

Agreement Is Not Truth: Characterizing and Controlling Consistent-but-Wrong Reasoning in LLMs

Minje Kim¹, Keeheon Lee^{1*}, Chi-Guhn Lee²

¹Yonsei University, Seoul, Republic of Korea ²University of Toronto, Toronto, ON, Canada
minjekim12@yonsei.ac.kr, keeheon@yonsei.ac.kr, cglee@mie.utoronto.ca

Abstract

Self-consistency treats agreement among sampled chains of thought as evidence that the answer is correct. We show that this reading fails predictably. Across 5 reasoning datasets and 5 generators, *consistent-but-wrong* (CBW) rates track the task rather than the generator, and the pattern holds across agreement thresholds and sample budgets. On StrategyQA, at least 15% of answers are wrong at 90% self-agreement. We prove that no procedure reading only the answer distribution can always separate a question whose majority is right from one whose majority is wrong. We propose Risk-Controlled Self-Consistency (RISC), which combines the majority ratio with a verifier signal and calibrates an acceptance threshold by Learn-then-Test. It certifies the risk target in domain at finite sample size, and accepts nothing when no threshold qualifies. Against self-consistency alone, pooled selective error falls 3.01pp at 0.50 coverage; per-task gains are mixed. At 0.9 coverage, a local generator with a lightweight verifier costs $\sim 9\times$ less than an all-API baseline and shows no detectable selective-error gap on four of five datasets. It trails on MATH500. Higher agreement is not always higher accuracy. Calibration narrows the gap, but more sampling usually does not.

1 Introduction

When a large language model (LLM) produces the same answer across independently sampled chains of thought (Wei et al., 2022; Kojima et al., 2022), that agreement is usually read as confidence. Self-consistency (SC) (Wang et al., 2023) and its budget-adaptive variants (Aggarwal et al., 2023) draw K samples and return the majority. Test-time-compute scaling reinforces the same reading, treating more samples as a substitute for a larger model (Brown et al., 2024; Snell et al., 2025). We

ask whether that confidence tracks correctness. Not reliably.

Existing certificates guarantee something weaker than correctness. Adaptive Consistency (Aggarwal et al., 2023) and related certified-stopping procedures guarantee that the returned answer is the asymptotic majority, not that the majority is correct. Neither covers the case here: a model that is highly self-consistent and wrong (Stechly et al., 2025), even when its verbalized confidence is calibrated separately (Kadavath et al., 2022; Tian et al., 2023; Lin et al., 2022).

Theorem 1 (Section 3) states that no procedure reading only the SC answer distribution can separate a modal-correct question from a modal-wrong one. Corollary 1 turns that into a floor: inside a class of questions sharing one answer distribution, no SC-only selective rule drives conditional error below the class’s wrong-mode share. Such classes are not hypothetical. In our cohort, 98% of high-agreement instances share an answer distribution with an oppositely labeled instance, and the wrong-mode share inside those classes peaks at 23.5%. Beating that floor requires a signal the answer distribution does not carry.

We define $\text{CBW}@ \tau$, the consistent-but-wrong rate at agreement threshold τ , and measure it on 8,222 valid (question, generator) instances across five reasoning datasets and five generators, open-weight and proprietary. CBW is structured by task: per-task variation in $\text{CBW}@0.8$ exceeds per-generator variation by $\sim 7\times$. The threshold does not explain the pattern: StrategyQA still errs on at least 15% of questions where at least 90% of valid extracted answers agree. Nor does the sample budget: raising K from 4 to 16 moves $\text{CBW}@0.8$ by at most 1.5 points on four of five datasets. Reasoning training lowers the rate but leaves the task pattern in place: two frontier reasoning models from different vendors still answer StrategyQA incorrectly 8.5%

* Corresponding author.

and 11.9% of the time at 90% agreement, against 1.4% each on GSM8K.

Section 5 introduces Risk-Controlled Self-Consistency (RISC). RISC scores each question by combining the majority ratio with a verifier signal (Cobbe et al., 2021; Lightman et al., 2024; Uesato et al., 2022). Learn-then-Test (LTT) (Angelopoulos et al., 2025; Bates et al., 2021) then calibrates an acceptance threshold with finite-sample in-domain selective-risk validity. The pooled gain is largest where an accepted wrong answer costs most, cutting selective error by 3.01pp at 0.50 coverage under both paired and question-level cluster bootstrap, and it fades as coverage rises (Section 5.3). Per-task effects are mixed. RISC lowers error on GSM8K, MATH500, and ARC and raises it on StrategyQA and CommonsenseQA (Table 3). Calibration survives a change of generator but not a change of task (violation rates 15–20% against 31–63%), and per-task Mondrian recalibration restores validity.

One strong verifier is enough at our default operating point. A noisier verifier costs coverage at strict targets, while a high-quality ensemble buys lower risk at a larger API bill. At 0.9 selective coverage, Qwen2.5-7B with RISC costs $\sim 9\times$ less in metered API spend than GPT-5.4-mini majority voting and shows no statistically detectable selective-error gap on four of five datasets. MATH500 keeps a +7.94pp gap in favor of GPT-5.4-mini. RISC calibrates whether to accept the vote; it does not change the vote.

High-confidence errors have been reported before (Madaan et al., 2023; Welleck et al., 2023; Stechly et al., 2025; Kuhn et al., 2023; Manakul et al., 2023; Tan et al., 2025). Our contribution has three parts. The first is CBW@ τ , measured across 5×5 generator-task cells, establishing a task structure robust to τ , to sample budget, and to reasoning training. The second is RISC, a verifier-informed risk-control procedure that cuts metered API cost $\sim 9\times$ against API majority voting, using a local generator and a lightweight verifier, with per-task recalibration under distribution change. The third is a wrong-mode-share lower bound on SC-only selective error inside classes sharing one answer distribution (Cor. 1), with the measurement showing they are the common case.

2 Background and Related Work

2.1 Self-consistency and test-time sampling

Self-consistency (Wang et al., 2023) aggregates K chain-of-thought samples and returns the majority answer. Budget-adaptive variants refine the sampling rule. Adaptive Consistency (Aggarwal et al., 2023) stops early under a bound on the probability of disagreeing with the asymptotic majority, and related stopping-time analyses certify the same probability. These guarantees concern agreement: K samples identify the modal answer. Neither guarantees correctness, and Theorem 1 makes the distinction precise.

A second family changes which answer wins. Tree-structured search (Yao et al., 2023; Hao et al., 2023; Wan et al., 2024), reasoning-aware adaptive sampling (Wan et al., 2025), and ranked voting (Wang et al., 2025) all act on the aggregation step. Confidence-weighted voting (Taubenfeld et al., 2025) is the closest of these to our setting: it reweights each chain by a confidence estimate and returns a reweighted point prediction, with no risk statement attached. RISC leaves the vote untouched and calibrates whether that vote should be accepted. Section 5.4 measures both schemes on the same instances.

2.2 Verification, abstention, and selective prediction

A separate line attaches an external scoring head to candidate answers or traces. Process- and outcome-supervised verifiers (Cobbe et al., 2021; Lightman et al., 2024; Uesato et al., 2022; Wang et al., 2024a,b; Lu et al., 2024) score correctness at the step or answer level. Multi-agent debate (Du et al., 2024; Khan et al., 2024) reads cross-model disagreement as a verification signal. Revision and reflection methods (Madaan et al., 2023; Shinn et al., 2023; Welleck et al., 2023) refine a candidate answer instead. Empirical work asks whether verifiers detect high-confidence errors (Lightman et al., 2024; Stechly et al., 2025). Our setup is orthogonal to verifier design, since a stronger verifier simply supplies a stronger signal to the same acceptance score.

Selective prediction is the natural statistical framing. The classical setup (Geifman and El-Yaniv, 2017; El-Yaniv and Wiener, 2010) pairs a classifier with a selection function, bounding conditional error at maximal coverage. Distribution-free finite-sample versions come from conformal prediction

(Vovk et al., 2005) and its risk-controlling variants, RCPS (Bates et al., 2021) and Learn-then-Test (Angelopoulos et al., 2025). RISC sits closest to selective prediction under LTT calibration, and departs in what it accepts: a self-consistency majority answer, scored by agreement plus a verifier signal. Inference-time scoring and verifier-guided decoding (Khalifa et al., 2023; Xie et al., 2023) also add information beyond the answer distribution, but steer generation with it. RISC calibrates acceptance of an answer already produced.

2.3 Consistent-but-wrong failures

CBW has been observed in step-verifier work (Stechly et al., 2025; Lightman et al., 2024), refinement work (Madaan et al., 2023; Welleck et al., 2023), calibration studies (Kadavath et al., 2022; Tian et al., 2023; Xiong et al., 2024; Jiang et al., 2021), and shortcut-learning analyses (Geirhos et al., 2020). Tan et al. (2025) study CBW errors in their own right and report that they survive model scaling and blunt the detectors built to catch them. Our question starts where that observation ends: what can still be certified once the errors are consistent. Theorem 1 and Corollary 1 formalize the gap under the SC-only information set. We then characterize it across 5×5 generator-task cells with bootstrap CIs, and calibrate an accept/abstain layer carrying a finite-sample certificate and an admissibility fallback.

3 When Self-Consistency Cannot Certify Correctness

A procedure that observes only the self-consistency distribution cannot certify the correctness of its modal answer, even in principle. The obstruction is familiar from selective classification, and self-consistency is one instance (Geifman and El-Yaniv, 2017). A rule that never sees the label cannot separate two instances it scores identically. We state the result for the answer distribution μ , the information set self-consistency certificates use.

Setup. Let $\mu_g(q) \in \Delta(\mathcal{A})$ denote the answer distribution induced by generator g on question $q \in \mathcal{Q}$, with $y(q) \in \mathcal{A}$ the ground-truth label, and work in the population limit. An *SC-only certifier* $\mathcal{C} : \Delta(\mathcal{A}) \rightarrow \{\text{accept}, \text{abstain}\}$ depends only on μ and commits to $\text{mode}(\mu)$ when it accepts. Adaptive Consistency (Aggarwal et al., 2023), certified-stopping rules, and majority-vote thresholding are all SC-only in this sense. We measure $\mathcal{C}$ by cov-

erage $c(\mathcal{C}) = \Pr[\mathcal{C} = \text{accept}]$ and selective error $\beta(\mathcal{C}) = \Pr[\text{mode}(\mu_g(q)) \neq y(q) \mid \mathcal{C} \text{ accepts}]$, the standard objects of selective prediction (Geifman and El-Yaniv, 2017; El-Yaniv and Wiener, 2010) and conformal risk control (Vovk et al., 2005; Bates et al., 2021).

3.1 Theorem: SC-only certificates cannot distinguish truth from agreement

Theorem 1 (Indistinguishability of correctness under SC-only signals). There exist a generator g and two distinct labeled task instances $z_1 = (q_1, y_1)$ and $z_2 = (q_2, y_2)$ with $y_1 \neq y_2$ such that $\mu_g(q_1) = \mu_g(q_2) = \mu$, $\text{mode}(\mu) = y_1$, and $\text{mode}(\mu) \neq y_2$. Any SC-only certifier $\mathcal{C}$ produces the same output on z_1 and z_2 : if it accepts, it is correct on the modal-correct instance z_1 and incorrect on the modal-wrong instance z_2 ; if it abstains, it forgoes coverage on a modal-correct instance. (Proof in Appendix A.)

The theorem rules out every post-hoc rule on μ alone in the worst case: thresholds on the majority ratio, entropy, or any other histogram functional. Existing self-consistency certificates (Aggarwal et al., 2023) certify properties of μ alone, and Corollary 1 quantifies the error floor.

3.2 Corollary: a selective-risk lower bound

Corollary 1. Let $\mathcal{T}_\mu$ be the indistinguishable subfamily sharing μ from Theorem 1, with z_1 - and z_2 -type instances in proportion p_1, p_2 ($p_1 + p_2 > 0$). For any SC-only certifier $\mathcal{C}$ that accepts on μ with positive probability, the conditional selective error within $\mathcal{T}_\mu$ obeys $\beta_\mu(\mathcal{C}) := \Pr[\text{mode}(\mu) \neq y(q) \mid \mathcal{C} \text{ accepts}, q \in \mathcal{T}_\mu] \geq p_2/(p_1 + p_2)$, with $\beta_\mu = 0$ unattainable at positive coverage on $\mathcal{T}_\mu$ unless p_2 vanishes.

Operationally, an SC-only certifier cannot drive its conditional selective error below the wrong-mode share of the class. Lowering coverage does not help, because the certifier abstains on correct instances at the same rate. The empirical $\text{Err@}\tau$ rates of Section 4 are the finite-sample, ρ -thresholded analogue of this floor on the $\rho \geq \tau$ subset. StrategyQA makes it concrete, holding $\geq 15\%$ high-agreement error at $\tau=0.9$ (parseable MATH500 is treated separately in Appendix E.5).

How often the premise holds. A worst-case construction matters only if the equivalence classes it needs are populated. We test this on the cohort of Section 4. Group every instance by its

answer-distribution signature, the number of distinct extracted answers together with the majority vote count at $K=16$. At high agreement this signature nearly determines $\hat{\mu}$. Call a signature class *mixed* when it holds both a modal-correct and a modal-wrong instance, so that an SC-only certifier must decide both the same way. Of the 5,369 high-agreement instances ($\rho \geq 0.9$), 98.3% lie in a mixed class, ranging from 92.1% on MATH500 to 99.9% on CommonsenseQA. The wrong-mode share inside those classes, the floor of Corollary 1, is 8.8% pooled and runs from 2.0% on GSM8K to 23.5% on StrategyQA. Indistinguishability at high agreement is therefore the ordinary case in our data.

Where the wrong-mode share is small, the bound is nearly tight, as on GSM8K with $\sim 2\%$ conditional error at $\rho \geq 0.9$. Where it is not small, beating the floor needs a signal v that correlates with $y(q)$ conditional on μ . Candidates are external verifiers (Cobbe et al., 2021; Lightman et al., 2024; Uesato et al., 2022), debate and self-checking (Du et al., 2024; Khan et al., 2024; Manakul et al., 2023; Saunders et al., 2022), and model-internal confidence (Appendix E.10). RISC takes external verifiers as a feature in a calibrated selective-risk layer with finite-sample validity (Section 5).

4 Characterizing Consistent-but-Wrong Reasoning

The limitation of Section 3 has an empirical counterpart. Across five datasets and five generators, we measure how often self-consistency returns a high-agreement, incorrect answer, and how that rate moves with the agreement threshold and the sample budget. Consistent-but-wrong answers are not anecdotal failures in this cohort.

4.1 Setup

We evaluate five datasets (GSM8K (Cobbe et al., 2021), StrategyQA (Geva et al., 2021), MATH500 (Hendrycks et al., 2021), ARC-Challenge (Clark et al., 2018), CommonsenseQA (CSQA) (Talmor et al., 2019)) and five generators: Qwen2.5-7B and Qwen2.5-14B (Qwen Team, 2024), Llama-3.1-8B (Grattafiori et al., 2024), Mistral-7B-Instruct-v0.3 (Jiang et al., 2023; Mistral AI, 2024), and GPT-5.4-mini (OpenAI, 2026). For each question q we draw $K = 16$ chain-of-thought samples, extract final answers, and return the majority $\hat{y}(q)$. The *majority ratio* $\rho(q) \in [0, 1]$ divides the count

agreeing with the majority by the number of valid extractions. Parsing failures are excluded from the denominator, and instances with no extracted majority are dropped. Restricting to $K_{\text{valid}} \geq 12$ leaves four of five datasets within 0.5pp of the headline rate, and only MATH500 is denominator-sensitive (Appendix E.22).

Per-cell sizes run from 100 to 500 questions, for 8,222 valid instances in total. BIG-Bench Hard (BBH) serves as a sixth open-weight-only dataset (Appendix E.29). MATH500 golds are LaTeX expressions on which string equality fails, so we recover answers with a SymPy-based check that retains $\sim 96\%$ of items (Appendix E.5). Model access, prompts, and extraction rules are in Appendix B and C. We measure CBW at an agreement threshold τ in two ways. The unconditional rate is $\text{CBW@}\tau = \Pr[\rho \geq \tau \wedge \hat{y} \neq y]$. The conditional error on high-agreement instances is $\text{ERR@}\tau = \Pr[\hat{y} \neq y \mid \rho \geq \tau]$, the finite-sample analogue of the wrong-mode floor in Corollary 1.

4.2 High-Agreement Errors Are Task-Structured

Figure 1 reports $\text{CBW@}0.8$ for every cell. The task drives the differences between cells. Generator changes move the differences far less. Mean $\text{CBW@}0.8$ runs from 2.1% on GSM8K to 20.8% on StrategyQA, with MATH500 at 3.4%, ARC at 6.2%, and CommonsenseQA at 9.6%. The MATH500 rate uses SymPy recovery and rises to 4.2% once the very small Mistral $\times$ MATH500 cell ($n_h=7$) is dropped, which leaves the ordering unchanged (Appendix E.5, E.26). MATH500 stays below StrategyQA however its unparseable items are treated (Appendix E.11).

Generator spread is far narrower than dataset spread, from 6.7% for Llama-3.1-8B to 9.6% for Qwen2.5-7B. The standard deviation of mean $\text{CBW@}0.8$ across datasets is about seven times the standard deviation across generators. A two-way variance decomposition agrees: $\eta_{\text{task}}^2=0.91$ at $F(4, 16)=50.7$ against $\eta_{\text{gen}}^2=0.02$. A question-clustered bootstrap puts $\Pr[\eta_{\text{task}}^2 > 5\eta_{\text{gen}}^2]=1.0$ (Appendix E.25). Task identity, not generator identity, predicts CBW. GPT-5.4-mini and Mistral both land inside the per-task variation. Our cohort spans four model families, one of them proprietary. On the open-weight subset, BBH reaches 8.4% mean $\text{CBW@}0.8$, between ARC at 5.6% and StrategyQA at 20.6%. A Qwen3-8B generator reproduces the same ordering (Appendix E.29, E.27).

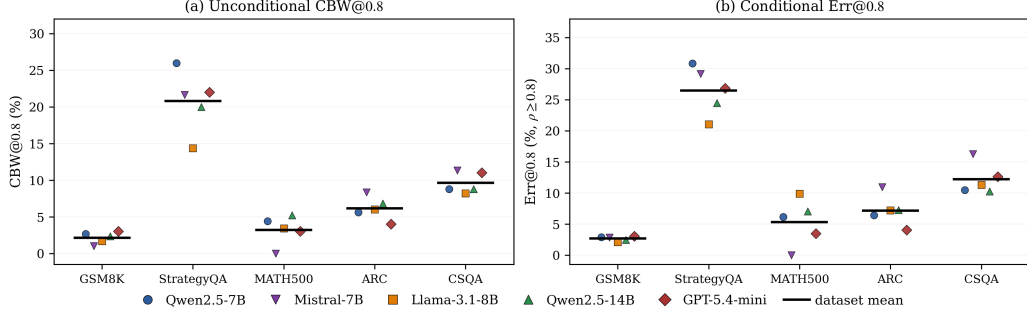


Figure 1: Unconditional CBW@0.8 (a) and conditional Err@0.8 (b) across 5 datasets and 5 generators; each marker is one (dataset, generator) cell and the black bar is the dataset mean. The dataset ordering is preserved across both metrics. Per-dataset extraction rules (Appendix C.3), the very small Mistral $\times$ MATH500 cell with $n_h=7$ (Appendix E.26), the per-cell heatmap, and the two-way variance decomposition (task $\eta^2=0.91$, generator $\eta^2=0.02$) are in Appendix D.1, E.25.

4.3 Agreement does not uniformly converge to truth

Test-time compute scaling suggests that the gap between agreement and truth should close as agreement grows (Brown et al., 2024; Snell et al., 2025). Figure 2 tests that suggestion by stratifying questions on the majority ratio and reporting conditional error within each bin. Whether the gap closes depends on the task. On GSM8K, conditional error decays cleanly toward $\sim 2\%$ in the $[0.9, 1.0]$ bin on every generator. On StrategyQA the error in that bin stays between 15.0% and 29.7%. The parseable MATH500 subset carries substantial error at intermediate bins, with per-generator $\rho \geq 0.9$ error between 2.5% and 11.7% once the very small Mistral $\times$ MATH500 cell is excluded (Appendix E.5). A StrategyQA answer the model repeats at least 90% of the time is therefore still wrong at those rates. This is the empirical content of "agreement is not truth." Theorem 1 explains it: where high- ρ correct and high- ρ wrong instances coexist, agreement alone cannot separate them.

4.4 CBW is not a sampling artifact

If CBW were a finite-sample fluctuation, more samples should resolve it. Four of five datasets change by less than 1.5 percentage points as K' grows from 4 to 16, recomputing CBW@0.8 from the first $K' \in \{4, 8, 16\}$ samples per question (Appendix D.3): GSM8K 1.7% $\rightarrow$ 2.1%, StrategyQA 19.9% $\rightarrow$ 20.8%, ARC 5.8% $\rightarrow$ 6.2%, CommonsenseQA 8.2% $\rightarrow$ 9.6%. Only MATH500 shows a substantial decrease (from $\sim 7\%$ at $K'=4$ to 3.4% at $K'=16$ under SymPy extraction). Increasing K is not a universal remedy within practical self-consistency budgets.

Generator	GSM8K		StrategyQA	
	CBW	Err	CBW	Err
Cohort mean (5 gen.)	2.1	~ 2	20.8	15.0–29.7
R1-Distill-Qwen-7B	4.0	3.1	20.0	27.1
DeepSeek-V4-Pro	1.7	1.4	11.3	8.5
Claude Sonnet 4.6	1.3	1.4	11.5	11.9

Table 1: CBW@0.8 and Err@ $\rho \geq 0.9$ in percent for three reasoning-trained generators, with the five-generator cohort mean of Section 4 for reference. The task gap is preserved on every row; its magnitude shrinks with capability. Protocol and per-cell counts are in Appendix E.28.

4.5 Reasoning training attenuates CBW without removing it

A natural reading of the cohort result is that CBW is an artifact of instruction-tuned generators which reasoning training will remove. It is not. We repeat the identical $K=16$ protocol on three reasoning-trained generators: DeepSeek-R1-Distill-Qwen-7B (Guo et al., 2025), an RL-distilled reasoner at the smallest scale in our cohort; DeepSeek-V4-Pro in thinking mode (DeepSeek-AI, 2026); and Claude Sonnet 4.6 with extended thinking (Anthropic, 2026), from a second frontier vendor. We use GSM8K and StrategyQA, the two datasets that bracket the cohort’s CBW range. The DeepSeek models run $n=300$ per dataset; Claude Sonnet 4.6 runs $n=150$ on GSM8K and $n=200$ on StrategyQA.

Capability moves the level and leaves the structure intact (Table 1). Both frontier models are accurate, reaching 96.7–98.0% majority-vote accuracy on GSM8K and 83.3–85.0% on StrategyQA. On StrategyQA both are still wrong 8.5% and 11.9% of the time at $\rho \geq 0.9$, against 1.4% each

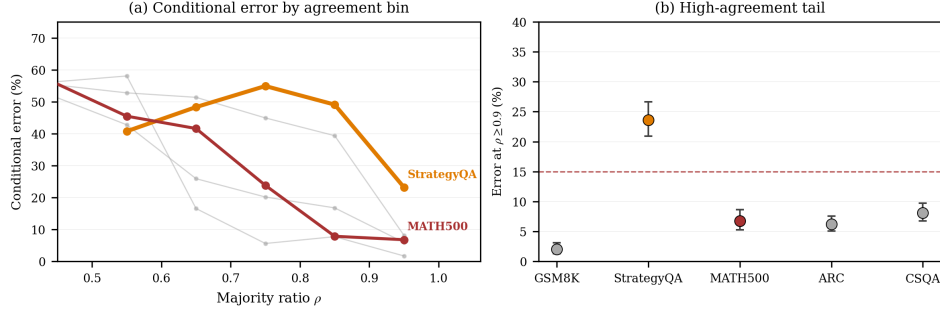


Figure 2: (a) Conditional error $\Pr[\hat{y} \neq y \mid \rho \in \text{bin}]$ averaged across generators, one line per dataset; StrategyQA’s curve begins at the $[0.5, 0.6)$ bin because its yes/no answer space makes $\rho < 0.5$ impossible. (b) Conditional error at $\rho \geq 0.9$ pooled across generators with Wilson 95% CI; the dashed line marks 15%. Per-generator detail in Appendix D.2.

on GSM8K. The task gap thus reproduces on a small distilled reasoner and on two frontier reasoners from different vendors, which also rules out a single-vendor artifact. The StrategyQA rate attenuates, from a cohort mean of 20.8% CBW@0.8 to $\sim 11.4\%$ at the frontier, and stays well above the low-CBW floor that GSM8K sets. Stronger reasoning mitigates CBW. It does not retire the need to verify what is left.

4.6 Robustness to the threshold choice

Sweeping $\tau \in \{0.5, 0.6, \dots, 1.0\}$ preserves the dataset ordering across the entire range, and StrategyQA retains 15.7% unconditional CBW even at $\tau = 0.9$ (Appendix D.3). The ordering also holds at generator temperatures $T \in \{0.3, 0.7, 1.0\}$, so neither τ nor decoding drives the ordering (Appendix E.16).

A manual audit of 143 wrong StrategyQA answers at $\rho \geq 0.75$ on three generators (Qwen2.5-14B, Llama-3.1-8B, GPT-5.4-mini) shows recurring failure modes: multi-step retrieval gaps, implicit-reasoning shortcuts, and lexical or negation priors. Appendix E.6 gives the five-class rubric breakdown. An LLM second-rater agrees poorly on that fine-grained taxonomy (Appendix E.7), so we read the category percentages as heuristic diagnostics.

How much of the StrategyQA rate belongs to the benchmark itself is uncertain; our two audits bracket it. Our manual audit attributes 4.9% of the audited errors to questionable gold labels while the LLM second-rater attributes 66.4%, at a rater agreement of $\kappa=0.076$ (Appendix E.7). We therefore bracket the gold-artifact share at $[4.9\%, 66.4\%]$. The task-structured finding does not turn on where in that bracket the truth lies. Discounting Strat-

egyQA CBW@0.8 by the full 66.4% still leaves $\approx 7.0\%$, above GSM8K (2.1%), MATH500 (3.4%), and ARC (6.2%). Dropping StrategyQA from the comparison entirely still leaves the other four datasets spread from 2.1% to 9.6%. StrategyQA therefore illustrates the high-CBW end of the range, and the finding survives without it.

Two further checks argue against noise. 40 of the 95 unique audited questions fail on at least two of the three generators, which points to shared difficulty (Appendix E.12). All four pairwise gaps between StrategyQA and the other datasets at $\rho \geq 0.9$ survive Bonferroni correction at $\alpha = 0.05/4$ (Appendix E.12). Together these results refine what agreement is worth. On a low-CBW task such as GSM8K, high agreement is a useful empirical signal. On StrategyQA that signal is structurally limited. A deployment needs not more agreement but a calibrated relation between agreement and correctness, which the next section builds.

5 Risk-Controlled Self-Consistency

A deployment cannot set one agreement threshold and call it a guarantee if agreement is task-dependent. It needs a signal from outside the answer histogram and a calibration step that turns that signal into a risk statement. RISC is that layer: a verifier-informed rule that accepts or refuses the majority answer (Figure 3). Throughout, *control* means a held-out calibration guarantee at a target the practitioner sets.

5.1 From Agreement Signals to Acceptance Scores

Each question receives a score that decides whether its SC majority answer is accepted (Figure 3). We compare SC-only $s = \rho$, single-verifier scores

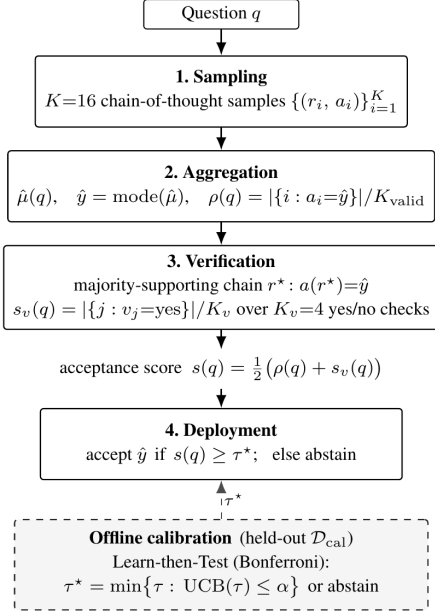


Figure 3: **RISC pipeline.** The generator produces $K=16$ chains, from which we take the majority answer $\hat{y}$ and the majority ratio $\rho(q)$. A verifier reads one majority-supporting chain r^* and returns the yes-rate $s_v(q)$ over $K_v=4$ checks. RISC scores by $s(q) = \frac{1}{2}(\rho(q) + s_v(q))$, LTT picks τ^* whose upper confidence bound satisfies $\text{UCB}(\tau^*) \leq \alpha$ on held-out calibration data, and deployment accepts $\hat{y}$ iff $s(q) \geq \tau^*$.

Main properties

Selective-risk control at a target α with finite-sample validity, under exchangeability.
 An admissibility check that abstains on all inputs when no threshold certifies α .
 A smaller pooled risk-coverage area than agreement alone on our cohort.

Scope

The accepted answer is the generator’s majority answer, passed through unchanged.
 Accuracy is bounded by the candidate pool: the K chains have to contain the correct answer.
 Validity holds in-domain; task shift calls for per-task recalibration.
 The size of the error reduction varies with task and coverage (Table 3).

Table 2: What the certificate covers and the conditions it holds under.

s_v from Phi-3.5-mini (Abdin et al., 2024), GPT-4o-mini (OpenAI, 2024a), and GPT-5.4-mini, and RISC ensembles that average ρ with a verifier yes-rate over $K_v = 4$ queries. The verifier reads one majority-supporting chain r^* , the first sample whose extracted answer matches $\hat{y}(q)$ (Appendix C.3). It reads that chain in full. Stripping the rationale worsens the area under the risk-

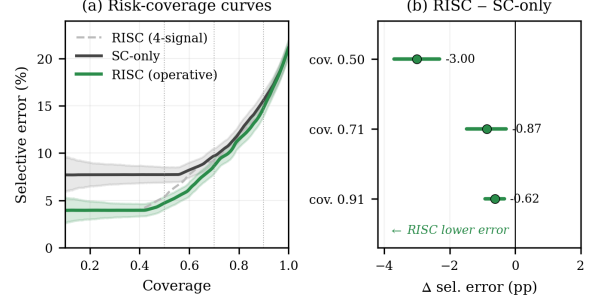


Figure 4: Risk-coverage curves with 95% paired-bootstrap CI ($n_{\text{boot}}=2000$, $N=8,222$). The gain is clearest at conservative coverage and shrinks toward high coverage, but excludes zero at all three levels under both bootstrap schemes (Appendix E.23), and it depends on the task (Table 3).

coverage curve (AURC, lower is better) $9.6 \rightarrow 14.4\%$, and a misleading rationale collapses the yes-rate $76.0 \rightarrow 1.1\%$ (Appendix E.14, E.20). Critique-style prompts improve AURC but weaken the LTT certificate (Appendix E.15). We keep rationale-included yes/no scoring, whose rankings are stable across verifier temperature (Appendix E.19).

5.2 Calibration via Learn-then-Test

Given score s and target α , we seek a threshold τ^* such that “accept iff $s(q) \geq \tau^*$ ” has selective error $\leq \alpha$ with probability $\geq 1 - \delta$ over the calibration draw under exchangeability between calibration and deployment. We obtain τ^* by Learn-then-Test (Angelopoulos et al., 2025; Bates et al., 2021) on a held-out $\mathcal{D}_{\text{cal}}$, in the manner of conformal selective prediction (Romano et al., 2019; Lei et al., 2018; Kamath et al., 2020; Park et al., 2022). Each candidate τ in the grid $T = \{s(q) : q \in \mathcal{D}_{\text{cal}}\}$ is one hypothesis test. We compute the Clopper-Pearson upper confidence bound (UCB) on calibration selective error at confidence $1 - \delta/|T|$ under Bonferroni correction, and return the smallest τ whose bound clears the target, $\tau^* = \min\{\tau : \text{UCB}(\tau) \leq \alpha\}$. When no τ clears α , the procedure returns no admissible threshold and abstains on everything. We use 50/50 splits, $R=10$ seeds, and $\delta=0.10$. RCPS on the same scores also controls finite-sample selective error, and LTT is tighter at $R=10$ (Appendix E.8, E.21). Exchangeability holds in-domain, and it is exactly what the task shift of Section 6 breaks.

5.3 Pooled Risk-Coverage Results

Figure 4 shows pooled risk-coverage curves with 95% paired-bootstrap CIs (2,000 resamples,

$N=8,222$). Coverage here is the fraction of questions accepted, a different quantity from the agreement threshold ρ of Section 4. Against SC-only, RISC’s error delta is -3.01pp $[-3.75, -2.29]$ at 0.50 coverage, -0.87pp $[-1.46, -0.30]$ at 0.71, and -0.63pp $[-0.91, -0.32]$ at 0.91. All three exclude zero under both the paired and the question-level cluster bootstrap (Appendix E.23). The improvement is task-dependent (Table 3); Table 2 states what the certificate covers. The low-coverage end matters most, because an accepted wrong answer costs more there than a refused right one. RISC also refuses the right questions: 49.8% of abstentions fall on answers that would have been wrong, a $2.36\times$ enrichment over the base error rate (Appendix E.24).

5.4 Where the Improvement Comes From

The gain belongs to the combination. On the score-combination subcohort ($N=6,154$; Appendix E.2), ranking by agreement alone gives AURC 10.17% $[9.18, 11.15]$ and ranking by the verifier score alone gives 10.65% $[9.66, 11.66]$, while their unweighted mean gives 7.76% $[6.95, 8.54]$. The verifier alone has the weaker point estimate, and averaging it with agreement still produces the best score. Learned combiners over the same two features do not beat that average, with logistic at 7.94% and an isotonic generalized additive model at 7.91%, both inside the average’s interval. We keep the unweighted mean as the simplest option that captures the gain.

Weighting the vote by the generator’s own confidence does not buy the same gain. We regenerate open-weight self-consistency with per-path token confidence on three generators and two datasets ($N=1,799$), then score a confidence-weighted vote of the form used by CISC (Taubenfeld et al., 2025). Weighting changes the winning answer on 11 of 1,799 questions and moves selective risk by 0.07pp (AURC 11.75 $\rightarrow$ 11.68). Ties dominate the confidence-weighted ranking, since 65.2% of instances sit at $\rho=1.0$, so we also randomize the tie order. Over 500 permutations the difference from SC-only is 0.03pp $[-1.56, +1.68]$, indistinguishable from zero (Appendix E.10). The external verifier moves selective risk by over two points. Confidence is information the generator already had. The verifier supplies information the generator did not, which is what Theorem 1 says is needed.

A single operating point hides how much the effect varies by task. Moving from agreement alone to RISC changes AURC by -0.57 on GSM8K,

-2.65 on MATH500, -0.57 on ARC, -1.82 on StrategyQA, and $+0.33$ on CommonsenseQA. Integrated over coverage, RISC improves four of five tasks, CommonsenseQA excepted; StrategyQA flips sign at the coverage-0.7 point of Table 3, so its two curves cross (Appendix D.6).

Which verifier supplies that information decides whether the combination helps at all. On the full cohort ($N=8,222$), agreement alone gives AURC 9.41% $[8.63, 10.19]$. Adding GPT-5.4-mini gives 7.16% $[6.51, 7.78]$, an interval strictly below agreement alone. Adding GPT-4o-mini gives 9.05% $[8.35, 9.79]$, an interval that is not (Appendix E.17, E.18). A verifier that does not clear it is screened out by its held-out certificate, and Section 5.5 reports what that costs in coverage.

5.5 Calibration Validity and Verifier Choice

In-domain LTT meets its target on splits that Bonferroni admits. At $\alpha=0.10$, SC-only records 4.27% selective error at 87.2% coverage and RISC 3.55% at 78.2% coverage, both with 0.0% violations, inside the $\delta=0.10$ budget. The target $\alpha=0.20$ is also met (App. Fig. 9), whereas ranking on s_v alone violates it at 2.8% (Appendix B.3). Admissibility is the binding constraint at strict targets. Bonferroni admits 23% of splits at $\alpha=0.10$ and none at $\alpha=0.05$, and Holm admits 36% at $\alpha=0.10$ (Appendix E.4). All-split coverage, counting refused splits as zero, is $\approx 14\%$ at per-cell $n_{\text{cal}} \approx 50\text{--}250$ (Appendix B.3). Pooled calibration accepts more than per-cell calibration, at 69.3% coverage and 8.36% selective error, but it is not certified cell by cell (Appendix E.24, E.30; protocol map in Table 7). We take majority ratio + GPT-5.4-mini as the operative variant (Section 5.4). Three cross-vendor verifiers (Haiku 4.5 (Anthropic, 2025), Sonnet 4.6 (Anthropic, 2026), GPT-4o (OpenAI, 2024b)) land within 1.1pp of that variant, Sonnet 4.6 below it in AURC. The open-weight Phi-3.5-mini trails by 1.6pp on its available subcohort, and a three-vendor ensemble reaches 6.22% (Appendix E.17, E.18). Verifier quality binds at strict α and per-cell calibration size decides admissibility; the score-combination and correction choices barely matter (Appendix E.2–E.4).

6 Boundaries, Transfer, and Cost

A calibration fitted on one task should not transfer to another if CBW is structured by task (Section 4). We test four regimes at $\alpha=0.10$: in-domain (50/50

split), cross-generator, leave-one-task-out (LOTO), and cross-task over 20 ordered pairs. Changing the generator breaks calibration far less than changing the task. Violations reach 15–20% across generators, 31–37% for LOTO, and 62.5% for SC-only cross-task (Fig. 8). The verifier does not restore validity: under direct cross-task shift RISC violates on 37.5% of admitted splits against SC-only’s 62.5%, and under LOTO on 36.8% against its 30.8%. A better verifier is therefore not the fix; the broken assumption is exchangeability. The same split appears at coverage 0.7 (Table 3, curves in Appendix D.6).

Dataset	SC-only err	RISC err	Δ
GSM8K	4.0%	2.1%	−1.9pp
StrategyQA	23.9%	25.0%	+1.1pp
MATH500	21.6%	19.3%	−2.2pp
ARC	5.9%	4.7%	−1.3pp
CSQA	9.7%	12.0%	+2.3pp

Table 3: Per-task selective error at coverage 0.7, averaged over five generators (mean across cells). Δ : RISC–SC-only. Full risk-coverage curves in Appendix D.6.

6.1 What Recalibration Costs

Per-task recalibration is the remedy for task shift. It is inexpensive in labels. Calibrating on a held-out half of the target task admits a threshold on 75 of 100 replicates and violates α on none of the 75, at mean coverage 72.8%. Pooled calibration on the other four tasks admits 76 of 100 and violates on 34.2% of those, at mean coverage 57.9% (Appendix E.13). Calibrating on the target task therefore recovers both validity and coverage, while calibration data from other tasks recovers neither.

More calibration labels help against only one of the two blockers: a calibration budget too small for the confidence bound, not a selective-error floor sitting above α . A sweep at the strict target $\alpha=0.05$ separates the two (Appendix E.30). On GSM8K cells, raising n_{cal} from 150 to 250 moves admission from 0 of 10 splits to as many as 3 of 10. The obstruction there was partly the budget, which more labels start to remove. On StrategyQA cells, no n_{cal} we test admits any split, because the conditional error at high agreement exceeds 0.05 and Corollary 1 places the achievable selective error above α . Set α above the task’s measured high-agreement error. A persistent refusal to certify is evidence about the task itself, and otherwise RISC abstains on everything.

6.2 Cost-Risk Tradeoff

Qwen2.5-7B with RISC costs $\sim \$1.2$ per 1,000 questions against $\sim \$10.5$ for GPT-5.4-mini majority voting, at the 0.9 selective operating point (not the $\alpha=0.10$ LTT threshold). On four of five datasets the per-task paired-bootstrap CI for the error difference straddles zero, so no gap is detectable at this sample size. MATH500 is the exception and keeps a +7.94pp [+2.26, +12.69] gap in favor of GPT-5.4-mini (Appendix D.8). The verifier bill is dominated by its input, since each of the $K_v=4$ calls carries a full chain and returns one word, while GPT-5.4-mini majority voting pays the output rate on $K=16$ generated chains. The MATH500 gap points to a limit of base-generator capacity. These figures exclude local-GPU amortization, energy, and the abstention fallback (Appendix D.8).

7 Conclusion

Self-consistency is an agreement mechanism, not a certificate of correctness. The agreement-correctness gap is structured by task, robust to the agreement threshold, and attenuated by reasoning training without being closed, so closing it takes information the answer histogram does not carry. RISC supplies that information under a finite-sample certificate, abstaining when no threshold certifies the target. At 0.9 coverage it shows no statistically detectable selective-error gap against GPT-5.4-mini majority voting on four of five datasets, at $\sim 9\times$ lower API cost. Reasoning systems should report whether their agreement is calibrated to correctness.

Limitations

Cohort. Two Qwen models, one Llama, one Mistral and GPT-5.4-mini, so task-versus-generator claims are within-cohort. We check generality on BBH, a newer Qwen3-8B generator and the Mistral member (Appendix E.29, E.27, E.26). Broader families remain untested.

Datasets and parsing. About 4% of MATH500 items stay unparseable under our SymPy extraction, mostly matrix and multi-part golds, and we exclude them. The filter never reads model predictions, only parseability, so MATH500 results hold for parseable items. Per-cell sizes run from 100 to 500 (Appendix B), so GPT-5.4-mini intervals are wider and small cells such as Mistral $\times$ MATH500 are read descriptively.

Benchmark-gold correctness. “Wrong” means disagreement with the gold label, not adjudicated error. Some high-agreement StrategyQA disagreements may reflect debatable golds, and an LLM second-rater blamed annotation artifact far more often than our human coder ($\kappa=0.076$) (Appendix E.6, E.7). StrategyQA CBW@ τ therefore measures high-confidence disagreement with gold.

Risk-control regime. Learn-then-Test with Bonferroni is conservative: at $n_{\text{cal}} \approx 150$ and $\alpha = 0.05$ no non-trivial threshold is admitted, and $n_{\text{cal}}=250$ admits one only where the floor of Cor. 1 does not bind (Appendix E.30). Cross-task calibration breaks outright (Section 6): no single calibration covers every task.

Verifier dependence and capability ceiling. A weak verifier costs coverage at strict targets. Scoring is rationale-conditioned, so RISC controls risk as measured on the visible chain, which hidden reasoning never enters. Answer-only ablations degrade AURC by $\sim 3.6\text{pp}$ and lose admissibility (Appendix E.14, E.20). RISC acts only on acceptance: where base accuracy is low, as on MATH500 and StrategyQA, selection cannot produce an answer the generator never proposed. Decoding is fixed at one setting per generator and one per verifier (Appendix E.16, E.31). Verifier selection and training and a broader decoding sweep are future work, and combining RISC with budget-adaptive generation could shift the cost-risk envelope further.

Ethics Statement

Deployment risks. The principal risk is applying RISC outside the regime where its guarantee holds. The bound assumes exchangeability between calibration and deployment. Under generator shift violation rates reach 15 to 20%, and under task shift 31 to 63% (Section 6); per-task recalibration restores validity (Appendix E.13). Calibrated abstention is not a safety certificate: where base accuracy is low RISC lowers selective error only by abstaining more, and cannot lift correct-answer coverage above the generator’s ceiling. An accepted answer carries the in-domain per-task bound at the chosen target and nothing more, in education, medicine, law and scientific question answering.

Use of pre-existing artifacts. The five benchmarks are public, released for research evaluation, and our use matches that intent. We redistribute none of them, only our analysis code and per-question scores derived under the benchmarks’

original licenses (Appendix B). Open-weight models (Tables 4, 5) are used under their authors’ licenses and the proprietary models under their providers’ Terms of Service. The items are English mathematics problems, multi-step yes/no strategy questions, and science and commonsense multiple choice. None carry personally identifying information about non-public individuals or content usually considered offensive in NLP research. We inherit the published curation unchanged apart from the prediction-independent MATH500 parseability filter of Section 4.1, and score correctness against gold labels.

Human subjects. No external annotation, human-subjects study or IRB review was involved; evaluation is automatic against gold answers. The one manual step is an author inspection of 143 high-agreement-but-wrong StrategyQA items under a five-class rubric, with no statistical claim (Appendix E.6, E.7); those labels stay with the authors.

Code and data release. Code, derived artifacts and analysis scripts are released under a permissive open-source license at <https://github.com/mjdddol/agreement-is-not-truth>: item IDs aligned to each benchmark’s gold, extracted answers and majority ratios ρ at $K=16$, verifier yes-rates s_v at $K_v=4$ for every (verifier, question) pair, the chain-selection rule r^* and prompt schema (Appendix C), and the LTT, bootstrap and figure code. We withhold only the raw chain-of-thought text, whose licensing is ambiguous, so exact verifier re-execution needs the chains regenerated. Every automatically computed number here reproduces from the artifact; the rubric audit does not.

Acknowledgments

This research was supported by the Basic Research Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Education, Science and Technology (2710108964); in part by a grant from the Institute for AI and Social Innovation at Yonsei University; in part by the Yonsei University Office of Research Affairs through the Yonsei T.R.U.S.T. (Yonsei Transdisciplinary Research Union for a Sustainable Tomorrow) program (Project Y; project no. 2025-22-0461); and in part by the Institute of Convergence Science (ICONS), Yonsei University.

Use of AI assistants. AI assistants were used for LaTeX formatting and grammar polishing. Ex-

perimental design, theoretical results, implementation, and scientific claims are the authors’ own. The use of LLMs as *verifiers* inside the experimental pipeline (Section 5, Appendix B) is a separate experimental choice, unrelated to manuscript writing.

References

- Marah Abdin, Jyoti Aneja, Hany Awadalla, et al. 2024. [Phi-3 technical report: A highly capable language model locally on your phone](#). *arXiv preprint arXiv:2404.14219*.
- Pranjal Aggarwal, Aman Madaan, Yiming Yang, and Mausam. 2023. [Let’s sample step by step: Adaptive-consistency for efficient reasoning and coding with LLMs](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12375–12396, Singapore. Association for Computational Linguistics.
- Anastasios N. Angelopoulos, Stephen Bates, Emmanuel J. Candès, Michael I. Jordan, and Lihua Lei. 2025. [Learn then test: Calibrating predictive algorithms to achieve risk control](#). *The Annals of Applied Statistics*, 19(2):1641–1662.
- Anthropic. 2025. Introducing Claude Haiku 4.5. <https://www.anthropic.com/news/claude-haiku-4-5>. Accessed: 2026-04-27.
- Anthropic. 2026. Introducing Claude Sonnet 4.6. <https://www.anthropic.com/news/claude-sonnet-4-6>. Accessed: 2026-05-10.
- Stephen Bates, Anastasios N. Angelopoulos, Lihua Lei, Jitendra Malik, and Michael I. Jordan. 2021. [Distribution-free, risk-controlling prediction sets](#). *Journal of the ACM*, 68(6):43:1–43:34.
- Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V. Le, Christopher Ré, and Azalia Mirhoseini. 2024. [Large language monkeys: Scaling inference compute with repeated sampling](#). *arXiv preprint arXiv:2407.21787*.
- Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. [Think you have solved question answering? try ARC, the AI2 reasoning challenge](#). *arXiv preprint arXiv:1803.05457*.
- 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*.
- DeepSeek-AI. 2026. DeepSeek API documentation. <https://api-docs.deepseek.com/>. Accessed: 2026-07-09.
- Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. 2024. [Improving factuality and reasoning in language models through multiagent debate](#). In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, Proceedings of Machine Learning Research. PMLR.
- Ran El-Yaniv and Yair Wiener. 2010. [On the foundations of noise-free selective classification](#). *Journal of Machine Learning Research*, 11:1605–1641.
- Yonatan Geifman and Ran El-Yaniv. 2017. [Selective classification for deep neural networks](#). In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. 2020. [Shortcut learning in deep neural networks](#). *Nature Machine Intelligence*, 2(11):665–673.
- Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. 2021. [Did Aristotle use a laptop? a question answering benchmark with implicit reasoning strategies](#). *Transactions of the Association for Computational Linguistics (TACL)*, 9:346–361.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, et al. 2024. [The Llama 3 herd of models](#). *arXiv preprint arXiv:2407.21783*.
- Daya Guo, Dejian Yang, Haowei Zhang, et al. 2025. [DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning](#). *Nature*.
- Shibo Hao, Yi Gu, Haodi Ma, Joshua Hong, Zhen Wang, Daisy Wang, and Zhiting Hu. 2023. [Reasoning with language model is planning with world model](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8154–8173, Singapore. Association for Computational Linguistics.
- 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 Track on Datasets and Benchmarks (NeurIPS Datasets and Benchmarks)*.
- Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Léo Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. [Mistral 7B](#). *arXiv preprint arXiv:2310.06825*.
- Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. 2021. [How can we know when language models know? on the calibration of language models for question answering](#). *Transactions of the Association for Computational Linguistics (TACL)*, 9:962–977.

- Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, and 17 others. 2022. [Language models \(mostly\) know what they know](#). *arXiv preprint arXiv:2207.05221*.
- Amita Kamath, Robin Jia, and Percy Liang. 2020. [Selective question answering under domain shift](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5684–5696, Online. Association for Computational Linguistics.
- Muhammad Khalifa, Lajanugen Logeswaran, Moontae Lee, Honglak Lee, and Lu Wang. 2023. [GRACE: Discriminator-guided chain-of-thought reasoning](#). In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15299–15328, Singapore. Association for Computational Linguistics.
- Akbir Khan, John Hughes, Dan Valentine, Laura Ruis, Kshitij Sachan, Ansh Radhakrishnan, Edward Grefenstette, Samuel R. Bowman, Tim Rocktäschel, and Ethan Perez. 2024. [Debating with more persuasive LLMs leads to more truthful answers](#). In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, Proceedings of Machine Learning Research. PMLR.
- Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. [Large language models are zero-shot reasoners](#). In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. [Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation](#). In *International Conference on Learning Representations (ICLR)*.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. [Efficient memory management for large language model serving with PagedAttention](#). In *Proceedings of the 29th Symposium on Operating Systems Principles (SOSP)*. ACM.
- Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J. Tibshirani, and Larry Wasserman. 2018. [Distribution-free predictive inference for regression](#). *Journal of the American Statistical Association*, 113(523):1094–1111.
- Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. [Let's verify step by step](#). In *International Conference on Learning Representations (ICLR)*.
- Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. [Teaching models to express their uncertainty in words](#). *Transactions on Machine Learning Research (TMLR)*.
- Jianqiao Lu, Zhiyang Dou, Hongru Wang, Zeyu Cao, Jianbo Dai, Yingjia Wan, Yunlong Feng, and Zhi-jiang Guo. 2024. [AutoPSV: Automated process-supervised verifier](#). In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegrefe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. [Self-refine: Iterative refinement with self-feedback](#). In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. [SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9004–9017, Singapore. Association for Computational Linguistics.
- Mistral AI. 2024. Mistral-7B-Instruct-v0.3 model card. <https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3>. Accessed: 2026-05-10.
- OpenAI. 2024a. GPT-4o mini: Advancing cost-efficient intelligence. <https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/>. Accessed: 2026-04-27.
- OpenAI. 2024b. Hello GPT-4o. <https://openai.com/index/hello-gpt-4o/>. Accessed: 2026-04-27.
- OpenAI. 2026. Introducing GPT-5.4 mini and nano. <https://openai.com/index/introducing-gpt-5-4-mini-and-nano/>. Accessed: 2026-05-10.
- Sangdon Park, Edgar Dobriban, Insup Lee, and Osbert Bastani. 2022. [PAC prediction sets under covariate shift](#). In *International Conference on Learning Representations (ICLR)*.
- Qwen Team. 2024. [Qwen2.5 technical report](#). *arXiv preprint arXiv:2412.15115*.
- Yaniv Romano, Evan Patterson, and Emmanuel J. Candès. 2019. [Conformalized quantile regression](#). In *Advances in Neural Information Processing Systems (NeurIPS)*.
- William Saunders, Catherine Yeh, Jeff Wu, Steven Bills, Long Ouyang, Jonathan Ward, and Jan Leike. 2022. [Self-critiquing models for assisting human evaluators](#). *arXiv preprint arXiv:2206.05802*.
- Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. [Reflexion: Language agents with verbal reinforcement learning](#). In *Advances in Neural Information Processing Systems (NeurIPS)*.

- Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. 2025. [Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning](#). In *International Conference on Learning Representations (ICLR)*.
- Kaya Stechly, Karthik Valmeekam, and Subbarao Kambhampati. 2025. [On the self-verification limitations of large language models on reasoning and planning tasks](#). In *International Conference on Learning Representations (ICLR)*.
- Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, and Jason Wei. 2023. [Challenging BIG-bench tasks and whether chain-of-thought can solve them](#). In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13003–13051, Toronto, Canada. Association for Computational Linguistics.
- Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. [CommonsenseQA: A question answering challenge targeting commonsense knowledge](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.
- Hexiang Tan, Fei Sun, Sha Liu, Du Su, Qi Cao, Xin Chen, Jingang Wang, Xunliang Cai, Yuanzhuo Wang, Huawei Shen, and Xueqi Cheng. 2025. [Too consistent to detect: A study of self-consistent errors in LLMs](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 4755–4765, Suzhou, China. Association for Computational Linguistics.
- Amir Taubenfeld, Tom Sheffer, Eran Ofek, Amir Feder, Ariel Goldstein, Zorik Gekhman, and Gal Yona. 2025. [Confidence improves self-consistency in LLMs](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 20090–20111, Vienna, Austria. Association for Computational Linguistics.
- Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher Manning. 2023. [Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5433–5442, Singapore. Association for Computational Linguistics.
- Jonathan Uesato, Nate Kushman, Ramana Kumar, H. Francis Song, Noah Y. Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. 2022. [Solving math word problems with process- and outcome-based feedback](#). *arXiv preprint arXiv:2211.14275*.
- Vladimir Vovk, Alex Gammerman, and Glenn Shafer. 2005. *Algorithmic Learning in a Random World*. Springer.
- Guangya Wan, Yuqi Wu, Jie Chen, and Sheng Li. 2025. [Reasoning aware self-consistency: Leveraging reasoning paths for efficient LLM sampling](#). In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3613–3635, Albuquerque, New Mexico. Association for Computational Linguistics.
- Ziyu Wan, Xidong Feng, Muning Wen, Stephen Marcus McAleer, Ying Wen, Weinan Zhang, and Jun Wang. 2024. [AlphaZero-like tree-search can guide large language model decoding and training](#). In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, Proceedings of Machine Learning Research. PMLR.
- Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. 2024a. [Math-shepherd: Verify and reinforce LLMs step-by-step without human annotations](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9426–9439, Bangkok, Thailand. Association for Computational Linguistics.
- Weiqin Wang, Yile Wang, and Hui Huang. 2025. [Ranked voting based self-consistency of large language models](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 14410–14426, Vienna, Austria. Association for Computational Linguistics.
- Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. [Self-consistency improves chain of thought reasoning in language models](#). In *International Conference on Learning Representations (ICLR)*.
- Zihan Wang, Yunxuan Li, Yuexin Wu, Liangchen Luo, Le Hou, Hongkun Yu, and Jingbo Shang. 2024b. [Multi-step problem solving through a verifier: An empirical analysis on model-induced process supervision](#). In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7309–7319, Miami, Florida, USA. Association for Computational Linguistics.
- 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)*.
- Sean Welleck, Ximing Lu, Peter West, Faeze Brahman, Tianxiao Shen, Daniel Khashabi, and Yejin Choi. 2023. [Generating sequences by learning to self-correct](#). In *International Conference on Learning Representations (ICLR)*.

Yuxi Xie, Kenji Kawaguchi, Yiran Zhao, James Xu Zhao, Min-Yen Kan, Junxian He, and Michael Qizhe Xie. 2023. [Self-evaluation guided beam search for reasoning](#). In *Advances in Neural Information Processing Systems (NeurIPS)*.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2024. [Can LLMs express their uncertainty? an empirical evaluation of confidence elicitation in LLMs](#). In *International Conference on Learning Representations (ICLR)*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. [Qwen3 technical report](#). *arXiv preprint arXiv:2505.09388*.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. [Tree of thoughts: Deliberate problem solving with large language models](#). In *Advances in Neural Information Processing Systems (NeurIPS)*.

A Appendix: Proof of Theorem 1

We prove the theorem via a *task-family construction*. The construction makes no claim about which natural questions exhibit the indistinguishable structure. It shows only that any procedure depending solely on the answer distribution must decide the same way on a constructible pair of instances with different ground-truth labels. Such a procedure therefore cannot uniformly certify correctness over a task family that contains both.

Theorem 1 (restated). *Let $\mathcal{C} : \Delta(\mathcal{A}) \rightarrow \{\text{accept, abstain}\}$ be any function depending only on its argument. There exist a distribution $\mu \in \Delta(\mathcal{A})$ with a unique mode $a^* = \text{mode}(\mu) \in \mathcal{A}$ and two labeled questions $z_1 = (q_1, y_1 = a^*)$ and $z_2 = (q_2, y_2)$ with $y_2 \neq a^*$ and $\mu_g(q_1) = \mu_g(q_2) = \mu$, such that $\mathcal{C}$ produces the same output on z_1 and z_2 .*

Proof. Fix any $\mu \in \Delta(\mathcal{A})$ with a unique mode and let $a^* = \text{mode}(\mu)$. Consider two labeled questions sharing the same generator-induced answer distribution but differing in their ground-truth labels:

$$z_1 = (q_1, y_1 = a^*), \quad z_2 = (q_2, y_2 \in \mathcal{A} \setminus \{a^*\}),$$

$$\text{with } \mu_g(q_1) = \mu_g(q_2) = \mu.$$

Such a family is well defined: a labeled task instance is specified by a question, the generator-induced answer distribution under g , and a ground-

truth label. The mapping from questions to answer distributions is determined by g , and the mapping from questions to ground-truth labels is determined by the task definition. The two mappings are not constrained to coincide. Concretely, z_1 and z_2 can be drawn from any task family in which two distinct labeled questions induce the same answer distribution under g but differ in their ground-truth labels. The high-agreement subset of StrategyQA motivates that regime empirically (Section 4), where the quantitative analogue is a finite-sample, ρ -thresholded approximation of exact equal- μ pairs.

By assumption, $\mathcal{C}$ depends only on its argument. Since $\mu_g(q_1) = \mu_g(q_2) = \mu$, we have

$$\mathcal{C}(\mu_g(q_1)) = \mathcal{C}(\mu) = \mathcal{C}(\mu_g(q_2)).$$

Thus $\mathcal{C}$ produces the same accept/abstain output on z_1 and z_2 . If $\mathcal{C}$ accepts, it commits to a^* on both instances: correctly on z_1 (since $y_1 = a^*$) and incorrectly on z_2 (since $y_2 \neq a^*$). If $\mathcal{C}$ abstains, it forgoes coverage on z_1 even though the modal answer would have been correct. $\square$

Remark (randomized rules). The argument extends verbatim to randomized SC-only rules $\mathcal{C} : \Delta(\mathcal{A}) \rightarrow \mathcal{P}(\{\text{accept, abstain}\})$ whose accept/abstain distribution depends only on μ : the induced acceptance distribution is identical at z_1 and z_2 , so conditioning on $\{\text{accept}\}$ preserves the $z_1:z_2$ proportions used in Corollary 1. We state Theorem 1 for deterministic $\mathcal{C}$ since the randomized extension adds no power.

Remark (scope of the bound). In the two-type construction the Corollary’s bound is tight ($\beta_\mu = p_2/(p_1+p_2)$ exactly). In more general $\mathcal{T}_\mu$ subclasses with $r > 2$ type proportions $\{p_i\}_{i=1}^r$ where some types are correctly modal at a^* and others are not, the same argument yields $\beta_\mu \geq \sum_{i: y_i \neq a^*} p_i$. The Corollary therefore states a wrong-mode-share *limitation* on SC-only certifiers within an indistinguishable subclass. Its scope is that subclass. The empirical $\text{Err}@_\tau$ rates of Section 4 are its finite-sample, ρ -thresholded analogue, and Appendix E.12 reports the StrategyQA $\rho \geq 0.9$ instance.

Corollary 1 (restated). *Let $\mathcal{T}_\mu$ be the indistinguishable subfamily of instances sharing answer distribution μ , with z_1 - and z_2 -type instances in proportion $p_1, p_2 \geq 0$, $p_1 + p_2 > 0$. For any SC-only certifier $\mathcal{C}$ that accepts on μ with positive probability, the conditional selective error within*

$\mathcal{T}_\mu$ is bounded below by $\beta_\mu(\mathcal{C}) := \Pr[\hat{y} \neq y \mid \mathcal{C} \text{ accepts}, q \in \mathcal{T}_\mu] \geq p_2/(p_1 + p_2)$.

Proof. Conditional on the event $\{\mathcal{C} = \text{accept}, q \in \mathcal{T}_\mu\}$, the distribution of z_1 - versus z_2 -type instances inherits the subfamily’s prior proportions p_1, p_2 , since $\mathcal{C}$ depends only on μ (which is identical for both types) and not on the type itself. On every accepted z_2 -instance the procedure commits to $a^* \neq y_2$ and is therefore wrong. On every accepted z_1 -instance it is correct. The conditional selective error within $\mathcal{T}_\mu$ is therefore

$$\beta_\mu(\mathcal{C}) = \frac{p_2}{p_1 + p_2}.$$

The only way for $\mathcal{C}$ to lower this conditional error is to abstain on the entire indistinguishable subfamily $\mathcal{T}_\mu$, but $\mathcal{C}$ cannot do so selectively, since it cannot distinguish z_1 from z_2 from μ alone, and abstaining on $\mathcal{T}_\mu$ forgoes coverage on z_1 -type instances at the same rate. $\square$

The theorem and corollary describe an arbitrary $\mathcal{C}$ acting on the indistinguishable subfamily $\mathcal{T}_\mu$. The empirical $\text{Err}@ \tau$ rates of Section 4 are the analogue of the conditional wrong-mode share $p_2/(p_1 + p_2)$, measured on agreement-thresholded subsets of natural reasoning datasets that approximate $\mathcal{T}_\mu$ classes in finite samples. Corollary 1 meets its empirical analogue on the high-agreement subset of StrategyQA in exactly that sense.

B Appendix: Reproducibility Details

B.1 Generators

We use five generators (Table 4). The four open-weight models are served locally via vLLM (Kwon et al., 2023), with Qwen2.5-7B, Llama-3.1-8B, and Mistral-7B in bfloat16 and Qwen2.5-14B under 4-bit bitsandbytes/NF4 quantization so that it fits on a single RTX A5000. The proprietary GPT-5.4-mini is accessed through the OpenAI Chat Completions API under the OpenAI Terms of Service. We use the same chain-of-thought prompt across all generators (Appendix C). Decoding values are fixed a priori from SC-standard practice (Wang et al., 2023) or are fully constrained by the API, and we run no generator-decoding hyperparameter search. Method-design choices (verifier type, rationale inclusion, score combination, K_v) are reported as ablations in Appendix E, with the operative variant specified in Section 5.5. For each (question, generator) cell we draw $K = 16$ chains and extract a final answer by a per-dataset extraction rule (Appendix C.3). Cells with no extracted

majority answer are excluded from the analysis (Section 4.1).

B.2 Verifiers

Six verifiers appear in our experiments (Table 5). Three of them are used in Section 5. The other three appear only in appendix ablations, where they test cross-vendor robustness (Appendix E.17) and rule out a temperature-based explanation (Appendix E.18). All six share one sampling regime: $K_v=4$ yes/no queries per question with a fixed prompt (Appendix C), the queries differing only in non-deterministic sampling. For the API verifiers (GPT-4o, GPT-4o-mini, Claude Haiku 4.5, Claude Sonnet 4.6) we set max_tokens=8, sufficient for the visible yes/no token, and 16 for the locally served Phi-3.5-mini. For the GPT-5.4-mini reasoning verifier, max_completion_tokens caps the sum of hidden reasoning and visible output. We set it to 100 so that a reasoning prefix can precede the yes/no token if the model emits one, which gives a cohort-averaged billed output of ~ 5 tokens per call (Table 6). When we say the verifier “conditions on the chain” or that RISC is “rationale-conditioned,” we mean the *visible rationale text emitted in the model response*. For reasoning-model API calls such as GPT-5.4-mini, hidden reasoning tokens may be billed, but they are never observed, stored, or passed to the verifier.

B.3 Calibration Procedure

For each (generator, dataset) cell we form an acceptance score per question (Section 5.1) and apply Learn-then-Test (Section 5.2) on a held-out 50/50 split with $R = 10$ random seeds. The threshold grid is the set of distinct scores observed in the calibration set, and the Bonferroni correction divides the failure probability δ uniformly across grid candidates, giving per-test confidence $1 - \delta/|T|$. We report observed test-set selective error and coverage averaged over the R splits.

B.4 Cost Computation

We compute metered API cost using OpenAI’s per-token rate for GPT-5.4-mini (input \$0.75/M, output \$4.50/M). Per question, generation costs $K \times (\bar{T}_{\text{in}} \cdot r_{\text{in}} + \bar{T}_{\text{out}} \cdot r_{\text{out}})$ at $K=16$ and verification costs $K_v \times (\bar{T}_{\text{in}}^v \cdot r_{\text{in}} + \bar{T}_{\text{out}}^v \cdot r_{\text{out}})$ at $K_v=4$. Output token counts include hidden reasoning tokens, which GPT-5.4-mini bills at the output rate. Token averages are measured from the endpoint’s own usage fields over the released prompts at the

Generator	Access (License)	Decoding
Qwen2.5-7B-Instruct	local vLLM bf16 (Apache 2.0)	$T=0.7, p=0.95, \text{max_tok } 384\text{--}768$
Llama-3.1-8B-Instruct	local vLLM bf16 (Llama 3.1 CL)	$T=0.7, p=0.95, \text{max_tok } 384\text{--}768$
Qwen2.5-14B-Instruct	local vLLM bnb-4bit (Apache 2.0)	$T=0.7, p=0.95, \text{max_tok } 384\text{--}768$
Mistral-7B-Instruct-v0.3	local vLLM bf16 (Apache 2.0)	$T=0.7, p=0.95, \text{max_tok } 384\text{--}768$
GPT-5.4-mini	OpenAI API (proprietary)	$T=1.0^*, \text{max_compl_tok } 768^\ddagger$

Table 4: Generator decoding settings. Symbols: T = sampling temperature; p = top- p (nucleus); max_tok = maximum output tokens (visible only for the open-weight non-reasoning models, where 384–768 visible tokens fit a chain plus extracted answer); max_compl_tok = OpenAI max_completion_tokens. *The OpenAI API forces $T=1.0$ for GPT-5.4-mini and rejects temperature/top_p overrides; this is the only generator setting that differs across the cohort, and we match it as closely as the API allows. ‡ For GPT-5.4-mini (a reasoning-family model), max_completion_tokens caps the *total* of hidden reasoning tokens plus visible output; we set it to 768, matching the upper end of the open-weight budget. The cohort-averaged billed output is ~ 131 tokens/call (Table 6), and the endpoint reports no reasoning tokens for these prompts. Per-generator detail for Mistral-7B-Instruct-v0.3 is reported in Appendix E.26.

Verifier	Access (License)	T	Used in
Phi-3.5-mini-instruct	local vLLM bf16 (MIT)	0.5	Section 5, Appendix E.2, E.30
GPT-4o-mini	OpenAI API	0.5	Section 5, Appendix E.19
GPT-5.4-mini	OpenAI API	1.0*	Section 5
GPT-4o	OpenAI API	0.5	Appendix E.18
Claude Haiku 4.5	Anthropic API	0.5 ‡	Appendix E.17
Claude Sonnet 4.6	Anthropic API	0.5 ‡	Appendix E.17

Table 5: All verifiers run with $K_v=4$ yes/no queries per question. The API verifiers use max_tokens=8 and the locally served Phi-3.5-mini uses 16; the GPT-5.4-mini reasoning verifier uses max_completion_tokens=100 to allow a reasoning prefix (see prose above). *The OpenAI API forces $T=1.0$ for GPT-5.4-mini. ‡ The Anthropic API default is $T=1.0$; we explicitly override it to $T=0.5$ to match the other mini-tier verifiers.

released caps, and absolute values vary by dataset: MATH500 generation is the longest at ~ 294 output tokens per call, CommonsenseQA the shortest at ~ 50 . Because both pipelines bill on the same GPT-5.4-mini rate schedule, the cost ratio is invariant under uniform price drift. A differential change between input and output rates would shift it, since generation is output-bound and verification input-bound.

Pipeline	Calls/Q	Avg in/out tok./call	Cost/1k Q
<i>Qwen2.5-7B+RISC (verifier API only):</i>			
GPT-5.4-mini verify	$K_v=4$	$\sim 363/\sim 5$	\$1.17
<i>GPT-5.4-mini majority voting (full API):</i>			
GPT-5.4-mini gen.	$K=16$	$\sim 85/\sim 131$	\$10.47

Table 6: API cost breakdown. The two pipelines sit at opposite ends of the rate card: generation pays sixteen short prompts and long completions, verification four long prompts and a one-word completion. Cost calculation: verifier $4 \times (363 \cdot 0.75 + 5 \cdot 4.50) / 10^6 \approx \$1.17/1k$ Q; generator $16 \times (85 \cdot 0.75 + 131 \cdot 4.50) / 10^6 \approx \$10.47/1k$ Q; ratio $\approx 8.9 \times$. Qwen2.5-7B+RISC charges only the verifier; the generator runs locally and is excluded.

B.5 Datasets, Languages, and Licenses

All five reasoning benchmarks are public and English-language. We summarize each below with its domain, the published license under which it was released, and the per-cell sample sizes used in our experiments. We do not redistribute any of these datasets. We use the published versions as released by their authors.

- **GSM8K** (Cobbe et al., 2021): grade-school math word problems; English; MIT License; 300 questions per cell on Qwen2.5-7B, Qwen2.5-14B, Llama-3.1-8B, and Mistral-7B-Instruct; 100 on GPT-5.4-mini.
- **StrategyQA** (Geva et al., 2021): multi-step strategy yes/no commonsense questions; English; MIT License; same per-cell sizes as GSM8K (300 on the four open-weight generators including Mistral, 100 on GPT-5.4-mini).
- **MATH500** (Hendrycks et al., 2021): competition mathematics (the standard 500-question subset); English; MIT License; 500 questions per cell on Qwen2.5-7B, Llama-3.1-8B, and Qwen2.5-14B; 300 per cell on Mistral-7B; 100 on GPT-5.4-mini. Answer extraction uses

SymPy-based equivalence (Appendix E.5); $\sim 4\%$ of items remain unparseable under this protocol (matrix and multi-part golds) and are excluded.

- **ARC-Challenge** (Clark et al., 2018): elementary-and-middle-school science multiple choice; English; CC-BY-SA-4.0; 500 per cell on Qwen2.5-7B, Llama-3.1-8B, and Qwen2.5-14B; 300 on Mistral-7B; 100 on GPT-5.4-mini.
- **CommonsenseQA** (Talmor et al., 2019): general commonsense multiple choice; English; MIT License; 500 per cell on Qwen2.5-7B, Llama-3.1-8B, and Qwen2.5-14B; 300 on Mistral-7B; 100 on GPT-5.4-mini.

After applying the SymPy-based MATH500 extraction and excluding question-generator instances with no extracted majority answer, the analysis covers 8,222 valid (question, generator) instances across the five datasets and five generators (Section 4.1). The 50/50 calibration/test split with $R = 10$ random seeds described in Section 5.2 is applied within each (generator, dataset) cell.

B.6 Compute Infrastructure

Local generation and verification (the four open-weight generators Qwen2.5-7B, Llama-3.1-8B, Qwen2.5-14B, Mistral-7B and the Phi-3.5-mini-instruct verifier) ran via vLLM (Kwon et al., 2023) on a single NVIDIA RTX A5000 (24 GB VRAM): Qwen2.5-7B-Instruct, Llama-3.1-8B-Instruct, Mistral-7B-Instruct, and Phi-3.5-mini-instruct in bfloat16, and Qwen2.5-14B-Instruct under 4-bit bitsandbytes weight quantization (NF4) so that the 14B-parameter model fits on this device alongside short-prompt inference traffic. The proprietary API endpoints (GPT-5.4-mini as both generator and verifier; GPT-4o-mini and GPT-4o as appendix-only verifiers; Anthropic Haiku 4.5 and Sonnet 4.6 as appendix-only verifiers) are accessed remotely and use no local compute beyond request scheduling. The reported analysis covers $\sim 10,622$ *scored valid* (question, generator) instances (8,222 from the 5×5 generator-task cohort, 1,200 from the BBH $\times 4$ open-weight subcohort at 300 items per cell, and 1,200 *additional* samples from the n_{cal} -sweep of Appendix E.30 (200 new items per cell on GSM8K and StrategyQA across three open-weight generators, pooled with the 300 items per cell from the 5×5 cohort to

reach the ~ 500 per-cell pool reported there); raw generation attempts are slightly higher because no-extraction and unparseable-gold MATH500 items are filtered after generation. The dominant compute cost is chain-of-thought sampling at $K = 16$ samples per (question, generator) instance over these instances, plus $K_v = 4$ verifier queries per accepted-or-considered question; end-to-end the experiment runs on a single A5000 for the local-model components, with metered API usage reported in Appendix D.8.

B.7 Software Stack

The pipeline is implemented in Python 3.10 and uses vLLM (Kwon et al., 2023) for open-weight serving, the official openai Python client for API access, and standard scientific Python libraries for analysis: numpy and scipy (scipy.stats for Wilson confidence intervals and paired-bootstrap resampling), statsmodels (Clopper-Pearson upper bound for the LTT step), pandas (per-cell tabulation), scikit-learn (the held-out logistic-regression combiner reported in Appendix E.2), and matplotlib (figure rendering). All of these libraries are released under permissive open-source licenses (BSD-3-Clause for numpy/scipy/scikit-learn/pandas/statsmodels; the Matplotlib license, BSD-style; Apache 2.0 for vLLM); pinned versions for reproducing our environment are listed in the environment.yml included with the released code (Ethics Statement).

C Appendix: Prompts and Extraction Rules

C.1 Generation Prompt

We use a single chain-of-thought prompt template across all generators and datasets:

```
Question: {question}
Think step by step, then conclude with
a line in the form "Answer:
{final_answer}".
```

For multiple-choice datasets (ARC-Challenge, CommonsenseQA), the prompt additionally lists the answer choices verbatim before "Think step by step". For StrategyQA, the final answer must be one of yes/no. For GSM8K and MATH500, the final answer is a numeric or symbolic expression.

C.2 Verifier Prompt

For each (question, candidate answer, candidate reasoning) tuple, we ask each verifier $K_v = 4$ times:

Question: {question}
Candidate reasoning: {reasoning}
Candidate answer: {candidate_answer}
Is the candidate answer correct? Reply
with
exactly one word: yes or no.

The verifier’s empirical yes-rate over K_v queries is the verifier acceptance score $s_v(q) \in \{0, 0.25, 0.5, 0.75, 1.0\}$.

C.3 Extraction Rules

We extract the final answer from each chain-of-thought sample as follows:

- **GSM8K:** Take the last numeric token in the sample (matching `[-]?[d+(\.d+)?]`; fall back to the rightmost number after "Answer:" if present.
- **StrategyQA:** Match `(?i)\b(yes|no)\b` after the last "Answer:" marker; if absent, return None.
- **MATH500:** Extract the content of the last `\boxed{...}` expression with nested-brace handling, then compare to the gold via SymPy-based mathematical equivalence (LaTeX→SymPy via `latex2sympy2`, with normalization for fractions/roots/degrees/tuples and a numerical fallback; full procedure in Appendix E.5). $\sim 4\%$ of MATH500 items remain unparseable under this protocol, mostly matrix-environment and multi-part goods. These are excluded from the analysis. The filter is determined by gold structure and is prediction-independent.
- **ARC-Challenge:** Match the letter (A/B/C/D) following the last "Answer:" marker, falling back to the most recently mentioned choice letter.
- **CommonsenseQA:** Same as ARC, with choice labels from the question.

A sample whose extracted answer is None is treated as missing for self-consistency aggregation but still counted toward $K = 16$. Question-generator instances with no extracted majority answer are excluded from the analysis.

Majority aggregation: canonicalization and tie-breaking.

For self-consistency aggregation, extracted answers are canonicalized before counting. Categorical answers (StrategyQA yes/no; ARC and CSQA letter choices) are compared by exact string equality after lowercasing and whitespace stripping. GSM8K numeric answers use absolute tolerance $|a_i - a_j| < 10^{-4}$. MATH500 prediction–prediction equivalence uses the same SymPy-based protocol used for prediction–gold comparison (LaTeX→SymPy via `latex2sympy2`, normalization for fractions/roots/degrees/tuples, numerical fallback): two predictions whose extractions are SymPy-equivalent (e.g., $\frac{1}{2}$ and 0.5, or $\sqrt{2}$ and $2^{1/2}$) are counted as agreeing. Majority-count ties are broken deterministically by *first occurrence in original sampling order*, matching the same first-occurrence rule used for verifier rationale selection. Exact ties at $K=16$ are rare ($< 1\%$ of cells in our cohort) and are recorded explicitly in the released per-question artifacts.

Verifier rationale selection. For verifier scoring, the representative chain r^* is the *first sample in original sampling order* whose extracted answer belongs to the majority equivalence class: exact normalized string equality for categorical answers (StrategyQA yes/no; ARC/CSQA letter choices), absolute numeric tolerance $|a - \hat{y}| < 10^{-4}$ for GSM8K numeric answers, and the same SymPy-equivalence predicate used for MATH500 aggregation. If no sample matches under the corresponding predicate, we fall back to the text of the first sample. This rule is applied identically across all five generators, both the open-weight models served by local vLLM and GPT-5.4-mini through its API. We do not score chains by token log-probability.

D Additional Empirical Evidence

D.1 Per-Cell CBW Heatmap

Figure 5 reports the full 5×5 per-cell view of the dot-plot summary in Figure 1 (Section 4), giving the exact CBW@0.8 and Err@0.8 values for every (dataset, generator) cell. The heatmap is provided for readers who want to look up individual cells. The dataset-level pattern and the task-dominant variance structure are summarized in the main-text figure.

D.2 Per-Generator Conditional Error

Figure 6 reports the detailed per-generator version of Figure 2(a), with Wilson 95% CI and faint con-

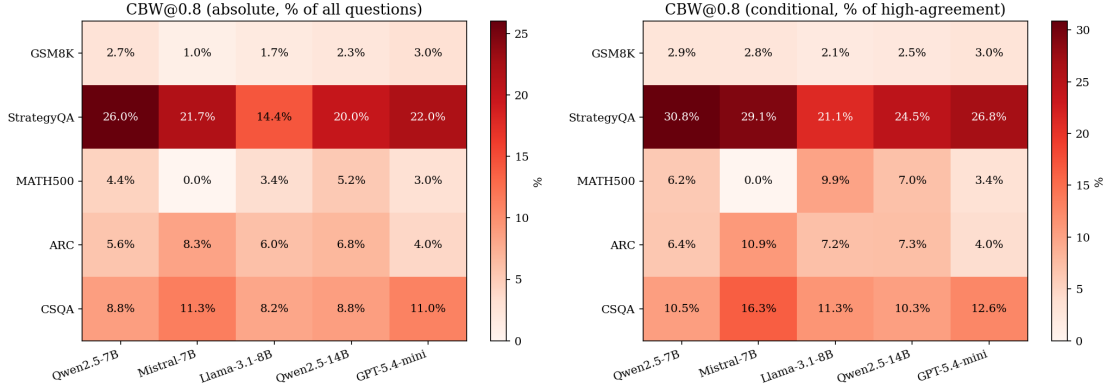


Figure 5: Per-cell unconditional CBW@0.8 (left) and conditional Err@0.8 (right) across 5 datasets and 5 generators. The Mistral×MATH500 cell has only $n_h=7$ high-agreement instances (Appendix E.26); MATH500 uses SymPy-based answer recovery (Appendix E.5).

necting lines. Bins with fewer than 5 valid examples are omitted; an omitted bin indicates insufficient support. The per-generator decomposition reveals that the CBW pattern is not a pooling artifact: on StrategyQA, every generator individually retains substantial conditional error in the highest- ρ bin, with Qwen2.5-7B and Mistral-7B both above 25% at $\rho \geq 0.9$. On GSM8K and ARC, all five generators converge cleanly to single-digit conditional error at $\rho \geq 0.9$. CommonsenseQA exhibits noisier curves owing to the smaller per-bin sample sizes, but the dataset-level ranking is preserved.

D.3 Robustness: Sample Budget and Threshold

Figure 7 reports the two robustness analyses referenced in Sections 4.4 and 4.6. (a) Mean CBW@0.8 across generators is stable on four of five datasets as K grows from 4 to 16; only MATH500 decreases substantially, consistent with its parsing-coverage sensitivity. (b) Sweeping the agreement threshold $\tau \in [0.5, 1.0]$ preserves dataset ordering. StrategyQA retains 15.7% CBW even at $\tau = 0.9$; MATH500 decreases on the parseable subset, consistent with the parsing-coverage analysis in Appendix E.5. Together these two views establish that the headline ordering is not an artifact of either the chosen threshold $\tau = 0.8$ or the budget $K = 16$.

D.4 Calibration-Transfer Violation Rates

Figure 8 reports the empirical risk-violation rate at $\alpha = 0.10$ across four calibration-transfer regimes referenced in Section 6. In-domain (50/50 split) violations stay within the $\delta = 10\%$ budget; cross-generator violations rise to 15–20%; leave-one-task-out and direct cross-task violations rise to 31–

63%. Per-task Mondrian recalibration restores validity (Appendix E.13). These results motivate the Section 6 claim that RISC’s finite-sample selective-risk guarantee is in-domain only and that per-task recalibration is required under task shift.

D.5 Target-Risk Validity Across Settings

Figure 9 reports the calibration validity referenced in Section 5.5. Observed-violation rates remain within the $\delta = 10\%$ LTT budget at $\alpha \in \{0.10, 0.20\}$ for SC-only and both RISC variants. The GPT-5.4-mini verifier used in isolation shows mild over-violation at $\alpha = 0.20$ (2.8% across admitted splits, against 0.6% for the operative variant), which reinforces the value of combining ρ with a verifier. The left panel plots target α against observed selective error. The right panel plots achieved coverage. SC-only and the operative RISC variant ($\rho + s_v^{\text{GPT-5.4-mini}}$) both stay below the $y = x$ line at the tested target risks, indicating valid in-domain risk control. Coverage achieved at the same α separates more sharply than calibration validity: the operative variant achieves 84% coverage at $\alpha = 0.20$ and 78% at $\alpha = 0.10$ (the 78.2% admission-conditional headline of Section 5.5), while a variant that adds the lower-quality GPT-4o-mini signal to the score drops to 68% at $\alpha = 0.10$, consistent with the Section 5.5 finding that adding a noisier verifier can reduce coverage at strict targets.

D.6 Per-Task Risk-Coverage Curves

Figure 10 gives the per-task curves behind Table 3. The ordering of the two methods depends on the task at every coverage level, and the 0.7 operating point reported in the body is one slice of it: RISC

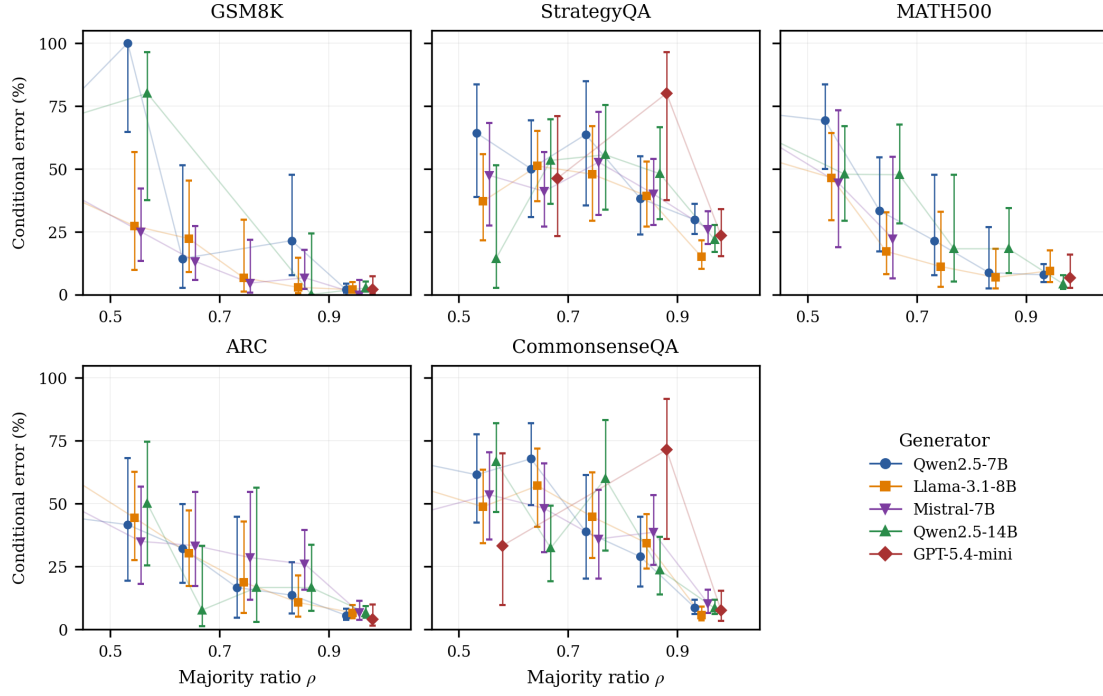


Figure 6: Per-generator conditional error $\Pr[\hat{y} \neq y \mid \rho \in \text{bin}]$ across 5 datasets and 5 generators, with Wilson 95% confidence intervals. Bins with fewer than 5 valid examples are omitted; faint connecting lines guide the eye between bins.

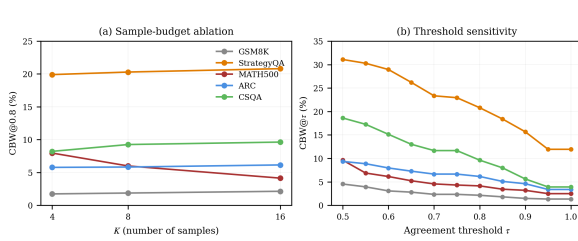


Figure 7: Robustness analyses. (a) Sample-budget ablation: mean CBW@0.8 across generators as a function of $K \in \{4, 8, 16\}$. (b) Threshold sensitivity: mean CBW@ τ across generators for $\tau \in [0.5, 1.0]$.

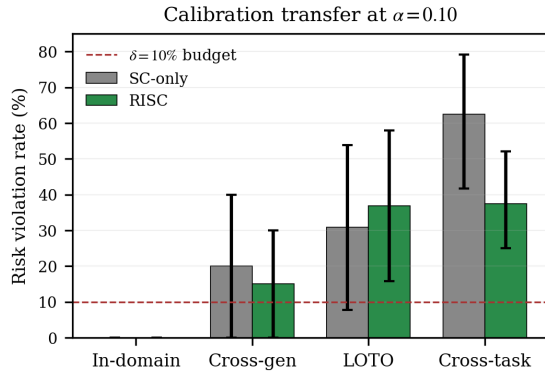


Figure 8: Empirical risk-violation rate at $\alpha = 0.10$ across four calibration-transfer regimes; 95% bootstrap CI; dashed line: LTT budget.

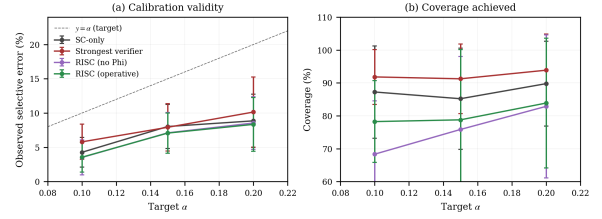


Figure 9: LTT calibration validity. Left: target α versus observed selective error. Right: achieved coverage. Observed-violation rates remain within the $\delta = 10\%$ budget at the target $\alpha \in \{0.10, 0.20\}$.

dominates SC-only on GSM8K, MATH500, and ARC, and does not on StrategyQA and CommonsenseQA, which are the two tasks where the verifier adds least separation over agreement.

D.7 Verifier Combination Ablation

Figure 11 reports the full verifier-ablation results referenced in Section 5.5. AURC (lower is better) is minimized by $\rho + s_{5.4}$, the operative variant of Section 5.4, with the all-signal ensemble inside its bootstrap CI at 7.54% and SC-only at 9.41%. Coverage at $\alpha = 0.10$ (higher is better) follows the same ordering, at 75.7% for $\rho + s_{5.4}$ against 71.6% for SC-only, and adding the GPT-4o-mini verifier degrades it to 65.0%. The pattern motivates the verifier-quality observation reported in

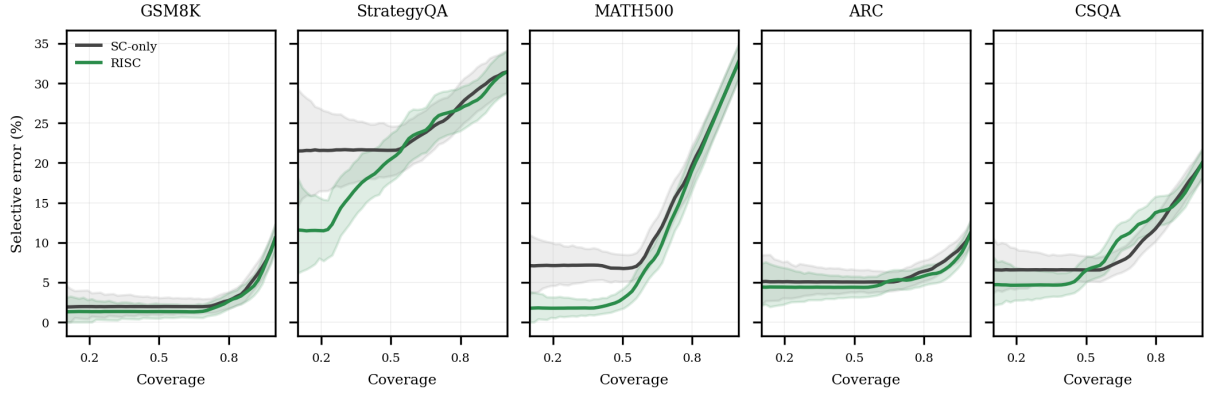


Figure 10: Per-task risk-coverage curves for SC-only vs. RISC (maj + GPT-5.4-mini verifier) on the five-dataset cohort; 95% paired-bootstrap CI bands over 1,000 resamples.

Section 5.5: adding a noisier verifier can extend the operating envelope on AURC while shrinking it on strict-target coverage. The optimal combination therefore depends on the target operating point, and the practical recipe is to select verifier candidates by held-out quality.

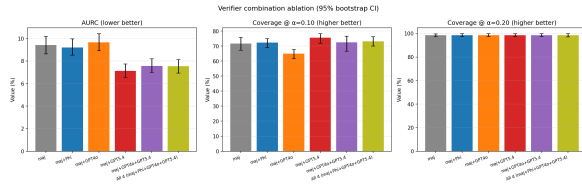


Figure 11: Verifier-combination ablation across six signal subsets. (a) Risk-coverage area (AURC, lower is better). (b) Coverage achievable at $\alpha = 0.10$ (higher is better). 95% bootstrap CIs over 1,000 resamples; cohort size varies per combo (Phi-3.5-mini-T=0.5 and GPT-4o-mini verifier outputs were not collected on the Mistral-7B cells, so combos requiring those signals restrict to the 4-generator subcohort, $N=6,724$; combos using only ρ and the GPT-5.4-mini verifier pool the full 5-generator cohort, $N=8,222$).

D.8 Cost-Quality Per Dataset

Figure 12 reports the per-dataset cost-quality comparison referenced in Section 6.2. Per-dataset Δ (Qwen2.5-7B+RISC – GPT-5.4-mini) at coverage 0.9: GSM8K +1.11pp [−2.96, +4.44], StrategyQA +6.16pp [−5.19, +17.05], ARC +2.20pp [−3.11, +6.67], CSQA −0.92pp [−9.78, +7.33] (all 95% paired-bootstrap CIs straddle zero), and MATH500 +7.94pp [+2.26, +12.69] (excludes zero, favoring GPT-5.4-mini). On four of five datasets the local-generator pipeline shows no statistically detectable selective-error gap at this sample size relative to GPT-5.4-mini majority voting

at $\sim 9\times$ lower metered API cost; on MATH500 a residual capability gap appears at 0.9 coverage and widens further at full coverage on the two hardest tasks (StrategyQA and MATH500).

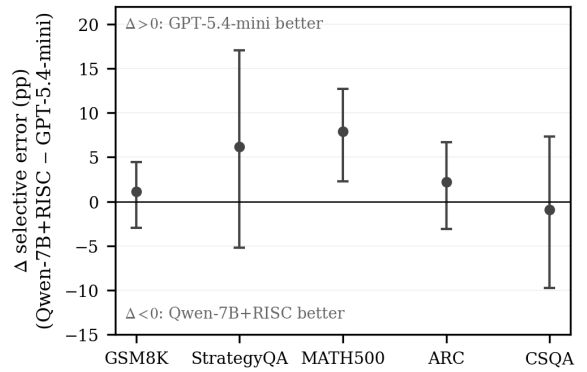


Figure 12: Cost-quality at a fixed 0.9 coverage operating point across 5 datasets with 95% bootstrap CIs (not the $\alpha=0.10$ LTT-calibrated threshold). GPT-5.4-mini majority vote ($K = 16$, $\sim \$10.5$ per 1k questions) vs. Qwen2.5-7B+RISC ($K_v = 4$ GPT-5.4-mini verification queries, $\sim \$1.2$ per 1k questions).

E Appendix: Extended Empirical Analyses

E.1 Protocol Map for Coverage / Error Quantities

Several coverage and selective-error quantities appear under different cohorts and protocols. Table 7 maps each quantity to its underlying cohort and protocol and indicates which are directly comparable to the LTT-validity result of Section 5.5.

E.2 Score-Combination Ablation

Beyond the unweighted mean, we evaluate two additional monotone combinations of (ρ, s_v) that

Reported number	Cohort	Protocol	Comparable to Section 5.5?	Section
78.2% cov. / 3.55% err. / 0.0% viol.	full $N=8,222$ (25 cells)	per-cell 50/50 LTT, $R=10$; admission-conditional mean (23% Bonferroni admission at $\alpha=0.10$)	yes (matches Section 5.5)	Sec-
69.3% accept. / 8.36% sel. err.	full $N=8,222$	single pooled-threshold descriptive diagnostic	no (Appendix E.24)	
83.6% cov. at $\alpha=0.10$	triple-coverage subset $n=1,858$	empirical max coverage at $\leq \alpha$ error	no (Appendix E.14)	
68.8% cov. at $\alpha=0.10$	triple-coverage subset $n=1,858$	LTT-admitted Bonferroni–Clopper–Pearson coverage	no (Appendix E.15)	
81.2% cov. at $\alpha=0.10$	full cohort $N=8,222$	paired pooled bootstrap	no (Appendix E.17, E.18)	

Table 7: Protocol map for coverage and selective-error quantities across protocols and cohorts. The LTT-validity number from Section 5.5 (78.2%/3.55%) is the only headline directly comparable to the $\alpha=0.10$ LTT budget. The others use different cohorts (no-Mistral, triple-coverage) or different protocols (single-threshold pooled diagnostic, empirical max-coverage). The pooled-threshold diagnostic of Appendix E.24 is the one most likely to be misread as comparable to the Section 5.5 headline, and is explicitly disclaimed there.

preserve LTT validity without learning weights: product $s = \rho \cdot s_v$ and minimum $s = \min(\rho, s_v)$. The score-combination subcohort here is (Qwen2.5-14B, Llama-3.1-8B, Mistral-7B-Instruct, GPT-5.4-mini; $N=6,154$), distinct from the no-Mistral verifier-availability subcohort used in cross-vendor and cross-prompt ablations. Figure 13 reports AURC and coverage at $\alpha = 0.10$ for the three combinations and the two individual signals.

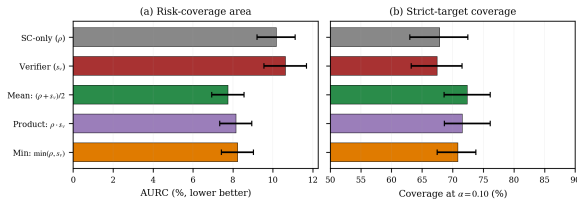


Figure 13: Score-combination ablation. AURC (lower is better) and coverage at $\alpha = 0.10$ (higher is better) for the three monotone combinations and each individual signal. The unweighted mean leads on both metrics, and its margin over the product and the minimum survives paired resampling.

The unweighted mean is the best of the three (AURC 7.76% [6.95, 8.54] against product 8.14% [7.33, 8.95] and min 8.23% [7.39, 9.03]), and its advantage survives paired resampling: -0.38pp $[-0.73, -0.01]$ over the product and -0.47pp $[-0.82, -0.11]$ over the minimum. Coverage at $\alpha = 0.10$ orders the same way (mean 72.4%, product 71.6%, min 70.9%). Averaging keeps information that the product and the minimum discard, since both collapse toward whichever signal is smaller. We also evaluate two learned monotone scores fit with 5-fold cross-fitting on the same

subcohort ($N=6,154$): a constrained logistic regression on (ρ, s_v) , and a cyclic isotonic GAM $s = f_1(\rho) + f_2(s_v)$ with f_1, f_2 non-decreasing fit by backfitting. Both stay within bootstrap CI of the unweighted mean (logistic 7.94% [7.13, 8.78], isotonic GAM 7.91% [7.11, 8.76], vs. mean 7.76% [6.95, 8.54]), with paired-bootstrap AURC differences of -0.18pp $[-0.61, +0.25]$ for logistic and -0.15pp $[-0.56, +0.28]$ for the isotonic GAM, neither excluding zero. $\text{Cov}@ \alpha=0.10$ shows the same pattern (logistic 73.6%, isotonic GAM 73.4%, mean 72.4%; CIs overlap). Verifier-alone ranking on this subcohort gives AURC 10.65% and $\text{Cov}@ \alpha=0.10$ 67.5%, worse than every (ρ, s_v) combination. We therefore report the unweighted mean as the operative variant: a non-parametric monotone learner over the same two features does not detect additional structure on our cohort, while ranking by ρ alone or s_v alone is decisively dominated.

E.3 Verifier-Budget Sensitivity (K_v)

The verifier query budget K_v is a cost-variance lever. We simulate Bernoulli draws of size $K_v \in \{1, 2, 4, 8, 16\}$ from each question’s empirical yes-rate, treating the four-query empirical rate as the underlying parameter, and recompute RISC AURC and $\text{Cov}@ \alpha = 0.10$ at each budget (Figure 14).

AURC decreases from 8.17% ($K_v = 1$) to 7.61% ($K_v = 4$) and 7.40% ($K_v = 16$). Coverage at $\alpha = 0.10$ rises from 71.3% to 74.4% and saturates near 75.1%. The marginal gain from $K_v = 4$ to $K_v = 8$ is well within the bootstrap CI. We adopt $K_v = 4$ as a cost-effective operating point and note

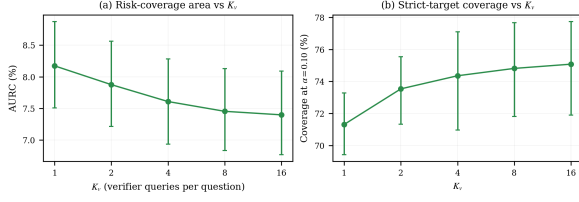


Figure 14: Verifier-budget sensitivity. AURC and coverage at $\alpha = 0.10$ as a function of K_v , with 95% bootstrap CI. Returns to additional queries diminish beyond $K_v = 4$, supporting our default budget choice.

that this analysis treats the simulated draws as exchangeable across queries. A direct independence audit using paraphrased prompts or alternative verifier variants is outside the scope of this sensitivity analysis.

E.4 Less Conservative Calibration: Holm Step-Down

The Bonferroni correction over a $|T|$ -sized threshold grid is conservative. The Holm step-down family of multiple-testing corrections tightens the bound on the most permissive thresholds. Figure 15 reports the fraction of (generator, dataset) cells in which a valid threshold is admitted at each target α , under Bonferroni and Holm step-down.

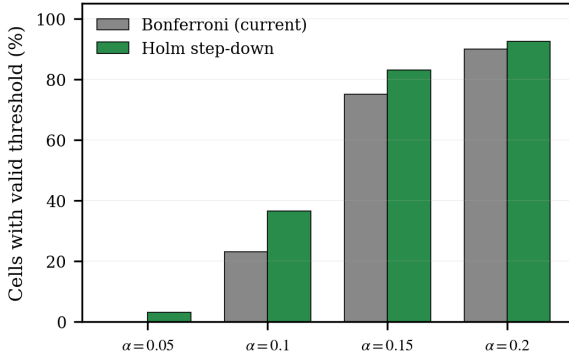


Figure 15: Fraction of cells with valid threshold under Bonferroni vs. Holm step-down LTT, for $\alpha \in \{0.05, 0.10, 0.15, 0.20\}$. Holm admits a usable threshold in slightly more cells but does not recover $\alpha = 0.05$ at our calibration sizes ($n_{\text{cal}} \approx 150$).

At $\alpha = 0.05$, Bonferroni admits no valid threshold in any cell, and Holm admits a threshold in only $\sim 3\%$ of cells (with substantial test-set violations). Strict targets therefore require larger calibration budgets. At $\alpha = 0.10$, Holm increases the usable-cell rate from 22% to 30% at near-zero violations on our test sets (0% for Bonferroni and 1.3% for Holm; both well within the $\delta = 0.10$ budget); at $\alpha = 0.20$, both procedures admit a

threshold on $\geq 85\%$ of cells. We report Bonferroni in Section 5.5 for transparency. Holm is a drop-in replacement that yields a small efficiency gain at the moderate- α regime where our results concentrate.

E.5 MATH500 SymPy Recovery and Parseability Sensitivity

Procedure. Plain regex matching `\boxed{[^\^]+\}` fails on nested braces (e.g., `\boxed{(3,\frac{\pi}{2})}` produces the truncated string `(3,\frac{\pi`, which then mismatches the gold) and would retain only $\sim 67\%$ of MATH500 items. Our protocol uses a nested-brace boxed extractor and a SymPy-based equivalence check (LaTeX $\rightarrow$ SymPy via `latex2sympy2`, normalization for `\frac/\sqrt/degree/tuple` forms, numerical fallback). Across the five generators this retains $\sim 96\%$ of items as parseable. The remaining $\sim 4\%$ are matrix-environment and multi-part golds where SymPy comparison fails or is ambiguous (e.g., `\begin{pmatrix}`... , set conditions like “ $x \in [-2, 7]$ ”).

Per-cell numbers. The MATH500 analysis cohort is $\sim 1,823$ parseable instances across the five generators. Mean CBW@0.8 is 3.36%, with per-generator values 4.70% (Qwen2.5-7B), 5.54% (Qwen2.5-14B), 3.47% (Llama-3.1-8B), 0.00% (Mistral, $n_h=7$ only), 3.06% (GPT-5.4-mini). Conditional error at $\rho \geq 0.9$ ranges from 2.5% (GPT-5.4-mini) to 11.7% (Llama-3.1-8B), excluding the Mistral cell with $n_{h \geq 0.9} = 2$. MATH500 stays below StrategyQA in every regime.

Parseability-frontier profile. The percentages below characterize items not parseable under plain string-equality matching (used as a comparison reference only); under the SymPy protocol of Section 4 only $\sim 4\%$ of items remain genuinely excluded. Beyond the token-length comparison (mean tokens 34.5 vs. 29.8 on Qwen2.5-7B), excluded items concentrate on Precalculus (75.0% excluded), Geometry (51.2%), and Intermediate Algebra (39.2%); Number Theory (3.2%), Prealgebra (25.6%), and Algebra (25.8%) are under-represented ($\chi^2=81.2$, $\text{df}=6$, $p \approx 2 \times 10^{-15}$). The mechanism is transparent: every gold containing a polar/Cartesian tuple ($n=19$), a `\matrix` environment ($n=7$), or a `\pi` literal ($n=10$) is excluded under regex extraction (100%); `\frac` (97.1%) and `\sqrt` (95.8%) almost as reliably. SymPy recovery resolves the `\frac/\sqrt/tuple/\pi` cases. Ma-

trix golds remain unrecovered. By difficulty, the parseability-frontier rate increases monotonically with level (20.9% at L1 to 44.8% at L5; $\chi^2=12.6$, $df=4$, $p=0.013$); under SymPy this gradient flattens substantially. The qualitative claim that high-agreement error is task-dependent and GSM8K/ARC/CommonsenseQA remain substantially below StrategyQA is preserved across both regimes.

E.6 StrategyQA Qualitative Failure Modes

To understand *why* $CBW@_\tau$ remains substantial on StrategyQA at high self-consistency thresholds, we manually inspect all high-agreement-but-wrong instances at the relaxed threshold $\rho \geq 0.75$ (i.e. $\geq 12/16$ samples agree on a wrong answer). The qualitative inspection is restricted to the three generators (Qwen2.5-14B, Llama-3.1-8B, GPT-5.4-mini) for which full reasoning traces were retained under the logging protocol. The five-generator quantitative high-agreement tail (including Mistral-7B and Qwen2.5-7B) is reported separately in Appendix E.12. This yields 143 instances: QWEN2.5-14B-INSTRUCT 70, LLAMA-3.1-8B-INSTRUCT 51, and GPT-5.4-MINI 22, spanning 95 unique questions. 143 at $\rho \geq 0.75$ is the largest sample available without diluting the high-agreement precondition: even at $\rho \geq 0.6$ the union across these three generators contains only 191 high-agreement errors. For each item we read the question, the gold label, the generator’s majority answer, and at least one full chain-of-thought, then assigned a single failure-mode category using an inductively developed five-class rubric, finalized before the breakdown below was computed. The categories are stable across generators; their mixing weights are not.

The aggregate breakdown (per-instance, $n = 143$) is: *multi-step retrieval failure* 53.8% (one bridging fact retrieved correctly but the other either hallucinated or missed, and the error propagates with high consensus because the correct intermediate step is genuinely hard to recover); *implicit-reasoning shortcut* 27.3% (3+ step chains collapsed to a plausible 1-step shortcut, e.g. “most expensive color = blue” ignoring the gold’s specific frame); *lexical or negation prior* 13.3% (modal cues such as “hypothetically” or embedded negations like “Is it impossible. . .” flip the surface heuristic); *annotation artifact / questionable gold* 4.9% (the gold label is debatable on inspection, e.g. a literal-numeric question where the gold disagrees with a basic con-

version or textbook definition); and *common-sense vs. encyclopedia frame conflict* 0.7% (the literal common-sense answer is the opposite of the gold’s encyclopedic frame). Per-generator mixtures differ: QWEN2.5-14B splits 48.6% retrieval / 31.4% shortcut / 12.9% lexical-negation, LLAMA-3.1-8B runs 56.9% retrieval / 23.5% shortcut / 15.7% lexical-negation, and GPT-5.4-MINI is the most retrieval-dominated at 63.6% / 22.7% / 9.1%, consistent with stronger surface-form parsing in the larger and more recent models.

Cross-tabulating across generators, 40 of the 95 unique questions (42%) are wrong on ≥ 2 of the 3 generators at $\rho \geq 0.75$, and 8 are wrong on all three. Shared structural failure of this magnitude indicates that the high-agreement error is driven by properties of the question: when a StrategyQA item fools self-consistency on QWEN2.5-14B, it is roughly five times more likely than chance to also fool LLAMA-8B or GPT-5.4-MINI.

Restricting to the strictest agreement bin $\rho \geq 0.9$ (a subset of this sample) gives mixing weights consistent with the aggregate: *multi-step retrieval* and *implicit-reasoning shortcut* together still account for over 70% of high-agreement errors, with *annotation artifact*, *frame conflict*, and *lexical/negation prior* occupying smaller shares. The qualitative story, in which the dominant failure mode is structural, therefore does not depend on the $\rho \geq 0.75$ relaxation.

Three illustrative items (paraphrased; gold and majority predictions verbatim):

- **Q:** “Is sunscreen unhelpful for the condition that killed Bob Marley?” **Gold:** Yes. **Majority pred. (15/16, Llama):** No. *Negation parsing:* Marley died of acral lentiginous melanoma, a UV-independent subtype; chains correctly identified “melanoma” but the embedded negation “unhelpful” inverted the answer relative to the standard sun-causes-melanoma script.
- **Q:** “Is it impossible for pigs to use pig latin?” **Gold:** Yes. **Majority pred. (16/16, Llama):** No. *Negation parsing:* every chain answers the unwrapped question (“can pigs use pig latin?”) and reports its negation, leaving the original double-negative unanswered.
- **Q:** “Does water have viscosity?” **Gold:** No. **Majority pred. (16/16, Llama; 15/16,**

Qwen2.5-14B): Yes. *Probable annotation artifact*: water has nonzero viscosity (~ 1 cP at 20°C) by any standard physics definition; the gold appears to encode a different intended frame that the question does not surface. Both generators independently produce the textbook answer with high consensus.

These observations suggest that a nontrivial portion of the high conditional error at $\rho \geq 0.75$ reflects recurring item-level difficulty, such as hard bridging retrieval, modal and negation cues, or premature shortcutting. Because the fine-grained five-class taxonomy has low independent-rater agreement (Cohen’s $\kappa=0.076$, Appendix E.7), we treat the category proportions as heuristic diagnostics. The LLM second-rater attributes a much larger share to annotation artifact than the human coder, so the $\sim 5\%$ gold-label-noise reading is a label-conservative lower bound. The independent observation that 42% of audited items fail across ≥ 2 of the three audited generators is consistent with item-level shared difficulty, but on its own does not adjudicate which fraction is structural-difficulty versus gold-label artifact.

E.7 Inter-Rater Agreement on StrategyQA Failure Modes

Methodology. A potential concern with the failure-mode taxonomy in Appendix E.6 is that the 143 high-agreement-wrong StrategyQA instances were tagged by a single author applying a heuristic rubric. To stress-test the breakdown, we collect a second, independent rating from **Claude Sonnet 4.6** (Anthropic API, `claude-sonnet-4-6`, temperature 0.5, $K=1$ query per instance, total API spend \$0.24). For each of the 143 instances we send the question, gold label, model’s majority answer, one representative chain-of-thought, and the verbatim five-category rubric (multi-step retrieval, annotation artifact, lexical/negation prior, frame conflict, implicit shortcut), and ask the model to assign exactly one category. For eight outputs that were not parseable under the one-label schema, we repeated the rating call with a stricter one-label instruction; all 143 ratings were ultimately parseable. Raw second-rater outputs are released alongside the codebase.

Cohen’s κ and confusion matrix. Treating the human coder and the LLM as two raters over the five-class label space, observed agreement is $p_o=0.224$ against chance agreement $p_e=0.160$,

yielding Cohen’s $\kappa=0.076$. On the Landis–Koch scale this is in the *poor* band ($\kappa<0.20$). The 5×5 confusion matrix (rows = human coder, columns = LLM rater) is shown in Table 8. Per-category

human ↓	LLM rater (Claude Sonnet 4.6)					total
	m.s.r.	art.	lex.	frm.	i.s.	
multi_step_retrieval	21	49	1	4	2	77
annotation_artifact	0	7	0	0	0	7
lexical_negation_prior	4	12	0	2	1	19
frame_conflict	0	1	0	0	0	1
implicit_shortcut	5	26	0	4	4	39
total	30	95	1	10	7	143

Table 8: Inter-rater confusion matrix between the human coder and the Claude Sonnet 4.6 second rater on 143 high-agreement-wrong StrategyQA items. Diagonal entries indicate agreement; abbreviations: m.s.r. = multi-step retrieval, art. = annotation artifact, lex. = lexical/negation prior, frm. = frame conflict, i.s. = implicit shortcut.

agreement rates (diagonal over human-coder row total) are 27.3% for multi-step retrieval, 100% for annotation artifact, 0% for lexical/negation prior, 0% for frame conflict, and 10.3% for implicit shortcut.

Where the disagreement concentrates and what survives. The disagreement is structural: the LLM rater applies the `annotation_artifact` label to $95/143=66.4\%$ of items, versus $7/143=4.9\%$ for the human coder, and a binary "is-artifact / not-artifact" reduction yields $\kappa=0.05$ that is essentially noise. In other words, when shown a high-agreement-wrong StrategyQA item with a reasoned chain of thought, Claude Sonnet 4.6 is sympathetic to the model under inspection and prefers to fault the gold label; the human coder, who labels artifact only on explicit gold-debatable cues, does not. A 5-class κ of 0.076 does not provide independent confirmation of the human coder’s breakdown, and the headline annotation-artifact rate ($\sim 5\%$) is therefore best read as a lower bound from a label-conservative pipeline. However, two robustness signals do emerge. First, restricted to the 48 items where the LLM does *not* default to `annotation_artifact`, exact agreement with the human coder rises to $25/48=52.1\%$, indicating that when the LLM commits to a non-artifact failure mode it concentrates on the same multi-step / implicit-shortcut failures the human coder flags. Second, the LLM’s own non-artifact distribution (multi-step retrieval 30, implicit shortcut 7, frame conflict 10, lexical/negation 1) preserves

multi-step retrieval as the dominant failure mode, matching the human coder’s qualitative ordering. We therefore treat the qualitative narrative in Appendix E.6 (that residual StrategyQA errors at our operating point are dominated by multi-hop retrieval and implicit-reasoning shortcuts) as supported, while flagging that the precise share of items reasonably attributable to gold-label noise is highly rater-dependent and our $\sim 5\%$ estimate is conservative.

E.8 LTT vs. RCPS

A natural alternative to LTT is Risk-Controlling Prediction Sets (RCPS, Bates et al., 2021). LTT searches the threshold grid $T = \{s_i : i \in D_{cal}\}$ and tests each candidate with a Clopper–Pearson upper bound at confidence $1 - \delta/|T|$, paying an explicit Bonferroni correction. RCPS instead applies a single pointwise Hoeffding–Bentkus (HB) bound at confidence $1 - \delta$. Under Theorem 1 and §3.2 of Bates et al. (2021) the bound is uniformly valid along the monotone-in- t loss path, so no grid correction is required. Both yield finite-sample, distribution-free selective-risk control.

We apply both calibration procedures to the per-question acceptance scores $s = (\rho + s_v)/2$ from Section 5.5 (GPT-5.4-mini verifier), averaging 10 random 50/50 cal/test splits per cell with $\delta = 0.10$, on the score-combination subcohort (Qwen2.5-14B, Llama-3.1-8B, Mistral-7B-Instruct, GPT-5.4-mini; $N=6,154$). Of 20 (dataset, generator) cells, 18 had ≥ 5 wrong samples and were retained. GSM8K and ARC paired with GPT-5.4-mini are excluded (only 4 wrong samples each). At $\alpha=0.10$, LTT admits a threshold on 17.2% of runs and RCPS on 58.3%. Conditional on admission, test selective error is 3.6% (LTT) vs. 5.5% (RCPS), coverage 79.0% vs. 69.3%, and violation rates are 0.0% and 5.7%. At $\alpha=0.20$, LTT/RCPS admit 67.2%/86.7% of runs with selective error 8.6%/12.5%, coverage 79.3%/76.4%, and violations 0.8%/9.0% (Table 9).

α	method	valid	sel. err.	cov.	viol.
0.10	LTT	17.2%	3.6%	79.0%	0.0%
0.10	RCPS	58.3%	5.5%	69.3%	5.7%
0.20	LTT	67.2%	8.6%	79.3%	0.8%
0.20	RCPS	86.7%	12.5%	76.4%	9.0%

Table 9: LTT vs. RCPS, 18 cells $\times$ 10 seeds, $\delta=0.10$.

Both methods keep empirical violation rates within the $\delta=0.10$ budget, with selective errors

below the corresponding α targets. RCPS is admissible on more than three times as many runs at $\alpha=0.10$ because it avoids the $|T|$ -fold Bonferroni penalty. LTT is more conservative, leaving more runs without a threshold but achieving lower selective error and higher coverage when it commits. We retain LTT as the default for transparency about multiple-testing structure with modest calibration sets, and note RCPS as a valid drop-in with very similar empirical behavior.

E.9 Answer-Distribution-Only Auxiliary Signals

We test whether scalar agreement signals *other than* the majority ratio ρ (already computable from the same $K=16$ samples) might serve as effective acceptance scores under our LTT framework, with no new compute. We evaluate four answer-distribution-only signals (entropy of the answer histogram, top-vs-runner-up margin, number of unique answers, and a $\rho+$ (-entropy) ensemble) alongside the verifier-augmented score $\frac{1}{2}\rho + \frac{1}{2}s_v$ as a reference. To keep this appendix directly comparable with the score-combination ablation of Appendix E.2, we follow that section’s exact protocol: scores are evaluated by pooling all 6,154 generator $\times$ dataset question instances on the GPT-5.4-mini-verifier-available sub-cohort (5 datasets $\times$ 4 of the 5 generators of Section 4.1; Qwen2.5-7B is excluded from this ablation cohort), ranking them by score, and integrating the empirical risk $1 - \text{acc}(K)$ against coverage K/N to obtain AURC. We also report selective coverage at the controlled risk level $\alpha=0.10$ defined as the largest cumulative coverage at which the cumulative error remains below α . Confidence intervals come from 1,000 paired bootstrap resamples over the pooled instances.

Score	AURC (%)	Cov. at $\alpha=0.10$ (%)
ρ alone	10.18 [9.21, 11.12]	67.90 [63.00, 72.47]
–entropy	10.09 [9.11, 11.03]	67.57 [62.59, 70.85]
margin	9.88 [8.90, 10.82]	69.20 [64.61, 73.84]
–num_unique	10.66 [9.67, 11.60]	59.01 [55.65, 62.61]
$\frac{1}{2}\rho + \frac{1}{2}(-\text{entropy})$	9.98 [9.02, 10.93]	67.85 [62.61, 71.55]
$\frac{1}{2}\rho + \frac{1}{2}s_v$ (verifier ref.)	7.76 [6.95, 8.54]	72.38 [68.64, 76.21]

Table 10: Pooled AURC and selective coverage at the controlled risk level $\alpha=0.10$ across the 5 datasets and the GPT-5.4-mini-verifier-available sub-cohort of 4 of the 5 generators of Section 4.1 ($N=6,154$ pooled question instances; Qwen2.5-7B is excluded from this ablation cohort). 95% paired bootstrap CIs in brackets (1,000 resamples).

None of the four answer-distribution-only signals improves over ρ alone: entropy and margin sit within bootstrap noise of ρ (10.09% and 9.88% AURC vs. ρ 's 10.18%), num_unique is strictly worse (10.66%, non-overlapping CI vs. margin), and the $\frac{1}{2}\rho + \frac{1}{2}(-\text{entropy})$ ensemble yields no detectable gain (9.98%). All five answer-only families are decisively dominated by the verifier-augmented score $\frac{1}{2}\rho + \frac{1}{2}s_v$ at 7.76% AURC, a 2.12-to-2.90-point absolute reduction with non-overlapping 95% CIs against *every* answer-only family (verifier upper CI 8.95% vs. best answer-only lower CI 8.90% for margin). At the strict-target coverage criterion the four broadly acceptable answer-only signals (ρ , $-\text{entropy}$, margin, $\rho + \text{entropy}$) already match the verifier reference within bootstrap noise around 68%, so the verifier's advantage shows up as a uniformly steeper risk-coverage curve, that is, a lower AURC. The one outlier is $-\text{num_unique}$, which loses roughly 10 points of strict-target coverage. Overall, *verifier-augmented RISC remains the operative variant*. No answer-distribution-only signal substitutes for an external verifier semantically grounded in the question-answer pair.

E.10 Confidence-Weighted Voting Baseline

Setup. Confidence-weighted voting requires per-path confidence, which the pooled cohort of Section 4 does not store, so we regenerate self-consistency with token-level scores retained. We use the three open-weight generators for which log-probabilities are available (Qwen2.5-7B, Qwen2.5-14B, Llama-3.1-8B) on GSM8K and StrategyQA, at $K=16$ and $n=300$ per cell with the decoding settings of Section 4.1, giving $N=1,799$ valid instances. Each chain receives confidence $c = \exp(\log p)$, the exponentiated mean token log-probability, and the vote of each chain is weighted by c in the manner of CISC (Taubenfeld et al., 2025). We then score selective risk by the confidence-weighted margin and compare it with the plain majority ratio on the same instances.

Ranking signal	AURC (%)	Acc (%)
SC-only (majority ratio ρ)	11.75	81.16
Confidence-weighted vote	11.68	81.38

Table 11: Confidence-weighted voting against SC-only on the regenerated subcohort ($N=1,799$). Acc is the accuracy of the returned answer; AURC is the area under the risk-coverage curve, lower being better.

Result. Confidence weighting changes the returned answer on 11 of 1,799 questions and moves accuracy by 0.22pp. Its effect on selective risk is 0.07pp of AURC (Table 11). Because 65.2% of these instances sit at $\rho=1.0$, the ranking contains many ties, and the AURC of a tie-heavy ranking depends on the order in which tied instances are swept. We therefore repeat the comparison under 500 random tie orders: SC-only averages 11.55 [10.49, 12.79], the confidence-weighted vote averages 11.51 [10.44, 12.76], and their paired difference averages 0.03pp $[-1.56, +1.68]$. The comparison is therefore insensitive to the tie convention, and the conclusion is the same under all of them. The external verifier of Section 5.4 moves the same measure by roughly two points on its own subcohort, consistent with the answer-distribution auxiliaries of Appendix E.9, which likewise fail to improve on ρ . Signals derived from the generator's own distribution behave alike here, which is what Theorem 1 anticipates: they are all functions of information the answer histogram already summarizes.

E.11 MATH500 Parseability Sensitivity

Appendix E.5 describes the SymPy-based answer extraction protocol. Under it, $\sim 4\%$ of MATH500 items remain unparseable (matrix and multi-part golds). The upper bound that assumes *every* unparseable item is high-agreement and wrong is at most 0.67–11.40% across generators. We report three complementary CBW@0.8 estimates per generator over the full nominal MATH500 set: a *best-case* apparent risk on the parseable subset; a *middle* bound treating each unparseable item as 50/50 wrong; and the *worst-case* bound. The middle bound corresponds to the symmetric prior of zero information about whether unparseable answers happen to be high-agreement-but-wrong.

The qualitative ordering of CBW prevalence across our five datasets is robust to the parseability-treatment regime. Under the best-case (parseable-only) reading, MATH500 (0.00–5.54%; the 0.00% Mistral cell reflects its very small high-agreement subset, $n_h=7$, and not a calibrated estimate) is comparable to GSM8K (2.15%) and ARC-CSQA (6.12–7.77%, per-generator range on the five-generator cohort), and well below StrategyQA (pooled cell-level CBW@0.8 = 23.0% across the $5 \times 1 = 5$ StrategyQA cells with $n=1,299$ valid pooled questions; the mean-of-cell-means of Section 4, 20.8%, weighs each generator equally, not

Generator	Cell-retained frac.	BEST (%)	MID (%)	WORST (%)
Qwen2.5-7B	93.6%	4.70	7.60	10.80
Qwen2.5-14B	93.8%	5.54	8.30	11.40
Llama-3.1-8B	98.0%	3.47	4.40	5.40
Mistral-7B	99.3%	0.00	0.33	0.67
GPT-5.4-mini	98.0%	3.06	4.00	5.00
Range		0.00–5.54	0.33–8.30	0.67–11.40

Table 12: MATH500 CBW@0.8 under three parseability-treatment regimes per generator, using the SymPy-based answer extraction of Appendix E.5. “Cell-retained frac.” is the fraction of the cell’s evaluated items whose gold answer is recoverable via SymPy normalization *and* whose chains yield a majority answer (cohort sizes: 500 for Qwen2.5-7B/14B/Llama, 300 for Mistral-7B, 100 for GPT-5.4-mini). The $\sim 4\%$ figure of Section 4 refers to residual gold-answer unrecoverability under the SymPy protocol (gold-structure-conditioned, prediction-independent); cell-retained fractions additionally reflect no-majority/no-extraction cases on the generation side and are therefore generator-dependent. BEST = recovered apparent risk (numerator: recovered high-agreement-wrong; denominator: recovered count). MID = unrecoverable items count as 50/50 wrong (numerator: recovered HRW + $0.5 \times$ unrecoverable count; denominator: cell’s evaluated total). WORST = unrecoverable items all count as high-agreement-but-wrong.

each question; Appendix E.12). Under the middle bound, MATH500 (0.33–8.30%) is below StrategyQA on every generator. Even under the maximally pessimistic worst-case (every remaining unrecoverable item is high-agreement and wrong), MATH500 (0.67–11.40%) stays below StrategyQA on every generator. GSM8K, ARC, and CommonsenseQA also remain substantially below StrategyQA in every regime. The MATH500 ordering is therefore preserved across all parseability regimes.

Topic and difficulty profile of would-be-excluded items. Beyond the token-length comparison in Appendix E.5, we also characterize the 169 items *that are not parseable under plain string-equality matching* (qwen-7B canonical filter; gold unparseable without LaTeX normalization) by MATH500’s native subject tag and difficulty level. Under the SymPy protocol of Section 4, only $\sim 4\%$ (~ 20 items) remain genuinely excluded. The analysis below characterizes the parseability frontier; the residual excluded set of Section 4 is smaller. The exclusion rate varies sharply by subject: Precalculus 75.0% (42/56), Geometry 51.2% (21/41), and Intermediate Algebra 39.2% (38/97) are over-represented among the excluded, while Number Theory (3.2%, 2/62), Prealgebra (25.6%), and Algebra (25.8%) are under-represented. A χ^2

test of independence between subject and exclusion is highly significant ($\chi^2 = 81.2$, $df = 6$, $p \approx 2 \times 10^{-15}$). Inspection of surface statistics on the gold answer makes the mechanism transparent: every MATH500 item whose gold answer contains a polar/Cartesian tuple ($n = 19$), a $\backslash$ matrix environment ($n = 7$), or a $\backslash$ pi literal ($n = 10$) is excluded (100%); items whose gold contains $\backslash$ frac (97.1%) or $\backslash$ sqrt (95.8%) are excluded almost as reliably. Mean answer length is 12.5 characters for excluded vs 2.6 for included items. The exclusion is therefore a *format* filter, driven by multi-token LaTeX expressions that defeat string-equality extraction, but it correlates strongly with subject because Precalculus and Geometry disproportionately require such expressions as final answers.

By difficulty, exclusion increases monotonically with level: 20.9% (Level 1, 9/43), 26.7% (L2, 24/90), 33.3% (L3, 35/105), 32.0% (L4, 41/128), 44.8% (L5, 60/134). The level-exclusion association is significant but modest ($\chi^2 = 12.6$, $df = 4$, $p = 0.013$; Pearson $r = 0.147$ between level and the exclusion indicator). The level effect is largely mediated by the subject effect: Precalculus and Intermediate Algebra items skew toward higher levels, and these are the subjects whose gold answers most often take a non-extractable LaTeX form. After conditioning on subject, the included subset still spans all five difficulty levels with 74 Level-5 items retained, so our CBW estimates are not confined to easy problems. We conclude that the parseability filter induces a topic-shaped distributional shift (Precalculus/Geometry under-represented; Number Theory/Algebra over-represented) but only a mild difficulty shift, and the included 331-item subset preserves coverage across both axes. The robustness of the dataset-level CBW ordering to the BEST/MID/WORST regimes above further confirms that this shift does not drive our qualitative claims.

E.12 StrategyQA High-Agreement Tail: Per-Generator and Cross-Dataset Significance

We examine whether the high-agreement-but-wrong finding on StrategyQA at $\rho \geq 0.9$ is driven by a single outlier generator, and whether the cross-dataset gap survives multiple-comparison correction. Per generator, the conditional error rate at $\rho \geq 0.9$ on StrategyQA is: Qwen2.5-7B $65/219 = 29.68\%$ (95% Wilson CI [24.02, 36.04]); Qwen2.5-14B $48/220 = 21.82\%$

[16.87, 27.73]; Llama-3.1-8B 23/153 = 15.03% [10.23, 21.55]; Mistral-7B 45/173 = 26.01% [19.97, 33.10]; GPT-5.4-mini 18/77 = 23.38% [15.33, 33.95]. *All five generators* place their Wilson lower bound at or above 10%, and four of five exceed 15% even on the lower bound. The smallest cell (GPT-5.4-mini, $n=77$) remains in the same regime as the larger cells. The phenomenon is not driven by any single generator.

Pooled across five generators, the conditional error rates at $\rho \geq 0.9$ are: GSM8K 18/899 = 2.00% [1.27, 3.15]; MATH500 58/860 = 6.74% [5.25, 8.62]; ARC 91/1470 = 6.19% [5.07, 7.54]; CommonsenseQA 105/1298 = 8.09% [6.74, 9.69]; StrategyQA 199/842 = 23.63% [20.86, 26.65]. After Bonferroni correction at family-wise $\alpha=0.05$ across the four StrategyQA-vs-other comparisons (per-test $\alpha=0.0125$), *all four* pairwise gaps remain highly significant (every $p < 10^{-16}$, smallest gap +15.54 percentage points with Newcombe lower bound at +12.36). The StrategyQA high-agreement-error finding is statistically robust to multiple-comparison correction and is not attributable to any single generator’s outlier behavior.

E.13 Mondrian / Per-Task Calibration

We examine whether per-task (Mondrian) conditional calibration recovers in-domain LTT validity when pooled cross-task calibration fails. We replicate the cross-task transfer setup of Section 6 under two regimes, both using the operative RISC ensemble score $\frac{1}{2}\rho + \frac{1}{2}s_v$ and the same Bonferroni–Clopper–Pearson LTT machinery ($\alpha=0.10$, $\delta=0.10$) so that the only thing that varies is the calibration distribution. Under *pooled* calibration we set $\mathcal{D}_{\text{cal}}$ to the union of the four non-target tasks (subsampling to match a 50/50 split’s calibration size for fairness) and evaluate on a held-out 50% of the target task. Under *Mondrian* per-task calibration we calibrate on a held-out 50% of the *target* task and evaluate on the remaining 50%. We average over $R=20$ permutations of each split. Because both regimes use a single LTT-controlled score, no Bonferroni correction over candidate calibration sources applies.

Pooled cross-task calibration violates the nominal risk on $\sim 34\%$ of admitted replicates in the 5-cohort matrix (concentrated on StrategyQA, whose high-agreement-error regime is most distinct from the calibration mixture), while Mondrian

per-task calibration never violates (0/67 admitted) and preserves higher coverage. Mondrian calibration (Vovk et al., 2005), calibrating each task category against its own held-out data, is the principled deployment remedy when cross-task exchangeability fails.

E.14 Answer-Only Verifier Ablation

We test whether the verifier’s gain depends on seeing the candidate *rationale* or only the candidate *answer* via an ablation in which the GPT-5.4-mini verifier sees the rationale field stripped (prompt: “Question: q; Candidate answer: $\hat{y}$; Is the candidate answer correct? Reply with exactly one word: yes or no.”) on a stratified subset of 100 questions per (generator, dataset) cell on the no-Mistral verifier-availability subcohort (Qwen2.5-7B, Qwen2.5-14B, Llama-3.1-8B, GPT-5.4-mini), giving $n = 1,998$ questions, $K_v = 4$ queries each, 7,992 API calls total, and yielding the per-question score s_v^{ao} . The triple-coverage subset on which we compare to the rationale-included variant and to the critique variant (Appendix E.15) contains $n = 1,858$ questions (Table 15).

The answer-only verifier diverges substantially from the rationale-anchored original: the rank correlation collapses to $\rho = 0.186$, and AURC degrades from 7.4% to 11.0%. With $K_v = 4$, the empirical error rate at $s_v^{\text{ao}} = 1.0$ is 7.7% over 272 questions, close enough to $\alpha = 0.10$ that the Bonferroni-corrected upper bound clears it on no split, so no LTT-validated threshold is admitted in 0/10 pooled split-conformal trials. Wired into the RISC ensemble $(\rho + s_v^{\text{ao}})/2$, AURC rises from 5.60% (original ensemble) to 8.01% and no valid LTT threshold is found in any of 10 splits, against 10/10 for the original ensemble. The rationale therefore carries the signal: stripping it changes the verifier’s behavior qualitatively, so the gains of Section 5 hold conditional on the verifier seeing the chain. We retain the rationale-included prompt as the operative variant.

E.15 Critique-Style Verifier Prompt Variant

The binary yes/no prompt may be too lossy: the verifier produces no intermediate token before committing to its decision. We test a *critique* prompt that asks for explicit step-by-step error identification before the yes/no token: “Question: q; Candidate reasoning: r; Candidate answer: $\hat{y}$; Critique the reasoning step by step. Identify any errors. Then conclude with exactly one word: yes if the answer

Comparison	k_S/n_S	k_D/n_D	$\Delta\%$ (95% CI)	p -value	sig. @0.0125?
StrategyQA vs. GSM8K	199/842	18/899	+21.63 [18.66, 24.71]	$<10^{-30}$	yes
StrategyQA vs. MATH500	199/842	58/860	+16.89 [13.57, 20.21]	$<10^{-21}$	yes
StrategyQA vs. ARC	199/842	91/1470	+17.44 [14.38, 20.63]	$<10^{-30}$	yes
StrategyQA vs. CommonsenseQA	199/842	105/1298	+15.54 [12.36, 18.83]	$<10^{-22}$	yes

Table 13: Cross-dataset pairwise comparison of conditional error at $\rho \geq 0.9$ pooled over five generators. Newcombe hybrid score 95% CIs for the difference of two proportions and pooled-variance two-proportion z -tests. Bonferroni correction at family-wise $\alpha=0.05$ across 4 comparisons gives a per-test threshold of 0.0125.

Regime	Violation rate at $\alpha=0.10$	Mean coverage
Pooled cross-task ($\mathcal{D}_{\text{cal}} = \text{other tasks}$)	34.2%	57.9%
Mondrian per-task ($\mathcal{D}_{\text{cal}} = \text{target 50\%}$)	0.0%	72.8%

Table 14: LTT risk-control validity under pooled vs. Mondrian calibration, averaged over 5 targets $\times R=20$ split replicates per regime (violation rates among admitted thresholds; admittance is reported separately). Violation rate is the fraction of admitted replicates whose held-out test risk exceeds the nominal $\alpha=0.10$. Pooled regime admits 76/100 thresholds with violations concentrated on StrategyQA (19/19 of admitted) and CommonsenseQA (7/19). Mondrian admits 75/100 thresholds with 0 violations on every target it certifies, and it certifies none on StrategyQA, where the selective-error floor sits above α .

Verifier	AURC $\downarrow$	Cov@ $\alpha=0.10$ $\uparrow$	Spearman
Original (s_v , with rationale)	7.38	83.6%	1.000
Answer-only (s_v^{ao})	11.01	36.1%	0.186
Critique (s_v^{crit})	7.38	88.7%	0.580

Table 15: Sole-acceptance metrics on the triple-coverage subset ($n = 1,858$). Cov@ $\alpha=0.10$ is the empirical maximum coverage at which selective error stays ≤ 0.10 . Spearman is the rank correlation of the per-question verifier score with the original.

is correct, no if it is wrong.” We bound output to 200 tokens to control cost. The same stratified 100-per-cell subset is used ($n = 1,998$ questions, $K_v = 4$, 7,992 API calls), yielding s_v^{crit} (Table 16).

Ensemble	AURC $\downarrow$	Cov@ $\alpha=0.10$ $\uparrow$	LTT obs. err	LTT cov
$(\rho + s_v)/2$ (original)	5.60	87.2%	6.12%	68.8%
$(\rho + s_v^{\text{crit}})/2$	5.61	90.1%	5.69%	74.3%

Table 16: Ensemble-level comparison on the triple-coverage subset ($n = 1,858$). LTT columns: pooled 50/50 split-conformal calibration at $\alpha = 0.10$, $R = 10$ random splits, Bonferroni correction, conf = 0.90. Both ensembles admit a valid threshold in 10/10 splits.

The critique-style prompt is *closer in distribution* to the original (Spearman $\rho = 0.580$, $p < 10^{-100}$) than the answer-only variant ($\rho = 0.186$), and on the triple-coverage subset the two are indistinguishable on AURC (5.61 vs. 5.60). The critique variant gains a little on both coverage measures, reaching 90.1% empirical Cov@ $\alpha=0.10$ against 87.2% and 74.3% LTT-admitted coverage against

68.8%, and both admit a valid threshold on every split. Asking for a step-by-step critique before the yes/no token therefore neither helps nor hurts the ranking, and buys a modest amount of certified coverage. We retain the original prompt as the operative variant because the cost per query is lower by $\approx 8\times$ (5 vs. ≈ 140 output tokens) and the AURC difference (0.01 percentage points) is far inside the bootstrap confidence interval of zero. Practitioners willing to spend the additional output tokens may prefer the critique variant for the extra certified coverage.

E.16 Decoding-Parameter Sensitivity

We test whether the CBW@ τ and selective-error numbers depend on the decoding configuration. The headline configuration uses temperature $T = 0.7$ with top- $p = 0.95$ for the open-weight generators. Here we ablate $T \in \{0.3, 0.7, 1.0\}$ at fixed top- $p = 0.95$ on two open-weight generators (Qwen2.5-7B-Instruct, Llama-3.1-8B-Instruct). We re-sample $K = 16$ chains per question on a fixed $N = 200$ -question stratified subset (seed-42) of two contrastive datasets: StrategyQA (the high-CBW headline) and GSM8K (the low-CBW counterpoint). Same prompts, same answer-extraction rules, same per-dataset max_tokens. The $T = 0.7$ rows reuse the existing $K=16$ samples to avoid double-counting (Table 17).

Generator	Dataset	T	Err@0.8	Err@0.9
Qwen2.5-7B	GSM8K	0.3	3.8%	2.8%
Qwen2.5-7B	GSM8K	0.7	2.7%	1.7%
Qwen2.5-7B	GSM8K	1.0	3.3%	1.2%
Llama-8B	GSM8K	0.3	2.5%	2.9%
Llama-8B	GSM8K	0.7	1.9%	1.5%
Llama-8B	GSM8K	1.0	1.4%	0.8%
Qwen2.5-7B	StrategyQA	0.3	33.5%	31.3%
Qwen2.5-7B	StrategyQA	0.7	33.5%	30.6%
Qwen2.5-7B	StrategyQA	1.0	34.8%	29.9%
Llama-8B	StrategyQA	0.3	26.5%	22.2%
Llama-8B	StrategyQA	0.7	23.7%	19.1%
Llama-8B	StrategyQA	1.0	20.5%	15.1%

Table 17: Decoding-parameter sensitivity on a 200-question StrategyQA / GSM8K subset across $T \in \{0.3, 0.7, 1.0\}$. Err@0.8 is the conditional error among questions with $\rho \geq 0.8$; Err@0.9 is the analogous quantity at $\rho \geq 0.9$. Note that for binary-answer StrategyQA the two metrics largely coincide because most $\rho \geq 0.8$ instances also satisfy $\rho \geq 0.9$.

The pattern is robust to temperature. The headline pattern is robust to temperature. Across all six $\langle \text{generator}, T \rangle$ cells, StrategyQA’s high-agreement error (Err@0.8) stays in the 20.5–34.8% range while GSM8K’s stays in the 1.4–3.8% range. The gap between the two datasets exceeds a factor of five at every temperature. The two extremes within a $\langle \text{generator}, \text{dataset} \rangle$ cell differ by at most 6.0pp on StrategyQA-Llama-8B (the largest swing) and by at most 1.1pp on GSM8K. Conditional error at $\rho \geq 0.8$ on StrategyQA remains substantial across the full temperature range (33.5–34.8% on Qwen2.5-7B; 20.5–26.5% on Llama-8B), consistent with the reading that the consistent-but-wrong mode is not a sampling-temperature artifact. None of these swings rewrite the conclusions of Section 4: StrategyQA remains structurally CBW-prone across temperatures, GSM8K’s high-confidence error rate stays low across temperatures, and the qualitative ordering is unchanged. We retain $T = 0.7$ as the headline configuration while reporting this sweep as evidence that the RISC framing is not specific to a single sampling temperature.

E.17 Cross-Vendor Verifier Ablation

Methodology. To test whether the verifier signal s_v depends on the particular vendor (OpenAI GPT-5.4-mini), we run two Anthropic Claude models (Anthropic, 2025, 2026) (Claude Haiku 4.5, the smaller, faster Claude in the 4 family, and Claude Sonnet 4.6, the more capable Claude in the same family) as verifiers on the full

5-generator cohort (Qwen2.5-7B, Llama-3.1-8B, Mistral-7B-Instruct, Qwen2.5-14B, GPT-5.4-mini; $N=8,222$), the same cohort used in Section 5 (Table 18). Each verifier was queried with $K_v=4$ at temperature=0.5 using the identical prompt of Appendix C. The Anthropic API default (1.0) was overridden explicitly. We compute pooled risk-coverage with bootstrapped AURC and coverage at $\alpha=0.10$ (1,000 resamples) and LTT validity over $R=10$ random cal/test splits at $\delta=0.10$.

The framework is vendor-agnostic. The framework is *vendor-agnostic*: swapping OpenAI’s GPT-5.4-mini for Anthropic’s Sonnet-4.6 changes pooled AURC by only -0.23pp ($7.16 \rightarrow 6.93$), and Haiku-4.5 by $+0.41\text{pp}$ ($7.16 \rightarrow 7.57$). All three single-vendor combinations preserve LTT validity at $\alpha=0.10$ (empirical violation rate at most 2.1% over $R=10$ random splits, within the $\delta=0.10$ budget), and the $(\rho+s_v)/2$ score dominates the SC-only baseline (9.41% AURC) regardless of which vendor supplies s_v . Within Anthropic, Sonnet 4.6 marginally outperforms Haiku 4.5 on AURC (6.93 vs 7.57), but both fall within bootstrap CI of each other and of the GPT-5.4-mini reference, so we read this as cross-vendor parity. The Spearman correlations between verifiers ($\rho_{H,G}=0.643$, $\rho_{S,G}=0.570$, $\rho_{H,S}=0.595$) are moderate: verifiers disagree often enough that simple averaging extracts new signal. The 3-vendor mean $\bar{s}_v^{H,S,G}$ combined with ρ achieves 6.22% AURC, a 0.71pp absolute improvement over the best single-vendor variant and a 3.19pp improvement over the SC-only baseline, while still satisfying LTT validity at $\alpha=0.10$. This is consistent with the broader literature on judge-ensembling and indicates a practical route to further risk-controlled gains in deployments where calling multiple verifiers is acceptable.

E.18 OpenAI GPT-4o as a Temperature-Controllable Verifier

Methodology. GPT-5.4-mini is fixed by the OpenAI API at $T=1.0$, while every other verifier in our cohort (Phi-3.5-mini-instruct, GPT-4o-mini, Claude Haiku 4.5, Claude Sonnet 4.6) runs at $T=0.5$. To make a temperature-only explanation for the cross-vendor comparison unlikely, we evaluate OpenAI GPT-4o (OpenAI, 2024b) as a temperature-controllable verifier at $T=0.5$, with the same prompt and same r^* rule. The matched-cohort numbers in Table 19 are computed on the full 5-generator cohort ($N=8,222$, same as Ap-

Variant	AURC (%)	Cov@ $\alpha=0.10$ (%)	LTT valid?	Spearman vs s_v^G
$(\rho + s_v^H)/2$, Haiku-4.5	7.57 [6.91, 8.26]	75.1 [70.5, 80.0]	yes (2.1%)	0.643
$(\rho + s_v^S)/2$, Sonnet-4.6	6.93 [6.30, 7.55]	80.5 [78.0, 83.4]	yes	0.570
$(\rho + s_v^G)/2$, GPT-5.4-mini (paper)	7.16 [6.51, 7.78]	75.5 [71.7, 78.3]	yes	1.000
$(\rho + s_v^{H,S,G})/2$, 3-vendor mean	6.22 [5.65, 6.79]	81.2 [79.1, 83.0]	yes	0.876
ρ only (SC baseline)	9.41 [8.63, 10.19]	71.6 [67.2, 75.7]	yes	—

Table 18: Cross-vendor verifier ablation on the full 5-generator cohort ($N=8,222$). Bootstrapped 95% CIs in brackets. “LTT valid?” indicates LTT validity at $\alpha=0.10$, $\delta=0.10$ over $R=10$ random cal/test splits; the empirical violation rate is 0% except where noted (parenthetical). The Spearman column reports rank correlation between each verifier’s scalar score and the GPT-5.4-mini score on the same cohort.

pendix E.17) with all four verifier outputs (GPT-4o-mini, GPT-4o, GPT-5.4-mini, Claude Sonnet 4.6) available on every row.

Temperature is not the explanation. GPT-4o at $T=0.5$ attains 8.21% AURC, within 1.05pp of GPT-5.4-mini’s 7.16% at $T=1.0$ with overlapping bootstrap CIs. The forced- $T=1.0$ result is therefore not separated from the controllable- $T=0.5$ result by a temperature gap; the remaining differences reflect model identity. Cross-vendor agreement at matched T : GPT-4o (OpenAI) and Claude Sonnet 4.6 (Anthropic), both at $T=0.5$, achieve AURCs of 8.21 and 6.93 respectively with matched LTT validity, supporting cross-vendor interchangeability when temperature is matched (model identity still drives the remaining gap). Their 50/50 mean reaches 6.75%, a modest 0.18pp improvement over Sonnet 4.6 alone and within bootstrap CI of the 3-vendor ensemble of Appendix E.17, consistent with the saturating multi-vendor behavior reported there.

E.19 Verifier Temperature Sensitivity (GPT-4o-mini)

Methodology. In our cross-vendor ablation (Appendix E.17), GPT-5.4-mini is run at the provider-forced $T=1.0$ while Phi-3.5-mini-instruct, GPT-4o-mini, Claude Haiku 4.5, and Claude Sonnet 4.6 all use $T=0.5$. To make an explanation based solely on temperature unlikely, we sweep the verifier sampling temperature $T \in \{0.0, 0.3, 0.5, 0.7, 1.0\}$ using GPT-4o-mini, the lowest-cost temperature-controllable OpenAI verifier in our setup, while holding everything else fixed: the verify prompt is identical at the byte level to Appendix C, $K_v=4$, the rationale field r^* is the same majority-answer representative used throughout the experiments, and the cohort is a stratified subset of 100 questions per cell on Qwen2.5-14B/Llama-3.1-8B/GPT-5.4-mini generators $\times$ {ARC-C, CSQA, GSM8K,

MATH500, StrategyQA} ($N = 1,498$). Only temperature varies across the 5 runs; bootstrap CIs use 1000 paired resamples (Table 20).

Synthesis: temperature is not a confound.

The ensemble AURC varies by only $\Delta = 0.26$ percentage points across all five temperatures (range [7.82, 8.08]%), and every pairwise 95% bootstrap CI overlaps the $T=0.5$ point estimate of 7.98%. Per-question yes-rates are highly stable in rank: Spearman $\rho_S \geq 0.948$ between the $T=0.5$ baseline and every other temperature, with the consecutive-temperature correlations $\rho_S(T=0.7, T=1.0) = 0.949$ and $\rho_S(T=0.3, T=0.5) = 0.968$ both exceeding the cross-vendor ρ_S figures reported in Appendix E.18. The most pertinent contrast for the temperature-confound concern is $T=0.5$ versus $T=1.0$: the former gives 7.98% AURC, the latter 7.83%, and the paired bootstrap on $T=1.0$ minus $T=0.5$ AURC has 95% CI $[-0.59, +0.24]$ percentage points containing zero, so the bootstrap does not detect a difference on this cohort. This is direct evidence that GPT-5.4-mini’s headline AURC in Appendix E.17 is *not* a $T=1.0$ artefact: within a single OpenAI verifier model, going from $T=0.5$ to $T=1.0$ moves AURC by less than the cross-temperature noise band. The remaining gap between GPT-4o-mini and GPT-5.4-mini in our verifier comparison must therefore be attributed to verifier capability.

E.20 Misleading-Rationale Verifier Ablation

The companion to the answer-only ablation (Appendix E.14) asks the converse: if the verifier is shown a *misleading* rationale that does not justify the candidate answer, does it still accept the candidate? We perform a paired swap experiment on StrategyQA and GSM8K, the two cells with cleanly extractable majority answers. We draw on the no-Mistral verifier-availability subcohort, restricted to StrategyQA and GSM8K over Qwen2.5-

Verifier	T	AURC (%)	Cov@ $\alpha=0.10$ (%)	LTT valid?	ρ_S vs. GPT-5.4-mini
GPT-4o-mini	0.5	9.05 [8.35, 9.79]	63.6	yes	0.505
GPT-4o	0.5	8.21 [7.51, 8.89]	70.9	yes	0.588
GPT-5.4-mini	1.0	7.16 [6.51, 7.78]	75.5	yes	—
Claude Sonnet 4.6	0.5	6.93 [6.30, 7.55]	80.5	yes	0.570
(40 + SONNET)/2	0.5	6.75 [6.15, 7.34]	80.5	yes	—

Table 19: RISC ensemble $(\rho + s_v)/2$ on the matched pooled cohort. Brackets are 95% bootstrap CIs.

T	AURC (%) $\downarrow$	Cov@ $\alpha=0.10$ $\uparrow$	Spearman vs. $T=0.5$
0.0	8.07 [6.49, 9.71]	67.2% [60.9, 74.8]	0.961
0.3	8.08 [6.51, 9.68]	66.9% [60.5, 72.8]	0.968
0.5	7.98 [6.40, 9.66]	67.5% [59.8, 74.1]	1.000
0.7	7.82 [6.32, 9.48]	66.9% [60.8, 79.5]	0.965
1.0	7.83 [6.27, 9.50]	67.4% [60.1, 74.4]	0.948

Table 20: GPT-4o-mini verifier temperature sweep on the matched cohort ($N = 1,498$). Score is the RISC ensemble $(\rho + s_v^T)/2$. AURC and Cov@ $\alpha=0.10$ are pooled across the 15 (gen, ds) cells; brackets are 95% bootstrap CIs from 1000 resamples. Spearman is the per-question yes-rate rank correlation against the $T=0.5$ baseline (so the baseline row is 1.000 by definition).

7B, Qwen2.5-14B, Llama-3.1-8B, and GPT-5.4-mini. From each cell we keep the first 100 questions with high agreement ($\rho \geq 0.8$) and a successfully extracted majority answer, giving $n = 781$ paired questions. The two GPT-5.4-mini cells contribute only 82 and 99 qualifying questions. For each question Q with majority answer A and rationale R , we sample a partner question Q' from the same dataset whose own majority answer differs from A , and build the misleading prompt by substituting R' for R while keeping Q and A unchanged. We then query GPT-4o-mini at temperature 0.5 with $K_v = 4$ for a per-question score s_v^{mis} (3,124 API calls; \$0.12). The matched score s_v from the GPT-4o-mini verifier on the unaltered (rationale, answer) pair is taken from the apiverifier labels released with the cohort (Table 21).

The verifier is overwhelmingly grounded in the rationale. Replacing the candidate’s rationale with one taken from a different question reduces the yes-rate from 74.3% to 1.1%, with the per-question score dropping by 0.73 on average and flipping in 77.0% of pairs (it rises in only 0.1%). The Spearman rank correlation between the two scores is 0.053, so swapping the rationale leaves a score that is essentially independent of the original one. What survives is not an inverted signal but no signal: the misleading score correlates with correctness at $\rho_s = +0.045$ against $+0.061$ for the original score on this high-agreement subset, and the ensemble

$(\rho + s_v^{\text{mis}})/2$ lands within noise of ρ alone (AURC 8.77% vs. 9.04%). Appendix E.14 points the same way, since stripping the rationale degraded AURC from 7.4% to 11.0%. The verifier is reading the rationale. The gains reported in Section 5 therefore come from rationale-conditioned verification.

E.21 LTT Split-Sensitivity (R=10 vs R=50)

The $R=10$ random calibration/test splits of Section 5.5 carry their own noise, and this appendix measures it. We replicate the calibration-validity analysis at $R=50$ (seeds 0–49) on the identical operative score $s = \frac{1}{2}\rho + \frac{1}{2}s_v$. Everything else is held fixed: the Bonferroni–Clopper–Pearson construction with per-threshold confidence $1 - \delta/|\mathcal{T}|$ at $\delta=0.10$ and $\mathcal{T} = \{s(q) : q \in \mathcal{D}_{\text{cal}}\}$, and the 50/50 cal/test split. The pool is the no-Mistral verifier-availability subcohort, $N=6,724$ valid instances over 20 cells. We exclude the Mistral-7B cells here because the finding, that $R=10$ already lies inside the $R=50$ 95% CI, does not depend on the larger pool. The only manipulated quantity is the number of seed replicates, so any difference between $R=10$ and $R=50$ is attributable to Monte-Carlo split noise.

The two sample sizes give the same conclusion. At $\alpha=0.10$ the violation rate is 0.00% under both regimes, since no admitted split exceeded the nominal risk on its test fold. At $\alpha=0.20$ it drops from 0.67% at $R=10$ to 0.41% at $R=50$, a difference of 0.26pp that sits well inside the $R=50$ 95% CI of [0.00%, 0.86%], with a split-level standard error of 0.23%. Mean test coverage and mean test selective error also agree to within ≤ 0.34 pp across the two regimes. Equivalently, the $R=10$ split-level mean lies inside the $R=50$ 95% CI for both target risks. We report the $R=10$ figures in Section 5.5 and the $R=50$ replication here for completeness on split-noise. The replication operates on the per-question scores already computed in Section 5 and adds no API calls or GPU inference (wall-time 24s).

Setting	yes-rate $\bar{s}_v$	$\rho + s_v$ ensemble AURC	Cov@ $\alpha=0.10$	LTT@ $\alpha=0.10$
Original (s_v , true rationale)	74.3%	9.17 [6.74, 11.87]	65.2% [38.9, 76.3]	0/10
Misleading (s_v^{mis} , swap)	1.1%	9.34 [6.90, 12.15]	70.0% [3.8, 96.0]	0/10
$\Delta = s_v - s_v^{\text{mis}}$	mean 0.731, median 1.000, $\Delta > 0$ in 77.0%, $\Delta = 0$ in 22.9%, $\Delta < 0$ in 0.1%			
Spearman(s_v, s_v^{mis})	0.053 (rank order is destroyed; swapping the rationale produces an essentially independent score)			

Table 21: Misleading-rationale paired comparison on the high-agreement StrategyQA + GSM8K subset ($n = 781$). AURC and Cov@ $\alpha=0.10$ are computed on the pooled subset using the same trapezoidal-AURC pipeline as Appendix E.2 (1,000 bootstrap replicates, 95% percentile CI). LTT@ $\alpha=0.10$ counts pooled 50/50 split-conformal trials in which the calibrated threshold yields test error ≤ 0.10 (10 random splits, seed 0). The ensemble $(\rho + s_v)/2$ row uses each verifier in turn.

α	metric	$R=10$	$R=50$	$R=50$ split-level 95% CI
0.10	test selective error	3.58%	3.54%	—
	test coverage	80.12%	79.93%	—
	violation rate	0.00%	0.00%	[0.00%, 0.00%]
0.20	test selective error	8.16%	8.15%	—
	test coverage	87.55%	87.49%	—
	violation rate	0.67%	0.54%	[0.14%, 1.08%]

Table 22: LTT calibration validity under $R=10$ vs $R=50$ random 50/50 cal/test splits, pooled over the 20-cell cohort and target risk levels $\alpha \in \{0.10, 0.20\}$. The 95% CI is computed on the seed-level violation rate from the $R=50$ run, where each seed contributes one across-cells violation rate (normal-approximation, $\text{SE} = \sigma/\sqrt{R}$, lower bound clipped at 0). All $R=10$ point estimates fall within the $R=50$ 95% CI; mean coverage and selective error agree to within 0.34pp.

E.22 Denominator Sensitivity: Minimum Valid Extractions

The majority ratio $\rho(q)$ uses valid extractions in the denominator, so a question with $K_{\text{valid}} < K=16$ has a coarser-grained ρ that could in principle inflate or depress CBW@0.8. We test this by recomputing per-cell CBW@0.8 restricted to questions with $K_{\text{valid}} \geq k_{\min}$ for $k_{\min} \in \{1, 12, 14, 16\}$, and taking the cell-mean across the five generators per dataset (Table 23).

Dataset	$k \geq 1$	$k \geq 12$	$k \geq 14$	$k \geq 16$
GSM8K	2.13%	2.13%	2.15%	2.10%
StrategyQA	20.81%	20.66%	20.44%	20.27%
MATH500	3.36%	1.49%	1.32%	0.78%
ARC	6.15%	6.14%	6.17%	6.15%
CSQA	9.63%	9.63%	9.63%	9.60%
Cohort N	8,222	7,692	7,466	6,854

Table 23: Mean CBW@0.8 across five generators per dataset under minimum-valid-extraction filters $K_{\text{valid}} \geq k_{\min}$. The 83.4% of pooled questions have $K_{\text{valid}}=16$; the remainder are concentrated in MATH500 cells (extraction failures on the LaTeX sub-distribution before SymPy normalization succeeds).

Four of five datasets are essentially flat across the $k_{\min}$ sweep (within $\sim 0.5\text{pp}$), so the headline CBW@0.8 values for GSM8K, StrategyQA, ARC, and CSQA are not artefacts of inflated ρ on small- K_{valid} questions. MATH500 drops substantially at strict $k_{\min}=16$ (from 3.36% to 0.78%), which is consistent with the parseability-conditioned framing for MATH500 in Section 4: items where several samples emit unrecoverable LaTeX expressions tend to also be the hardest items, so restricting to $K_{\text{valid}}=16$ removes much of MATH500’s high-agreement-wrong tail. Crucially, the dataset ranking (StrategyQA $\gg$ CSQA $>$ ARC $\gtrsim$ MATH500/GSM8K) is preserved at every $k_{\min}$ tested. The qualitative finding that StrategyQA is the high-CBW outlier is therefore not driven by extraction-failure-induced denominator effects.

E.23 Question-Level Cluster Bootstrap

Because the same questions are evaluated by all five generators, the $N=8,222$ (question, generator) instances are not independent under within-question generator effects. We complement the paired bootstrap underlying Figure 4 with a question-level cluster bootstrap: the resample unit is the (*dataset, question id*) cluster, giving 2,098 unique questions across the five datasets, and all five generators’ realizations of the same question move together in each resample. We use 2,000 resamples and report the same three reference coverages (Table 24).

All three gains survive clustering ($\Delta < 0$ at 95% under both schemes). Widening is mildest at high coverage, where the interval barely moves, and largest at moderate coverage, where the cluster interval reaches $[-1.58, -0.13]$ against the paired $[-1.46, -0.30]$. Dependence between the five generators that answer the same question therefore costs precision without changing any sign.

Coverage	Instance-level CI	Cluster CI
0.50	$[-3.75, -2.29]$	$[-3.92, -2.09]$
0.71	$[-1.46, -0.30]$	$[-1.58, -0.13]$
0.91	$[-0.91, -0.32]$	$[-0.94, -0.31]$

Table 24: 95% bootstrap CIs for $\Delta = \text{RISC error} - \text{SC-only error}$ at three reference coverages, under instance-level resampling ($N=8,222$) versus question-level cluster resampling ($n_{\text{cluster}}=2,098$). 2,000 resamples each. The conservative-coverage gain (0.50) is robust under both schemes. The high-coverage gain (0.91) excludes zero under both paired and cluster bootstrap, as does the moderate-coverage gain (0.71).

E.24 Where Does RISC Abstain?

A natural diagnostic question for a selective-prediction system is whether abstentions land on *hard* or *wrong* instances (where deferral is operationally useful) or on easy correct instances (where deferral wastes coverage). This appendix reports a *pooled-threshold descriptive diagnostic*: a single LTT-calibrated τ^* is computed on the pooled cohort and applied to every (question, generator) instance, yielding a single accept/abstain split. This is a different protocol from the in-domain LTT validity numbers of Section 5.5, which average coverage and error over the admitted subset of $25 \times R=10$ cell-splits, at 23% admission under Bonferroni at $\alpha=0.10$. The headline numbers here, 69.3% acceptance and 8.36% selective error on the pooled cohort, are therefore not directly comparable to the 78.2% coverage and 3.55% error reported there for the operative variant. We provide a descriptive pooled-threshold diagnostic on the same pooled $N=8,222$ cohort used in Section 5 (5 datasets $\times$ 5 generators = 25 cells, the GPT-5.4-mini verifier available on every cell) with the operative RISC score $s = \frac{1}{2}\rho + \frac{1}{2}s_v^{\text{GPT-5.4-mini}}$. We compute the LTT-calibrated operative threshold τ^* on the pooled cohort following the protocol of Section 5.5 (Clopper-Pearson upper bound on a 50/50 cal/test split at confidence $\delta=0.10$, $R=10$ random splits, take the median across LTT-valid splits), obtaining $\tau^*=0.7500$ at $\alpha=0.10$ (valid in 10/10 splits). We then label each pooled question as *abstained* ($s < \tau^*$) or *accepted* ($s \geq \tau^*$) and break the abstained pool down per task and per ρ -bucket. We use the pooled τ^* as the per-cell operating threshold. Per-cell calibration sets, at $n \sim 100\text{--}500$, fall below the size at which the Bonferroni-corrected Clopper-Pearson bound admits a non-trivial threshold at our α on most cells, matching the $n_{\text{cal}} \approx 150$

infeasibility regime of Section 5.5. The pooled τ^* is also the deployment-relevant quantity, since a practitioner calibrates once on a labeled mixture and then deploys.

The abstention pool is enriched for lower-confidence and higher-error instances. In-bucket abstention rates rise monotonically as self-consistency ρ falls, from 9.7% at $\rho=1$ to 100% at $\rho \leq 0.25$. The latter bucket necessarily abstains because $s \leq (0.25+1)/2 = 0.625 < \tau^*$. RISC therefore defers most often when the generator exhibits stronger disagreement. On higher-error tasks, abstentions are more error-targeted: 66.5% of MATH500 abstentions correspond to incorrect underlying answers, and CSQA and StrategyQA also exceed their task-level base error rates. On high-baseline-accuracy tasks such as ARC and GSM8K, most abstentions necessarily fall on correct underlying answers, since at $\sim 89\text{--}97\%$ base accuracy any uniform threshold concentrates the bulk of its abstentions on correct items. A practitioner deploying RISC at $\alpha=0.10$ on this mixture abstains on 30.7% of inputs. Of those, 49.8% (1,258/2,528) would have been wrong in expectation, a $2.36\times$ enrichment over the cohort base error rate of 21.1%. The accepted 69.3% reaches 91.6% accuracy, which is 8.36% selective error. These pooled figures describe the mixture; the per-cell, split-averaged LTT certificate appears in Section 5.5. The pattern matches the intent of the calibration: abstentions concentrate on disagreement (low ρ) and on tasks with high intrinsic difficulty (MATH500, StrategyQA, CSQA), while easy and self-consistent items pass through with little cost.

E.25 Statistical Test of Task vs. Generator Effects on CBW

Methodology. Section 4 reports that, within our cohort, the standard deviation of mean CBW@0.8 across the five datasets is approximately seven times the standard deviation across the five generators (numerically, 0.0671 vs. 0.0092, ratio $7.32\times$). To put this dominance on a formal footing we run a balanced two-way analysis of variance on the per-cell CBW@0.8. The two crossed fixed factors are task, at 5 levels (ARC, CommonsenseQA, GSM8K, MATH500, StrategyQA), and generator, at 5 levels (Qwen2.5-7B, Llama-3.1-8B, Qwen2.5-14B, Mistral-7B-Instruct, GPT-5.4-mini). Together they give the full $5 \times 5 = 25$ cells of our cohort. Each cell contributes a single CBW@0.8 value (count of questions with $\rho \geq 0.8$ and ma-

Task	N	Base err. (%)	Abstain (%)	Useful (%)	Wasteful (%)	Sel. acc. (%)
ARC	1,900	11.1	17.5	35.1	64.9	94.0
CSQA	1,900	20.1	31.6	39.5	60.5	88.9
GSM8K	1,300	10.5	17.1	46.8	53.2	96.9
MATH500	1,823	32.7	43.0	69.3	30.7	94.8
StrategyQA	1,299	31.4	45.3	43.6	56.4	78.7
Pooled	8,222	21.1	30.7	49.8	50.2	91.6

Table 25: Per-task abstention behavior at the LTT-calibrated operative threshold $\tau^*=0.7500$ ($\alpha=0.10$, $\delta=0.10$). This diagnostic uses a single pooled operating threshold and is not directly comparable to the per-cell, split-averaged LTT validity figures of Section 5.5. *Useful* is the fraction of abstentions whose underlying answer was wrong; *Wasteful* is the fraction whose answer was correct (these sum to 100%). *Selective accuracy* is the accuracy of the accepted fraction; selective error = 1–selective accuracy. Aggregated across all 2,705 abstentions, in-bucket abstention rates rise monotonically with disagreement ($\rho=1$ instances are abstained at the lowest rate and $\rho\leq 0.25$ instances are nearly always abstained), with the $\rho=1$ subset contributing the smallest share and the $\rho\leq 0.5$ subset contributing the largest.

jority incorrect, divided by the parseable cell size, with parse-failed MATH500 questions excluded as in Section 4.1). We fit $\text{CBW} \sim C(\text{task}) + C(\text{generator})$ with `statsmodels ols` and read out Type-II sums of squares via `anova_lm`. Effect size is reported as $\eta^2 = \text{SS}_{\text{factor}}/\text{SS}_{\text{total}}$. The interaction term is not estimable in the saturated 5×5 design with one observation per cell and is absorbed into the residual (Table 26).

Factor	df	F	p	η^2
Task	4	50.69	6.89×10^{-9}	0.911
Generator	4	0.95	0.462	0.017
Residual	16	—	—	0.072

Table 26: Two-way ANOVA on per-cell CBW@0.8 ($N=25$ cells, 5 tasks $\times$ 5 generators). Type-II sums of squares; η^2 is the proportion of total CBW@0.8 variance explained by each factor.

Result. The task factor accounts for $\eta_{\text{task}}^2=91.1\%$ of total CBW@0.8 variance, the generator factor for only $\eta_{\text{gen}}^2=1.7\%$, and the residual 7.2%. The ratio $\eta_{\text{task}}^2/\eta_{\text{gen}}^2 \approx 54$ corroborates the SD-ratio diagnostic ($\approx 7\times$, which compares standard deviations; squaring gives $\approx 54\times$, consistent within rounding) reported in Section 4. Because the design has one observation per (task, generator) cell, the task–generator interaction is absorbed into the residual. We therefore read this two-way decomposition as a descriptive variance partition, and note that F -statistic p -values $p_{\text{task}}=6.89 \times 10^{-9}$ and $p_{\text{gen}}=0.462$ are reported as descriptive accompaniments under that caveat. Within this 5×5 cohort essentially all between-cell CBW@0.8 variation is task-driven. The same diagnostic

restricted to the four-generator subcohort that excludes Mistral-7B gives $\eta_{\text{task}}^2=0.910$ and $\eta_{\text{gen}}^2=0.023$, so task-dominance is not specific to the particular five-generator pooling. We retain the cohort-scoped phrasing of the Limitations section: the test is conditional on our 5×5 design and does not extrapolate beyond it.

Question-clustered bootstrap. Finite per-cell question counts inject sampling variability into a partition built on one observation per cell. To quantify how much of it survives, we run a clustered bootstrap over $B=1,000$ replicates. Each replicate resamples questions within each (task, generator) cell with replacement, recomputes CBW@0.8 per cell, refits the two-way ANOVA, and records the resulting η^2 shares. The bootstrap distribution of η_{task}^2 has mean 0.869 with 95% CI [0.796, 0.929], and the distribution of η_{gen}^2 has mean 0.025 with 95% CI [0.007, 0.052]. The task SD / generator SD ratio on per-bootstrap cell means has mean $6.46 \times$ with 95% CI [3.95, 11.15]. The dominance is uniform across the bootstrap: $\Pr[\eta_{\text{task}}^2 > \eta_{\text{gen}}^2]=1.000$ and $\Pr[\eta_{\text{task}}^2 > 5\eta_{\text{gen}}^2]=1.000$ over the 1,000 replicates, so task-dominance is not an artifact of the single-observation-per-cell point estimate but is preserved under question-level resampling within cells. The no-Mistral four-generator subcohort gives consistent bootstrap values ($\eta_{\text{task}}^2=0.867$ [0.782, 0.941], $\eta_{\text{gen}}^2=0.031$ [0.008, 0.066]).

E.26 Per-Generator Detail: Mistral-7B-Instruct

Setup. This appendix isolates the Mistral-family member of our cohort, Mistral-7B-Instruct-v0.3, and reports its per-dataset numbers alongside the

mean of the other four generators (Qwen2.5-7B / 14B, Llama-3.1-8B, GPT-5.4-mini). Mistral runs the same main-experiment pipeline at fixed $K=16$, $T=0.7$, and $\text{top-}p=0.95$. It draws $K=16$ chains per question on all five benchmarks (StrategyQA, GSM8K, MATH500, ARC-Challenge, CSQA), over the same seed-42 question subsets, with the same per-dataset max_tokens budgets, the same answer-extraction rules, and the same GPT-5.4-mini API verifier at $K_v=4$. Its five cells complete the $5 \times 5 = 25$ -cell cohort matrix (Table 27).

Where Mistral conforms and where it strains.

Mistral conforms to the task-structured CBW pattern of Section 4. StrategyQA is its high-CBW outlier (CBW@0.8=21.7% vs. a 20.60% mean over the other four generators on the same dataset) and GSM8K remains low-CBW (CBW@0.8=1.0% vs. 2.42%). The qualitative ordering “open-ended factual reasoning produces consistent-but-wrong errors at higher rates than mechanically verifiable arithmetic” is therefore not restricted to the Qwen / Llama / GPT-5 portion of the cohort. Mistral runs at lower overall SC accuracy than the rest of the cohort (mean SC@ $K=64.1\%$ vs. 78.9% on the parseable cohort), with MATH500 most affected (SC@ $K=23.2\%$ on the parseable subset, only 7 high-agreement questions). We retain this cell in the cohort matrix because the high-disagreement, low-accuracy regime contributes to cross-family generality of the task-structured finding. RISC AURC on Mistral tracks its weaker base accuracy (e.g., 49.0% on MATH500). On the four standard-accuracy Mistral cells RISC stays within $1.0\text{--}2.5\times$ the corresponding cohort-mean AURC, again matching the task ranking. The Mistral cells also shift absolute pooled error rates where its base accuracy is much lower. MATH500 is the clearest case: the low parseable SC@ K raises the pooled SC-only error to 21.2%, while RISC’s $\Delta = -1.2\text{pp}$ on this cell matches the small-improvement pattern seen on the rest of the cohort.

E.27 Newer Qwen-Family Generator: Qwen3-8B

Setup. We additionally evaluate Qwen3-8B (Yang et al., 2025) (Qwen3 family, April 2025) on a 100-question-per-dataset subset, holding everything else fixed: same seed-42 question sampling, same five datasets (StrategyQA, GSM8K, MATH500, ARC-Challenge, CSQA), same $K=16$, $T=0.7$, $\text{top-}p=0.95$, same per-dataset

max_tokens , same answer-extraction rules, and the same GPT-5.4-mini API verifier with $K_v=4$. Qwen3 “thinking mode” is disabled to match the non-reasoning protocol of the Qwen2.5 cohort. This appendix reports the resulting per-dataset numbers alongside the mean of the five Section 4 generators (Qwen2.5-7B/14B, Llama-3.1-8B, Mistral-7B-Instruct, GPT-5.4-mini) on the same dataset (Table 28).

Qwen3-8B reproduces the task pattern.

Qwen3-8B reproduces the task-structured CBW pattern of the five-generator cohort: StrategyQA remains the high-CBW outlier (24.0% vs. cohort 20.8%), GSM8K/ARC/CSQA stay in single digits and within 1pp of the cohort means (2.0%/7.0%/8.0% vs. 2.1%/6.1%/9.6%), and the dataset rank order is preserved. The task-structured finding therefore generalizes from the 2024 Qwen2.5 family to the 2025 Qwen3 release; the five-generator cohort of Section 4 is retained as the controlled comparison.

E.28 Reasoning-Trained and Frontier Generators

Protocol. The three generators of Section 4.5 use the sampling protocol of Section 4.1 without modification: $K=16$ chains per question, temperature 0.6, $\text{top-}p$ 0.95, the prompts of Appendix C, and the same extraction and majority rules. DeepSeek-R1-Distill-Qwen-7B runs locally in bf16 with a 6,144-token generation limit. DeepSeek-V4-Pro runs through its provider API endpoint deepseek-v4-pro in thinking mode, sampled on 2026-07-09, with an 8,192-token limit, raised to 16,384 on GSM8K where long solutions were truncating. Claude Sonnet 4.6 runs through its provider API with extended thinking at a 2,048-token thinking budget. Question sets are drawn from the same public test splits used in Section 4 under a fixed seed.

Extraction validity. Truncation and extraction failures are negligible on every cell, so no cell rests on a thin denominator: truncated samples are 0.21% and 0.15% of the 4,800 GSM8K and StrategyQA samples for R1-Distill-Qwen-7B, 0.08% and 0.02% for DeepSeek-V4-Pro, and 0.00% for Claude Sonnet 4.6. Unextractable answers are at most 0.31% on any cell. Recomputing CBW with truncated samples dropped from the majority pool leaves every reported value unchanged (Table 29).

Dataset	N_{val}	SC@ K	CBW@0.8 (n_h)	ERR@0.9	RISC AURC	Cohort CBW@0.8
GSM8K	300	74.0%	1.0% (107)	0.0%	5.31%	2.42%
StrategyQA	300	66.7%	21.7% (223)	26.0%	19.76%	20.60%
MATH500	298	23.2%	0.0% (7) [†]	0.0%	49.01%	3.36%
ARC-C	300	82.0%	8.3% (229)	6.7%	5.21%	5.60%
CSQA	300	75.0%	11.3% (209)	10.3%	10.36%	9.20%

Table 27: Mistral-7B-Instruct-v0.3 per-dataset numbers, with the mean CBW@0.8 over the other four generators on the same dataset for comparison. N_{val} is the number of questions with parseable gold and at least one extractable sample; n_h is the count of questions with $\rho \geq 0.8$. RISC AURC is the area under the risk-coverage curve when ranking by the average of ρ and the GPT-5.4-mini verifier p_{yes} (lower is better). [†]Only 7 of the 298 parseable Mistral×MATH500 questions reach $\rho \geq 0.8$ due to high disagreement (SC@ K =23.2% on the parseable subset; $\sim 0.7\%$ of the 300 sampled MATH500 items remain unparseable under SymPy extraction); the 0% rate reflects this very small high-agreement subset.

Dataset	N_{val}	SC@ K	CBW@0.8 (n_h)	ERR@0.9	RISC AURC	Cohort CBW@0.8
GSM8K	100	92.0%	2.0% (91)	1.1%	0.74%	2.1%
StrategyQA	100	66.0%	24.0% (79)	29.6%	22.60%	20.8%
MATH500	84	86.9%	8.3% (71)	10.4%	4.46%	3.4%
ARC-C	100	93.0%	7.0% (98)	5.3%	2.93%	6.1%
CSQA	100	89.0%	8.0% (87)	9.1%	7.59%	9.6%

Table 28: Qwen3-8B (Qwen3 family, April 2025) per-dataset validation results on 100 questions per task, with the five-generator cohort mean CBW@0.8 from Section 4 on the same dataset for comparison. N_{val} is the number of questions with parseable gold and at least one extractable sample; n_h is the count with $\rho \geq 0.8$. RISC AURC is the area under the risk-coverage curve when ranking by $\frac{1}{2}(\rho + s_v^{\text{GPT-5.4-mini}})$ (lower is better).

Generator	Task	Acc	n_h	CBW	Err
R1-Distill-7B	GSM8K	87.0	225	4.0	3.11
	SQA	62.3	96	20.0	27.08
V4-Pro	GSM8K	96.7	289	1.67	1.38
	SQA	83.3	247	11.33	8.50
Sonnet 4.6	GSM8K	98.0	147	1.33	1.36
	SQA	85.0	177	11.50	11.86

Table 29: Per-cell detail behind Table 1. Acc is majority-vote accuracy at $K=16$, n_h the number of high-agreement instances ($\rho \geq 0.9$), CBW is CBW@0.8, and Err is Err@ $\rho \geq 0.9$; all in percent except n_h . SQA denotes StrategyQA.

Generator	Task	0.5	0.6	0.7	0.8	0.9	1.0
R1-Distill-7B	GSM8K	9.0	5.7	4.0	4.0	2.3	1.3
	SQA	37.7	33.0	24.0	20.0	8.7	5.0
V4-Pro	GSM8K	3.3	3.3	2.3	1.7	1.3	1.0
	SQA	16.7	15.3	12.0	11.3	7.0	5.3
Sonnet 4.6	GSM8K	2.0	1.3	1.3	1.3	1.3	1.3
	SQA	15.0	13.0	12.5	11.5	10.5	9.0

Table 30: CBW@ τ in percent across the threshold sweep $\tau \in \{0.5, \dots, 1.0\}$ for the three reasoning-trained generators. The StrategyQA-over-GSM8K ordering holds at every threshold on every generator, matching the cohort behavior of Section 4.6.

Reading the numbers. The change of generator family leaves two properties intact. The task ordering holds at every threshold in Table 30, and

the conditional error at $\rho \geq 0.9$ stays an order of magnitude apart between the two tasks even for the two frontier models. What changes is the level: StrategyQA CBW@0.8 falls from a cohort mean of 20.8% to 11.33% and 11.50% at the frontier, and the frontier GSM8K cells are already near their floor. Claude Sonnet 4.6 also serves as a verifier in Appendix E.17. The two roles use disjoint inputs, since the verifier scores chains produced by the cohort generators.

E.29 Generalizing the Cohort: BIG-Bench Hard

Setup. Our five-dataset cohort (GSM8K / StrategyQA / MATH500 / ARC-C / CSQA) covers a specific slice of reasoning genres. To test whether the task-structured CBW pattern extends to a more heterogeneous benchmark, we add BIG-Bench Hard (BBH; Suzgun et al., 2023), a 27-task diverse-reasoning benchmark, as a sixth dataset on the same pipeline: $K=16$ chains per question at $T=0.7$, top- $p=0.95$, max_tokens=384, the same chain-of-thought prompt, and the same GPT-5.4-mini API verifier with $K_v=4$. We sample 300 questions per open-weight generator (Qwen2.5-7B, Llama-3.1-8B, Qwen2.5-14B, Mistral-7B) across the 27 BBH sub-tasks. The BBH evaluation is re-

stricted to the open-weight subset of the cohort, which provides the cross-family check across Mistral, Qwen, and Llama at uniform per-cell sampling.

Where BBH lands. For comparability with BBH, the ARC-C and StrategyQA reference values below are computed on the same open-weight subset used here (four open-weight generators, excluding GPT-5.4-mini) at the same Err@0.8 threshold. Table 31 reports Err@0.9 separately. BBH places between StrategyQA and ARC-C in the CBW hierarchy of the cohort: its mean CBW@0.8 (8.4%) and mean Err@0.8 ($\sim 19.5\%$) are above ARC-C (5.6%/7.3%) and below StrategyQA (20.6%/27.5%). The task-structured pattern of Section 4 is preserved: the strongest generator (Qwen2.5-14B, SC@ $K=82.0\%$) has the lowest BBH RISC AURC (8.17%) and the weakest generator (Mistral-7B, SC@ $K=50.7\%$) the highest (36.24%); same-base-accuracy pairs (Qwen2.5-7B vs. Llama-3.1-8B, both $\sim 66\%$) differ by 0.73pp on RISC AURC, again matching the cohort behavior of Section 4. The verifier itself buys nothing on this dataset. Mean RISC AURC (19.98%) sits alongside mean SC-only AURC (19.79%), the two move in opposite directions across the four generators, and every per-cell bootstrap interval overlaps. BBH is a sixth task on which the per-task effects of Section 6 come out flat rather than positive. We do not see BBH overturning the “CBW is task-structured” claim within our cohort: across the $4 \times 6 = 24$ open-weight generator–dataset cells (the four open-weight generators on BBH and on the five Section 4 datasets), per-task variance dominates per-generator variance on $\rho \geq 0.8$ conditional error.

Scope of the BBH analysis. BBH is internally heterogeneous (boolean expressions, logical deduction, formal fallacies, geometric shapes, and more). We do not break out per-sub-task CBW outside this appendix because $300/27 \approx 11$ questions per sub-task is underpowered for stable per-cell rates. We report aggregate-level results and leave per-sub-task analysis, which would need a substantially larger per-sub-task sample, to future work.

E.30 Calibration-Budget Expansion: When Does $\alpha=0.05$ Become Feasible?

Setup. At $n_{\text{cal}} \approx 150$, the LTT calibration of Section 5.5 does not admit non-trivial thresholds at $\alpha = 0.05$. To distinguish a calibration-budget

artifact from a structural property of the underlying task, this appendix uses a larger per-cell pool of ~ 500 questions on GSM8K and StrategyQA across three open-weight generators (Qwen2.5-14B, Llama-3.1-8B, Mistral-7B-Instruct; Qwen2.5-7B is omitted from this sweep). We then sweep $n_{\text{cal}} \in \{75, 150, 250\}$ at $\alpha=0.05$, $\delta=0.10$, with a 50/50 cal/test split and $R=10$ random seeds, counting how often LTT returns a non-trivial threshold (“feasible”) (Table 32).

Two regimes. Two qualitatively different regimes emerge.

On the lower-error task (GSM8K, base SC@ $K=74\text{--}95\%$ across the three sweep generators), expanding the calibration budget from $n_{\text{cal}}=150$ to $n_{\text{cal}}=250$ moves $\alpha=0.05$ from never admitted (0/10 on every cell) to admitted on a minority of seeds (0–3/10): the $\alpha=0.05$ infeasibility on lower-error tasks is at least partly a calibration-budget effect, and a practitioner with a larger calibration set begins to reach the strict target.

On the higher-error task (StrategyQA, base SC@ $K \sim 67\text{--}71\%$ across the three sweep generators), $\alpha=0.05$ remains infeasible at every n_{cal} we test. This is consistent with Corollary 1: the selective-error floor on a task family with high wrong-mode prevalence p_2 is bounded below by $p_2/(p_1+p_2)$, and on StrategyQA this conditional rate exceeds 0.20 (Section 4). No n_{cal} in the range we tested fixes this. LTT correctly refuses to certify $\alpha=0.05$ when the underlying selective-error floor is above α , regardless of calibration set size. On higher-error tasks, the $\alpha=0.05$ infeasibility is therefore structural (Corollary 1), and a larger n_{cal} does not relax it.

E.31 Verifier Temperature Sensitivity (Phi-3.5-mini, all datasets)

Setup. Appendix E.19 reports a temperature sweep on GPT-4o-mini across one matched cohort. To check that the temperature-robustness conclusion is not specific to that single verifier or cell, we sweep the local Phi-3.5-mini-instruct verifier at sampling temperature $T \in \{0.0, 0.3, 0.7, 1.0\}$ on the Qwen2.5-7B generator across all five datasets (GSM8K / StrategyQA / MATH500 / ARC-C / CSQA). The verifier prompt, $K_v=4$ regime, and rationale-conditioning r^* rule are held fixed, and only temperature varies. Across the pooled Qwen2.5-7B subset (≈ 500 valid questions, about 100 per dataset), we make four temperature settings

Generator	N	SC@ K	CBW@0.8 (n_h)	ERR@0.9	SC-only AURC	RISC AURC
Qwen2.5-7B	300	68.0%	14.0% (179)	12.0%	18.25%	17.39%
Llama-3.1-8B	300	65.0%	4.7% (117)	6.8%	16.36%	18.12%
Qwen2.5-14B	300	82.0%	8.0% (218)	6.0%	7.47%	8.17%
Mistral-7B	300	50.7%	7.0% (67)	25.9%	37.08%	36.24%
4-gen mean		66.4%	8.4%	12.7%	19.79%	19.98%

Table 31: BBH per-generator numbers. n_h is the count of questions with $\rho \geq 0.8$. SC-only AURC ranks by ρ ; RISC AURC ranks by $(\rho + s_v^{\text{GPT}5})/2$ as in Section 5. Both are bootstrap means over 1,000 resamples.

Cell	n_{pool}	$n_{\text{cal}}=75$	$n_{\text{cal}}=150$	$n_{\text{cal}}=250$
<i>Easy task (low base error):</i>				
GSM8K $\times$ Qwen2.5-14B	500	0/10	0/10	3/10
GSM8K $\times$ Llama-3.1-8B	500	0/10	0/10	3/10
GSM8K $\times$ Mistral-7B	500	0/10	0/10	0/10
<i>Hard task (high base error):</i>				
StrategyQA $\times$ Qwen2.5-14B	500	0/10	0/10	0/10
StrategyQA $\times$ Llama-3.1-8B	499	0/10	0/10	n/a
StrategyQA $\times$ Mistral-7B	500	0/10	0/10	0/10

Table 32: LTT $\alpha=0.05$ feasibility (#non-trivial thresholds returned out of $R=10$ random 50/50 cal/test splits) as a function of n_{cal} . “n/a” = $n_{\text{pool}}/2 < n_{\text{cal}}$. GSM8K admits a threshold only at $n_{\text{cal}}=250$, and then on a minority of seeds. For the tested StrategyQA cells, $\alpha=0.05$ never becomes feasible at any n_{cal} tested (the StrategyQA $\times$ Llama-3.1-8B cell at $n_{\text{cal}}=250$ is excluded by pool-size as flagged “n/a”).

$\times K_v=4$ verifier calls per item, yielding roughly 8 K Phi calls overall (Table 33).

Dataset	$T=0.0$	$T=0.3$	$T=0.7$	$T=1.0$	Spread
GSM8K	3.40%	3.40%	3.40%	3.40%	0.00pp
StrategyQA	30.49%	29.92%	29.02%	29.10%	1.47pp
MATH500	6.23%	6.45%	6.23%	6.28%	0.22pp
ARC-C	11.88%	12.05%	12.05%	12.05%	0.17pp
CSQA	10.89%	10.70%	10.69%	10.69%	0.20pp

Table 33: Phi-3.5-mini-instruct verifier AURC ($\downarrow$ better) on Qwen2.5-7B chains across five datasets and four sampling temperatures (Table 34). Spread = max – min over T . All five datasets show spread $\leq 1.47\text{pp}$, far smaller than the cohort’s cross-generator noise band, replicating the cell-level conclusion of Appendix E.19 on a five-dataset sweep.

Verifier temperature is not a confounder. Phi-3.5-mini verifier scores are stable across sampling temperature on all five datasets in our cohort: AURC moves by $\leq 1.47\text{pp}$ across $T \in \{0.0, 0.3, 0.7, 1.0\}$ (largest gap on StrategyQA, the noisiest benchmark in the cohort), and consecutive-temperature Spearman correlations average 0.967 over fifteen comparisons. This generalizes the single-cell GPT-4o-mini conclusion of Appendix E.19 to five-dataset coverage on the local

Dataset	$\rho_S(T_{0.0}, T_{0.3})$	$\rho_S(T_{0.3}, T_{0.7})$	$\rho_S(T_{0.7}, T_{1.0})$
GSM8K	1.000	1.000	1.000
StrategyQA	0.875	0.904	0.922
MATH500	0.978	0.978	0.998
ARC-C	0.929	>0.999	0.999
CSQA	0.917	>0.999	1.000

Table 34: Spearman correlation of per-question Phi yes-rate between consecutive sampling temperatures. Mean ρ_S across all 15 pairs is 0.967. The lowest correlation (StrategyQA $T_{0.0} \rightarrow T_{0.3}$, $\rho_S=0.875$) is the noisiest yes/no surface and still consistent with stable cross-temperature ranking.

Phi verifier, and supports the claim in Section 5.1 that verifier yes-rates are stable across sampling temperature. Verifier sampling temperature is not a confounder of the RISC results in Section 5.5. The cross-vendor and forced- T comparisons in Appendices E.17 and E.18 are unlikely to be artifacts of verifier sampling temperature.