealed preferences, or preference axioms such as transitivity (Chen et al., 2023; Jia et al., 2024; Song et al., 2025; Liu et al., 2025). A third line audits or improves rational behaviour by enforcing belief consistency, logical preference consistency, debiasing, or rational thought prompting (Kassner et al., 2023; Echterhoff et al., 2024; Koo et al., 2024; Gou et al., 2024). Recent surveys and benchmarks consolidate these views into broader notions of rational agents with consistency, grounding, preference orderability, and evidence-aligned decision making (Jiang et al., 2025; Zhou et al., 2025).

Although prior work has shown that LLMs exhibit rationality failures such as inconsistency, cognitive biases, and violations of decision-theoretic principles, it has not explicitly characterised these failures as utility loss that persists after value alignment. In contrast, our work mathematically formalises rationality as an inference-time utility gap between a deployed reasoning strategy and its rational counterpart, thereby separating irrational reasoning from value misalignment.

## 3 Preliminaries

Let $\mathcal{X}$ denote the space of input prompts and $\mathcal{V}$ a finite vocabulary. The reasoning space is $\mathcal{Z} = \bigcup_{T\geq 1} \mathcal{V}^T$, where $T$ denotes the length of a reasoning path $z = (z_1, \ldots, z_T)$. In reasoning, a frozen language model $\pi_\theta$ in a policy set

$\Pi = \{\pi_\theta : \mathcal{X} \to \Delta(\mathcal{Z})\}$ generates a reasoning path $z \in \mathcal{Z}$ by sequentially sampling tokens according to a conditional probability on $x \in \mathcal{X}$: $\pi_\theta(z \mid x) = \prod_{t=1}^{T} \pi_\theta(z_t \mid x, z_{<t})$.

Given a reasoning problems $D = \{\mathbf{x}_i\}_{i=1}^{M}$, where $\mathbf{x}_i = (x_i, y_i^+)$ of $M$ input questions $x_i \in \mathcal{X}$ and preferred (or verifiable) answer $y_i^+ \in \mathcal{Y}$, we define a reasoning strategy $d_\theta(\cdot|x_i) \triangleq d(\pi_\theta(\cdot|x_i))$ where $\mathcal{D} = \{d : \Pi \to \Pi\}$, including temperature sampling, self-consistency, and varying context length. For each input question $x_i$, a frozen language model generates a reasoning path $z_i$ or answer $y_i = g(z_i)$ from an extraction function $g : \mathcal{Z} \to \mathcal{Y}$ by reasoning strategy $d_\theta(\cdot|x_i)$.

Let a $\mathcal{O}$ be the outcome space, a verifier is modelled as an outcome distribution $P : \mathcal{X} \times \mathcal{Y} \times \mathcal{Y} \to \Delta(\mathcal{O})$ evaluates the generated answer and returns an outcome $o_i \sim P(\cdot|\mathbf{x}_i, y_i)$. This formulation covers preference-based tasks, where outcomes may be stochastic due to variation across annotators, judges, or repeated evaluations. In verifiable reasoning tasks, such as mathematical reasoning and code generation, the outcome distribution degenerates to a Dirac distribution as $\delta_{f(\mathbf{x}_i, y_i)}(o_i) : P(o_i = f(\mathbf{x}_i, y_i)|\mathbf{x}_i, y_i) = 1$, where $f : \mathcal{X} \times \mathcal{Y} \times \mathcal{Y} \to \mathcal{O}$ is a deterministic verifier. Let $U : \mathcal{O} \to \mathbb{R}$ be a utility function defined on the outcome space, assigning a utility value $U(o) \in [0, 1]$ to each evaluation outcome $o_i \in \mathcal{O}$.

## 4 Rationality of LLM reasoning

This section defines the rationality measurement for LLM reasoning.

### 4.1 Rationality measures

We first define (perfectly) rational reasoning.

**Definition 1** (rational reasoning)**.** *A reasoning path $z^\circ$ with an answer $y^\circ = g(z^\circ)$ is called perfectly rational, if it maximises the expected utility function $U : \mathcal{O} \to \mathbb{R}$ over the outcome distribution $P(\cdot|\mathbf{x}, y)$ of any reasoning problem $\mathbf{x}$:*

$$y^\circ \in \arg\max_{y \in \mathcal{Y}} \mathbb{E}_{o \sim P(\cdot|\mathbf{x}, y)} U(o).$$

**Remark 1.** *In value alignment, $\pi_\theta$ is updated to align with $U$ in expectation. In contrast, this paper studies the rationality of a frozen language model $\pi_\theta$ at reasoning time through its reasoning strategy $d_\theta$, isolating the effect of the reasoning from that of training.*

LLM reasoning is not always rational. We define a rational value risk to quantify the discrepancy of expected utility between an LLM reasoning strategy and its rational counterpart.

**Definition 2** (rational value risk)**.** *For any reasoning question $\mathbf{x} = (x, y^+)$ drawn from a distribution $\rho$, let $y^\circ$ denote the answer of rational reasoning under $d_\theta^\circ(\cdot \mid x)$, and let $y$ be an answer drawn from a reasoning strategy $d_\theta(\cdot \mid x)$. We define the rational value risk $\mathcal{R}(d_\theta)$ of $d_\theta$ under a utility function $U$ as follows,*

$$\mathbb{E}_{\mathbf{x} \sim \rho, o \sim P(\cdot|\mathbf{x}, y^\circ)} U(o) - \mathbb{E}_{\mathbf{x} \sim \rho, y \sim d_\theta, o \sim P(\cdot|\mathbf{x}, y)} U(o).$$

It is usually infeasible to compute the perfectly rational reasoning and rational value risk, because of finite resources. We thus define compute-bounded rational reasoning given a compute budget of $K$ samplings.

**Definition 3** (compute-bounded rational reasoning)**.** *Let $d_\theta$ denote the reasoning strategy of a frozen language model. For any reasoning problem $\mathbf{x}$, let $y_1, \ldots, y_K \overset{iid}{\sim} d_\theta(\cdot \mid x)$, where each $y_k = g(z_k)$ is the extracted answer from an independently and identically distributed (iid) sampled reasoning path $z_k$. For each sampled answer $y_k$, suppose we obtain $L$ independently and identically distributed evaluation outcomes, $o_{k,1}, \ldots, o_{k,L} \overset{iid}{\sim} P(\cdot \mid \mathbf{x}, y_k)$. We define an empirical expected utility of $y_k$ as*

$$\widehat{U}_L(\mathbf{x}, y_k) = \mathbb{1}\left[\frac{1}{L}\sum_{l=1}^{L} o_{k,l} \geq \frac{1}{2}\right].$$

*A reasoning path $\hat{z}_K^\circ$ with extracted answer $\hat{y}_K^\circ = g(\hat{z}_K^\circ)$ is called* compute-bounded rational *if*

$$\hat{y}_K^\circ \in \arg\max_{1 \leq k \leq K} \widehat{U}_L(\mathbf{x}, y_k).$$

A Monte Carlo estimator is defined below to estimate the rational value risk.

**Definition 4** (empirical rational value risk)**.** *Given a set of reasoning problem $\{\mathbf{x}_i\}_{i=1}^{M}$, let$\{\hat{y}_{i,k}\}_{k=1}^{K} \overset{iid}{\sim} d_\theta(\cdot \mid x_i)$ denote $K$ sampled answers for each input $x_i$. Let $\hat{y}_{i,K}^\circ \in \arg\max_{1 \leq k \leq K} \widehat{U}_L(\mathbf{x}_i, \hat{y}_{i,k})$ be the extracted answer of compute-bounded rational reasoning among the $K$ sampled candidates in Definition 3. The empirical rational value risk $\widehat{\mathcal{R}}_{M,K,L}(d_\theta)$ under a utility function $U$ is defined as follows.*

$$\frac{1}{MK}\sum_{i=1}^{M}\sum_{k=1}^{K}\left[\widehat{U}_L(\mathbf{x}_i, \hat{y}_{i,K}^\circ) - \widehat{U}_L(\mathbf{x}_i, y_{i,k})\right],$$

▷ **Does $\widehat{U}_L(\mathbf{x}_i, \hat{y}^{\circ}_{i,K})$ reduce to pass@$\kappa$, when verifiers are deterministic?**

**Answer:** Let's look at pass@$\kappa$ first:

$$\text{pass@}\kappa = \frac{1}{M}\sum_{i=1}^{M}\left[1 - \frac{\binom{N-c_i}{\kappa}}{\binom{N}{\kappa}}\right],$$

where $N \geq \kappa$ samples are drawn per problem and $c_i = \sum_{j=1}^{N} \mathbb{1}[y_{i,j} = y_i^+]$ is the number of correct samples (Chen et al., 2021). It estimates the probability of solving a verifiable reasoning problem $\mathbf{x}_i$ within $k \in [K]$ attempts.

In contrast, our $\widehat{U}_L(\mathbf{x}_i, \hat{y}^{\circ}_{i,K})$ measures the maximum empirical utility within the candidate set of size $K$, i.e., $\max_{k\in[K]} \widehat{U}_L(\mathbf{x}_i, y_{i,k})$.

Further, when the verifiers or utility $U(o_i)$ is not deterministic, the rational reasoning $\hat{y}^{\circ}_{i,K}$ is defined by the highest utility, not necessarily the exact correct response.

### 4.2 Theoretical guarantee on estimation

This section studies the estimation error in empirically computing the rationality value risk. Detailed proofs are given in Appendix A.

We first decompose the estimation error into three components: candidate approximation error, prompt sampling error, and verification error.

**Lemma 1** (estimation error decomposition)**.** *The estimation error decomposes as*

$$\begin{aligned}&\mathcal{R}(d_\theta) - \widehat{\mathcal{R}}_{M,K,L}(d_\theta)\\ &= \underbrace{\mathcal{R}(d_\theta) - \mathcal{R}_K(d_\theta)}_{\text{(I) candidate approximation}} + \underbrace{\mathcal{R}_K(d_\theta) - \overline{\mathcal{R}}_{M,K}(d_\theta)}_{\text{(II) prompt sampling}}\\ &\quad + \underbrace{\overline{\mathcal{R}}_{M,K}(d_\theta) - \widehat{\mathcal{R}}_{M,K,L}(d_\theta)}_{\text{(III) verification}}\end{aligned}$$

The three terms are bounded below.

**Lemma 2** (candidate approximation error)**.** *Given a $K$ compute budget, let $A_K(d_\theta)$ denote the candidate approximation error $\mathcal{R}(d_\theta) - \mathcal{R}_K(d_\theta)$ and $A_K(d_\theta) \geq 0$. If $U(o) \in [0,1]$, the expected utility of rational reasoning $\mathbb{E}_{\mathbf{x}\sim\rho, o\sim P(\cdot|\mathbf{x},y^\circ)}U(o) = 1$. Let $p_x \triangleq \Pr_{\hat{y}\sim d_\theta(\cdot|\mathbf{x})}[\mathbb{E}_{o\sim P(\cdot|\mathbf{x},\hat{y})}U(o) = 1]$, the candidate approximation error of the reasoning strategy $d_\theta$ is*

$$A_K(d_\theta) = \mathbb{E}_{\mathbf{x}\sim\rho}\left[(1-p_x)^K\right].$$

**Lemma 3** (prompt sampling error)**.** *For any $\delta \in (0,1)$, with probability at least $1-\delta$,*

$$\left|\mathcal{R}_K(d_\theta) - \overline{\mathcal{R}}_{M,K}(d_\theta)\right| \leq \sqrt{\frac{\log(2/\delta)}{2M}}.$$

**Lemma 4** (verification error)**.** *For each candidate $(\mathbf{x}_i, \hat{y}_{i,k})$, define the verifier's subjective preference $q_{i,k} = \mathbb{E}[o_{i,k,l} \mid \mathbf{x}_i, \hat{y}_{i,k}]$, where $o_{i,k,l} \overset{\text{iid}}{\sim} P(\cdot \mid \mathbf{x}_i, \hat{y}_{i,k})$. For any $\delta \in (0,1)$, with probability at least $1-\delta$, if $|q_{i,k} - 1/2| > \sqrt{\log(2MK/\delta)/2L}$, for all $i \in [M]$ and $k \in [K]$, we have*

$$\left|\overline{\mathcal{R}}_{M,K}(d_\theta) - \widehat{\mathcal{R}}_{M,K,L}(d_\theta)\right| = 0.$$

*If the verifier is deterministic, this term is zero.*

**Remark 2.** *Lemma 4 allows verifier outcomes in $\{0, 0.5, 1\}$, since they are bounded in $[0,1]$.*

We thus have the following theorem.

**Theorem 1** (estimation error bound)**.** *Assume $U(o) \in [0,1]$. Assume the verifier's subjective preference satisfies $|q_{i,k} - 1/2| > \epsilon_L$ for all $i \in [M]$ and $k \in [K]$. Then, with probability at least $1-\delta$,*

$$\mathcal{R}(d_\theta) - \widehat{\mathcal{R}}_{M,K,L}(d_\theta) \leq A_K(d_\theta) + \sqrt{\frac{\log(4/\delta)}{2M}}.$$

*If the verifier is deterministic, then the same result holds.*

**Practical insights.** The term $A_K(d_\theta)$ is unavoidable: it measures the utility loss incurred when the finite candidate set fails to contain a high-utility answer. In binary deterministic tasks, $A_K(d_\theta)$ decays exponentially in $K$ whenever $p_x > 0$; however, if the model never samples a high-utility answer for some prompts, $p_x = 0$, sampling more candidates cannot remove this error. For stochastic verifiers, $q_{i,k}$ denotes a probability that the verifier assigns a positive preference to candidate $\hat{y}_{i,k}$. Majority voting can reduce its variance, but it cannot remove verifier bias.

**Sample complexity.** To make the statistical error at most $\epsilon$, if $\min_{i,k} |q_{i,k} - 1/2| \geq \epsilon$, it suffices to take $M = O\big(\epsilon^{-2}\log(1/\delta)\big)$ and $L = O\big(\epsilon^{-2}\log(MK/\delta)\big)$, together with a sampling budget $K$ such that $A_K(d_\theta) \leq \epsilon$. In binary tasks with $p_x \geq p_{\min} > 0$, this requires $K = O\big(p_{\min}^{-1}\log(1/\epsilon)\big)$. Thus, estimating rational value risk depends not only on the number of prompts and verifier samples, but also on whether high-utility answers are reachable under the reasoning strategy $d_\theta$.

## 5 Experiments

Extensive experiments are conducted to verify four empirically verifiable hypotheses, which well establish that *there is irrationality on top of misalignment in LLM reasoning.*

### 5.1 Empirically verifiable hypotheses

**H1: Rational value risk is widespread.** LLMs can generate high-utility answers but fail to deploy them consistently. This gap would appear across different models and benchmarks, including conversational and verifiable reasoning tasks.

**H2: Value alignment methods can reduce, but cannot eliminate, rational value risk.** Value alignment shifts the model distribution toward higher-utility answers, but the deployed reasoning strategy can still select lower-utility outputs. Thus, rational value risk remains after alignment stages such as SFT, DPO, and RLVR.

**H3: Rational value risk is highly sensitive to inference-time reasoning strategy.** For the same frozen model, reasoning strategies can change both its sampled candidate answers and the final deployed answer. Therefore, rationality should be measured with a specified reasoning strategy.

**H4: Longer reasoning improves rationality with diminishing returns.** Increasing the reasoning length can help the model reduce rational value risk, but this effect diminishes after a certain reasoning budget.

### 5.2 Implementation details

**Setup.** We evaluate the rationality of LLM reasoning on two task types: **(1) conversational tasks**: given a conversation dataset $D = \{\mathbf{x}_i\}_{i=1}^{M}$, where $\mathbf{x}_i = (x_i, y_i^+)$, a stochastic verifier compares the LLM's answer $y$ with human preferred answer $y_i^+$ and return an outcome $o_i \in \{0, 0.5, 1\}$, corresponding to win, tie, or lose, respectively, where $o_i \sim P(\cdot|\mathbf{x}_i, y_i)$. Specifically,

$$o_i = \begin{cases} 1, & y_i \succ y_i^+ \quad \text{(win)} \\ 0.5, & y_i \approx y_i^+ \quad \text{(tie)} \\ 0, & y_i \prec y_i^+ \quad \text{(lose)} \end{cases},$$

where $\succ$, $\prec$, and $\approx$ denote verifier preference, non-preference, and indifference, respectively.

The stochastic verifier can be from the LLM itself, reflecting its subjective preference in $P(\cdot|\mathbf{x}_i, y_i)$, or from larger-scale models as external verifiers. **(2) verifiable reasoning tasks:** we consider binary utility functions defined by answer correctness: $U(f(\mathbf{x}_i, y_i)) = \mathbb{1}\left[y_i = y_i^+\right]$.

**Datasets and benchmarks.** For conversational evaluation, we measure the rational value risk on two benchmarks: UltraFeedback (Cui et al., 2024) and AlpacaEval (Dubois et al., 2024). For verifiable reasoning, we use three widely used benchmarks in LLM development and evaluation: GSM8K (Cobbe et al., 2021), MATH (Hendrycks et al., 2021), and HumanEval (Chen et al., 2021). In addition, we adopt one reasoning benchmark as a deployment dataset: MathArena (Balunović et al., 2025). *Its release dates are later than or close to those of the evaluated models*, while it is more difficult than other benchmarks, inducing a distribution shift between the development inputs, $\mathbf{x} \sim \rho$, and (unseen or harder) deployment inputs, $\mathbf{x} \sim \rho^{\dagger}$. Detailed descriptions are in Appendix E.

**Models.** Experiments for H1-H4 use open-weight models, Qwen2.5-7B-Instruct (Qwen et al., 2025), Llama-3.1-8B-Instruct (Grattafiori et al., 2024), Llama-3.1-Tülu-3-8B-(SFT, DPO, RLVR) (Lambert et al., 2024), Qwen2.5-72B-Instruct (Qwen et al., 2025), and Llama-3.1-Tülu-3-70B-(SFT, DPO, RLVR) (Lambert et al., 2024), and proprietary models, GPT-5.2 (OpenAI, 2025), GPT-5.5 (OpenAI, 2026), and DeepSeek-V4-Flash APIs (DeepSeek-AI, 2026) For the external stochastic verifier in conversational tasks, we use the Qwen2.5-14B-Instruct (Qwen et al., 2025) for the 7-8B models and DeepSeek-v4-flash API (DeepSeek-AI, 2026) for the 72B model.

**Configuration.** Unless otherwise specified, $K$=64 reasoning paths are sampled per prompt using temperature sampling with $\tau$=1.0, without top-$p$ or top-$k$ truncation. We consider two verifier settings on UltraFeedback and AlpacaEval: (1) self-as-verifier, where the evaluated model acts as its own verifier, and (2) external verifier using a larger-scale language model. Specifically, we use Qwen2.5-14B-Instruct as the verifier for the 7-8B models with verifier budget $L$=5, and DeepSeek-V4-Flash as the verifier for the Qwen2.5-72B model with verifier budget $L$=3. Estimator calibration and verifier details are presented in Appendix C and Appendix D, respectively.

**Reproducibility and budget.** Experiments are performed with vLLM 0.6+ on six NVIDIA A800 80GB GPUs and APIs including GPT 5.2, GPT 5.5, and DeepSeek-v4-flash. The experiments require approximately 62.2 GPU-hours. API cost is $320 in total, with GPT 5.2($117), GPT 5.5($117), and DeepSeek-v4-flash($86). Additional implementation details are provided in Appendix B.

Table 1: Rational value risk across LLMs on conversational and development benchmarks. Qwen2.5-72B-Instruct is running with compute budget $K$=32; values are reported as mean ± 95% bootstrap confidence interval. Bold indicates the smallest RVR per dataset. (Self.: self-as-verifier; Ext.: external verifier.)

| | | Llama-3.1-8B-Instruct | | | Qwen2.5-7B-Instruct | | | Tülu-3-8B-RLVR | | | Qwen2.5-72B-Instruct | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Task | Dataset | REU | AEU | RVR | REU | AEU | RVR | REU | AEU | RVR | REU | AEU | RVR |
| Conversation | UltraFeedback (self.) | 0.996 | 0.525 | 0.470 ±0.011 | 0.995 | 0.536 | 0.459 ±0.012 | 0.973 | 0.515 | 0.457 ±0.014 | 0.954 | 0.794 | **0.160** ±0.014 |
| | UltraFeedback (ext.) | 0.950 | 0.457 | 0.492 ±0.019 | 0.964 | 0.585 | 0.379 ±0.016 | 0.919 | 0.557 | 0.363 ±0.018 | 0.903 | 0.696 | **0.207** ±0.012 |
| | AlpacaEval (self.) | 1.000 | 0.691 | 0.309 ±0.022 | 0.998 | 0.660 | 0.338 ±0.022 | 1.000 | 0.751 | 0.249 ±0.023 | 0.998 | 0.989 | **0.009** ±0.006 |
| | AlpacaEval (ext.) | 0.998 | 0.792 | 0.207 ±0.029 | 0.998 | 0.872 | 0.126 ±0.022 | 1.000 | 0.910 | 0.090 ±0.019 | 0.983 | 0.922 | **0.062** ±0.013 |
| Verifiable reasoning | GSM8K | 0.990 | 0.780 | 0.210 ±0.013 | 0.990 | 0.906 | 0.085 ±0.010 | 0.984 | 0.861 | 0.123 ±0.012 | 0.986 | 0.958 | **0.027** ±0.007 |
| | MATH | 0.949 | 0.470 | 0.479 ±0.020 | 0.993 | 0.898 | 0.095 ±0.013 | 0.967 | 0.658 | 0.309 ±0.021 | 0.995 | 0.954 | **0.041** ±0.009 |
| | HumanEval | 0.848 | 0.431 | 0.417 ±0.053 | 0.933 | 0.749 | **0.184** ±0.044 | 0.848 | 0.398 | 0.450 ±0.051 | 0.927 | 0.730 | 0.197 ±0.055 |
| | MathArena | 0.117 | 0.003 | **0.114** ±0.081 | 0.383 | 0.087 | 0.297 ±0.102 | 0.133 | 0.008 | 0.125 ±0.084 | 0.500 | 0.183 | 0.317 ±0.097 |

Table 2: Decomposition of total utility discrepancy between true answer and actual reasoning across all evaluated models on MathArena benchmark.

| Size | Model | 1−REU | RVR | %RVR |
|---|---|---|---|---|
| 7–8B | Qwen2.5-7B | 0.617 | 0.297 | 0.325 |
| | Tülu-3-8B-RLVR | 0.867 | 0.125 | 0.126 |
| | Llama-3.1-8B | 0.883 | 0.114 | 0.114 |
| 70–72B | Qwen2.5-72B | 0.500 | 0.317 | 0.388 |
| | Tülu-3-70B-RLVR | 0.600 | 0.340 | 0.361 |
| APIs | DeepSeek-V4-Flash | 0.150 | 0.264 | 0.637 |
| | GPT-5.2 | 0.150 | 0.360 | 0.706 |
| | GPT-5.5 | 0.183 | 0.248 | 0.575 |

Table 3: Rational value risk of Tülu-3-8B family across SFT, DPO, and RLVR stages. Values are reported as mean ± 95% bootstrap confidence interval. (Ext.: external verifier.

| Dataset | | SFT | DPO | RLVR |
|---|---|---|---|---|
| UltraFeedback (ext.) | REU | 0.819 | 0.923 | 0.919 |
| | AEU | 0.278 | 0.550 | 0.557 |
| | RVR | 0.541 ±0.022 | 0.373 ±0.018 | 0.363 ±0.018 |
| AlpacaEval (ext.) | REU | 0.972 | 1.000 | 1.000 |
| | AEU | 0.481 | 0.903 | 0.910 |
| | RVR | 0.491 ±0.033 | 0.097 ±0.019 | 0.090 ±0.019 |
| GSM8K | REU | 0.990 | 0.986 | 0.984 |
| | AEU | 0.588 | 0.858 | 0.861 |
| | RVR | 0.402 ±0.015 | 0.129 ±0.012 | 0.123 ±0.012 |
| MATH | REU | 0.922 | 0.966 | 0.967 |
| | AEU | 0.276 | 0.639 | 0.658 |
| | RVR | 0.646 ±0.019 | 0.327 ±0.021 | 0.309 ±0.021 |
| HumanEval | REU | 0.860 | 0.854 | 0.848 |
| | AEU | 0.148 | 0.250 | 0.398 |
| | RVR | 0.712 ±0.047 | 0.603 ±0.050 | 0.450 ±0.051 |

The code is at https://github.com/EVIEHub/LLM-Rationality.

### 5.3 Experimental results

**H1: Rational value risk is widespread.** Table 1 shows the expected utility by rational reasoning (REU), the expected utility by actual reasoning (AEU), and their difference, rational value risk (RVR), across conversational, mathematical reasoning, and code generation benchmarks. We report the relative contribution of rational value risk as $\%\mathrm{RVR} = \frac{\mathrm{RVR}}{1-\mathrm{AEU}}$. We observe a consistently positive rational value risk across all evaluated settings. It ranges from 0.027 for Qwen2.5-72B on GSM8K to 0.492 for Llama-3.1-8B on UltraFeedback under external verification. On conversational and standard reasoning benchmarks, many models achieve high REU but have lower AEU. This indicates that high-utility answers are often present in the sampled candidate set, but the deployed reasoning strategy does not consistently select them.

Table 2 decomposes the total utility discrepancy $(1 - \mathrm{AEU})$ on MathArena into two parts: (1) the unreachable-utility gap $(1-\mathrm{REU})$, which measures the failure to sample a high-utility answer, and (2) RVR, which measures the failure to deploy a high-utility candidate. For smaller models, the dominant issue is limited finite candidates: high-utility answers cannot be sampled at all. For instance, Tülu-3-8B-RLVR has unreachable-utility discrepancy $(1 - \mathrm{REU})$ of 0.867 and rational value risk of 0.125, so rational value risk accounts for only 12.6% of the total discrepancy. In contrast, larger models and proprietary models improve the ability to sample high-utility answers, but their rational value remains large. GPT-5.2 reduces $(1 - \mathrm{REU})$ to 0.150, but still has rational value risk of0.360, meaning that rational value risk accounts for 70.6% of the total discrepancy.

These results confirm that rational value risk is widespread across both conversational and verifiable reasoning tasks. They also reveal two deployment bottlenecks. For weaker models, the main

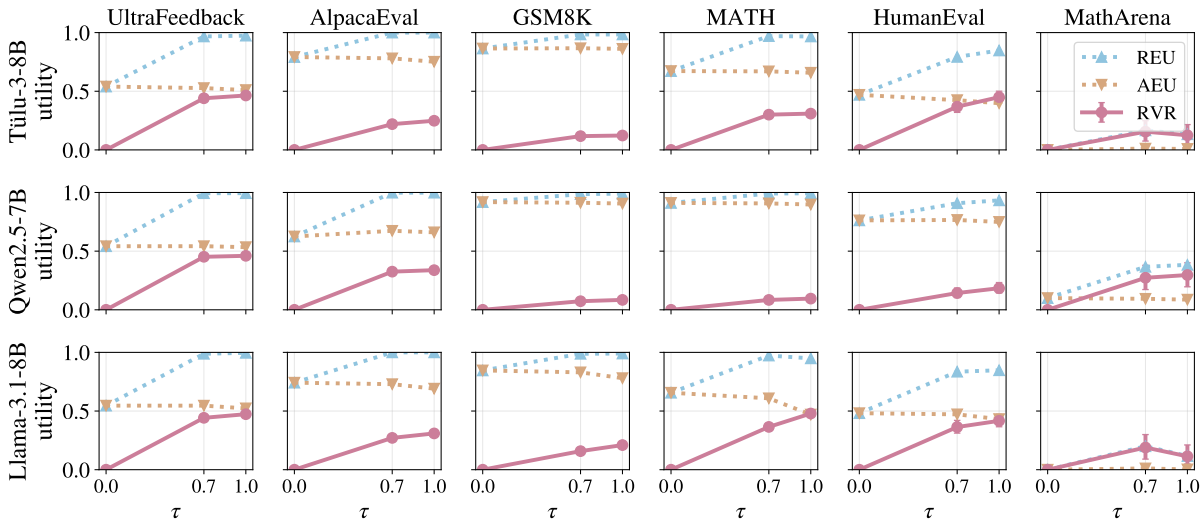


Figure 1: Effects of sampling temperature $\tau = \{0.0, 0.7, 1.0\}$ on rational value risk of Tülu-3-8B-RLVR (Tülu-3-8B), Qwen2.5-7B-Instruct (Qwen2.5-7B), and Llama-3.1-8B-Instruct (Llama-3.1-8B).

Table 4: Decomposition of total utility discrepancy between true answer and actual reasoning across all evaluated models along value alignment pipeline of Tülu-3 models on MathArena benchmark.

| Model | Stage | 1−REU | RVR | %RVR |
|---|---|---|---|---|
| Tülu-3-8B | SFT | 0.817 | 0.179 | 0.179 |
| | DPO | 0.783 | 0.205 | 0.208 |
| | RLVR | 0.867 | 0.125 | 0.126 |
| Tülu-3-70B | SFT | 0.717 | 0.259 | 0.265 |
| | DPO | 0.700 | 0.238 | 0.254 |
| | RLVR | 0.600 | 0.340 | 0.361 |

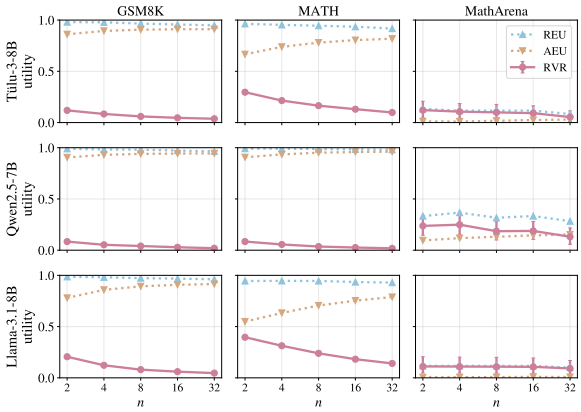


Figure 2: Effects of self-consistency budget $n = \{2, 4, 8, 16, 32\}$ on rational value risk of Tülu-3-8B-RLVR (Tülu-3-8B), Qwen2.5-7B-Instruct (Qwen2.5-7B), and Llama-3.1-8B-Instruct (Llama-3.1-8B).

limitation is the inability to sample high-utility answers. For stronger models, high-utility answers become more reachable, but the model may still fail to deploy them rationally. In this context, rational value risk becomes a central bottleneck in deployment-time reasoning.

**H2: Value alignment reduces but does not eliminate rational value risk.** Table 3 reports REU, AEU, and RVR for the Llama-3.1-Tülu-3-8B family across SFT, DPO, and RLVR stages on conversational and verifiable reasoning benchmarks. We observe that post-training generally improves AEU and reduces rational value risk. On GSM8K, REU remains almost stable across the alignment stages, from 0.990 at SFT to 0.984 at RLVR, while AEU increases from 0.588 to 0.861. As a result, rational value risk decreases from 0.402 to 0.123. These results indicate that value alignment can improve the actual deployed answers from SFT to DPO, reducing the rational value risk. However, rational value risk remains after DPO. After RLVR, Tülu-3-8B-RLVR still has rational value risk of 0.309 on MATH and 0.450 on HumanEval. The improvement from DPO to RLVR is also limited on several benchmarks. For example, rational value risk decreases only from 0.129 to 0.123 on GSM8K, and from 0.097 to 0.090 on AlpacaEval.

Table 4 further decomposes the total utility discrepancy (1 − AEU) on MathArena across the Tülu-3 alignment pipeline. We observe a different pattern on this deployment benchmark. For Tülu-3-70B, the unreachable-utility gap (1 − REU) decreases from 0.717 at SFT to 0.600 at RLVR, indicating an improved ability to sample high-utility answers. At the same time, rational value risk increases from 0.259 to 0.340, and its relative contri-

bution rises from 26.5% to 36.1%. This means that alignment improves the model's capacity to produce high-utility candidates, but it does not guarantee that the deployed answer has the highest utility within the sampled candidate set.

However, rationality improvement of Tülu-3-8B is not clear across the alignment stages, suggesting that value alignment is less effective on challenging benchmarks at 8B scale.

**H3: Rational value risk is highly sensitive to inference-time reasoning strategy.** Figure 1 reports the effect of sampling temperatures $\tau \in \{0, 0.7, 1.0\}$, and Figure 2 reports the effect of self-consistency with voting budgets $n \in \{2, 4, 8, 16, 32\}$ across three models.

Figure 1 shows that increasing temperature generally improves REU, especially on MATH and MathArena, indicating that stochastic sampling allows the model to sample better candidate answers. However, AEU does not improve at the same rate, leading to increasing rational value risk. For example, on MATH with Llama-3.1-8B-Instruct, increasing temperature from $\tau = 0$ to $\tau = 1.0$ raises REU from 0.67 to 0.95, while rational value risk increases to about 0.48. Figure 2 shows a different pattern for self-consistency. Increasing the voting budget generally reduces rational value risk because it improves AEU while REU remains stable. For Tülu-3-8B-RLVR on MATH, increasing the self-consistency budget from $n = 2$ to $n = 32$ reduces rational value risk from about 0.29 to below 0.10. Similar trends appear on GSM8K and MATH, but the improvement is weaker on MathArena.

These results indicate rational value risk depends strongly on the inference-time reasoning strategy. Temperature sampling improves the diversity of the sampled candidate set at the cost of rational value risk. In contrast, self-consistency with more voting budget contributes to rational reasoning.

**H4: Longer reasoning improves rationality with diminishing returns.** Figure 3 illustrates the effect of reasoning length $T \in \{0, 64, 128, 256, 512, 1024, 2048\}$.

Rational value risk does not decrease monotonically with longer reasoning across all models. For example, on GSM8K of Tülu-3-8B-RLVR, it increases from 0.080 at $T = 0$ to 0.456 at $T = 64$, and then decreases to 0.107 at $T = 2048$. The pattern is similar on MATH: its rational value risk increases from 0.048 at $T = 0$ to 0.393 at $T = 128$, before decreasing when $T \geq 256$. In addition,

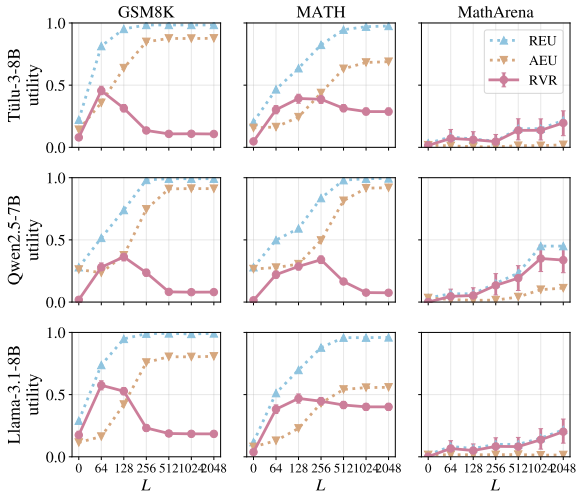


Figure 3: Rational value risk of of Tülu-3-8B-RLVR (Tülu-3-8B), Qwen2.5-7B-Instruct (Qwen2.5-7B), and Llama-3.1-8B-Instruct (Llama-3.1-8B) under varying reasoning lengths $T$.

the benefit of longer reasoning becomes smaller at longer reasoning. On GSM8K and MATH of these three models, their REU and AEU are close to their best values after $T \geq 256$ and $T \geq 1024$, respectively. The MathArena benchmark across three models shows that longer reasoning slightly improves REU, but AEU remains close to zero, while their rational value risk continues to increase.

These results indicate that an appropriate reasoning-token budget can improve the rationality of LLM reasoning, while avoiding unnecessary token cost. However, simply extending the reasoning length is insufficient for harder deployment reasoning tasks.

## 6 Conclusions

This work identifies *rational value risk* as an inference-time failure that can persist after value alignment: the utility gap between a model's deployed reasoning strategy and its rational counterpart. We decompose the estimation error of this risk into finite candidates, finite prompts, and imperfect verifiers, and evaluate it across open and proprietary LLMs on conversational, mathematical, and code-generation benchmarks. Our results show that rational value risk is widespread, reduced but not eliminated by post-training, highly sensitive to reasoning strategy, and only partly mitigated by longer reasoning, suggesting rationality as a distinct evaluation dimension complementary to value alignment.

## Limitations

**Rationality over final answers vs. trajectories.** In this work, we focus on utility defined over final answers, which is the standard evaluation target in preference alignment, mathematical reasoning, and code generation. This choice allows rational value risk to be measured using existing outcome-based evaluators, such as preference judges, exact-match verifiers, unit tests, or symbolic checkers. We acknowledge that an alternative formulation could instead assign utility directly to reasoning paths or intermediate reasoning steps. Such a process-level notion of rationality would assess whether the model follows a valid and efficient trajectory toward its answer, and could reveal failures that are hidden when only the final response is evaluated.

However, defining and verifying utilities over reasoning processes introduces additional challenges, including how to compare multiple valid solution paths, how to score partially correct intermediate steps, and how to evaluate latent or unfaithful reasoning traces. We therefore leave process-level rationality outside the scope of this work and focus on final-answer rationality as a broadly applicable and empirically measurable setting.

**New algorithms for improved rationality.** Another limitation of this work is that it does not propose a new inference or training algorithm for reducing rational value risk. Instead, our goal is to provide a formal definition, estimator, and empirical diagnosis of irrationality in LLM reasoning. This makes the framework primarily evaluative rather than prescriptive: it identifies when and where a model fails to realise high-utility answers, but does not by itself specify the optimal intervention.

Nevertheless, we identified several future applications. Rational value risk can serve as an objective for designing inference-time algorithms, such as verifier-guided search, adaptive sampling, self-consistency, or compute allocation strategies that explicitly minimise unrealised utility. It may also inform post-training by distinguishing failures caused by value misalignment from failures caused by irrational reasoning under an already aligned value function. More broadly, the framework can be used as a diagnostic tool for comparing models, reasoning strategies, and deployment settings, and as a target metric for developing LLMs that not only know what is valuable but also act more reliably to realise it.

## Ethics considerations

This work is fundamental research. All experiments use publicly available datasets and benchmarks; no human subjects or sensitive data are involved. No direct negative societal impacts are identified.

## Acknowledgements

K. Qian was supported in part by the UKRI Grant EP/Y03516X/1 for the UKRI Centre for Doctoral Training in Machine Learning Systems (https://mlsystems.uk/).

## References

David Abel. 2019. Concepts in bounded rationality: Perspectives from reinforcement learning. Master's thesis, Brown University.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, and 32 others. 2022. Constitutional AI: Harmlessness from AI feedback. *arXiv preprint arXiv:2212.08073*.

Mislav Balunović, Jasper Dekoninck, Ivo Petrov, Nikola Jovanović, and Martin Vechev. 2025. MathArena: Evaluating LLMs on uncontaminated math competitions. In *Advances in Neural Information Processing Systems (NeurIPS) Track on Datasets and Benchmarks*.

Marcel Binz and Eric Schulz. 2023. Using cognitive psychology to understand GPT-3. *Proceedings of the National Academy of Sciences*, 120(6):e2218523120.

Oliver Brady, Paul Nulty, Lili Zhang, Tomás E. Ward, and David P. McGovern. 2025. Dual-process theory and decision-making in large language models. *Nature Reviews Psychology*.

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

Yiting Chen, Tracy Xiao Liu, You Shan, and Songfa Zhong. 2023. The emergence of economic rationality of GPT. *Proceedings of the National Academy of Sciences*, 120(51):e2316205120.

Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. In

*Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4302–4310, Red Hook, NY, USA. Curran Associates Inc.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Julian Coda-Forno, Marcel Binz, Jane X. Wang, and Eric Schulz. 2024. CogBench: A large language model walks into a psychology lab. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 9076–9108. PMLR.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2024. UltraFeedback: Boosting language models with scaled AI feedback. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*.

DeepSeek-AI. 2026. DeepSeek-V4: Towards highly efficient million-token context intelligence. https://huggingface.co/deepseek-ai/DeepSeek-V4-Pro/blob/main/DeepSeek_V4.pdf.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B. Hashimoto. 2024. Length-controlled AlpacaEval: A simple way to debias automatic evaluators. In *Conference on Language Modeling (COLM)*.

Jessica Maria Echterhoff, Yao Liu, Abeer Alessa, Julian McAuley, and Zexue He. 2024. Cognitive bias in decision-making with LLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 12640–12653, Miami, Florida, USA. Association for Computational Linguistics.

Benjamin Patrick Evans, Leo Ardon, and Sumitra Ganesh. 2025. Modelling bounded rational decision-making through wasserstein constraints. *arXiv preprint arXiv:2504.03743*.

Tian Gou, Boyao Zhang, Zhenglie Sun, Jing Wang, Fang Liu, Yangang Wang, and Jue Wang. 2024. Rationality of thought improves reasoning in large language models. In *Knowledge Science, Engineering and Management*, pages 343–358, Singapore. Springer Nature Singapore.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Thilo Hagendorff, Sarah Fabi, and Michal Kosinski. 2023. Human-like intuitive behavior and reasoning biases emerged in large language models but disappeared in ChatGPT. *Nature Computational Science*, 3:833–838.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the MATH dataset. In *Advances in Neural Information Processing Systems (NeurIPS) Track on Datasets and Benchmarks*.

Jingru Jia, Zehua Yuan, Junhao Pan, Paul E. McNamara, and Deming Chen. 2024. Decision-making behavior evaluation framework for LLMs under uncertain context. In *Advances in Neural Information Processing Systems*, volume 37.

Bowen Jiang, Yangxinyu Xie, Xiaomeng Wang, Yuan Yuan, Zhuoqun Hao, Xinyi Bai, Weijie J. Su, Camillo Jose Taylor, and Tanwi Mallick. 2025. Towards rationality in language and multimodal agents: A survey. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3656–3675, Albuquerque, New Mexico. Association for Computational Linguistics.

Nora Kassner, Oyvind Tafjord, Ashish Sabharwal, Kyle Richardson, Hinrich Schuetze, and Peter Clark. 2023. Language models with rationality. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 14190–14201, Singapore. Association for Computational Linguistics.

Ryan Koo, Minhwa Lee, Vipul Raheja, Jong Inn Park, Zae Myung Kim, and Dongyeop Kang. 2024. Benchmarking cognitive biases in large language models as evaluators. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 517–545, Bangkok, Thailand. Association for Computational Linguistics.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V. Miranda, Alisa Liu, Nouha Dziri, Xinxi Lyu, Yuling Gu, Saumya Malik, Victoria Graf, Jena D. Hwang, Jiangjiang Yang, Ronan Le Bras, Oyvind Tafjord, Chris Wilhelm, Luca Soldaini, and 4 others. 2024. Tülu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*.

Andrew K. Lampinen, Ishita Dasgupta, Stephanie C. Y. Chan, Hannah R. Sheahan, Antonia Creswell, Dharshan Kumaran, James L. McClelland, and Felix Hill. 2024. Language models, like humans, show content effects on reasoning tasks. *PNAS Nexus*, 3(7):pgae233.

Yinhong Liu, Zhijiang Guo, Tianya Liang, Ehsan Shareghi, Ivan Vulić, and Nigel Collier. 2025. Align-

ing with logic: Measuring, evaluating and improving logical preference consistency in large language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 38518–38539. PMLR.

Olivia Macmillan-Scott and Mirco Musolesi. 2024. (Ir)rationality and cognitive biases in large language models. *Royal Society Open Science*, 11(6):240255.

Simon Malberg, Roman Poletukhin, Carolin M. Schuster, and Georg Groh. 2025. A comprehensive evaluation of cognitive biases in LLMs. In *Proceedings of the 5th International Conference on Natural Language Processing for Digital Humanities*, pages 578–613, Albuquerque, USA. Association for Computational Linguistics.

OpenAI. 2025. Introducing GPT-5.2. https://openai.com/index/introducing-gpt-5-2/. System card: https://cdn.openai.com/pdf/3a4153c8-c748-4b71-8e31-aecbde944f8d/oai_5_2_system-card.pdf.

OpenAI. 2026. GPT-5.5 system card. https://deploymentsafety.openai.com/gpt-5-5/gpt-5-5.pdf.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35.

Kejiang Qian, Amos Storkey, and Fengxiang He. 2026. Rationality measurement and theory for reinforcement learning agents. *Preprint*, arXiv:2602.04737.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Kiwon Song, James M. Jennings III, and Clintin P. Davis-Stober. 2025. Benchmarking the rationality of AI decision making using the transitivity axiom. *Preprint*, arXiv:2502.10554.

Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. 2020. Learning to summarize from human feedback. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Peter Sunehag and Marcus Hutter. 2015. Rationality, optimism and guarantees in general reinforcement learning. *Journal of Machine Learning Research*, 16(40):1345–1390.

Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, YaGuang Li, Hongrae Lee, Huaixiu Steven Zheng, Amin Ghafouri, Marcelo Menegali, Yanping Huang, Maxim Krikun, Dmitry Lepikhin, James Qin, and 41 others. 2022. Lamda: Language models for dialog applications. *Preprint*, arXiv:2201.08239.

Leslie G. Valiant. 1995. Rationality. In *Proceedings of the Eighth Annual Conference on Computational Learning Theory (COLT)*, pages 3–14. ACM.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pages 24824–24837.

Nicolas Yax, Hernán Anlló, and Stefano Palminteri. 2024. Studying and improving reasoning in humans and machines. *Communications Psychology*, 2(51).

Zhilun Zhou, Jing Yi Wang, Nicholas Sukiennik, Chen Gao, Fengli Xu, Yong Li, and James Evans. 2025. Rationality check! benchmarking the rationality of large language models. *arXiv preprint arXiv:2509.14546*.

## A Proofs

For notational brevity, define the expected utility of answer $y$ to input question answer pairs $\mathbf{x} = (x, y^+)$ as

$$V(\mathbf{x}, y) \coloneqq \mathbb{E}_{o \sim P(\cdot|\mathbf{x},y)}[U(o)].$$

Throughout this subsection, we assume $U(o) \in [0, 1]$, and hence $V(\mathbf{x}, y) \in [0, 1]$.

Recall that $y^\circ$ denotes the rational reasoning answer. The rational value risk in Definition 2 can be written equivalently as

$$\mathcal{R}(d_\theta) = \mathbb{E}_{\mathbf{x}\sim\rho}\left[V(\mathbf{x}, y^\circ) - \mathbb{E}_{y\sim d_\theta(\cdot|\mathbf{x})}V(\mathbf{x}, y)\right].$$

Since the rational reasoning answer is generally inaccessible, Definition 3 uses a compute-bounded rational answer selected from $K$ sampled candidates. We therefore introduce the corresponding compute-bounded population risk $\mathcal{R}_K(d_\theta)$:

$$\mathbb{E}_x\mathbb{E}_{\hat{y}_{1:K}}\left[\max_{1\le k\le K} V(\mathbf{x}, \hat{y}_k) - \frac{1}{K}\sum_{k=1}^{K} V(\mathbf{x}, \hat{y}_k)\right],$$

where $x \sim \rho$, $\hat{y}_{1:K} = (\hat{y}_1, \ldots, \hat{y}_K) \sim d_\theta(\cdot \mid \mathbf{x})$.

Given $M$ prompts and $K$ sampled answers per prompt, define the empirical compute-bounded risk$\overline{\mathcal{R}}_{M,K}(d_\theta)$ computed with the true expected utility as

$$\frac{1}{M}\sum_{i=1}^{M}\left[\max_{1\le k\le K} V(\mathbf{x}_i, \hat{y}_{i,k}) - \frac{1}{K}\sum_{k=1}^{K} V(\mathbf{x}_i, \hat{y}_{i,k})\right].$$

The empirical estimator in Definition 4 replaces $V$ with the empirical utility $\widehat{U}_L(\mathbf{x}_i, \hat{y}_{i,k})$

$$\frac{1}{L}\sum_{l=1}^{L} U(o_{i,k,l}), o_{i,k,l} \overset{\text{iid}}{\sim} P(\cdot \mid \mathbf{x}_i, \hat{y}_{i,k}).$$

We write this estimator $\widehat{\mathcal{R}}_{M,K,L}(d_\theta)$ as

$$\frac{1}{M}\sum_{i=1}^{M}\left[\max_{1\le k\le K} \widehat{U}_L(\mathbf{x}_i, \hat{y}_{i,k}) - \frac{1}{K}\sum_{k=1}^{K} \widehat{U}_L(\mathbf{x}_i, \hat{y}_{i,k})\right].$$

This is equivalent to Definition 4, because $\hat{y}^\circ_{i,K}$ is any maximizer of $\widehat{U}_L(\mathbf{x}_i, \hat{y}_{i,k})$ over $k \in [K]$.

### A.1 Proof of Lemma 1

*Proof.* Adding and subtracting $\mathcal{R}_K(d_\theta)$ and $\overline{\mathcal{R}}_{M,K}(d_\theta)$ gives

$$\begin{aligned}
&\mathcal{R}(d_\theta) - \widehat{\mathcal{R}}_{M,K,L}(d_\theta) \\
&= \mathcal{R}(d_\theta) - \mathcal{R}_K(d_\theta) \\
&\quad + \mathcal{R}_K(d_\theta) - \overline{\mathcal{R}}_{M,K}(d_\theta) \\
&\quad + \overline{\mathcal{R}}_{M,K}(d_\theta) - \widehat{\mathcal{R}}_{M,K,L}(d_\theta).
\end{aligned}$$

This proves the claim. □

### A.2 Proof of Lemma 2

*Proof.* By definition,

$$\mathcal{R}(d_\theta) = \mathbb{E}_{\mathbf{x}\sim\rho}\left[V(\mathbf{x}, y^\circ) - \mathbb{E}_{y\sim d_\theta(\cdot|\mathbf{x})}V(\mathbf{x}, y)\right],$$

and

$$\mathcal{R}_K(d_\theta) = \mathbb{E}_{\mathbf{x}\sim\rho}\mathbb{E}_{\hat{y}_{1:K}\sim d_\theta(\cdot|\mathbf{x})}\left[\max_{1\le k\le K} V(\mathbf{x}, \hat{y}_k) - \frac{1}{K}\sum_{k=1}^{K} V(\mathbf{x}, \hat{y}_k)\right].$$

Since $\hat{y}_1, \ldots, \hat{y}_K$ are iid samples from $d_\theta(\cdot \mid \mathbf{x})$,

$$\mathbb{E}_{\hat{y}_{1:K}|\mathbf{x}}\left[\frac{1}{K}\sum_{k=1}^{K} V(\mathbf{x}, \hat{y}_k)\right] = \mathbb{E}_{y\sim d_\theta(\cdot|\mathbf{x})}V(\mathbf{x}, y).$$

Therefore,

$$\begin{aligned}\mathcal{R}(d_\theta) - \mathcal{R}_K(d_\theta) &= \mathbb{E}_{\mathbf{x}\sim\rho}\left[V(\mathbf{x}, y^\circ) - \mathbb{E}_{\hat{y}_{1:K}|\mathbf{x}}\max_{1\le k\le K} V(\mathbf{x}, \hat{y}_k)\right] \\ &= A_K(d_\theta).\end{aligned}$$

Since $y^\circ$ maximizes $V(\mathbf{x}, y)$ over $\mathcal{Y}$,

$$V(\mathbf{x}, y^\circ) \ge \max_{1\le k\le K} V(\mathbf{x}, \hat{y}_k)$$

for every candidate set. Thus $A_K(d_\theta) \ge 0$.

For the binary case, suppose $V(\mathbf{x}, y^\circ) = 1$ and let

$$p_x = \Pr_{\hat{y}\sim d_\theta(\cdot|\mathbf{x})}[V(\mathbf{x}, \hat{y}) = 1].$$

Then $\max_{1\le k\le K} V(\mathbf{x}, \hat{y}_k) = 0$ if and only if all $K$ sampled candidates have utility zero, which occurs with probability $(1 - p_x)^K$. Hence

$$V(\mathbf{x}, y^\circ) - \mathbb{E}_{\hat{y}_{1:K}|\mathbf{x}}\max_{1\le k\le K} V(\mathbf{x}, \hat{y}_k) = (1 - p_x)^K.$$

Taking expectation over $\mathbf{x} \sim \rho$ completes the proof. □

### A.3 Proof of Lemma 3

*Proof.* For each prompt and its preferred answer $\mathbf{x}_i$, define

$$G_i \coloneqq \max_{1\le k\le K} V(\mathbf{x}_i, \hat{y}_{i,k}) - \frac{1}{K}\sum_{k=1}^{K} V(\mathbf{x}_i, \hat{y}_{i,k}).$$

Since $V(\mathbf{x}, y) \in [0, 1]$, we have $G_i \in [0, 1]$. Moreover, $G_1, \ldots, G_M$ are iid because the prompts and candidate sets are sampled independently. By definition,

$$\mathcal{R}_K(d_\theta) = \mathbb{E}[G_i], \qquad \overline{\mathcal{R}}_{M,K}(d_\theta) = \frac{1}{M}\sum_{i=1}^{M} G_i.$$

Hoeffding's inequality gives

$$\Pr\left(\left|\overline{\mathcal{R}}_{M,K}(d_\theta) - \mathcal{R}_K(d_\theta)\right| \ge \epsilon\right) \le 2\exp(-2M\epsilon^2).$$

Setting the right-hand side to $\delta$ yields

$$\epsilon = \sqrt{\frac{\log(2/\delta)}{2M}}.$$

This proves the result. □

### A.4 Proof of Lemma 4

*Proof.* For a fixed candidate $(\mathbf{x}_i, \hat{y}_{i,k})$,

$$\widehat{U}_L(\mathbf{x}_i, \hat{y}_{i,k}) = \mathbb{1}\left[\frac{1}{L}\sum_{l=1}^{L} o_{i,k,l} \geq \frac{1}{2}\right],$$

where

$$o_{i,k,l} \overset{\text{iid}}{\sim} P(\cdot \mid \mathbf{x}_i, \hat{y}_{i,k}).$$

We define $q_{i,k} = \mathbb{E}[o_{i,k,l} \mid \mathbf{x}_i, \hat{y}_{i,k}]$, and $V(\mathbf{x}_i, \hat{y}_{i,k}) = \mathbb{1}\left[q_{i,k} \geq \frac{1}{2}\right]$. Since $o_{i,k,l} \in [0,1]$, Hoeffding's inequality gives

$$\Pr\left(\left|\frac{1}{L}\sum_{l=1}^{L} o_{i,k,l} - q_{i,k}\right| \geq \epsilon\right) \leq 2\exp(-2L\epsilon^2).$$

Taking a union bound over all $MK$ candidates, with probability at least $1-\delta$,

$$\max_{i,k}\left|\frac{1}{L}\sum_{l=1}^{L} o_{i,k,l} - q_{i,k}\right| \leq \epsilon_L,$$

where

$$\epsilon_L = \sqrt{\frac{\log(2MK/\delta)}{2L}}.$$

If

$$|q_{i,k} - 1/2| > \epsilon_L,$$

then the majority vote is stable, i.e.,

$$\widehat{U}_L(\mathbf{x}_i, \hat{y}_{i,k}) = V(\mathbf{x}_i, \hat{y}_{i,k}).$$

Hence, on this event, for every $i$,

$$\left|\max_k \widehat{U}_L(\mathbf{x}_i, \hat{y}_{i,k}) - \max_k V(\mathbf{x}_i, \hat{y}_{i,k})\right| = 0,$$

and

$$\left|\frac{1}{K}\sum_{k=1}^{K} \widehat{U}_L(\mathbf{x}_i, \hat{y}_{i,k}) - \frac{1}{K}\sum_{k=1}^{K} V(\mathbf{x}_i, \hat{y}_{i,k})\right| = 0.$$

Thus,

$$\left|\overline{\mathcal{R}}_{M,K}(d_\theta) - \widehat{\mathcal{R}}_{M,K,L}(d_\theta)\right| = 0.$$

If the verifier is deterministic, then $\widehat{U}_L(\mathbf{x}, y) = V(\mathbf{x}, y)$ for every candidate, so the term is zero. □

### A.5 Proof of Theorem 1

*Proof.* By Lemma 1,

$$\begin{aligned}
&\mathcal{R}(d_\theta) - \widehat{\mathcal{R}}_{M,K,L}(d_\theta) \\
&= \mathcal{R}(d_\theta) - \mathcal{R}_K(d_\theta) + \mathcal{R}_K(d_\theta) - \overline{\mathcal{R}}_{M,K}(d_\theta) \\
&\quad + \overline{\mathcal{R}}_{M,K}(d_\theta) - \widehat{\mathcal{R}}_{M,K,L}(d_\theta).
\end{aligned}$$

By Lemma 2,

$$\mathcal{R}(d_\theta) - \mathcal{R}_K(d_\theta) = A_K(d_\theta).$$

Applying Lemma 3 with failure probability $\delta/2$, we have with probability at least $1-\delta/2$,

$$\mathcal{R}_K(d_\theta) - \overline{\mathcal{R}}_{M,K}(d_\theta) \leq \sqrt{\frac{\log(4/\delta)}{2M}}.$$

Applying Lemma 4 with failure probability $\delta/2$, we have with probability at least $1 - \delta/2$,

$$\overline{\mathcal{R}}_{M,K}(d_\theta) - \widehat{\mathcal{R}}_{M,K,L}(d_\theta) = 0.$$

By a union bound, both events hold simultaneously with probability at least $1 - \delta$. Combining the three bounds gives

$$\mathcal{R}(d_\theta) - \widehat{\mathcal{R}}_{M,K,L}(d_\theta) \leq A_K(d_\theta) + \sqrt{\frac{\log(4/\delta)}{2M}}.$$

If the verifier is deterministic, Lemma 4 gives zero evaluator error. Applying Lemma 3 with failure probability $\delta$ proves the deterministic-verifier version. □

## B Experimental details

This appendix provides additional details on the experimental configuration, prompt construction, evaluation procedures, calibration analyses, and supplementary breakdowns used in the main experiments.

### B.1 Decoding configuration

All experiments use a common sampling interface based on vLLM. Unless otherwise specified, we sample $K = 64$ candidate answers per prompt with

$$(\texttt{temperature}, \texttt{top\_p}, \texttt{top\_k}) = (1.0, 1.0, -1),$$

where $\texttt{top\_k} = -1$ disables top-$k$ truncation. This default configuration is used for H1, H2, and H4. H3 varies the inference-time reasoning strategy by changing the sampling temperature and the self-consistency budget. For greedy decoding, we use $\tau = 0$ with $K = 1$, since multiple greedy samples would be identical.

Table 5 summarises the decoding configuration for each experiment.

**Stop tokens.** We do not configure additional stop tokens. For instruct models, the corresponding chat template already includes the assistant-end token, and generation stops when the EOS token is produced. For the Tülu-3 base model used in the few-shot H2 setting, no chat template is applied and generation is controlled by `max_tokens`.

### B.2 Prompt construction

**Chat-mode prompts.** For each chat-mode model and dataset pair, we apply the model's HuggingFace chat template to a fixed system–user message pair. The system prompt is dataset-specific and specifies the required answer format. The user prompt contains the input question or problem. This design keeps the prompting protocol consistent across models while allowing task-specific answer-format instructions.

```
GSM8K system: Solve the following math problem
step by step. After your reasoning, write the
final numeric answer on its own line in the
format: #### <answer>

MATH system: Solve the following math problem.
Show your reasoning, then put your final
answer in □.

HumanEval system: Complete the following
Python function. Return ONLY the function
definition; do not include explanations or
examples.

MathArena system: Solve the following
competition math problem. Show your reasoning
step by step, then put your final answer
inside □.

UltraFeedback system: (empty string)
AlpacaEval system: (empty string) - in
both, each generator/verifier applies its own
default system prompt.

User templates:
GSM8K : "question"
MATH : "problem"
MathArena : "problem"
HumanEval : "prompt"
UltraFeedback : "prompt"
AlpacaEval : "instruction"
```

**Few-shot prompts for the Tülu-3 base model.** The Tülu-3 base model does not use a chat template. In the H2 base-stage experiments, we prepare a fixed few-shot context to the user question. For GSM8K, we use the standard 8-shot chain-of-thought prompt from Wei et al. (2022). For MATH, we use a 4-shot algebra prompt, matching the algebra-only split used in our experiments.

```
GSM8K 8-shot block (header lines only; full
text in repo):
Question: There are 15 trees in the grove.
...
Answer: ... The answer is 6.

Question: If there are 3 cars in the parking
lot ...
Answer: ... The answer is 5.
(... 6 more shots ...)

Question: Olivia has $23. She bought five
bagels for $3 each. ...
Answer: ... The answer is 8.

MATH 4-shot block (algebra-only, header lines
only):
Problem: Find the sum of all integer values
of n such that 20/(2n-1) is an integer.
Solution: ... The sum of these values is 1 +
0 + 3 + (-2) = 2.

Problem: How many positive 3-digit integers
have all digits different?
Solution: ... 9 * 9 * 8 = 648.

Problem: A right cylinder ... lateral surface
area?
Solution: ... 2 * pi * 2 * 5 = 20π.

Problem: Compute sqrt((31)(30)(29)(28)+1).
Solution: ... = 869.
```

**Conversation-evaluation prompts.** For UltraFeedback, the verifier compares the generated re-

Table 5: Decoding hyperparameters used in the experiments.

| Experiment | $\tau$ | top-$p$ | top-$k$ | $K$ | max_tokens | Notes |
|---|---|---|---|---|---|---|
| H1 | 1.0 | 1.0 | $-1$ | 64 | 1024 | UltraFeedback uses $K = 32$, max_tokens $= 512$ for 7-8B models and Qwen2.5-72B uses max_tokens $=$ 2048. |
| H2 | 1.0 | 1.0 | $-1$ | 64 | 1024 | Same decoding setting across SFT, DPO, and RLVR stages. |
| H3 direct, $\tau = 0$ | 0.0 | 1.0 | $-1$ | 1 | 1024 | Greedy decoding. |
| H3 direct, $\tau = 0.7$ | 0.7 | 1.0 | $-1$ | 64 | 1024 | Stochastic sampling. |
| H3 direct, $\tau = 1.0$ | 1.0 | 1.0 | $-1$ | 64 | 1024 | Default stochastic sampling. |
| H3 self-consistency | 1.0 | 1.0 | $-1$ | 64 | 1024 | Voting budget $n \in \{2, 4, 8, 16, 32\}$. |
| H4 | 1.0 | 1.0 | $-1$ | 64 | $T$ | $T \in \{0, 64, 128, 256, 512, 1024, 2048\}$. |
| Self-as-verifier calls | 0.7 | 1.0 | $-1$ | 1 | 8 | $L = 5$ verifier calls per candidate. |

sponse with the reference response and returns win, tie, or lose. We use $L = 5$ verifier calls per candidate and return the outcomes by majority vote. Candidate and reference positions are randomised across calls to reduce position bias. Invalid evaluations are mapped to ties.

```
System: You are evaluating two responses to a
user's question. Pick the better one. Answer
with exactly one character: A if Response A
is better, B if Response B is better, T if
they are equally good.

User: User question: x
- Response A - a
- Response B - b

Which is better? Answer with one letter (A, B,
or T):
```

### B.3 Reproducibility

Experiments are run with Python 3.11, vLLM 0.6.3, PyTorch 2.5.1, CUDA 12.4, and Transformers 4.45.2. The main open-weight model experiments are run on NVIDIA A800 80GB GPUs. All reported confidence intervals are 95% prompt-bootstrap intervals with $B = 1000$ resamples. We use a fixed seed for each headline cell and cache generations by model, dataset, decoding configuration, prompt template, and sampling budget, so repeated runs reuse the same generated candidates when the configuration is unchanged.

### B.4 GPU hours

The sampling stage requires 62.2 GPU-hours in total. Verification and bootstrap aggregation are performed separately and are not included in the GPU-hour count. Table 6 reports the breakdown by hypothesis.

Table 6: GPU-hour breakdown for each experiment. Verification and bootstrap aggregation are not included.

| Hypothesis | Number of cells | GPU-hours |
|---|---|---|
| H1 | 42 | 25.3 |
| H2 | 33 | 7.7 |
| H3 | 140 | 20.0 |
| H4 | 28 | 9.2 |
| **Total** | 243 | 62.2 |

## C Estimator calibration

We calibrate the empirical rational value risk estimator along the three terms in Theorem 1: the number of sampled candidates $K$, the number of verifier calls $L$, and the number of prompts $M$.

### C.1 Compute budget $K$

To assess the effect of the compute budget, we compute the estimator at truncated budgets $K' \in \{1, 2, 4, 8, 16, 32, 64\}$ using the same cached candidate set. For each $K'$, we compute

$$\mathrm{REU} = \frac{1}{M} \sum_{i=1}^{M} \max_{k \le K'} \widehat{U}_L(\mathbf{x}_i, y_{i,k}),$$

$$\mathrm{AEU} = \frac{1}{MK'} \sum_{i=1}^{M} \sum_{k=1}^{K'} \widehat{U}_L(\mathbf{x}_i, y_{i,k}),$$

and

$$\widehat{\mathcal{R}}_{K'} = \mathrm{REU} - \mathrm{AEU}.$$

Table 7 reports the effect of compute budget $K$ on REU, AEU, and $\widehat{\mathcal{R}}_{K'}$ in verifiable reasoning tasks when $L = 1$. The increase from $K = 32$ to $K = 64$ is small across the reported settings, indicating that $K = 64$ provides a practical compute budget for estimating rational value risk.

Table 7: Effect of compute budget $K$ on REU, AEU, and $\widehat{\mathcal{R}}_{K'}$. Each entry is the prompt mean at budget $K'$. The last column of each $\widehat{\mathcal{R}}_{K'}$ row includes the 95% prompt-bootstrap confidence interval.

| Model | Dataset | Metric | $K$=1 | $K$=2 | $K$=4 | $K$=8 | $K$=16 | $K$=32 | $K$=64 |
|---|---|---|---|---|---|---|---|---|---|
| Tülu-3-8B-RLVR | GSM8K | REU | 0.858 | 0.916 | 0.946 | 0.959 | 0.973 | 0.980 | 0.984 |
| | | AEU | 0.858 | 0.861 | 0.861 | 0.861 | 0.863 | 0.861 | 0.861 |
| | | $\widehat{\mathcal{R}}_K$ | 0.000 | 0.055 | 0.085 | 0.098 | 0.110 | 0.118 | 0.123 ± 0.012 |
| | MATH | REU | 0.682 | 0.786 | 0.846 | 0.896 | 0.928 | 0.948 | 0.967 |
| | | AEU | 0.682 | 0.669 | 0.664 | 0.662 | 0.661 | 0.660 | 0.658 |
| | | $\widehat{\mathcal{R}}_K$ | 0.000 | 0.117 | 0.182 | 0.234 | 0.267 | 0.288 | 0.309 ± 0.021 |
| | HumanEval | REU | 0.463 | 0.573 | 0.646 | 0.665 | 0.768 | 0.823 | 0.848 |
| | | AEU | 0.463 | 0.445 | 0.398 | 0.355 | 0.362 | 0.381 | 0.398 |
| | | $\widehat{\mathcal{R}}_K$ | 0.000 | 0.128 | 0.248 | 0.309 | 0.406 | 0.442 | 0.450 ± 0.051 |
| Qwen2.5-7B-Instruct | GSM8K | REU | 0.907 | 0.944 | 0.961 | 0.974 | 0.978 | 0.985 | 0.990 |
| | | AEU | 0.907 | 0.904 | 0.907 | 0.907 | 0.906 | 0.907 | 0.906 |
| | | $\widehat{\mathcal{R}}_K$ | 0.000 | 0.039 | 0.054 | 0.067 | 0.072 | 0.078 | 0.085 ± 0.010 |
| | MATH | REU | 0.904 | 0.953 | 0.974 | 0.982 | 0.985 | 0.991 | 0.993 |
| | | AEU | 0.904 | 0.901 | 0.903 | 0.900 | 0.898 | 0.897 | 0.898 |
| | | $\widehat{\mathcal{R}}_K$ | 0.000 | 0.051 | 0.071 | 0.082 | 0.087 | 0.094 | 0.095 ± 0.013 |
| | HumanEval | REU | 0.762 | 0.805 | 0.866 | 0.896 | 0.915 | 0.927 | 0.933 |
| | | AEU | 0.762 | 0.762 | 0.761 | 0.758 | 0.750 | 0.755 | 0.749 |
| | | $\widehat{\mathcal{R}}_K$ | 0.000 | 0.043 | 0.105 | 0.138 | 0.165 | 0.172 | 0.184 ± 0.044 |
| Llama-3.1-8B-Instruct | GSM8K | REU | 0.781 | 0.879 | 0.929 | 0.958 | 0.972 | 0.983 | 0.990 |
| | | AEU | 0.781 | 0.777 | 0.782 | 0.782 | 0.781 | 0.780 | 0.780 |
| | | $\widehat{\mathcal{R}}_K$ | 0.000 | 0.102 | 0.147 | 0.176 | 0.191 | 0.203 | 0.210 ± 0.013 |
| | MATH | REU | 0.458 | 0.631 | 0.761 | 0.844 | 0.897 | 0.929 | 0.949 |
| | | AEU | 0.458 | 0.473 | 0.471 | 0.472 | 0.472 | 0.469 | 0.470 |
| | | $\widehat{\mathcal{R}}_K$ | 0.000 | 0.158 | 0.290 | 0.372 | 0.425 | 0.460 | 0.479 ± 0.020 |
| | HumanEval | REU | 0.476 | 0.561 | 0.689 | 0.726 | 0.780 | 0.817 | 0.848 |
| | | AEU | 0.476 | 0.448 | 0.454 | 0.443 | 0.437 | 0.431 | 0.431 |
| | | $\widehat{\mathcal{R}}_K$ | 0.000 | 0.113 | 0.235 | 0.283 | 0.343 | 0.386 | 0.417 ± 0.053 |

Table 8: Effect of $L$ on $\widehat{\mathcal{R}}_K$ at $K = 32$ for the H1 experiment: Tülu-3-RLVR on UltraFeedback. (Std. is the standard deviation.)

| $L'$ | $\widehat{\mathcal{R}}_K$ | Std. |
|---|---|---|
| 1 | 0.362 | 0.0027 |
| 3 | 0.363 | 0.0021 |
| 5 | 0.363 | 0.0000 |
| 7 | 0.362 | 0.0022 |
| 9 | 0.363 | 0.0020 |

Table 9: Bootstrap CI width under different prompt sampling numbers $M$.

| $M'$ | CI half-width | $\sqrt{M'}$·half-width |
|---|---|---|
| 50 | 0.0795 | 0.5624 |
| 100 | 0.0347 | 0.3469 |
| 200 | 0.0291 | 0.4116 |
| 500 | 0.0178 | 0.3973 |
| 1000 | 0.0149 | 0.4696 |
| 1319 | 0.0129 | 0.4698 |

### C.2 Verifier calls $L$

For deterministic verifiers, such as GSM8K, MATH, HumanEval, and MathArena, the verifier output is fixed for a given answer. So repeated verifier calls are not required, i.e., $L = 1$.

In conversational benchmarks, verifier calls $L$ may impact the evaluations from a stochastic verifier. For UltraFeedback, we re-aggregate the recorded verifier calls at different verifier calls $L'$. Table 8 shows that the rational value risk estimation is stable across $L'$, indicating that verifier-call variance is not the dominant source of uncertainty in this setting, supported by the Lemma 4.

### C.3 Prompt sampling number $M$

We assess prompt sampling error by bootstrapping subsets of prompts at different sizes $M'$. Table 9 shows that the confidence interval decreases as $M'$ grows, which is consistent with the $1/\sqrt{M}$ behaviour in Theorem 1.

Table 10: Self-as-verifier diagnostics on the UltraFeedback of H1 experiment. A-pick rate is the fraction of non-tie verifier calls in which the response in position A is selected. Krippendorff's $\alpha$ measures consistency among the $L = 5$ repeated verifier calls for the same candidate answer. It is computed as $1 - D_o/D_e$, where $D_o$ is the observed disagreement among verifier calls and $D_e$ is the average disagreement under random verifier evaluations. A larger value indicates more consistent verifier evaluations. A value close to $0.5$ indicates small position bias.

| Verifier | A-pick rate | Krippendorff's $\alpha$ | Mean $\lvert$margin$\rvert$ |
|---|---|---|---|
| Tülu-3-8B-RLVR | 0.502 | 0.599 | 3.63 |
| Qwen2.5-7B-Instruct | 0.501 | 0.595 | 3.67 |
| Llama-3.1-8B-Instruct | 0.502 | 0.639 | 3.82 |

Table 11: Extraction functions and their conditional-correctness rates of the GSM8K

| Model | #### $N$ | | \boxed{$N$} | | "the answer is $N$" | | Last number | | No number | | $M{\cdot}K$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | Fires | Acc. | Fires | Acc. | Fires | Acc. | Fires | Acc. | Fires | Acc. | |
| Tülu-3-8B-RLVR | 99.9 | 86.2 | $< 0.1$ | 62.5 | — | — | $< 0.1$ | 25.0 | — | — | 84416 |
| Qwen2.5-7B-Instruct | 84.9 | 90.6 | 14.7 | 90.9 | — | — | 0.4 | 72.1 | — | — | 84416 |
| Llama-3.1-8B-Instruct | 1.6 | 74.9 | $< 0.1$ | 78.3 | 22.0 | 83.4 | 76.3 | 76.5 | $< 0.1$ | — | 84416 |

# D Benchmark details

## D.1 Verifiable reasoning tasks

**GSM8K.** We extract the final numeric answer using a priority order over common answer formats, including the canonical GSM8K answer marker, boxed answers, explicit "the answer is" statements, and the last numeric expression in the output. The predicted and reference answers are normalised before numerical comparison.

**MATH and MathArena.** We evaluate mathematical answers using symbolic equivalence checking. The predicted answer is extracted from the final boxed expression when available. If symbolic parsing fails, the candidate is marked incorrect, except for MathArena where we additionally use a strict string-equality fallback. This fallback is conservative, since it can only reject more answers than symbolic equivalence.

**HumanEval.** Code generation tasks are evaluated by executing the generated solution against the provided unit tests in an isolated subprocess with time limits. A candidate receives utility $1$ only if all tests pass, and utility $0$ otherwise.

## D.2 Conversational tasks

For UltraFeedback, we use a self-as-verifier and an external verifier to compare each generated response with the reference response $y^+$, respectively. The verifier returns win, tie, or lose, which is mapped to utilities $1$, $0.5$, and $0$. We use multiple verifier calls and return the outcome by majority vote. The positions of generated and reference responses are randomised to reduce position bias. Table 10 reports diagnostics for the self-as-verifiere setting and shows that the A/B position bias is small.

## D.3 Answer-extraction diagnostics

We further analyse answer-extraction failures for GSM8K and MATH. For GSM8K, Table 11 reports the used extraction function $g(\cdot)$ and the conditional correctness of each function. The residual unparseable rate is below $0.1\%$, indicating that the extractor covers almost all generated numeric answers.

For MATH, Table 12 shows that most failures come from missing final boxed answers, rather than symbolic parser errors. This means that the main source of $U = 0$ is the model's failure to provide the required final-answer format or a correct answer.

# E Difficulty breakdown of mathematical benchmarks

We provide a difficulty breakdown of MATH and HumanEval benchmarks to investigate the pattern of rational value risk on varied difficulty levels.

For MATH, we use the official difficulty levels from 1 to 5. For HumanEval, which has no native difficulty label, we group problems by the token length of the reference solution into short, medium, and long groups. Table 13 reports rational value risk for each difficulty level.

The results show that rational value risk appears across difficulty levels. It is not concentrated only

Table 12: MATH verify failure rates, reported as percentages of candidate answers. Non-semantic failure modes are marked as incorrect.

| Model | No \boxed{} | Empty GT | Parse error | Parse empty | Verify exception | Incorrect | Correct | $M \cdot K$ |
|---|---|---|---|---|---|---|---|---|
| Tülu-3-8B-RLVR | 4.9 | — | — | $< 0.1$ | — | 29.3 | 65.8 | 64000 |
| Qwen2.5-7B-Instruct | 2.6 | — | — | $< 0.1$ | — | 7.7 | 89.8 | 64000 |
| Llama-3.1-8B-Instruct | 27.7 | — | — | $< 0.1$ | — | 25.2 | 47.0 | 64000 |

Table 13: Rational value risk for each difficulty level at $K = 64$.

| Dataset | Bucket | $n$ | Tülu-3-8B-RLVR | Qwen2.5-7B-Instruct | Llama-3.1-8B-Instruct |
|---|---|---|---|---|---|
| MATH | L1 | 113 | 0.066 | 0.009 | 0.202 |
| | L2 | 170 | 0.174 | 0.039 | 0.339 |
| | L3 | 226 | 0.253 | 0.049 | 0.449 |
| | L4 | 234 | 0.320 | 0.088 | 0.557 |
| | L5 | 257 | 0.545 | 0.218 | 0.648 |
| HumanEval | Short | 56 | 0.392 | 0.093 | 0.337 |
| | Medium | 53 | 0.463 | 0.212 | 0.443 |
| | Long | 55 | 0.495 | 0.249 | 0.472 |

in easy problems or only in the hardest problems. On MATH, the risk generally increases with difficulty, especially for weaker models. On HumanEval, the risk is also present across all solution-length groups.

## F Failure-case analysis

We also inspect prompts where the sampled candidate set contains at least one correct answer, but the model has low average utility across samples. Formally, we select prompts satisfying

$$\mathrm{REU}(x_i) = 1 \quad \text{and} \quad \mathrm{AEU}(x_i) \leq 0.1.$$

These cases illustrate the core phenomenon measured by rational value risk: the model can generate a correct or high-utility answer within $K$ samples, but the deployed reasoning strategy still concentrates on lower-utility answers. Table 14 reports the number of such prompts in H1 experiment.

Table 14: Number of failure-case prompts satisfying $\mathrm{REU} = 1$ and $\mathrm{AEU} \leq 0.1$.

| Model | Dataset | Count |
|---|---|---|
| Tülu-3-8B-RLVR | GSM8K | 35 / 1319 |
| Tülu-3-8B-RLVR | MATH | 68 / 1000 |
| Tülu-3-8B-RLVR | HumanEval | 29 / 164 |
| Qwen2.5-7B-Instruct | GSM8K | 22 / 1319 |
| Qwen2.5-7B-Instruct | MATH | 15 / 1000 |
| Qwen2.5-7B-Instruct | HumanEval | 9 / 164 |
| Llama-3.1-8B-Instruct | GSM8K | 45 / 1319 |
| Llama-3.1-8B-Instruct | MATH | 115 / 1000 |
| Llama-3.1-8B-Instruct | HumanEval | 23 / 164 |

## G Statements on AI Assistants in Research and Writing

We used LLM assistants for language polishing, theory refinement, and code refactoring during the development of this work. All scientific claims, theoretical results, experimental results, and analyses were produced and verified by the authors; no AI-generated text was included without author review.