How Much Best-of-N Regret Can Reranking a Shortlist Recover?

Anonymous authors
Paper under double-blind review

Abstract

A second stage confined to a proxy-score shortlist can improve Best-of-N only through the candidates it may return. How much of the regret can it remove? Measuring regret against the gold-best candidate in the same draw, we split it exactly into regret on shortlist misses and probability-weighted admission and ordering losses. The weighted ordering loss is the largest expected gain over Best-of-N of any selector confined to the shortlist that keeps the Best-of-N pick on misses, and a gold oracle attains it. The spread of the proxy residual bounds it, in distinct forms for i.i.d. and for finite-pool sampling. On 45 generator, proxy, gold and task configurations at a budget of 64, the median share of regret a second stage can recover rises from 10.4 per cent with one expected shortlist member to 52.0 with four and 87.7 with sixteen under randomised-size subset sampling. The sampling protocol and the gold also move this share. On the 35 configurations carrying a rule-checkable gold, replacing the learned gold lowers the median from 10.5 to 8.3 per cent, and the fall recurs in all seven datasets of a seventeen-pool extension. On MATH-500, replacing the grader that both verifies and evaluates turns a 1.7 percentage-point shortlist advantage over a call-matched random subset into a 1.6-point deficit at two verifier calls. Regrading the original selections alone leaves the advantage positive. Under exact verification, checking the runners-up with the Best-of-N pick as fallback beats the shortlist at matched verifier calls. The median paired accuracy difference is positive in all 54 benchmark, budget and width cells.

1 Introduction

Best-of-N can return a worse answer than one its own draw contains. A natural remedy is a stronger selector, one that combines weak verifiers (Saad-Falcon et al., 2025), resolves a score-margin shortlist by pairwise comparison (Zhao et al., 2025) or spends costly verification on the top-ranked candidates (Dughmi et al., 2026). Run as a second stage, such a selector chooses within a shortlist: a proxy reward model ranks the draw and a threshold or rank cut-off keeps its top candidates. If the better answer has already been excluded, no improvement to the second stage recovers it. At the budget and shortlist width in use, how much of Best-of-N regret remains recoverable once the shortlist is fixed?

We answer by comparing three candidates of the same draw: the Best-of-N pick, the best shortlisted candidate and the best drawn candidate, with the shortlist cut at a proxy-score threshold. When the draw holds a shortlisted candidate, the gold-reward gap between the best drawn and the best shortlisted candidate is the admission loss, and the gap between the best shortlisted candidate and the pick is the ordering loss. The two losses, weighted by the probability of such a draw, and the regret on draws with no shortlisted candidate add up to Best-of-N regret. The weighted ordering loss is exactly what a gold oracle confined to the shortlist gains. We measure how it changes with shortlist width and compare the threshold shortlist with a fixed top-k list.

On logged candidate pools three further choices shape the measured value: how a budget-N draw and its shortlist are taken from the pool fixes the policy measured, the gold defines the regret, and a grader that also verifies decides which candidate is returned as well as its grade. We vary the width and the protocol on 45 configurations, the gold on 35 and MATH-500's grader (Figures 1 and 2).

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

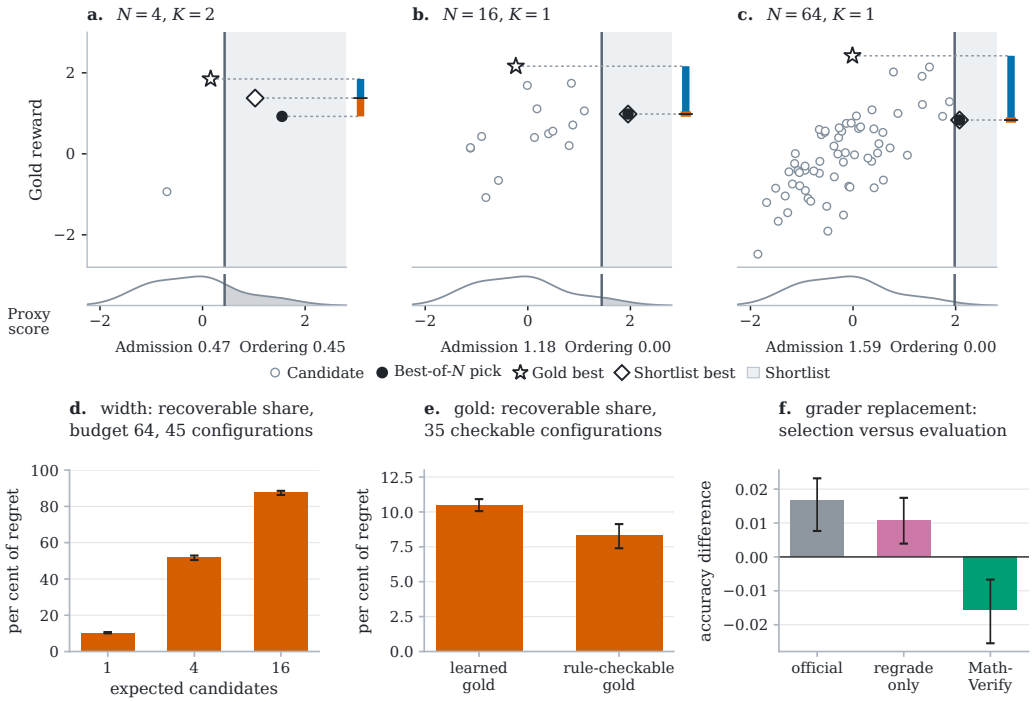


Figure 1: **The shortlist splits regret; width sets how much of it is recoverable, and the gold and the grader move the measurement. (a–c)** Three illustrative synthetic draws at one expected candidate, all with $K \geq 1$ shortlist members; bars split regret into admission (blue) and ordering (orange) loss. **(d–f)** Budget 64, primary protocol (Section 3). **(d)** Width map: median recoverable share over the 45 configurations. **(e)** Median recoverable share at one expected candidate over the 35 rule-checkable configurations under each gold. **(f)** Shortlist minus a call-matched random subset at two calls, median over three MATH-500 pools and two reward models, when each grader selects and grades or when Math-Verify regrades the official selection. Math-Verify replaces the official extractor and equivalence rule. Whiskers are 95% prompt-cluster intervals.

Contributions.

- **The oracle ceiling.** An exact three-part split of Best-of-N regret around a threshold shortlist, an oracle ceiling for any second stage confined to it, and residual-variance certificates for i.i.d. and finite-pool sampling. The recoverable share is small at one expected candidate and rises steeply with width, from a median 10.4 per cent of regret to 87.7 at sixteen at a budget of 64.
- **What the measurement depends on.** The sampling protocol, the gold and the grader each move the measured value. Rebasing on a rule-checkable gold lowers the recoverable share by paired medians of 2.0 to 9.2 points on 35 configurations and, at width one, in all seven datasets of a seventeen-pool extension. On MATH-500, regrading fixed selections separates a change of evaluation from a change of the returned candidate. The shortlist's lead over a call-matched random subset survives the regrade and reverses at two calls when the new grader also selects.
- **The runners-up rule.** With an exact verifier and the Best-of-N pick kept as the unverified fallback, verifying the candidates ranked second to $(k{+}1)$-th has the accuracy of top-$(k{+}1)$ coverage. Its median paired accuracy difference over the threshold shortlist is positive in all 54 benchmark, budget and width cells, also when the shortlist skips the pick and widens to the same expected calls.

2 Related Work and Positioning

Shortlists and second stages. Weak verifiers can be combined (Saad-Falcon et al., 2025), a score-margin shortlist can be resolved by pairwise comparison (Zhao et al., 2025), and Dughmi et al. (2026) search adaptively, generating and verifying until a candidate is verified correct. Our question

is how much loss a selector confined to a shortlist could remove, measured as returned-answer accuracy after a fixed candidate draw with the unverified Best-of-N pick retained as a fallback. The choice among second stages (Zhang et al., 2025a; Toshniwal et al., 2025; Singhi et al., 2025) is complementary, since the oracle ceiling bounds the gain of any of them confined to the shortlist. Selection plateaus below coverage under repeated sampling (Brown et al., 2024), imperfect verifiers limit what resampling reaches (Stroebl et al., 2026), and a binary any-of-k oracle gap bounds the gain of any fixed-pool selector over a first-sample or self-consistency reference (Hu, 2026). We split real-valued regret at a threshold shortlist, separate admission loss from ordering loss, and bound the ordering loss by the proxy residual's spread.

Golds and grading. Gao et al. (2023a) separate the proxy reward optimised against from the gold that assesses it, and grading reward models against verifiable outcomes is established practice (Frick et al., 2025; Malik et al., 2026; Wen et al., 2025; Kim et al., 2024). We measure what the choice of gold does to the split and rebase the recoverable share on a rule-checkable gold. Zhang et al. (2025b) find that the ranking of process-model aggregations reverses with the source of training labels; on MATH-500 the last-step aggregation's lead over the product for one process reward model is present under the official extractor and absent under Math-Verify (Section 6). Section B places the split among finite-sample Best-of-N bounds, concomitant laws, subset selection, n-best oracle curves and Gaussian-copula references.

3 The Shortlist Split, Its Oracle Ceiling and Its Certificate

Setup. Fix a prompt. A draw holds $N \geq 2$ candidates $Y_1, \ldots, Y_N$ from a policy. Each candidate has a proxy reward $\widehat{R}$ and a gold reward $R^\star$ that defines the regret and is hidden from Best-of-N and the shortlist. Best-of-N returns the drawn candidate $\widehat{Y}_N$ with the largest proxy reward, the *Best-of-N pick*. The drawn candidate with the largest gold reward is $Y_N^\star$, and the per-draw regret is $\Delta := R^\star(Y_N^\star) - R^\star(\widehat{Y}_N) \geq 0$, with mean $\Delta J(N)$.

Finite-pool measurements. Each prompt carries a logged pool of M candidates with recorded proxy and gold rewards, and every measurement is conditional on those pools. A budget-N draw is a uniform size-N subset of the prompt's pool, and we report the mean regret of that finite-pool policy. Section 5.1 relates it to the deployment quantity, the same mean under *fresh* generation.

Threshold shortlist. For a width $0 < c \leq N$, the pool shortlist S_m holds the pool's top m records by proxy score, at expected mass c/N, and the drawn shortlist is the part of S_m inside the draw. The *primary protocol* pairs the subset draw with a random-size shortlist: the pool-shortlist size is drawn as $m \sim \mathrm{Bin}(M, c/N)$, independently of the candidate subset; at proxy ties one uniform key per candidate settles both shortlist membership and the argmax. The number K_N of members a draw holds is then $\mathrm{Bin}(N, c/N)$, so c is an expected count: at $M = 256$, $N = 64$ and $c = 1$ (*one expected candidate*) the pool shortlist holds $\mathbb{E}[m] = 4$ records and a draw $\mathbb{E}[K_N] = 1$ member. Section 5.1 compares this construction with an inclusive pool-quantile shortlist, and Section A lists the construction and draw law behind each analysis.

A draw is a *hit* when it holds at least one shortlist member; write Z for the probability of that event. Every member outranks every non-member under the tie convention above, so the Best-of-N pick is a member on every hit ($\mathcal{E}_N$ and Z_N in Section C).

The split. The shortlist separates two sources of regret on a hit: a better drawn candidate excluded from it, and a worse one selected among those kept. Let M^{in} be the largest gold reward among the drawn members, M_N^{in} in Section C. Adding and subtracting it splits expected regret into three parts:

$$\Delta J(N) = \underbrace{\mathbb{E}[\Delta\, \mathbf{1}_{\mathrm{miss}}]}_{\text{regret on misses}} + Z\, \underbrace{\mathbb{E}\big[R^\star(Y_N^\star) - M^{\mathrm{in}} \mid \mathrm{hit}\big]}_{\Xi_N\ \text{(admission loss)}} + Z\, \underbrace{\mathbb{E}\big[M^{\mathrm{in}} - R^\star(\widehat{Y}_N) \mid \mathrm{hit}\big]}_{\Lambda_N\ \text{(ordering loss)}}. \qquad (1)$$

Admission loss Ξ_N compares the best drawn candidate with the best drawn member; ordering loss Λ_N compares that member with the pick. Both condition on a hit, so they enter total regret multiplied by Z, while the miss term already carries its own probability. The identity needs only $\mathbb{E}|R^\star| < \infty$; because the shortlist is a top set of the proxy order and holds the pick on every hit, both losses are non-negative.

The oracle ceiling. Hold the draw law, the shortlist and the reward pair fixed, and ask which part of Equation (1) can move when only the second-stage selector changes.

Proposition 3.1 (Oracle ceiling)**.** *Assume $\Delta J(N) < \infty$ and let H be any selector of finite expected regret returning a drawn shortlist member on hits and $\widehat{Y}_N$ on misses. Then $\mathbb{E}[R^\star(H) - R^\star(\widehat{Y}_N)] \le Z\Lambda_N$, with equality for the selector returning the member with the largest gold reward on hits.*

The *oracle ceiling* is $Z\Lambda_N$, an expected gain in gold-reward units, and the *recoverable share* is its ratio to a positive $\Delta J(N)$, quoted in per cent of regret (proofs in Section D). Across budgets and widths the recoverable share forms the *width map*. The shortlist's gold oracle attains the bound; even that oracle leaves regret on misses and admission loss untouched. The split and the oracle ceiling depend on the proxy only through its ranks (Ho et al., 1992), while the certificate below depends on its scale as well. Both also hold for a *fixed top-k list*, the reward model's k highest-scoring drawn candidates. That list always contains the pick, so its hit probability is one and its miss term vanishes, whereas a threshold shortlist's occupancy is random under the draw law.

A certificate for the ordering loss. Proposition 3.1 expresses the oracle ceiling through the gold-best member. We bound the conditional ordering loss instead by the spread of the proxy residual $\xi := \widehat{R} - R^\star$, where $\tau_{\xi,c/N}$ is its standard deviation over shortlist candidates.

Theorem 3.2 (Ordering certificate under i.i.d. draws)**.** *Let $Y_1, \dots, Y_N$ be i.i.d. from the policy. Assume $\mathbb{E}|R^\star| < \infty$ and $\mathbb{E}[\xi^2 \mid Y \in \mathcal{T}_{c/N}] < \infty$, and let one independent uniform key per candidate at a proxy atom randomise both shortlist membership and the argmax tie, so $\mathcal{T}_{c/N}$ is the policy's fixed top set of exact mass c/N. For every $N \ge 2$ and $c \in (0, N]$,*

$$\Lambda_N(c) \le c_c(N)\,\tau_{\xi,c/N}, \qquad c_c(N) = \frac{\sqrt{2\,(c - Z_c)(Z_c - b_c)}}{Z_c}, \tag{2}$$

where $Z_c = 1 - (1 - c/N)^N$ and $b_c = c\,(1 - c/N)^{N-1}$ are the probabilities that a draw holds at least one and exactly one member.

The bound is on expected ordering loss given a hit, to which only draws holding two or more members contribute; $Z_c - b_c$ is the probability of such a draw and $c - Z_c$ the expected occupancy beyond the first. At one expected candidate $c_1(N)$ stays below 0.698 at every finite budget, with limit $\sqrt{2(e-2)/(e-1)}$. Section D.2 proves Theorem 3.2.

Under the primary protocol the shortlist is S_m, the pool's top m records, with expectations over m and the draw, and given m the members are drawn without replacement. The certificate then averages the without-replacement bound over the shortlist size (Theorem D.8 in Section D). On the cohort of Section 4, ordering loss uses a median 0.273 to 0.332 of this bound at one expected candidate and budgets of 4 to 64 (Section D.2). On a fixed top-k list the pick's ordering gap is at most $\sqrt{2k}$ times the within-list spread with divisor k (Section D.3).

4 How Much Regret Does the Shortlist Leave Recoverable?

Evaluation data. The cohort has 45 configurations, each a generator, a proxy reward model, a learned gold reward model and a dataset, on 40 pools, 14 prompt lists, 15 datasets and 10 proxy-gold pairs. Its 14 generators come from six families, 26 configurations use a generator of 3B or fewer, and the five reward models, with checkpoints in Table 28, are Skywork-V0.2 (Liu et al., 2024), Skywork-V2-8B and Skywork-V2-3B (Liu et al., 2026), ArmoRM (Wang et al., 2024a) and RM-Mistral (Xiong et al., 2024). Pools cover up to 200 prompt rows with 256 candidates per prompt, 1,024 for the one 70B pool, and both scores are standardised per prompt (Sections C and K). Figures and tables carry 95% prompt-cluster bootstrap intervals, with replicate counts in Section A, and the text reports each estimate without its interval.

The width map. Under the primary protocol, at a budget of 64, the median recoverable share over the 45 configurations is 10.4% with one expected candidate, 52.0% with four and 87.7% with sixteen (Figure 1d). At one expected candidate it is 13.5% at a budget of 16 and 21.0% at 4. Each share, the ratio of a configuration's prompt-averaged moments, is the most that any second stage confined to the same shortlist can recover (Proposition 3.1). At width one, where a hit usually holds

a single member, most of the loss is beyond any shortlist-confined second stage; at sixteen most of it is within reach. As separate cohort medians, misses carry 39.3% of regret and admission loss 50.6%, admission being 83.0% of the regret on hits.

A fixed top-k list has a higher cohort-median recoverable share at every evaluated $k \geq 2$. The fixed top-k list has a higher cohort-median recoverable share than a threshold shortlist of the same expected size at every cell of Table 1, by 0.42 to 8.40 points, and all 45 configurations agree at six of the seven cells. At one candidate a top-1 list is the pick and recovers nothing, while the threshold shortlist's oracle ceiling is positive.

Table 1: **A fixed list of the reward model's top k drawn candidates has a higher recoverable share than a threshold shortlist of equal expected size at every evaluated budget and width.** Recoverable share of Best-of-N regret under the primary protocol, medians over 45 configurations with 95% prompt-cluster intervals. The gap is the median per-configuration gap. Positive counts configurations with a positive gap.

Budget	k	Fixed list (%)	Threshold shortlist (%)	Gap (points)	Positive
16	2	43.81 [42.86, 44.84]	34.98 [34.32, 36.24]	+8.40 [+8.08, +8.55]	45/45
16	4	73.70 [71.92, 74.42]	67.53 [65.98, 68.63]	+5.65 [+5.52, +5.75]	45/45
16	8	91.51 [90.47, 92.34]	90.81 [89.74, 91.63]	+0.72 [+0.66, +0.80]	44/45
64	2	33.27 [32.69, 34.63]	26.98 [26.04, 27.62]	+6.90 [+6.57, +7.08]	45/45
64	4	58.12 [56.38, 59.07]	51.97 [50.43, 52.90]	+5.95 [+5.83, +6.12]	45/45
64	8	75.54 [74.02, 76.77]	73.50 [72.09, 74.79]	+1.87 [+1.81, +1.92]	45/45
64	16	88.20 [86.77, 89.02]	87.66 [86.29, 88.56]	+0.42 [+0.39, +0.46]	45/45

5 Measuring the Recoverable Share on Logged Pools

The width map rests on the primary protocol and the learned gold, and changing either moves the recoverable share.

5.1 The sampling protocol

How a budget-N draw is taken from the logged pool, and how its shortlist is cut, set how many shortlist members the draw holds, hence every part of Equation (1). When a pool holds i.i.d. draws, averaging over all its size-N subsets is unbiased for any fresh-generation moment of the draw alone (Nakano et al., 2021; Gao et al., 2023a). Resampling with replacement instead lets one logged record fill several draw positions, which biases nonlinear statistics such as the maximum, and Chen et al. (2021) correct pass@k combinatorially for this reason. A shortlist cut on the pool also depends on the pool's records outside the draw, so that guarantee does not cover it. The inclusive construction cuts at the (cM/N)-th largest proxy score and keeps every candidate tied with it, so its realised mass can exceed c/N. Conditional on its size, its occupancy under a subset draw is hypergeometric, while the rule it stands for, pass every candidate above a score cut-off of mass c/N, implies $\mathrm{Bin}(N, c/N)$. The primary protocol keeps the subset draw and draws the pool-shortlist size as $\mathrm{Bin}(M, c/N)$, which reproduces that occupancy law exactly, with the pool's top records as the admitted candidates.

We compare three protocols (Section A): the inclusive shortlist drawn *with replacement* or as *plain subsets*, and the *primary* protocol. Section C.4 compares them with fresh generation in simulation.

Recoverable share under the three protocols. The three protocols disagree (Figure 2a). At a budget of 64 with one expected candidate, the cohort median recoverable share is 8.1% with replacement, 9.8% on plain subsets and 10.4% under the primary protocol, so here the draw law moves it more than the shortlist construction. The primary and plain-subset medians differ by $+0.60$, -1.20 and -0.16 points at one, four and sixteen.

Pool-size sensitivity depends on the protocol. We grow two GSM8K pools from 256 to 1,024 candidates: the cohort's 70B pool and a separately drawn Llama-3.2-3B pool on the same prompts. With replacement the budget-64 recoverable share rises 1.64 and 1.65 points while regret on misses does not move; on plain subsets regret on misses rises instead (Section C.4). Under the primary

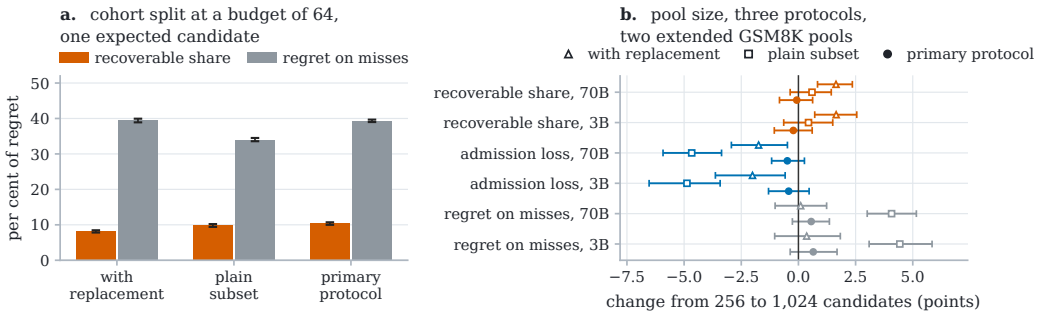


Figure 2: **The sampling protocol moves the split and its pool-size sensitivity.** At a budget of 64 with one expected candidate, **(a)** the median recoverable share and regret on misses over the 45 configurations and **(b)** the change in recoverable share, admission loss and regret on misses as two GSM8K pools grow from 256 to 1,024 candidates. Whiskers are 95% prompt-cluster intervals, paired across pool sizes in (b).

Table 2: **Rebasing on the rule-checkable gold lowers the recoverable share in every cell.** Median recoverable shares over the 35 rule-checkable configurations, in per cent of regret, on the same pools and draws under the primary protocol; the rule check uses official grading. The paired difference is the median per-configuration difference. Brackets are 95% prompt-cluster bootstrap intervals.

Budget	Expected candidates	Learned gold	Rule-checkable gold	Paired difference
64	1	10.5 [10.1, 10.9]	8.3 [7.4, 9.1]	−2.0 [−3.1, −1.2]
64	4	52.2 [51.1, 53.8]	43.0 [40.1, 47.1]	−9.2 [−12.4, −5.2]
64	16	88.6 [87.1, 89.3]	79.2 [76.3, 82.9]	−9.1 [−12.0, −4.7]
16	1	13.8 [13.3, 14.2]	11.0 [10.1, 11.8]	−2.4 [−3.5, −1.5]

protocol the recoverable share changes by -0.07 and -0.22 points, and the 95% intervals of its changes all include zero (Figure 2b).

5.2 The gold

The gold that defines the regret is a learned reward model throughout the cohort. Of the 45 configurations, 35 admit a gold a rule can check, 19 on GSM8K, 11 on MATH-500 and five across the code benchmarks and BBH; the other ten are preference, safety or open-ended tasks, or withhold their answer options.

Rebasing on a rule-checkable gold lowers the recoverable share. Rebasing the width map on the rule-checkable gold lowers the recoverable share in every cell of Table 2. At a budget of 64 with one expected candidate the median recoverable share over the 35 configurations falls from 10.5 to 8.3 per cent, and the paired difference is negative in 30 of the 35. On an expanded set of seventeen pools covering seven datasets and 21 proxy-gold pairs, 16 of these pairs outside the cohort, the paired difference is negative in every dataset (Section F.6).

Matching the binary pass counts isolates the labels. Binarising the learned gold at each prompt's checkable pass count matches the rule check's scale and per-prompt mass, so only the labels differ, and the binarised gold's recoverable share stays above the rule check's in all four cells of Table 2. Against the binarised gold, at a budget of 64 with one expected candidate, the oracle ceiling moves by -0.0003, not separated from zero, while total regret rises by 0.048 (Section F.2), so the recoverable share falls because unrecoverable regret grows.

6 When Grading Also Selects the Answer

Answer extraction changes measured accuracy and rankings (Yu et al., 2025; Molfese et al., 2025; Kydlíček et al., 2025); where the grader also verifies, it changes both the grade and which answer is returned. The MATH-500 pools were generated with no answer-format instruction under a 512-token output limit (Section K.2). On the four pools studied here the official grader, which takes

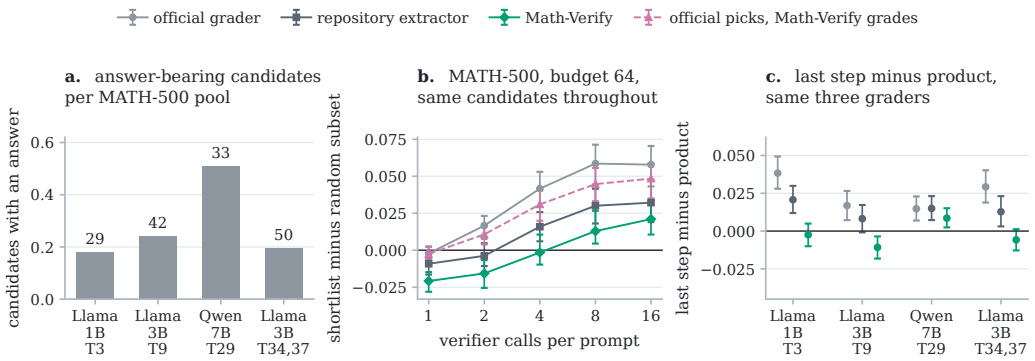


Figure 3: **Changing the grading pipeline moves two MATH-500 comparisons.** **(a)** Answer-bearing share of candidates under official grading; the number above each bar is how many of the 200 prompts have none. Ticks give the generator and the configuration ID (Table 29). **(b)** Shortlist minus call-matched random subset, budget 64, median over T3, T9 and T29 with two reward models. **(c)** Process-reward-model aggregation, last step minus product, each pool's mean over its 18 cells, two reward models at nine budget and width settings. Math-Verify replaces the official extractor and equivalence rule; the repository extractor is the lenient alternative of Section G.1. In (b) graders also select, except in the dashed line (official picks, Math-Verify grades); in (c) none does. Whiskers are 95% prompt-cluster bootstrap intervals.

the last boxed answer and grades it with the PRM800K equivalence checker (Lightman et al., 2024), recovers a boxed answer from between 0.178 and 0.507 of the candidates, and 29 to 50 of each pool's 200 prompts carry no parseable candidate at all (Figure 3a). Measured accuracy here is therefore partly format compliance. In a post hoc comparison on a stratified sample, a blinded language-model reading agrees with the official grade on 0.832 of candidates and with the Math-Verify (Kydlíček, 2026) grade on 0.988 (Section G.2).

Regrading fixed selections and replacing the verifier. We keep every candidate and prompt fixed and change only the grader. The official extractor reads a boxed answer from 0.281 of the four pools' $204{,}800$ candidates, and Math-Verify, which replaces the official grader's extractor and equivalence rule, extracts an expression from 0.9996, including every candidate the official extractor reads. On the three pools carrying verifier labels, T3, T9 and T29 in Figure 3a, we compare verifying the shortlist with verifying a call-matched uniformly random subset, each returning a passing candidate or else the Best-of-N pick. The grader is also the verifier the policy calls, so a different grader can return a different candidate. With the official grader selecting and grading, the shortlist leads by 0.017 at two calls and 0.042 at four. Regrading those same selections with Math-Verify leaves it ahead at every call count above one, by 0.011 with two calls, 0.031 with four and 0.048 with sixteen. When Math-Verify also selects, the lead is -0.016 at two calls and -0.0015 at four (Figure 3b).

A blinded audit by two language-model readers covers the official selection's returns on which the two extractors' grades differ under the official equivalence rule. Neither reader finds a wrong answer that the official extractor grades correct (Section G.3), and substituting their grades leaves the shortlist ahead at two and four calls. When Math-Verify's extractor also selects under that rule, the corrected lead on its own returns is -0.013 at two calls, with an interval containing zero, and $+0.049$ at four (Table 21).

Changing evaluation without changing selection. When a process reward model's step scores are aggregated, the reward model selects and the grader only evaluates. Aggregating the step scores of Qwen2.5-Math-PRM-7B (Zhang et al., 2025b) by their last step rather than their product is worth 0.023 in median accuracy over the four pools under official grading, positive on all four with all 18 reward-model, budget and width cells separated from zero. On the same candidates the last-step lead is -0.0041 under Math-Verify, positive on one pool with no cell separated (Figure 3c).

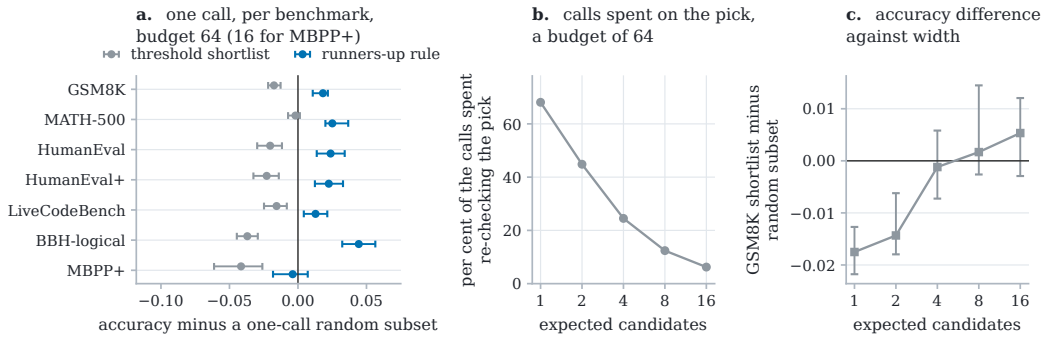


Figure 4: **Verification allocation with a retained Best-of-N fallback. (a)** Accuracy under the rule-checkable gold minus a random subset that verifies one candidate, at nominal width one for the inclusive threshold shortlist, which can spend more than one expected call, and at one call for the runners-up rule; median over each benchmark's configurations. **(b)** Share of the inclusive shortlist's expected calls spent on the pick and **(c)** shortlist minus random subset against width, GSM8K medians over 19 configurations. Budget 64; MBPP+ at budget 16, verified by its prompt-visible assertions and graded by the withheld tests. Whiskers are 95% prompt-cluster bootstrap intervals.

7 Verification Calls and the Best-of-N Fallback

The shortlist spends calls on the Best-of-N pick, which its fallback returns anyway. We compare second stages by returned-answer accuracy under an exact verifier, whose pass/fail label is the gold's, at a given cost, the expected number of candidates a rule submits for verification; every rule keeps the unverified Best-of-N pick as its fallback. At nominal width one the shortlist is the inclusive construction at $c = 1$, whose ties can lift its expected cost above one. There the shortlist trails a uniformly random one-call subset on six of the seven benchmarks (Figure 4a), from -0.016 on LiveCodeBench to -0.042 on MBPP+, while on MATH-500 the difference, -0.0018, is not separated from zero. On GSM8K the deficit is confined to one and two calls. Checking a candidate that the fallback would return anyway cannot change whether the returned answer is correct. The pick is inside a threshold shortlist whenever the draw hits it. At a budget of 64 checking the pick takes Z_c/c of the expected calls under the primary protocol, 63.5 per cent at width one down to 6.2 at sixteen. On the inclusive shortlist the GSM8K median runs from 68.1 down to 6.2 per cent, and the deficit fades in step (Figure 4b,c).

The fallback enlarges the set of possible outputs. For $1 \le k < N$, the runners-up rule verifies the drawn candidates the reward model ranks second to $(k+1)$-th, returns the highest-ranked one that passes, and keeps the Best-of-N pick when none does. The shortlist and the random subset keep the same unverified fallback. Under an exact verifier the rule is right exactly when one of the candidates ranked 1 to $k{+}1$ is correct, so its accuracy is top-$(k{+}1)$ coverage C_{k+1}: it verifies k candidates and can return any of $k+1$. The identity holds for any proxy ranking.

Against the shortlist at matched calls. At one call the runners-up rule beats a call-matched random subset on six of the seven benchmarks (Figure 4a), by 0.018 on GSM8K. Under an exact verifier the median paired accuracy difference between the rule and the threshold shortlist is positive in all 54 benchmark, budget and width cells. At one call it runs from 0.027 to 0.078, and from 0.007 to 0.027 when the shortlist skips the pick and widens to the same expected calls (Table 3). The rule also leads the fixed top-k list, the call-matched random subset and the Best-of-N pick in every cell; on the mathematics pools, where answer classes exist, it leads Best-of-Majority (Di et al., 2026) in every cell and zero-call majority voting in all but one, MATH-500 at a budget of 64 and one call (Section H.4).

The margin rests on an exact label and a retained fallback. A smaller gain over the shortlist remains at one and two calls when the pass/fail signal is MBPP's prompt-visible assertions and the gold the withheld MBPP+ tests (Sections H.2 and H.4).

When a process reward model scores the runners-up in place of an exact verifier, the rule that returns the one it scores highest discards the pick. At one call it returns the sole runner-up whatever its score,

Table 3: **At equal expected verifier calls, the runners-up rule leads the threshold shortlist in all 54 benchmark medians.** Median, over each benchmark's configurations (35 in all), of the paired accuracy difference, rule minus shortlist, in accuracy proportions (0.01 is one percentage point), under the official gold and an exact verifier; parentheses count the configurations above zero. The shortlist verifies its members other than the Best-of-N pick, keeping the pick when none passes, at the width c^* whose expected calls $c^* - Z_{c^*}$ equal k.

Budget	k	GSM8K	MATH-500	HumanEval	HumanEval+	LiveCodeBench	BBH-logical
16	1	+0.0133 (18/19)	+0.0067 (11/11)	+0.0141 (1/1)	+0.0141 (1/1)	+0.0089 (2/2)	+0.0213 (1/1)
16	2	+0.0109 (19/19)	+0.0065 (11/11)	+0.0132 (1/1)	+0.0128 (1/1)	+0.0080 (2/2)	+0.0217 (1/1)
16	4	+0.0047 (19/19)	+0.0033 (11/11)	+0.0063 (1/1)	+0.0047 (1/1)	+0.0047 (2/2)	+0.0127 (1/1)
16	8	+0.0012 (19/19)	+0.0004 (9/11)	+0.0010 (1/1)	+0.0018 (1/1)	+0.0004 (2/2)	+0.0046 (1/1)
64	1	+0.0139 (18/19)	+0.0135 (11/11)	+0.0144 (1/1)	+0.0146 (1/1)	+0.0113 (2/2)	+0.0271 (1/1)
64	2	+0.0116 (19/19)	+0.0138 (11/11)	+0.0158 (1/1)	+0.0140 (1/1)	+0.0102 (2/2)	+0.0269 (1/1)
64	4	+0.0084 (19/19)	+0.0085 (11/11)	+0.0099 (1/1)	+0.0111 (1/1)	+0.0054 (2/2)	+0.0207 (1/1)
64	8	+0.0031 (19/19)	+0.0033 (11/11)	+0.0037 (1/1)	+0.0044 (1/1)	+0.0022 (2/2)	+0.0097 (1/1)
64	16	+0.0008 (19/19)	+0.0012 (10/11)	+0.0011 (1/1)	+0.0015 (1/1)	+0.0005 (1/2)	+0.0046 (1/1)

and it trails the shortlist on GSM8K and MATH-500 (Table 24). Keeping the pick unless the highest-scoring runner-up clears a threshold fitted on the other half of the prompts turns the GSM8K deficit at a budget of 64 under product aggregation into a lead of 0.0093, positive in all 10 configurations. On MATH-500 the deficit vanishes under the last step and narrows under the product and the minimum (Section H.5).

8 Conclusion

A shortlist fixes which Best-of-N errors remain available to a second stage. The oracle ceiling measures that opportunity, and the residual certificates bound it under i.i.d. and finite-pool sampling. At a budget of 64 the recoverable share rises from 10.4% at one expected candidate to 87.7% at sixteen, and on logged pools it moves with the protocol and the gold. Where the grader also verifies, replacing it changes the returned answer as well as its grade. With an exact verifier and the Best-of-N pick kept as the unverified fallback, verifying the runners-up with k calls has the accuracy of top-$(k{+}1)$ coverage. A reranking gain therefore has to be read together with its candidate set, its gold and its fallback.

Ethics Statement

The study involves no human subjects and collects no new data from people. Its prompts come from the public benchmarks of Section K.1, including a safety benchmark of potentially harmful requests, and every candidate response is generated by a publicly released model. The main numerical archive holds candidate-level scores and identifiers and no prompt or response text, so none of the generated responses, including those to the safety prompts, is released with it.

Reproducibility Statement

Section A lists every main analysis with its draw law, shortlist construction, tie rule, gold, grader, fallback, call accounting and interval design; Section C gives the estimators, Section D the proofs, and Section K the datasets, generation and scoring settings and reward-model checkpoints. Code and data are at `https://anonymous.4open.science/r/ICLR27-20758`: the candidate-level scores and labels of every pool (no prompt or response text), the analysis code that recomputes the reported quantities from them, brute-force checks of the finite-pool identities, the unit-test evaluator and MBPP+ grader, and the generators of every figure and generated table.

AI Use Statement

A large language model assistant was used throughout this work: to refine the research questions, suggest analyses and interpret results; to help develop and check the theory and its proofs; to write

and revise the analysis, simulation and figure code; to recompute the reported numbers and check them against the text; to search the literature; and to edit the prose. The blinded adjudications of Sections G.2 and G.3 are language-model readings (Section G.2). The authors take responsibility for the paper's claims, proofs, analyses and references.

REFERENCES

Gholamali Aminian, Idan Shenfeld, Amir R. Asadi, Ahmad Beirami, and Youssef Mroueh. Best-of-N through the smoothing lens: KL divergence and regret analysis. In *International Conference on Learning Representations (ICLR)*, 2026. doi: 10.48550/arxiv.2507.05913. arXiv:2507.05913.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021. doi: 10.48550/arxiv.2108.07732.

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

Ahmad Beirami, Alekh Agarwal, Jonathan Berant, Alexander D'Amour, Jacob Eisenstein, Chirag Nagpal, and Ananda Theertha Suresh. Theoretical guarantees on the best-of-N alignment policy. In *International Conference on Machine Learning (ICML)*, 2025. doi: 10.48550/arxiv.2401.01879.

Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V. Le, Christopher Ré, and Azalia Mirhoseini. Large language monkeys: Scaling inference compute with repeated sampling. *arXiv preprint arXiv:2407.21787*, 2024. doi: 10.48550/arxiv.2407.21787.

Zheng Cai et al. InternLM2 technical report. *arXiv preprint arXiv:2403.17297*, 2024. doi: 10.48550/arXiv.2403.17297.

Alexandra Carpentier and Michal Valko. Simple regret for infinitely many armed bandits. In *International Conference on Machine Learning (ICML)*, 2015. arXiv:1505.04627.

Eugene Charniak and Mark Johnson. Coarse-to-fine n-best parsing and MaxEnt discriminative reranking. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 173–180, 2005. doi: 10.3115/1219840.1219862.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021. doi: 10.48550/arxiv.2107.03374.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. Chatbot Arena: An open platform for evaluating LLMs by human preference. In *International Conference on Machine Learning (ICML)*, 2024. doi: 10.48550/arxiv.2403.04132.

Robert Chin, Jonathan E. Rowe, Iman Shames, Chris Manzie, and Dragan Nešić. Ordinal optimisation for the Gaussian copula model. *arXiv preprint arXiv:1911.01993*, 2019. doi: 10.48550/arXiv.1911.01993.

Karl Cobbe, Vineet Kosaraju, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021. doi: 10.48550/arxiv.2110.14168.

Qiwei Di, Kaixuan Ji, Xuheng Li, Heyang Zhao, and Quanquan Gu. Best-of-majority: Minimax-optimal strategy for pass@k inference scaling. In *International Conference on Learning Representations (ICLR)*, 2026. OpenReview a6CVQpjbXq.

Florian E. Dorner, Yatong Chen, André F. Cruz, and Fanny Yang. ROC-n-reroll: How verifier imperfection affects test-time scaling. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=3Gy5mmyuxn. arXiv:2507.12399.

Shaddin Dughmi, Mahdi Haghifam, and Yusuf Hakan Kalayci. Adaptive generate-rank-verify: Inference-time search with costly verification. *arXiv preprint arXiv:2605.17609*, 2026. doi: 10.48550/arXiv.2605.17609.

Evan Frick, Tianle Li, Connor Chen, Wei-Lin Chiang, Anastasios N. Angelopoulos, Jiantao Jiao, Banghua Zhu, Joseph E. Gonzalez, and Ion Stoica. How to evaluate reward models for RLHF. In *International Conference on Learning Representations (ICLR)*, 2025. doi: 10.48550/arXiv.2410.14872. arXiv:2410.14872.

Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning (ICML)*, 2023a. doi: 10.48550/arxiv.2210.10760.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation. Zenodo, version v0.4.0, `https://zenodo.org/records/10256836`, 2023b.

Shanti S. Gupta. On some multiple decision (selection and ranking) rules. *Technometrics*, 7(2): 225–245, 1965. doi: 10.1080/00401706.1965.10490251.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks*, 2021. doi: 10.48550/arxiv.2103.03874.

Y. C. Ho, R. S. Sreenivas, and P. Vakili. Ordinal optimization of DEDS. *Discrete Event Dynamic Systems*, 2(1):61–88, 1992. doi: 10.1007/BF01797280.

Jiri Hron, Karl Krauth, Michael I. Jordan, and Niki Kilbertus. On component interactions in two-stage recommender systems. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. doi: 10.48550/arXiv.2106.14979. arXiv:2106.14979.

Jie Hu. Oracle gap and signal fidelity: A fixed-pool diagnostic for test-time collaboration. *arXiv preprint arXiv:2607.17531*, 2026. doi: 10.48550/arXiv.2607.17531.

Audrey Huang, Adam Block, Qinghua Liu, Nan Jiang, Akshay Krishnamurthy, and Dylan J. Foster. Is best-of-N the best of them? Coverage, scaling, and optimality in inference-time alignment. In *International Conference on Machine Learning (ICML)*, 2025. doi: 10.48550/arxiv.2503.21878. URL `https://proceedings.mlr.press/v267/huang25c.html`.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. LiveCodeBench: Holistic and contamination free evaluation of large language models for code. In *International Conference on Learning Representations (ICLR)*, 2025. doi: 10.48550/arxiv.2403.07974.

Jiaming Ji, Mickel Liu, Juntao Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. BeaverTails: Towards improved safety alignment of LLM via a human-preference dataset. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks*, 2023. doi: 10.48550/arxiv.2307.04657.

S. N. Joshi and H. N. Nagaraja. Joint distribution of maxima of concomitants of subsets of order statistics. *Bernoulli*, 1(3):245–255, 1995. doi: 10.3150/bj/1193667817.

Hadi Khalaf, Claudio Mayrink Verdun, Alex Oesterling, Himabindu Lakkaraju, and Flavio du Pin Calmon. Inference-time reward hacking in large language models. In *Advances in Neural Information Processing Systems (NeurIPS), Spotlight*, 2025. doi: 10.48550/arxiv.2506.19248.

Sunghwan Kim, Dongjin Kang, Taeyoon Kwon, Hyungjoo Chae, Jungsoo Won, Dongha Lee, and Jinyoung Yeo. Evaluating robustness of reward models for mathematical reasoning. *arXiv preprint arXiv:2410.01729*, 2024. doi: 10.48550/arXiv.2410.01729.

Thomas Kwa, Drake Thomas, and Adrià Garriga-Alonso. Catastrophic Goodhart: regularizing RLHF with KL divergence does not mitigate heavy-tailed reward misspecification. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. doi: 10.52202/079017-0466. arXiv:2407.14503.

Hynek Kydlíček. Math-Verify: Math verification library. Software, `https://github.com/huggingface/Math-Verify`, version 0.9.0, 2026.

Hynek Kydlíček, Alina Lozovskaya, Nathan Habib, and Clémentine Fourrier. Fixing the Open LLM Leaderboard with Math-Verify. Hugging Face blog, `https://huggingface.co/blog/math_verify_leaderboard`, 14 February, 2025.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, Noah A. Smith, and Hannaneh Hajishirzi. RewardBench: Evaluating reward models for language modeling. In *Findings of the North American Chapter of the Association for Computational Linguistics (NAACL Findings)*, 2025. doi: 10.18653/v1/2025.findings-naacl.96.

Eddie Landesberg. When LLM judge scores look good but best-of-N decisions fail. *arXiv preprint arXiv:2603.12520*, 2026. doi: 10.48550/arXiv.2603.12520.

T. W. Edward Lau and Y. C. Ho. Universal alignment probabilities and subset selection for ordinal optimization. *Journal of Optimization Theory and Applications*, 93(3):455–489, 1997. doi: 10.1023/A:1022614327007.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, et al. Holistic evaluation of language models. *Transactions on Machine Learning Research (TMLR)*, 2023. doi: 10.48550/arXiv.2211.09110. arXiv:2211.09110.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations (ICLR)*, 2024. doi: 10.48550/arxiv.2305.20050.

Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2022. doi: 10.18653/v1/2022.acl-long.229.

Chris Yuhao Liu, Liang Zeng, Jiacai Liu, Rui Yan, Jujie He, Chaojie Wang, Shuicheng Yan, Yang Liu, and Yahui Zhou. Skywork-Reward: Bag of tricks for reward modeling in LLMs. *arXiv preprint arXiv:2410.18451*, 2024. doi: 10.48550/arxiv.2410.18451.

Chris Yuhao Liu, Liang Zeng, Yuzhen Xiao, Jujie He, Jiacai Liu, Chaojie Wang, Rui Yan, Wei Shen, Fuxiang Zhang, Jiacheng Xu, Yang Liu, and Yahui Zhou. Skywork-Reward-V2: Scaling preference data curation via human-AI synergy. In *International Conference on Learning Representations (ICLR)*, 2026. doi: 10.48550/arxiv.2507.01352. arXiv:2507.01352.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. Is your code generated by ChatGPT really correct? Rigorous evaluation of large language models for code generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. doi: 10.48550/arxiv.2305.01210.

Yantao Liu, Zijun Yao, Rui Min, Yixin Cao, Lei Hou, and Juanzi Li. PairJudge RM: Perform best-of-N sampling with knockout tournament. *arXiv preprint arXiv:2501.13007*, 2025. doi: 10.48550/arXiv.2501.13007.

Saumya Malik, Valentina Pyatkin, Sander Land, Jacob Morrison, Noah A. Smith, Hannaneh Hajishirzi, and Nathan Lambert. RewardBench 2: Advancing reward model evaluation. In *International Conference on Learning Representations (ICLR)*, 2026. doi: 10.48550/arXiv.2506.01937. arXiv:2506.01937.

Chuan Meng, Negar Arabzadeh, Arian Askari, Mohammad Aliannejadi, and Maarten de Rijke. Ranked list truncation for large language model-based re-ranking. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 2024. doi: 10.1145/3626772.3657864.

Francesco Maria Molfese, Luca Moroni, Luca Gioffré, Alessandro Scirè, Simone Conia, and Roberto Navigli. Right answer, wrong score: Uncovering the inconsistencies of LLM evaluation in multiple-choice question answering. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 18477–18494. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-acl.950.

Youssef Mroueh. Information theoretic guarantees for policy alignment in large language models. In *Workshop on Aligning Reinforcement Learning Experimentalists and Theorists (ARLET) at the International Conference on Machine Learning (ICML), Oral*, 2024. doi: 10.48550/arxiv.2406.05883. arXiv:2406.05883.

Arpan Mukherjee, Marcello Bullo, Debabrota Basu, and Deniz Gündüz. Test-time verification via optimal transport: Coverage, ROC, and sub-optimality. In *International Conference on Learning Representations (ICLR)*, 2026. doi: 10.48550/arXiv.2510.18982. OpenReview BBDhQJh6GB; arXiv:2510.18982.

H. N. Nagaraja and H. A. David. Distribution of the maximum of concomitants of selected order statistics. *The Annals of Statistics*, 22(1):478–494, 1994. doi: 10.1214/aos/1176325380.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. WebGPT: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*, 2021. doi: 10.48550/arXiv.2112.09332. Appendix I, the without-replacement estimator of Best-of-N performance.

Roger B. Nelsen. *An Introduction to Copulas*. Springer Series in Statistics. Springer, New York, 2006. doi: 10.1007/0-387-28678-0.

Barry L. Nelson, Julie Swann, David Goldsman, and Wheyming Song. Simple procedures for selecting the best simulated system when the number of alternatives is large. *Operations Research*, 49(6):950–963, 2001. doi: 10.1287/opre.49.6.950.10019.

Gabriel Orlanski, Nicholas Roberts, Aws Albarghouthi, and Frederic Sala. Pareto optimal code generation. *arXiv preprint arXiv:2506.10056*, 2025. doi: 10.48550/arXiv.2506.10056. Version 2, February 2026.

Qwen Team. Qwen2.5-Math-PRM-7B model card. Hugging Face model repository `https://huggingface.co/Qwen/Qwen2.5-Math-PRM-7B`, revision 0610740, 2025.

Nazneen Rajani, Lewis Tunstall, Edward Beeching, Nathan Lambert, Alexander M. Rush, and Thomas Wolf. No Robots. Hugging Face Datasets, `https://huggingface.co/datasets/HuggingFaceH4/no_robots`, 2023.

Eshwar Reddy M and Sourav Karmakar. Proof-carrying cognition: Closing the verification gap with reality-settled reward. *arXiv preprint arXiv:2609.09776*, 2026. doi: 10.48550/arXiv.2609.09776.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level Google-proof Q&A benchmark. In *First Conference on Language Modeling (COLM)*, 2024. arXiv:2311.12022.

Jon Saad-Falcon, E. Kelly Buchanan, Mayee F. Chen, Tzu-Heng Huang, Brendan McLaughlin, Tanvir Bhathal, Shang Zhu, Ben Athiwaratkun, Frederic Sala, Scott Linderman, Azalia Mirhoseini, and Christopher Ré. Shrinking the generation-verification gap with weak verifiers. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. doi: 10.48550/arXiv.2506.18203. arXiv:2506.18203.

Shivalika Singh, Freddie Vargus, Daniel D'souza, Börje F. Karlsson, Abinaya Mahendiran, Wei-Yin Ko, Herumb Shandilya, Jay Patel, Deividas Mataciunas, Laura O'Mahony, et al. Aya dataset: An open-access collection for multilingual instruction tuning. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024. doi: 10.48550/arxiv.2402.06619.

Nishad Singhi, Hritik Bansal, Arian Hosseini, Aditya Grover, Kai-Wei Chang, Marcus Rohrbach, and Anna Rohrbach. When to solve, when to verify: Compute-optimal problem solving and generative verification for LLM reasoning. In *Conference on Language Modeling (COLM)*, 2025. doi: 10.48550/arXiv.2504.01005. arXiv:2504.01005.

Ved Sriraman and Adam Block. Revisiting the (sub)optimality of best-of-N for inference-time alignment. In *Conference on Learning Theory (COLT)*, 2026. URL https://proceedings.mlr.press/v336/sriraman26a.html. PMLR 336:5980–6028.

Benedikt Stroebl, Sayash Kapoor, and Arvind Narayanan. The limits of inference scaling through resampling. In *International Conference on Learning Representations (ICLR)*, 2026. doi: 10.48550/arxiv.2411.17501. arXiv:2411.17501.

Miraç Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, and Jason Wei. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In *Findings of the Association for Computational Linguistics (ACL Findings)*, 2023. doi: 10.18653/v1/2023.findings-acl.824.

James Theiler, Stephen Eubank, André Longtin, Bryan Galdrikian, and J. Doyne Farmer. Testing for nonlinearity in time series: the method of surrogate data. *Physica D: Nonlinear Phenomena*, 58(1–4):77–94, 1992. doi: 10.1016/0167-2789(92)90102-S.

Shubham Toshniwal, Ivan Sorokin, Aleksander Ficek, Ivan Moshkov, and Igor Gitman. GenSelect: A generative approach to best-of-N. In *2nd AI for Math Workshop at the International Conference on Machine Learning (ICML)*, 2025. doi: 10.48550/arXiv.2507.17797. arXiv:2507.17797.

Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. Interpretable preferences via multi-objective reward modeling and mixture-of-experts. In *Findings of the Empirical Methods in Natural Language Processing (EMNLP Findings)*, 2024a. doi: 10.18653/v1/2024.findings-emnlp.620.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations (ICLR)*, 2023. doi: 10.48550/arxiv.2203.11171. arXiv:2203.11171.

Yizao Wang, Jean-Yves Audibert, and Rémi Munos. Algorithms for infinitely many-armed bandits. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2008.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks*, 2024b. doi: 10.48550/arxiv.2406.01574.

Xueru Wen, Jie Lou, Yaojie Lu, Hongyu Lin, Xing Yu, Xinyu Lu, Ben He, Xianpei Han, Debing Zhang, and Le Sun. Rethinking reward model evaluation: Are we barking up the wrong tree? In *International Conference on Learning Representations (ICLR), Spotlight*, 2025. doi: 10.48550/arxiv.2410.05584.

Wei Xiong, Hanze Dong, Chenlu Ye, Ziqi Wang, Han Zhong, Heng Ji, Nan Jiang, and Tong Zhang. Iterative preference learning from human feedback: Bridging theory and practice for RLHF under KL-constraint. In *International Conference on Machine Learning (ICML)*, 2024. URL https://proceedings.mlr.press/v235/xiong24a.html. PMLR 235:54715–54754.

Rui Yang, Ruomeng Ding, Yong Lin, Huan Zhang, and Tong Zhang. Regularizing hidden states enables learning generalizable reward model for LLMs. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. arXiv:2406.10216.

Qingchen Yu, Zifan Zheng, Shichao Song, Zhiyu Li, Feiyu Xiong, Bo Tang, and Ding Chen. xFinder: Large language models as automated evaluators for reliable evaluation. In *International Conference on Learning Representations (ICLR)*, 2025. doi: 10.48550/arXiv.2405.11874. arXiv:2405.11874.

Zhuohao Yu, Zhiwei Steven Wu, and Adam Block. From curiosity to caution: Mitigating reward hacking for best-of-N with pessimism. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=EZn2TmBBfF. arXiv:2604.04648.

Huiming Zhang, Binghan Li, Wan Tian, and Qiang Sun. Tail-aware information-theoretic bounds for LLM alignment under heavy-tailed rewards. *arXiv preprint arXiv:2604.10727v2*, 2026a. doi: 10.48550/arxiv.2604.10727.

Junkai Zhang, Zihao Wang, Lin Gui, Swarnashree Mysore Sathyendra, Jaehwan Jeong, Victor Veitch, Wei Wang, Yunzhong He, Bing Liu, and Lifeng Jin. Chasing the tail: Effective rubric-based reward modeling for large language model post-training. In *International Conference on Learning Representations (ICLR)*, 2026b. doi: 10.48550/arxiv.2509.21500. arXiv:2509.21500.

Lunjun Zhang, Arian Hosseini, Hritik Bansal, Mehran Kazemi, Aviral Kumar, and Rishabh Agarwal. Generative verifiers: Reward modeling as next-token prediction. In *International Conference on Learning Representations (ICLR)*, 2025a. doi: 10.48550/arXiv.2408.15240. arXiv:2408.15240.

Zhenru Zhang, Chujie Zheng, Yangzhen Wu, Beichen Zhang, Runji Lin, Bowen Yu, Dayiheng Liu, Jingren Zhou, and Junyang Lin. The lessons of developing process reward models in mathematical reasoning. In *Findings of the Association for Computational Linguistics (ACL Findings)*, 2025b. doi: 10.18653/v1/2025.findings-acl.547.

Eric Zhao, Pranjal Awasthi, and Sreenivas Gollapudi. Sample, scrutinize and scale: Effective inference-time search by scaling verification. In *International Conference on Machine Learning (ICML)*, 2025. URL https://proceedings.mlr.press/v267/zhao25a.html. PMLR 267:77272–77309.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P Xing, et al. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks*, 2023. doi: 10.48550/arxiv.2306.05685.

A PROTOCOL REGISTER

Table 4 lists the paper's main analyses with the conventions that produced them: the law by which a budget-N draw is taken from the logged pool, how the shortlist is built, how ties are resolved, which gold defines correctness, which grader scores a candidate, what a rule returns when nothing it checked passes, how calls are counted, and the design behind every interval and count printed beside the result. Each row names where the analysis appears.

Two distinctions matter most: how the shortlist is built and which design produces each interval. The inclusive pool-quantile shortlist keeps every candidate scoring at least the (cM/N)-th largest, including those tied with the last of them, so its realised mass can exceed c/N. The random-size shortlist of the primary protocol (Section C.2) draws its size as $\mathrm{Bin}(M, c/N)$, so that its pool mass has mean c/N, and takes the top m of a total order broken by a uniform key. Redrawing that key moves a cell of Table 2 by up to 0.27 points. The call shares of Section 7 are measured on the inclusive construction, and the primary protocol's figure beside them, $Z_c/c = [1-(1-c/N)^N]/c$, is analytic. The cluster-robust design of Section J.3 is applied to the configuration-weighted mean. An interval beside a cohort median is a percentile interval of prompt-cluster bootstrap replicates of that median, recomputed inside each replicate.

Table 4: **The main analyses, each with the conventions that produced it.** Draw law *repl.* takes N i.i.d. pool records and *subset* a uniform size-N subset without replacement; *exact* is a closed-form finite-pool expectation and *MC* a Monte Carlo average over the stated number of draws per prompt (seed 7 for MC 400, 20260919 for MC 8,000, 20260929 for MC 4,000). Shortlist *incl.* is the inclusive pool-quantile construction, which keeps threshold ties, *random size* the random-size shortlist of the primary protocol and *exact mass* the fixed-count shortlist of Section C.2. *Key* is a uniform tie-break key at proxy atoms. Gold *learned* is the configuration's gold reward model and *check.* the rule-checkable gold; grader *official* is each dataset's official grader, which runs the tests on code and reads the answer letter on multiple-choice sets, and *alt.* the repository's alternative extractor. Second stage *kept* means the Best-of-N pick is the fallback, *none* that the analysis has no second stage; cost is expected second-stage calls per prompt. In the design column B is a bootstrap replicate count, preceded by the units it resamples, and *mean* the cluster-robust design on the configuration-weighted mean (Section J.3), applied one level at a time with $t(G-1)$ on a level's G clusters: by proxy-gold pair and by prompt list, or, on the rule-checkable configurations, by pair with the dataset clusters as sign counts. A triple such as 45/10/14 counts configurations, proxy-gold pairs and dataset clusters; HumanEval and HumanEval+ share one prompt list and count as one, so the 45 configurations' 15 datasets form 14 clusters and the 35 rule-checkable configurations' six form five. A to D and X in the last row are the conditions of Section I.

Analysis	Draw	Shortlist	Ties	Gold; grader	Second stage	Design
Width map (Section 4)	subset; exact	random size	key per budget	learned	none	prompt B 2,000; mean 45/10/14
Protocol comparison, pool growth (Section 5.1, Figure 2)	repl.; subset	incl.; random size	key; incl. keeps ties	learned	none	prompt B 2,000; paired in (b)
Split controls (Section E)	repl.; MC 400	incl.; exact mass	first index; shared key	learned	none	medians over 45
Fixed list against a threshold shortlist (Section 4, Table 1)	subset; exact	fixed list of k; random size	list: reward, draw position; threshold: key	learned	list holds the pick	prompt B 2,000; mean 45/10/14
Gold rebasing, 35 configurations (Section 5.2, Table 2)	subset; exact	random size	one key, both golds	check. vs learned, paired; official	none	paired prompt B 2,000; mean 35/10/5; short prompts dropped
Binarised learned gold (Section 5.2, Section F.2)	subset; exact	random size	as above	check., binarised, learned; official	none	paired prompt B 2,000; 35 configurations
Gold rebasing, 22 configurations (Section F.2)	repl.; exact	incl.	level mean gold	exact match, tests vs learned; last number, tests	none	prompt B 20,000; 22 configurations, 4 pools paired
Gold rebasing, seventeen pools (Section 5.2, Section F.6)	subset; exact	random size	one key per pool and prompt	check. vs learned, paired; official, tests, letter	none	mean by pair, pool, dataset; 351/21/7
Extraction rates, grader agreement (Section 6, Sections G.1 and G.2)	none	none	none	check.; official, alt., Math-Verify; audit by language-model reading	unread is incorrect	31 configurations, 4 pools; audit 160 candidates, stratified
Zero-call rules, majority voting, Best-of-Majority (Section G)	repl.; MC 400	none	smallest index; groups by size, mean reward, first member	check.; official	Best-of-N if no group	4 GSM8K, 3 MATH-500 configurations
Candidate-restriction controls (Section G.1)	subset; exact	incl.	key	official, answer-bearing, cap-dropped, lenient; official, alt.	kept; calls	prompt B 20,000; 31/10/3; short prompts dropped
Shortlist against a call-matched subset (Section 6, Figure 3b)	subset; exact	random size	key	check.; official, alt., Math-Verify	kept; calls, $c = k$	prompt B 20,000; 6 rows, 2 pairs
Audit of returned answers (Section 6, Section G.3)	subset; MC 8,000	random size	key	check.; official, Math-Verify; disagreements read by a language model	kept; calls 1, 2, 4	prompt B 20,000, returns resampled; 270 per selection
Cost of verifying a shortlist (Section H.3)	subset; exact	incl.	kept	official, MBPP+; official, tests	kept; calls by width c	prompt B 20,000
Process-model aggregation (Section 6, Figure 3c)	subset; exact	random size	key per pool and model	check.; official, alt., Math-Verify	kept; calls $= c$	prompt B 20,000; 4 pools, 2 pairs
Call shares, one-call deficit (Section 7, Figure 4)	subset; exact	incl.	kept at threshold	check.; official	kept; calls 1.014 vs 1.000	prompt B 20,000, configurations resampled; 35/10/5
Runners-up rule, exact verifier (Section H.4)	subset; MC 4,000	random size	reward, then shared key	check. is the verifier label; official	kept; calls $= c$; k at c^* without the pick	prompt and configuration B 2,000; 35/10/5
Runners-up rule, visible assertions (Section H.2)	subset; MC 4,000; $N = 16$	random size; incl. for charging	as above	MBPP+; tests	kept; incl. charged 3.57 for 2.00	task and configuration B 2,000; 4 units
Runners-up rule, process model, fallback discarded (Section H.4)	subset; MC 4,000	random size	as above	check., lenient, answer-bearing, cap-dropped; official, alt.	discarded; calls $= c$	prompt and configuration B 2,000; 18/2/2
Runners-up rule, process model, fallback matched (Section 7, Section H.5)	subset; MC 4,000	random size	as above	check.; official	kept below a cross-fitted threshold; calls $= c$	prompt and configuration B 2,000; 18/2/2

Analysis	Draw	Shortlist	Ties	Gold; grader	Second stage	Design
Reference departures (Section J)	repl.; MC 400	incl.	first index; reference keeps ties	learned	none	prompt B 2,000, both seeds; mean 45/10/14
Comparisons from the literature (Section I)	repl. (A, C); subset (B, X, D); exact	none	level mean gold; first maximum	binarised learned (A, B); check. (C, X, D); official, D answer-bearing on 17 of 35	class rules keep the pick	mean 35/10/5; short prompts dropped

Rows that report the same quantity under different conventions. The budget-64 recoverable share at one expected candidate is 10.38 per cent in the width map, under the primary protocol, and 8.13 in the split controls of Section E, on with-replacement draws with the inclusive shortlist. The rule-checkable gold regrades the width map on 35 configurations under the primary protocol (Section 5.2) and on 22 under with-replacement draws with the inclusive shortlist (Section F.2). The call-matched comparison of Section 6 and the aggregation contrast use the primary protocol. The call shares and the one-call deficit use the subset draw with the inclusive shortlist, and the fixed-count exact-mass shortlist of Section E and Table 10 sits on the with-replacement draws (Section C.2). The runners-up rule's shortfall under a process model appears twice, with the Best-of-N fallback discarded and with it matched.

B Further related work

Finite-sample Best-of-N bounds. Existing analyses differ in both the objective they bound and the error they measure. Beirami et al. (2025) analyse the KL cost of the Best-of-N policy, and Mroueh (2024, Cor. 1, Props. 3, 5) bounds reward improvement over the reference policy, once through the gold-reward scale and once through a transfer bound that uses $\|\widehat{R} - R^{\star}\|_{\infty}$. Aminian et al. (2026) use a β-tilted squared error whose Best-of-N limit is a squared sup-norm, Huang et al. (2025) combine marginal squared error with coverage relative to a comparator policy, and Zhang et al. (2026a) derive bounds under heavy-tailed rewards. Di et al. (2026) evaluate the error at the unobserved gold optimum and propose Best-of-Majority, a minimax-optimal rule for pass@k inference scaling that selects over the whole draw (Section G implements it with $k = 1$). Our split partitions realised regret on each draw at a proxy-score quantile, and the residual scale of Theorem 3.2 averages over the proxy's shortlist.

Pairwise error, reward hacking and error tails. Sriraman & Block (2026, Lem. 13) observe that squared error can change under rank-preserving proxy rescalings; their pairwise win-rate error gives win-rate guarantees for properly tuned Best-of-N, and their Proposition 15 gives a counterexample to a gold-regret bound based on that error under unbounded rewards. Our parts keep gold-reward magnitudes while remaining invariant to strictly increasing proxy rescalings. Under Best-of-N, gold reward can fall as optimisation against a proxy strengthens (Khalaf et al., 2025), and pessimistic selection mitigates this (Yu et al., 2026). Zhang et al. (2026b) compute exactly, at the KL-regularised optimum with a uniform gold distribution and deterministic misspecification maps, that ranking the best responses correctly suffices for near-optimal performance, and Kwa et al. (2024) prove that heavy-tailed reward error defeats KL regularisation. We measure, on finite Best-of-N draws from learned reward models, how regret divides between admission to a shortlist and ordering within it.

Verification, second stages and oracle gaps. Infinitely armed bandits distinguish sampling a good arm from identifying it (Wang et al., 2008; Carpentier & Valko, 2015). For repeated sampling, pass@N keeps rising while majority voting and reward-model selection plateau (Brown et al., 2024), and imperfect verifiers such as weak unit tests limit what resampling can reach (Stroebl et al., 2026); Saad-Falcon et al. (2025) combine weak verifiers to narrow that gap. Generative verifiers (Zhang et al., 2025a) and GenSelect (Toshniwal et al., 2025) can each serve as a second stage on a shortlist. Zhao et al. (2025) keep the responses whose average self-verification score lies within 0.05 of the best and break ties by pairwise comparison, Dughmi et al. (2026) rank candidates by a cheap reward and verify top-ranked ones under a cost for each call, and Orlanski et al. (2025) find outcome reward models most effective when lightweight filters prune candidates before expensive verification. In two-stage recommenders the interaction between nominators and ranker changes overall performance (Hron et al., 2021). Singhi et al. (2025) find self-consistency more compute-efficient than

generative verification for most budgets, and a pairwise knockout judge (Liu et al., 2025) selects over the whole draw rather than over a shortlist. Hu (2026) bounds the gain of any fixed-pool selector over a first-sample or self-consistency reference by a binary any-of-k oracle gap, and factors the realised gain through signal coverage, selection quality and harm. Our oracle ceiling and the share of it that a second stage realises follow the same logic of a recoverable amount and its realisation, for real-valued regret around a shortlist across widths and budgets. For binary rewards, Dorner et al. (2026) relate Best-of-N accuracy to the verifier's ROC and Mukherjee et al. (2026, App. O) split sub-optimality into sampling and verification error. Both of our losses are real-valued and measured within each draw against its best candidate: admission loss concerns which candidates the shortlist keeps and ordering loss the choice among them. Weighted by the hit probability, the ordering loss is the oracle ceiling of a second stage confined to the shortlist.

Concomitants, ordinal optimisation and reranking depth. For a bivariate sample ordered by its first coordinate, Nagaraja & David (1994) and Joshi & Nagaraja (1995) derive the laws of the largest concomitant among the top k observations and its joint law with the largest concomitant among the rest. On a draw that hits the shortlist, our admission loss is the positive part of the second maximum minus the first, zero when every drawn candidate is a member, here with a random number of shortlist members. Subset selection keeps a small subset containing the best population with a chosen probability (Gupta, 1965; Nelson et al., 2001), and ordinal optimisation formalises subsets of noisily ranked designs that contain good ones with high probability (Ho et al., 1992; Lau & Ho, 1997; Chin et al., 2019). Admission loss is the regret of such a screen, here by an imperfect score, that misses the best drawn candidate. Oracle F-scores of n-best parse lists rise from 0.897 to 0.968 as the list grows from one to 50 parses (Charniak & Johnson, 2005), and per-query truncation of a retrieved list sets the re-ranking depth (Meng et al., 2024). We give the regret-valued split and oracle ceiling of a reward-model shortlist across widths and budgets, with certificates for i.i.d. and finite-pool draws.

Matched references for selection quality. Surrogate data preserve chosen statistics of observed series to test for structure beyond them (Theiler et al., 1992). For a fixed budget and i.i.d. proxy-gold pairs with continuous margins under a Gaussian copula, the law of the gold score of the selected candidate depends only on the copula parameter and the gold distribution (Reddy M & Karmakar, 2026, Prop. 2). Landesberg (2026) shows that global judge correlation does not identify best-of-4 recovery over random choice, reporting a within-prompt correlation of 0.267 against a global 0.471 with 21.0% recovery against the 27% a Gaussian model at the within-prompt correlation implies. We match per-prompt rank agreement with duplicate candidates kept identical and measure the departure of each part of regret, per budget, with intervals over 45 configurations (Section J).

Reward-model evaluation. RewardBench (Lambert et al., 2025) and RewardBench 2 (Malik et al., 2026) benchmark reward models by their accuracy at picking preferred or correct responses, PPE (Frick et al., 2025) selects proxy metrics by their correlation with post-RLHF outcomes, and Wen et al. (2025) relate reward-model accuracy to downstream policy quality. Benchmark scores also depend on the grading pipeline. Standardised harnesses and mathematics graders fix the answer-extraction rule (Gao et al., 2023b; Liang et al., 2023; Kydlíček, 2026; Balunović et al., 2025), regular-expression extraction mis-scores correct answers (Yu et al., 2025; Molfese et al., 2025), and swapping one grader for another reorders a public leaderboard (Kydlíček et al., 2025). Section 6 measures how grading changes selection comparisons on fixed candidate pools. On identical draws, the preferred reward model and the helpful reranker change with the gold on GSM8K (Section F.1).

C Notation, the threshold shortlist and the draw law

C.1 Notation

Fix a prompt and a budget-N draw of candidates $Y_1, \ldots, Y_N$, $N \geq 2$. The symbols below do not depend on how the draw is taken; each statement names its draw law. Theorem 3.2 takes the draw i.i.d. from a reference policy π_{ref}, which for the with-replacement arm of Section C.3 is the uniform distribution on a logged pool of M candidates. The primary protocol takes a uniform size-N subset of that pool, whose candidates are exchangeable but not i.i.d., and Lemma D.6 and Theorem D.8 give its certificate. The proxy residual is $\xi := \widehat{R} - R^\star$ and $\xi_i := \xi(Y_i)$. Table 5 collects the symbols.

Probability formulas use continuous proxy scores. With atoms, attach independent uniform tie-break keys and use the resulting ranks for both the shortlist and the argmax. Gold ties in $Y_N^\star$ use the first draw index.

Table 5: **Symbols of the shortlist split and its certificate.** Ξ_N, Λ_N and $\mathcal{R}_N$ condition on the hit event $\mathcal{E}_N$; $\Delta J(N)$, the oracle ceiling $Z_N\Lambda_N$ and the unrecoverable regret are unconditional. All six are in gold units.

Symbol	Meaning
$R^\star$, $\widehat{R}$	Gold and proxy rewards.
$\widehat{Y}_N$, $Y_N^\star$	Best-of-N pick (the proxy-selected candidate) and gold-best candidate in the same draw of N candidates.
Δ, $\Delta J(N)$	Per-draw regret $R^\star(Y_N^\star) - R^\star(\widehat{Y}_N)$ and its mean.
$\mathcal{T}_{c/N}$, $\mu_{c/N}$	Threshold shortlist at mass c/N and the law of a candidate conditional on it, with $\mathcal{T}_{1/N}$ for one expected candidate.
$\mathcal{E}_N$, Z_N, Z_c, b_c	Hit event, on which the draw contains a shortlist member, and its probability, which is $Z_c = 1-(1-c/N)^N$ under $\mathrm{Bin}(N, c/N)$ occupancy; $b_c = c(1-c/N)^{N-1}$ is the probability of exactly one member.
K_N, M_N^{in}	Number of drawn shortlist members and their largest gold reward on $\mathcal{E}_N$.
$\mathcal{R}_N$	Regret on hits, $\mathbb{E}[\Delta \mid \mathcal{E}_N] = \Xi_N + \Lambda_N$.
Ξ_N, Λ_N	Admission loss and ordering loss, conditional on $\mathcal{E}_N$; $100\,\Xi_N/(\Xi_N+\Lambda_N)$ is the admission share, in per cent of regret on hits.
$Z_N\Lambda_N$	The oracle ceiling, an expected gain in gold units; divided by a positive $\Delta J(N)$ it is the recoverable share, in per cent of regret. Unrecoverable regret is $\Delta J(N) - Z_N\Lambda_N$.
ξ	Per-candidate proxy residual $\widehat{R} - R^\star$.
$\sigma_{\xi,c/N}$, $\tau_{\xi,c/N}$	Residual RMS and standard deviation under $\mu_{c/N}$, with $\tau_{\xi,c/N} \le \sigma_{\xi,c/N}$.
$c_c(N)$, $c_1(N)$	Coefficient of the ordering certificate in Theorem 3.2 at width c, and its value at one expected candidate, which stays uniformly below 0.698.

C.2 The threshold shortlist and its tie conventions

Under the inclusive construction, each prompt's shortlist keeps every candidate whose score is at least the linear-interpolation quantile of the M pool proxy scores at level $1 - c/N$. For the pool sizes and widths used here, all with $c < N$, cM/N is an integer and the interpolation position lies strictly between two order statistics, so the rule keeps every candidate scoring at least the (cM/N)-th largest score. That is exactly cM/N candidates when scores are distinct, plus the candidates tied with the last of them otherwise. The Best-of-N pick takes the first maximising draw index, and every shortlist member scores above every non-member, so the pick is a member on every hit. A keyed shortlist instead breaks proxy ties by one independent uniform key per candidate record, redrawn as the Ties column of Table 4 states, and uses the resulting order for the shortlist and for the argmax. Every member again outranks every non-member, so the pick is again a member on every hit.

Three keyed constructions appear, the first two at exact mass. The *population threshold* of Theorem 3.2 keeps every candidate above a proxy quantile at mass c/N. The *fixed-count* shortlist, used on the with-replacement draws in the controls of Section E and in Table 10, keeps the top cM/N records. N divides cM in every cell where it is used, so conditional on the keys it is the population threshold of the uniform law on the keyed pool, at mass exactly c/N, and Theorem 3.2 applies to it. The *random-size* shortlist of the primary protocol draws $m \sim \mathrm{Bin}(M, c/N)$ independently of the size-N subset and keeps the top m, at every width, which is the form Theorem D.8 assumes. Its pool mass m/M is random with mean c/N, and its draw occupancy is exactly $\mathrm{Bin}(N, c/N)$ (Theorem D.8). Each draw law carries its own coefficient (Remark D.7).

C.3 MONTE CARLO AND EXACT ESTIMATORS

The with-replacement arm. This arm samples candidates with replacement from each prompt's pool, with seed 7 and 400 draws per prompt at every budget up to 64; every width reads the same draws. Cohort statistics are medians over the 45 configurations.

Pooling, on every arm. Within a configuration, admission and ordering losses are ratios of moments summed over prompts and replicates, and Z and $\Delta J(N)$ are pooled the same way, so a prompt with zero regret enters the numerator and the denominator without forming a quotient of its own. Every reported share is formed once per configuration, over up to 200 prompt rows, where $\Delta J(N) > 0$, and a cohort median is taken over those per-configuration shares.

Exact finite-pool expectations. Exact expectations are computed under either draw law, with the pool indices $J_1, \dots, J_N$ drawn i.i.d. uniformly or as a uniform size-N subset. For one reward model and one prompt with M candidate records, write $t(a)$ for the probability that all N draws fall in a fixed set of a records,

$$t(a) = \Big(\frac{a}{M}\Big)^N \text{ (i.i.d. draws)}, \qquad t(a) = \binom{a}{N} \Big/ \binom{M}{N} \text{ (uniform subset)},$$

the second being zero for $a < N$. Let $U \subseteq \{1, \dots, M\}$ be an allowed subset. Rank records by the order that settles the argmax: the proxy score under the first-index rule of Section C.2, or the pair (proxy score, key) under a uniform key, whose levels each hold one record. Index the distinct levels of that order among records in U in increasing order by $\ell = 1, \dots, L$, let n_ℓ count the records of U at or below level ℓ, with $n_0 = 0$, and let $\bar{r}_\ell$ be the mean gold reward of the records of U at level ℓ. Then

$$\mathbb{E}\Big[R^\star(\widehat{Y}_N)\,\mathbf{1}\{J_1, \dots, J_N \in U\}\Big] = \sum_{\ell=1}^{L} \bar{r}_\ell \big[t(n_\ell) - t(n_{\ell-1})\big],$$

because the bracket is the probability that all draws lie in U with highest level ℓ, and the selected record is uniform over the records at that level. Under the first-index rule the draw order is exchangeable, so the first drawn record at the top proxy level is uniform over that level. Under a key each level is one record, and the selected gold is that of the drawn proxy-tied record the key ranks highest, so averaging gold within a proxy tie belongs to the first-index rule alone. Sorting by gold gives the corresponding oracle moment. For the largest gold reward among drawn shortlist members, let T be the model's shortlist, index the distinct gold levels among $U \cap T$ in increasing order by $v = 1, \dots, V$ with values g_v, let S_v hold the records of $U \cap T$ at or below level v, and set $a_v = |U \setminus T| + |S_v|$ with $a_0 = |U \setminus T|$. Then

$$\mathbb{E}\Big[\max_{i:\, J_i \in T} R^\star(Y_{J_i})\,\mathbf{1}\{J_1, \dots, J_N \in U,\ J_i \in T \text{ for some } i\}\Big] = \sum_{v=1}^{V} g_v \big[t(a_v) - t(a_{v-1})\big],$$

because $t(a_v)$ is the probability that every draw lies in U and none enters T above level v. A draw misses exactly when every index lies in the complement of T, so a moment restricted to hits is the moment with U the whole pool minus the same moment with U that complement. With the selected-gold and oracle moments restricted to hits in this way,

$$Z_N \Lambda_N = \mathbb{E}\big[M_N^{\text{in}} \mathbf{1}_{\mathcal{E}_N}\big] - \mathbb{E}\big[R^\star(\widehat{Y}_N) \mathbf{1}_{\mathcal{E}_N}\big], \qquad Z_N \Xi_N = \mathbb{E}\big[R^\star(Y_N^\star) \mathbf{1}_{\mathcal{E}_N}\big] - \mathbb{E}\big[M_N^{\text{in}} \mathbf{1}_{\mathcal{E}_N}\big].$$

The random-size shortlist takes T as the top m records of the keyed order, ranks the argmax by the same order with the key held fixed across m, and averages every moment over $m \sim \text{Bin}(M, c/N)$. Unconditional moments are averaged over prompts before dividing by an event probability. The fixed list of Table 1 orders a draw by proxy score and then by draw position, so its list and its Best-of-N pick break proxy ties uniformly at random, while the threshold column beside it uses the key. Ordering the list and its pick by that key moves no median of the table by more than 0.05 points and leaves every count of positive gaps unchanged.

Bootstrap intervals. Clusters are distinct prompt texts, or task identifiers for code pools; MT-Bench has 200 rows with 73 distinct texts, and repeated rows move together while all rows stay in

the point estimate. The width map, the three-protocol comparison of Figure 2a, the fixed-list comparison, the 35-configuration gold rebasing and its binarised learned gold, the split-sample comparison of Table 9 and the departures of Section J.2 use 2,000 percentile replicates with seed 20260918. Each replicate draws one weight per distinct prompt identifier for each of the 14 prompt lists, and every configuration on that list reads them at all its arms, budgets, widths and golds; the cohort median is recomputed inside each replicate. The paired pool-growth comparison of Figure 2b uses 2,000 replicates with the same seed, and the runners-up comparisons the 2,000-replicate designs of Table 4. The other shared-draw comparisons use 20,000 percentile replicates with seed 20260914. Each of those replicates draws one weight per distinct prompt identifier, and every configuration carrying the same prompt sequence reads the same weights at all its budgets, widths, golds, reward models and directions. In both designs configurations on different prompt sequences draw theirs independently.

C.4 Sampling and threshold calibration on logged pools

A logged pool of M candidates admits three protocols for forming a budget-N draw and its shortlist. Two share the inclusive pool-quantile shortlist and differ in the draw law, with replacement or as a uniform size-N subset; the third keeps the subset draw and changes the shortlist construction. They differ in how many shortlisted candidates a draw holds.

Sampling N candidates with replacement is the arm of Section C.3. It is conditional on the pool and lets one shortlisted record fill more than one slot. With 256 candidates and one expected candidate, 10.1% of 64-candidate draws repeat one of the four shortlisted records, and removing those repeats raises the recoverable share by a paired median of 1.68 points, positive in all 45 configurations. The rise is 0.51 at a budget of 16, 0.12 at a budget of 4 and 0.38 on the 1,024-candidate pool.

When a prompt's pool holds i.i.d. draws, each of its size-N subsets is, over the generation of the pool, an i.i.d. sample of size N, so averaging over them estimates without bias any fresh-generation moment that depends on the draw alone (Nakano et al., 2021; Gao et al., 2023a). A shortlist cut on the pool depends on the pool as well. A shortlist of fixed size cM/N inside a subset draw gives hypergeometric occupancy in place of the $\mathrm{Bin}(N, c/N)$ occupancy that a score cut-off implies, so it re-specifies the rule. The primary protocol keeps the subset draw and draws the shortlist size as $\mathrm{Bin}(M, c/N)$, at the cost of a uniform tie-break key at proxy atoms. A $\mathrm{Bin}(M, c/N)$ mixture of hypergeometric occupancies is $\mathrm{Bin}(N, c/N)$, so its occupancy law is exact; its members are the top m of the pool for an m drawn independently of the pool's scores. Its moments are exact expectations of that finite-pool policy conditional on the pool, and its shares are ratios of those moments.

On the two GSM8K pools grown from 256 to 1,024 candidates (Figure 2b), resampling with replacement repeats records, 11.4 per cent of the slots in a 64-candidate draw from 256 and 3.0 from 1,024, which leaves fewer distinct members to rank. On plain subsets the change in recoverable share is not separated from zero while regret on misses rises 4.07 and 4.43 points, the fixed pool-shortlist size meeting an occupancy the threshold rule does not imply.

A fourth arm cuts at the population quantile. Its threshold then does not depend on the pool, so on subsets of an i.i.d. pool each of its moments is unbiased for fresh generation against that deployment threshold, and its shares are ratios of unbiased moments. It needs the population quantile of the generator's score distribution, so it appears in simulation only.

Table 6 compares them with fresh generation on six quantities: the three parts of the split, the recoverable share, the admission share and total regret. Candidates are generated afresh, each arm is applied to a logged pool drawn from the same generator, and an arm covers a cell of a quantity when its interval contains the fresh-sampling value there. For each simulated pool, M draws come from a bivariate standard normal at latent correlation 0.5 or 0.7. The with-replacement and plain-subset arms z-score the gold within the pool, as the cohort does, while the randomised-size and population-threshold arms keep the latent gold of the fresh-sampling reference. Each pair therefore differs in one factor, the draw law in the first and the shortlist construction in the second, and a contrast across the pairs combines protocol and normalisation. A z-score rescales every loss in a pool by one positive factor, which moves the four quantities in percentage points only through each pool's weight in the pooled ratio and moves the two in gold standard deviations directly. Cells are correlation by M by budget by width, pooling is ratio-of-sums over pools, and intervals are 2,000-

replicate percentile bootstraps over pools. The reference is a direct Monte Carlo of Best-of-N on N population draws against the population threshold.

At pools of at least 256 candidates the primary protocol covers all 16 recoverable-share and ordering cells, as the population-threshold arm does on the same pools and latent gold, with a largest recoverable-share gap of 0.26 points against that arm's 0.54 (0.75 on 128-candidate pools), and fewer admission-loss, admission-share and regret-on-misses cells than that arm. Of the two z-scored arms, plain subsets cover more cells than resampling with replacement on every quantity but regret on misses, where they cover none against 11 of 16, and carry the larger gap on admission loss and regret on misses, while resampling with replacement carries the larger gap on the recoverable share, the admission share, ordering loss and total regret. Total regret does not involve the shortlist, so there the plain-subset and primary arms share one draw law and differ only in the gold normalisation; both cover 14 of 16 cells, plain subsets with the smaller largest gap. For regret on misses the primary protocol covers 6 of 16 cells against the with-replacement arm's 11; at the cohort's own pool size of 256 the two read 0 of 8 against 3 of 8 covered cells, with mean signed gaps of -0.60 and -0.27 points.

Table 6: **Simulation diagnostics under two gold-normalisation conventions.** The first two arms z-score the gold within each pool, as the cohort does, and the last two keep the latent gold, so a contrast across the two groups combines protocol and normalisation. Entries count the 16 cells (pools of 256 and 1,024 candidates) whose 95% pool-bootstrap interval contains the fresh-sampling value; parentheses give the largest absolute gap, in points or, for the last two rows, gold standard deviations.

	z-scored gold		Latent gold	
Quantity	With replacement	Plain size-N subset	Randomised size (primary)	Population threshold (diagnostic)
Recoverable share	3/16 (3.00)	6/16 (1.75)	16/16 (0.26)	16/16 (0.54)
Regret on misses	11/16 (0.80)	0/16 (6.62)	6/16 (2.16)	14/16 (0.66)
Admission loss	3/16 (3.80)	7/16 (7.49)	10/16 (2.19)	14/16 (0.57)
Admission share	3/16 (5.75)	4/16 (4.01)	13/16 (0.95)	16/16 (0.51)
Ordering loss	4/16 (0.05)	8/16 (0.03)	16/16 (0.004)	16/16 (0.004)
Total regret	8/16 (0.018)	14/16 (0.008)	14/16 (0.010)	14/16 (0.010)

The arm choice under non-Gaussian references. The simulation above draws its pools from a Gaussian copula, and Table 26 shows that family does not reproduce the cohort's split, so the arm comparison is repeated under three further families calibrated to the same population rank agreement: Student-t with three degrees of freedom, Frank and Clayton, on the same cells, the same pool counts and the same fresh-sampling reference. The randomised-size arm covers at least 6 of the 8 recoverable-share cells under every family at both pool sizes the cohort uses, and its recoverable-share gap against fresh sampling moves with the family (Table 7). At the cohort's pool size of 256 its largest recoverable-share gap runs from 0.19 points under the Gaussian to 1.15 under Student-$t(3)$, and at $M = 1{,}024$ from 0.14 to 0.23.

Table 7: **Recoverable-share agreement with fresh sampling across copula families and pool sizes.** Of eight recoverable-share cells (two latent correlations, budgets 16 and 64, widths 1 and 4), Cells covered counts those whose 95% bootstrap interval for the randomised-size arm contains the fresh-sampling value; Gap is the largest absolute gap from that value. All four families match the same population rank agreement.

	$M = 256$		$M = 1{,}024$
Reference family	Cells covered	Gap (points)	Cells covered
Gaussian	8	0.19	8
Student-$t(3)$	6	1.15	8
Frank	7	0.97	8
Clayton	7	0.58	8

The gaps in Tables 6 and 7 exceed the fresh-sampling reference's own Monte Carlo error by one to two orders of magnitude (Table 8), and each combines the arm's finite-pool bias with pool-sampling

spread. Two independent pool draws of the identical Gaussian design move the primary protocol's simulated recoverable share by up to 0.84 points at $M = 1{,}024$. In the simulation of Table 6 the primary protocol's largest gap falls to 0.4 points on admission loss and 0.6 on regret on misses at $M = 1{,}024$, and a run at $M = 16{,}384$ puts it below 0.3 points on every quantity in percentage points.

Table 8: **The simulation resolves gaps of a few tenths of a point.** The primary protocol's median 95% pool-bootstrap half-width over the eight $M = 256$ cells of Table 6 (2,000 pools each), and how many of them it covers when rerun with a new pool seed. The rerun's fresh-sampling reference has a Monte Carlo standard error of 0.011 points on the recoverable share and 0.021 on admission loss at $N = 64$, $c = 1$ and latent correlation 0.5.

Quantity	Half-width (points)	Cells covered on rerun
Recoverable share	0.29	8
Admission loss	0.31	4
Regret on misses	0.14	4
Admission share	0.37	6

The split-sample population-threshold arm, and why it moves the width map at width 1. The population-threshold arm needs a quantile of the generator's score distribution, which a logged pool does not carry. Its feasible version fits the threshold on one random half of the records and draws from the other half, averaged over eight independent splits; exchanging the two halves' roles within each split moves the budget-64, width-1 recoverable share by -0.03. Table 9 gives what that construction costs.

At a budget of 64 and width 1 the fitted threshold admits about three of the other half's 128 records where its rule asks for two. Exchangeability alone predicts this excess for the plug-in quantile. Drawing exchangeable records, with no pool data, gives 2.955, 8.870 and 32.497 admitted records against nominal 2, 8 and 32, and the cohort measures 2.996, 8.932 and 32.693. The predicted excess is 47.8, 10.9 and 1.6 per cent of nominal mass at widths 1, 4 and 16. At width 1 the mass step accounts for the difference in Table 9, while at widths 4 and 16 the rest component offsets much of that step. Fitting the threshold in sample, or on the full pool, leaves the realised size at 2.03 and the recoverable share at 8.47 and 9.71 per cent.

The two arms differ in pool and in shortlist: the primary protocol draws 64 of 256 records with a shortlist of 4.00 of them and the split-sample arm draws 64 of 128 with a realised shortlist of 3.00, so a draw holds one member in expectation under the first and about 1.5 under the second. Two comparisons that separate the halved pool from the fitted threshold give a difference of similar size. On the same half and the same splits the gap is $+7.95$ points, positive in all 45 configurations. In a simulation where the population threshold is known and fresh Best-of-N can be drawn, halving the pool with the threshold held known moves the width-1 recoverable share by $+0.45$ and -0.31 points at latent correlations of 0.5 and 0.7 and leaves its interval covering the truth. Fitting the threshold on the discarded half instead moves it $+8.39$ and $+9.55$ and puts the arm last of the four. The split-sample arm's 17.73 per cent measures a different shortlist policy, one with a larger realised mass.

Over the ten proxy-gold pairs of Section J.3, the wild cluster bootstrap p of the configuration-weighted mean difference is 0.001953 in both width-1 cells and 0.0039 at a budget of 64 and width 4, against 0.5586 and 0.4336 in the two widest cells, where five pairs sit on each side. At width 1 the mass component has the same p at both budgets, with every configuration positive, and the pool and rest components have p from 0.1016 to 0.5273.

Table 9: **At width 1 the mass step accounts for the difference.** Recoverable shares are cohort medians over the 45 configurations; *primary* is the primary protocol on the full pool. Differences are split-sample minus primary per configuration; the first is their median, with a 95% prompt-cluster bootstrap interval (2,000 replicates), the others configuration-weighted means. In turn, *pool* restricts the primary protocol to the half the split-sample arm draws from, *mass* matches its expected shortlist size to that arm's realised size, and *rest* completes the total.

		Recoverable share (%)		Paired difference (points)				
N	c	Primary	Split-sample	Total (median)	Pool	Mass	Rest	Total
64	1	10.38	17.73	$+7.60\,[+7.19, +7.95]$	-0.13	$+7.80$	-0.05	$+7.62$
64	4	51.97	52.99	$+1.31\,[+1.12, +1.73]$	-0.13	$+3.27$	-1.67	$+1.47$
64	16	87.66	87.63	$+0.10\,[-0.05, +0.20]$	$+0.02$	$+0.29$	-0.24	$+0.07$
16	1	13.47	15.81	$+2.65\,[+2.49, +2.81]$	-0.03	$+2.56$	$+0.12$	$+2.66$
16	4	67.53	67.05	$+0.07\,[-0.05, +0.26]$	-0.06	$+0.72$	-0.52	$+0.14$

D PROOFS

D.1 THE ORACLE CEILING

Proof of Proposition 3.1. Write $I := \{i \le N : Y_i \in \mathcal{T}\}$ for the indices of drawn shortlist members, $\mathcal{E} := \{I \neq \varnothing\}$ for the hit event with $Z := \Pr(\mathcal{E})$, and $M^{\mathrm{in}} := \max_{i \in I} R^\star(Y_i)$ on $\mathcal{E}$. The ordering loss is $\Lambda_N = \mathbb{E}[M^{\mathrm{in}} - R^\star(\widehat{Y}_N) \mid \mathcal{E}]$, non-negative because under the tie conventions of Section C.2 the Best-of-N pick $\widehat{Y}_N$ is a member on every hit.

Compare H with the pick on each event. On $\mathcal{E}$ the selector H returns a member, so $R^\star(H) - R^\star(\widehat{Y}_N) \le M^{\mathrm{in}} - R^\star(\widehat{Y}_N)$. On $\mathcal{E}^c$ we have $H = \widehat{Y}_N$ and the gain is zero. Both selectors have finite expected regret, which is what allows the expectation of their reward difference to be taken, and gives $\mathbb{E}[R^\star(H) - R^\star(\widehat{Y}_N)] \le Z\Lambda_N$.

For the bound to be attained, return a member with the largest gold reward on $\mathcal{E}$ and $\widehat{Y}_N$ otherwise. This selector meets the pointwise bound on every hit and has the same zero gain on every miss. Its regret is non-negative and never exceeds that of Best-of-N, whose expected regret is finite by assumption, so it too has finite expected regret and lies in the class. Equality follows.

D.2 THE SPLIT ON HITS AND THE ORDERING CERTIFICATE

At one expected candidate, draws with a single shortlist member account for most hits (Figure 5a), and the resulting coefficient stays bounded as the budget grows (panel b).

Use the sampling setup of Section 3 and the shortlist conventions of Section C.2. Let $I := \{i \le N : Y_i \in \mathcal{T}_{1/N}\}$ be the indices of the drawn shortlist members, so $K_N = |I|$ and $\mathcal{E}_N = \{K_N \ge 1\}$. On this event,

$$M_N^{\mathrm{in}} = \max_{i \in I} R^\star(Y_i), \qquad \Xi_N = \mathbb{E}\big[R^\star(Y_N^\star) - M_N^{\mathrm{in}} \,\big|\, \mathcal{E}_N\big], \qquad \Lambda_N = \mathbb{E}\big[M_N^{\mathrm{in}} - R^\star(\widehat{Y}_N) \,\big|\, \mathcal{E}_N\big]. \tag{3}$$

Proposition D.1 (Admission and ordering loss split regret on hits)**.** *For every $N \ge 2$,*

$$\mathcal{R}_N \;=\; \mathbb{E}\big[R^\star(Y_N^\star) - R^\star(\widehat{Y}_N) \,\big|\, \mathcal{E}_N\big] \;=\; \Xi_N + \Lambda_N, \tag{4}$$

and $\Xi_N \ge 0$, $\Lambda_N \ge 0$. These expectations of non-negative gaps are well defined in $[0, \infty]$.

Proof. On $\mathcal{E}_N$ the gold-best candidate $Y_N^\star$ maximises $R^\star$ over all N drawn candidates, while the Best-of-N pick lies in the shortlist, so $M_N^{\mathrm{in}} \ge R^\star(\widehat{Y}_N)$. Both gaps in the telescoping identity $R^\star(Y_N^\star) - R^\star(\widehat{Y}_N) = \big(R^\star(Y_N^\star) - M_N^{\mathrm{in}}\big) + \big(M_N^{\mathrm{in}} - R^\star(\widehat{Y}_N)\big)$ are therefore non-negative. Applying $\mathbb{E}[\,\cdot \mid \mathcal{E}_N]$ gives Equation (4) with $\Xi_N \ge 0$ and $\Lambda_N \ge 0$; and because the gaps are non-negative, their conditional expectations and the identity stay well defined in $[0, \infty]$ with no integrability hypothesis. The ordering gap is non-negative because the shortlist is an upper set of the proxy order, as is every shortlist in this paper.

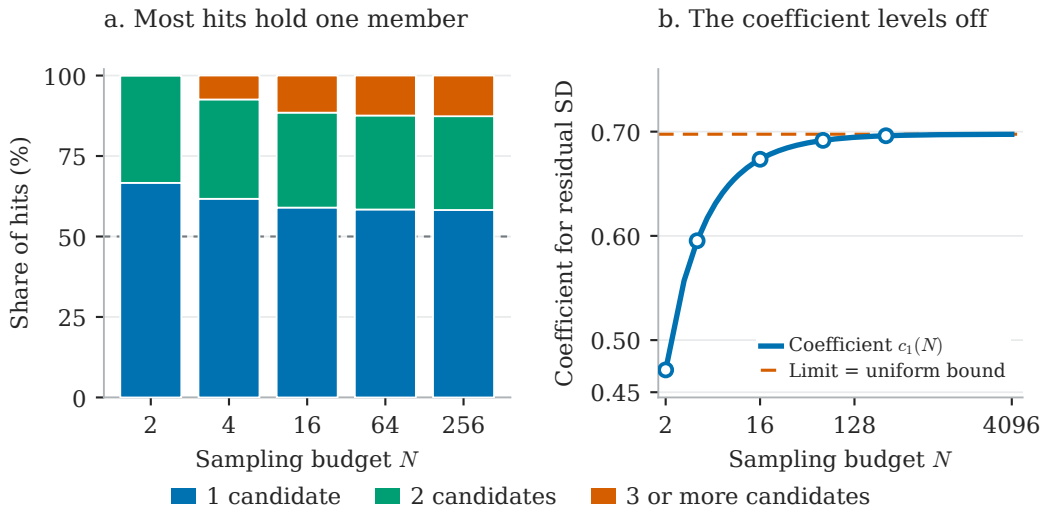


Figure 5: **At one expected candidate most hits hold a single shortlist member, and $c_1(N)$ stays below 0.698. (a)** Shortlist members on a hit under $\mathrm{Bin}(N, 1/N)$ occupancy; the singleton share tends to $1/(e-1) \approx 58.20\%$. **(b)** $c_1(N)$ of Equation (2), with circles at the budgets of panel a; its limit is $\sqrt{2(e-2)}/(e-1) \approx 0.69754$.

Theorem D.2 (Residual spread bounds ordering loss)**.** *Let $Y_1, \ldots, Y_N$ be i.i.d. copies of $Y \sim \pi_{\mathrm{ref}}$, and assume $\Pr[Y \in \mathcal{T}_{1/N}] = 1/N$, as for a continuous proxy distribution or a shortlist defined by randomised ranks at threshold atoms. Let $\xi := \widehat{R} - R^\star$ and let $\mu_{1/N}$ be the law of Y on this shortlist. Write $\tau_{\xi,1/N} := \sqrt{\mathrm{Var}_{\mu_{1/N}}(\xi)}$ for the residual standard deviation and $\sigma_{\xi,1/N} := \sqrt{\mathbb{E}_{\mu_{1/N}}[\xi^2]}$ for its RMS. If $\sigma_{\xi,1/N} < \infty$, then for every $N \geq 2$,*

$$\Lambda_N \;\leq\; c_1(N)\,\tau_{\xi,1/N} \;\leq\; c_1(N)\,\sigma_{\xi,1/N}, \qquad c_1(N) := \frac{\sqrt{2a_N(Z_N - b_N)}}{Z_N}, \tag{5}$$

where $a_N := (1-1/N)^N$, $b_N := (1-1/N)^{N-1}$, and $Z_N := 1 - a_N$ are respectively $\Pr[K_N = 0]$, $\Pr[K_N = 1]$, and $\Pr[K_N \geq 1]$ for $K_N \sim \mathrm{Bin}(N, 1/N)$. Uniformly over all finite budgets,

$$c_1(N) \;\leq\; c_1^\infty := \frac{\sqrt{2(e-2)}}{e-1} \;<\; 0.698, \qquad c_1(N) \longrightarrow c_1^\infty. \tag{6}$$

Let $\mathcal{G}_N$ consist of strictly increasing real-valued functions g on $\mathbb{R}$ with $g(\widehat{R}) - R^\star \in L^2(\mu_{1/N})$. This class contains the identity and is closed under additive shifts. Its maps preserve both losses, so

$$\Lambda_N \;\leq\; c_1(N) \inf_{g \in \mathcal{G}_N} \tau_{g(\widehat{R}) - R^\star, 1/N} \;=\; c_1(N) \inf_{g \in \mathcal{G}_N} \sigma_{g(\widehat{R}) - R^\star, 1/N}. \tag{7}$$

All residual scales are evaluated under the original $\mu_{1/N}$.

Proof. We first bound the ordering gap on a realised set of members, then average over the occupancy to prove Equation (5). The uniform coefficient bound and the rank-preserving optimisation follow.

From the ordering gap to residual spread. Work on $\mathcal{E}_N$ and condition on I. Every shortlist member ranks above every other candidate, using the same keys if membership is randomised. Thus the global proxy maximiser coincides with the best member: $\widehat{Y}_N = Y_j$ with $j := \arg\max_{i \in I} \widehat{R}(Y_i)$. Let $a := \arg\max_{i \in I} R^\star(Y_i)$, so $M_N^{\mathrm{in}} = R^\star(Y_a)$. Writing $R^\star = \widehat{R} - \xi$ and using $\widehat{R}(Y_j) \geq \widehat{R}(Y_a)$ by the definition of j,

$$\begin{aligned} M_N^{\mathrm{in}} - R^\star(\widehat{Y}_N) &= \big(\widehat{R}(Y_a) - \widehat{R}(Y_j)\big) + \big(\xi(Y_j) - \xi(Y_a)\big) \\ &\leq \xi(Y_j) - \xi(Y_a) \leq \max_{i \in I} \xi(Y_i) - \min_{i \in I} \xi(Y_i). \end{aligned} \tag{8}$$

For any $k \geq 2$ numbers, with $\bar{\xi} = k^{-1} \sum_{i=1}^{k} \xi_i$,

$$\max_i \xi_i - \min_i \xi_i \leq \sqrt{2 \sum_{i=1}^{k} (\xi_i - \bar{\xi})^2}.$$

Subtracting the sample mean leaves the range unchanged. Applying $(u - v)^2 \leq 2(u^2 + v^2)$ to the largest and smallest centred values gives this inequality. The i.i.d. draw enters at the next step and through the binomial law of K_N. Conditionally on $K_N = k$, the shortlist members are i.i.d. $\mu_{1/N}$, so $\mathbb{E}[\sum_{i=1}^{k} (\xi_i - \bar{\xi})^2 \mid K_N = k] = (k-1)\tau^2_{\xi,1/N}$, an expectation the residual second-moment hypothesis makes finite. Jensen's inequality therefore bounds their expected ordering gap by $\sqrt{2(k-1)}\,\tau_{\xi,1/N}$. For $k = 1$ both the range and the ordering gap are zero. Since the $K_N = 1$ term vanishes, $\mathbb{E}[\sqrt{K_N - 1} \mid K_N \geq 1] = \mathbb{E}[\sqrt{K_N - 1}\,\mathbf{1}\{K_N \geq 2\} \mid K_N \geq 1]$, and Cauchy–Schwarz applied to that product gives

$$\begin{aligned}
\Lambda_N &\leq \sqrt{2}\,\tau_{\xi,1/N} \mathbb{E}[\sqrt{K_N - 1} \mid K_N \geq 1] \\
&\leq \tau_{\xi,1/N} \sqrt{2 \Pr[K_N \geq 2 \mid K_N \geq 1]\, \mathbb{E}[K_N - 1 \mid K_N \geq 1]} \\
&= \tau_{\xi,1/N} \frac{\sqrt{2 a_N (Z_N - b_N)}}{Z_N},
\end{aligned}$$

where $\mathbb{E}[K_N] = 1$ implies $\mathbb{E}[K_N - 1 \mid K_N \geq 1] = Z_N^{-1} - 1 = a_N / Z_N$.

Uniform control of the coefficient. The two occupancy factors of the preceding bound are controlled separately. First, $a_N \leq e^{-1}$ gives $a_N / Z_N \leq (e-1)^{-1}$. Also

$$\frac{\Pr[K_N = k]}{\Pr[K_N = 1]} = \frac{1}{k!} \prod_{j=1}^{k-1} \frac{N - j}{N - 1} \leq \frac{1}{k!}, \qquad 1 \leq k \leq N,$$

so $Z_N / b_N \leq e - 1$ and hence $1 - b_N / Z_N \leq (e-2)/(e-1)$. Multiplying these two inequalities proves $c_1(N)^2 \leq 2(e-2)/(e-1)^2$ for every $N \geq 2$. Both a_N and b_N converge to e^{-1}, which gives the limit. Rational bounds on e give $c_1^{\infty} < 0.698$.

The same residual-range argument applies at any exact shortlist mass $p \in (0, 1]$, because nothing in it uses the value $1/N$ beyond the occupancy law, so only the occupancy factors change, and $K_N \sim \mathrm{Bin}(N, p)$ gives Equation (9). Randomised membership within the threshold atom, using the same independent uniform keys that break argmax ties, restores the exact-mass statement at that p.

Rank-preserving transformations. For Equation (7), fix $g \in \mathcal{G}_N$. For every r, order preservation gives $F_{g(\widehat{R})}(g(r)) = F_{\widehat{R}}(r)$ and the same equality for left limits. At an atom the randomised rank $F_{\widehat{R}}(r^-) + U\{F_{\widehat{R}}(r) - F_{\widehat{R}}(r^-)\}$, with independent $U \sim \mathrm{Unif}(0, 1)$, is also invariant. The upper-quantile shortlists and sample argmax therefore agree almost surely under the same keys. This preserves $I, K_N, \mathcal{E}_N, \mu_{1/N}, \Xi_N$ and Λ_N. Applying Equation (5) to $g(\widehat{R}) - R^{\star}$ proves the variance bound for each g. Taking the infimum gives the standard-deviation bound.

To equate the two infima, let $m_g := \mathbb{E}_{\mu_{1/N}}[g(\widehat{R}) - R^{\star}]$. The map $\widetilde{g}(r) := g(r) - m_g$ also belongs to $\mathcal{G}_N$ and satisfies $\sigma_{\widetilde{g}(\widehat{R}) - R^{\star}, 1/N} = \tau_{g(\widehat{R}) - R^{\star}, 1/N}$. Thus the RMS infimum is no larger than the standard-deviation infimum. The reverse inequality follows from $\tau_{g(\widehat{R}) - R^{\star}, 1/N} \leq \sigma_{g(\widehat{R}) - R^{\star}, 1/N}$ for each g.

The infimum is the same over nondecreasing maps: for any such h with finite residual norm, $g_\eta(r) := h(r) + \eta \arctan r$ is strictly increasing for $\eta > 0$ and $\|g_\eta(\widehat{R}) - h(\widehat{R})\|_{L^2(\mu_{1/N})} \leq \eta\pi/2$. This bound tends to zero as η decreases to zero. On a finite sample, isotonic regression attains the nondecreasing least-squares minimum, grouping equal original proxy scores. Because constant shifts are allowed, the fitted residuals have zero empirical mean and their RMS equals the infimal empirical standard deviation. This also holds for weighted least squares when fitting and residual scales use the same weights. The fitted map can merge distinct proxy scores into one level, so it may tie distinct proxy scores; its strictly increasing perturbations g_η approach its residual RMS while preserving all original ranks, so the fit computes the infimum over rank-preserving maps.

Corollary D.3 (The coefficient at a general shortlist mass)**.** *In the i.i.d. setting of Theorem D.2, let the shortlist carry exact mass $p \in (0,1]$, so that $K_N \sim \mathrm{Bin}(N,p)$, and let $\tau_{\xi,p}$ be the residual standard deviation under the law of a candidate on it. Then, for every $N \geq 2$,*

$$\Lambda_N \;\leq\; c_1(N,p)\,\tau_{\xi,p}, \qquad c_1(N,p) := \frac{\sqrt{2\,(1-p_0-p_1)(Np-1+p_0)}}{1-p_0}, \tag{9}$$

with $p_0 = (1-p)^N$ and $p_1 = Np(1-p)^{N-1}$. At $p = 1/N$ this is $c_1(N)$, and at $p = 1$, where the shortlist is the whole draw, it is $\sqrt{2(N-1)}$.

Remark D.4 (Singleton occupancy). The ordering gap vanishes when $K_N = 1$, since the single shortlist member is both the proxy's pick and the best member. On the conditioning event,

$$\Pr[K_N = 1 \mid K_N \geq 1] = \frac{(1-1/N)^{N-1}}{1-(1-1/N)^N} \;\xrightarrow[N\to\infty]{}\; \frac{e^{-1}}{1-e^{-1}} \;\approx\; 58.20\%. \tag{10}$$

The proof above shows that this probability is at least $(e-1)^{-1}$ at every $N \geq 2$. Thus the per-draw ordering gap can be nonzero on at most $1-(e-1)^{-1} \approx 41.80\%$ of the draws that reach the shortlist.

Remark D.5 (Rank invariance). The shortlist, the hit event, regret on misses and both losses depend on the proxy only through its ordering, as in the ordinal optimisation of Ho et al. (1992).

Proof of Theorem 3.2. The independent uniform keys give the shortlist the exact mass c/N that the statement assumes, and because the same keys decide shortlist membership and break the tie in the argmax, the Best-of-N pick is a member on every hit. With $\mathbb{E}|R^\star| < \infty$ and $\mathbb{E}[\xi^2 \mid Y \in \mathcal{T}_{c/N}] < \infty$ the three expectations $\mathcal{R}_N$, Ξ_N and Λ_N of Proposition D.1 are finite.

Apply Theorem D.2 at shortlist mass $p = c/N$, in the general-mass form of Corollary D.3. Its occupancy probabilities are $p_0 = \Pr[K=0] = (1-c/N)^N = 1 - Z_c$ and $p_1 = \Pr[K=1] = Np(1-p)^{N-1} = c(1-c/N)^{N-1} = b_c$, while $Np - 1 + p_0 = c - Z_c$, so

$$c_1(N, c/N) = \frac{\sqrt{2\,(1-p_0-p_1)(Np-1+p_0)}}{1-p_0} = \frac{\sqrt{2\,(Z_c-b_c)(c-Z_c)}}{Z_c} = c_c(N),$$

which is the coefficient of Equation (2) and reduces to $c_1(N)$ at $c = 1$. Its two factors are the probability that a hit holds at least two members, $\Pr[K \geq 2 \mid K \geq 1] = (Z_c - b_c)/Z_c$, and the expected occupancy of a hit beyond the first, $\mathbb{E}[K-1 \mid K \geq 1] = (c - Z_c)/Z_c$, since $\mathbb{E}[K] = c$. By Proposition D.1, regret on hits is then at most $\Xi_N + c_c(N)\tau_{\xi,c/N}$.

The draw law of the experiments. Each prompt carries a logged pool of M records with proxy scores $\widehat{R}$, gold rewards $R^\star$ and residuals $\xi := \widehat{R} - R^\star$. Fix the total order $(\widehat{R}, U)$ of Theorem 3.2, with one independent key $U \sim \mathrm{Unif}(0,1)$ per record used both for the shortlist and for the argmax. A draw is a uniformly random size-N subset D of the pool, taken *without replacement*, and $\widehat{Y}_N$ maximises the order over D. A shortlist is an upper set of the same order: the top m records, written S_m. Given S_m let $K := |D \cap S_m|$, and on $\{K \geq 1\}$ let

$$M^{\mathrm{in}} := \max_{i \in D \cap S_m} R^\star(y_i), \qquad \Lambda_m := \mathbb{E}\big[M^{\mathrm{in}} - R^\star(\widehat{Y}_N) \,\big|\, K \geq 1\big].$$

Write $\bar{\xi}_m := m^{-1}\sum_{i\in S_m}\xi_i$ and

$$\tau_m^2 := \frac{1}{m}\sum_{i\in S_m}(\xi_i - \bar{\xi}_m)^2, \qquad s_m^2 := \frac{1}{m-1}\sum_{i\in S_m}(\xi_i - \bar{\xi}_m)^2 = \frac{m}{m-1}\,\tau_m^2, \tag{11}$$

the population residual standard deviation over the finite shortlist and its Bessel-corrected counterpart, the second for $m \geq 2$ only; at $m = 1$ the population spread is $\tau_1 = 0$. With the pool as the population, τ_m is the finite-pool counterpart of $\tau_{\xi,c/N}$, and s_m differs from it by the divisor.

Lemma D.6 (Ordering certificate for a fixed shortlist under subset draws)**.** *Let $1 \leq m \leq M$ and $2 \leq N \leq M$. Then $K \sim \mathrm{Hyp}(M,m,N)$; write*

$$q_m := \frac{Nm}{M} = \mathbb{E}[K], \qquad Z_m := \Pr[K \geq 1] = 1 - \frac{\binom{M-m}{N}}{\binom{M}{N}},$$

$$b_m := \Pr[K = 1] = \frac{m\binom{M-m}{N-1}}{\binom{M}{N}}.$$

Then $\Lambda_1 = 0$*, and for every* $m \geq 2$

$$\Lambda_m \;\leq\; c_m^{\mathrm{fp}}\,\tau_m, \qquad c_m^{\mathrm{fp}} := \sqrt{\frac{m}{m-1}}\;\frac{\sqrt{2\,(q_m - Z_m)(Z_m - b_m)}}{Z_m}. \tag{12}$$

Equivalently, against the Bessel-corrected spread the factor disappears, since $c_m^{\mathrm{fp}}\tau_m = \beta_m s_m / Z_m$ *with* $\beta_m := \sqrt{2(q_m - Z_m)(Z_m - b_m)}$*. Uniformly in* M *and* N*,*

$$c_m^{\mathrm{fp}} \;\leq\; \sqrt{2\,\frac{m}{m-1}\Big(\frac{q_m}{1 - e^{-q_m}} - 1\Big)} \;<\; \sqrt{\frac{2m\,q_m}{m-1}} \;\leq\; 2\sqrt{q_m}. \tag{13}$$

Proof. Fix the pool and its total proxy order. The pathwise half of the ordering-certificate argument does not depend on the draw law. The shortlist is an upper set of the order the pick maximises, so $\widehat{Y}_N \in D \cap S_m$ on $\{K \geq 1\}$, the gap $M^{\mathrm{in}} - R^\star(\widehat{Y}_N)$ is at most $\max_{i \in D\cap S_m} \xi_i - \min_{i \in D\cap S_m} \xi_i$, and for $k \geq 2$ values that range is at most $\sqrt{2S}$ with S their centred sum of squares. Sampling without replacement changes the occupancy law, the expected centred sum of squares and the uniform bound on $\Pr[K = 0]$.

Occupancy. D is a uniformly random size-N subset of the pool, so $K = |D \cap S_m|$ is hypergeometric with the stated q_m, Z_m, b_m, and given $K = k$ the set $D \cap S_m$ is a simple random sample of size k drawn without replacement from S_m. For $m = 1$ this forces $K \leq 1$, the range is zero and $\Lambda_1 = 0$.

Conditional second moment. Let $k \geq 2$ and let $\xi_1, \ldots, \xi_k$ be the residuals of a simple random sample of size k without replacement from S_m, with mean $\bar{\xi}_{(k)}$. Since $\sum_{i \leq k}(\xi_i - \bar{\xi}_{(k)})^2 = \frac{1}{2k}\sum_{i,j \leq k}(\xi_i - \xi_j)^2$ and each ordered pair $i \neq j$ is a uniformly random ordered pair of distinct records of S_m,

$$\mathbb{E}\Big[\sum_{i \leq k}(\xi_i - \bar{\xi}_{(k)})^2\Big] = \frac{k(k-1)}{2k} \cdot \frac{1}{m(m-1)}\sum_{a \neq b}(\xi_a - \xi_b)^2 = (k-1)\,s_m^2 = (k-1)\,\tfrac{m}{m-1}\,\tau_m^2,$$

using $\frac{1}{m(m-1)}\sum_{a \neq b}(\xi_a - \xi_b)^2 = 2s_m^2$. This is the only place the factor $m/(m-1)$ enters. Jensen's inequality for the square root then bounds the expected gap given $K = k$ by $\sqrt{2(k-1)\,m/(m-1)}\,\tau_m$, which reads $0 \leq 0$ at $k = 1$.

Averaging over hits. Exactly as in Theorem D.2,

$$\begin{aligned}\Lambda_m &\leq \sqrt{\tfrac{2m}{m-1}}\,\tau_m\,\mathbb{E}\big[\sqrt{K-1}\,\mathbf{1}\{K \geq 2\} \mid K \geq 1\big] \\ &\leq \sqrt{\tfrac{2m}{m-1}}\,\tau_m \sqrt{\Pr[K \geq 2 \mid K \geq 1]\;\mathbb{E}[(K-1)\mathbf{1}\{K \geq 2\} \mid K \geq 1]},\end{aligned}$$

with $\Pr[K \geq 2 \mid K \geq 1] = (Z_m - b_m)/Z_m$ and, since $\mathbb{E}[K\mathbf{1}\{K \geq 1\}] = \mathbb{E}[K] = q_m$, $\mathbb{E}[(K-1)\mathbf{1}\{K \geq 2\} \mid K \geq 1] = (q_m - Z_m)/Z_m$. This is Equation (12).

Uniform bound. The step $a_N \leq e^{-1}$ of Theorem D.2, in its mass-p form, has no hypergeometric counterpart; for subset draws the bound $Z_m \geq 1 - e^{-q_m}$ below takes its place. If $N > M - m$ then no size-N subset avoids S_m, so $\Pr[K = 0] = 0$, $Z_m = 1$ and $Z_m \geq 1 - e^{-q_m}$ holds at once. Suppose therefore $N \leq M - m$, so that $M - m - j \geq 1$ for every $0 \leq j \leq N - 1$ and the factors below are all positive; only then may the termwise comparison be multiplied out. For $0 \leq j \leq N - 1$, $\frac{M-m-j}{M-j} = 1 - \frac{m}{M-j} \leq 1 - \frac{m}{M}$, so $\Pr[K = 0] = \prod_{j=0}^{N-1}\frac{M-m-j}{M-j} \leq (1 - \frac{m}{M})^N \leq e^{-q_m}$ and $Z_m \geq 1 - e^{-q_m}$. With $(Z_m - b_m)/Z_m \leq 1$ and $(q_m - Z_m)/Z_m = q_m/Z_m - 1$ this gives the first inequality of Equation (13). The second is $q_m < (1 + q_m)(1 - e^{-q_m})$, that is $e^{q_m} > 1 + q_m$, and the third is $m/(m-1) \leq 2$ for $m \geq 2$.

Remark D.7 (Why the i.i.d. coefficient does not apply to subset draws). Under a subset draw the finite-population factor and the occupancy law change together. Equation (12) applies $\sqrt{m/(m-1)}$ to the hypergeometric coefficient β_m/Z_m, not to $c_{q_m}(N)$, and $\sqrt{m/(m-1)}\,c_{q_m}(N)$ need not bound c_m^{fp}, because b_m/Z_m can fall below its binomial counterpart when N is close to M; at $(M, N, m) = (3, 3, 2)$ the ratio $c_m^{\mathrm{fp}}/(\sqrt{m/(m-1)}\,c_{q_m}(N))$ is 1.0987. Nor does $c_c(N)$ itself carry over, since the adversarial search below exceeds it under subset draws.

Theorem D.8 (Ordering certificate under a random shortlist size)**.** *Let $c \in (0, N]$, let $m \sim \mathrm{Bin}(M, c/N)$ be drawn independently of D, and let the shortlist be the top m records, so that $K \mid m \sim \mathrm{Hyp}(M, m, N)$. Then $K \sim \mathrm{Bin}(N, c/N)$ exactly: the hit event has the i.i.d. probability $Z_c = 1 - (1 - c/N)^N$ and $\Pr[K = 1] = b_c$. Let $\Lambda := \mathbb{E}[M^{\mathrm{in}} - R^\star(\widehat{Y}_N) \mid K \geq 1]$, averaged over both the size and the draw, and let*

$$w_m := \Pr[\, m \mid K \geq 1\,] = \frac{P(m) Z_m}{Z_c}, \qquad P(m) := \binom{M}{m} \Big(\frac{c}{N}\Big)^m \Big(1 - \frac{c}{N}\Big)^{M-m}.$$

Then, with c_m^{fp} and τ_m of Lemma D.6 and $c_m^{\mathrm{fp}} = 0$ for $m \leq 1$,

$$\Lambda \;\leq\; \sum_{m=2}^{M} w_m \, c_m^{\mathrm{fp}} \, \tau_m \;=\; \frac{1}{Z_c} \sum_{m=2}^{M} P(m) \sqrt{2 \, \tfrac{m}{m-1} (q_m - Z_m)(Z_m - b_m)} \; \tau_m. \tag{14}$$

Consequently

$$\Lambda \;\leq\; C(M, N, c)\, \bar{\tau}, \qquad C(M, N, c) := \Big(\sum_{m \geq 2} w_m \big(c_m^{\mathrm{fp}}\big)^2 \Big)^{1/2}, \qquad \bar{\tau} := \Big(\sum_{m \geq 1} w_m \tau_m^2 \Big)^{1/2}, \tag{15}$$

the quadratic means of the conditional coefficients and of the conditional residual spreads under the same law w. Against the Bessel-corrected spread s_m the coefficient is smaller and is controlled by the i.i.d. one:

$$\Lambda \;\leq\; \frac{1}{Z_c} \sum_{m \geq 2} P(m)\, \beta_m \, s_m, \qquad \frac{1}{Z_c} \sum_{m \geq 2} P(m)\, \beta_m \;\leq\; c_c(N). \tag{16}$$

In particular, if s_m does not vary with m, then $\Lambda \leq c_c(N)\, s$. Finally $C(M, N, c) \leq 2\sqrt{(c - Z_c)/Z_c}$, and $C(M, N, c)$ tends to $c_c(N)$ as $M \to \infty$ with N and c held fixed.

Proof. Occupancy. Because D is a uniformly random size-N subset, the law of $|D \cap S|$ depends on the shortlist S only through $|S|$, so $K \mid m$ is $\mathrm{Hyp}(M, m, N)$ although S_m is the deterministic top m. Write $p := c/N$ and mark each of the M records independently with probability p: the number of marks is $\mathrm{Bin}(M, p)$, given that number the marked set is a uniform subset of that size, and $|D \cap \{\text{marked}\}|$ is the number of marks among the N records of D, which is $\mathrm{Bin}(N, p)$ since the marks are i.i.d. and independent of D. Reading the same quantity in the two orders gives $\sum_m P(m) \Pr[K = j \mid m] = \binom{N}{j} p^j (1 - p)^{N-j}$ for every j. The marking is a device for evaluating that mixture; the shortlist is still the deterministic top m records. Hence $\Pr[K \geq 1] = Z_c$, $\Pr[K = 1] = b_c$, and $\sum_m P(m) Z_m = Z_c$, so w is a probability law and $\sum_m P(m) q_m = c$, $\sum_m P(m) b_m = b_c$.

Mixture bound. Condition on m and use $\Pr[K \geq 1 \mid m] = Z_m$. At $m = 0$ the shortlist is empty and $\Pr[K \geq 1 \mid m = 0] = 0$, so that term contributes nothing and Λ_0 is never needed. Hence

$$Z_c \Lambda = \sum_{m \geq 1} P(m)\, Z_m \, \Lambda_m \leq \sum_{m \geq 2} P(m)\, Z_m \, c_m^{\mathrm{fp}} \, \tau_m$$

by Lemma D.6, the $m \leq 1$ terms vanishing. Dividing by Z_c gives Equation (14). Equation (15) is Cauchy–Schwarz under w, taking the coefficient and the residual spread in the same weighted quadratic mean.

Bessel-corrected residual scale. Substituting $\tau_m = s_m \sqrt{(m-1)/m}$ in Equation (14) gives the first part of Equation (16). For the second, Cauchy–Schwarz on the sum and the three moment identities above give

$$\sum_m P(m)\, \beta_m \leq \sqrt{2 \sum_m P(m)(q_m - Z_m) \; \sum_m P(m)(Z_m - b_m)} = Z_c \, c_c(N).$$

Envelope and limit. In Equation (15), $m/(m-1) \leq 2$ and $(Z_m - b_m)/Z_m \leq 1$ give $C^2 \leq \frac{1}{Z_c} \sum_m P(m)\, 4(q_m - Z_m) = 4(c - Z_c)/Z_c$. For the limit, write $C^2 = \frac{1}{Z_c} \sum_{m \geq 2} P(m)\, Z_m \, (c_m^{\mathrm{fp}})^2$, an expectation under P rather than under the weights w, which themselves move with M. As

$M \to \infty$ with N, c fixed, $m/M \to c/N$ and $m \to \infty$ in P-probability, so $q_m \to c$, $m/(m-1) \to 1$ and the hypergeometric Z_m, b_m converge to the binomial Z_c, b_c, whence $Z_m(c_m^{\mathrm{fp}})^2 \to Z_c c_c(N)^2$ in probability. The envelope above bounds the summand uniformly in m and M, so the family is bounded and that convergence carries to the expectation. Hence $C \to c_c(N)$.

Remark D.9 (Values of the coefficients). At $M = 256$ and $N = 64$ the three coefficients are

Expected candidates c	1	2	4	8	16
Fixed shortlist, $c^{\mathrm{fp}}_{cM/N}$ (Equation (12))	0.68114	1.39796	2.48031	3.80045	5.52052
Random size, $C(M, N, c)$ (Equation (15))	0.85447	1.48392	2.48584	3.79919	5.52050
I.i.d., $c_c(N)$ (Equation (2))	0.69162	1.33888	2.38817	3.73875	5.47723

Each multiplies its own residual scale: at $c = 1$ the random-size coefficient is 0.85447 against the quadratic mean $\bar{\tau}$ and 0.68918 against a common Bessel-corrected spread s (Equation (16)). The bound of Equation (12) is attained at $M = N = 4$ and $m = 2$: give all four records proxy score 0, let the keys order them $1 < 2 < 3 < 4$ for both the shortlist and the argmax, and set their gold rewards to $(0, 0, 1, -1)$. Then $S_2 = \{3, 4\}$, the pick is record 4, the residuals on S_2 are -1 and 1, and $K \equiv 2$, $Z_2 = 1$, $\tau_2 = 1$, $\Lambda_2 = 2$ and $c_2^{\mathrm{fp}} = 2$. Equality rests on the proxy tie between the pick and the gold-best member; with strictly ordered scores $(0, \epsilon, 2\epsilon, 3\epsilon)$ the bound is $2 + \epsilon$ against $\Lambda_2 = 2$, so equality is approached as $\epsilon \to 0$. The large-pool limit of Theorem D.8 is approached from above: at $N = 64$, $c = 1$, $C = 0.85447$, 0.77355, 0.70197 and 0.69226 at $M = 256$, 512, 4096 and 65536, against $c_1(64) = 0.69162$.

Numerical checks of the split and the finite-pool certificates. Enumeration on small pools closes Equation (1) to 1.6×10^{-13} over 707 cells. Under the primary protocol, no configuration and no prompt of the 8,928 exceeds the mixture bound of Equation (14) at any of the 14 width and budget cells evaluated, and at one expected candidate ordering loss uses a median 0.273, 0.295 and 0.332 of it at budgets of 4, 16 and 64. An adversarial search over 28,000 pools of three to nine records, under uniform size-N subset draws, exceeds the i.i.d. constant $c_c(N)$ in 14,170 of the 79,200 random-size cells it generates, by a factor of up to 2.25, and in 11,915 of the 190,400 fixed-size cells, by up to 1.5538. It never exceeds the bounds of Equations (12) and (14), whose largest attained ratio there is 0.99999983.

The coefficient, the finite-population form and the slack. The coefficient $c_c(N)$ of Equation (2) is bounded by $\sqrt{2(c/(1-e^{-c})-1)} < \sqrt{2c}$ for every $N \geq c$, since $c_c(N)^2 = 2(c/Z_c - 1)(1 - b_c/Z_c)$ and $Z_c \geq 1 - e^{-c}$, and Table 10 gives its supremum over all $N > c$, exact in rational arithmetic for $N \leq 1024$ and enclosed by interval bounds on $(1 - c/N)^N$ beyond, its value at a budget of 64 and the share of it that ordering loss uses, at each of the five widths. At the degenerate budget $N = c \geq 2$, which Theorem 3.2 admits, the shortlist is the whole draw: $Z_c = 1$, $b_c = 0$ and $c_c(c) = \sqrt{2(c-1)}$, which the displayed bound covers and which sits at or above that supremum, by 5 per cent at two expected candidates and by less than half a per cent from four upwards. Under a subset draw the coefficients are those of Lemma D.6 and Theorem D.8 (Remark D.7). The bound is taken at the realised shortlist mass rather than the nominal width, since the inclusive shortlists the estimator uses keep candidates tied at the threshold. No configuration-cell exceeds either form of it, 0 of 495 cells of configuration, width and budget. On the with-replacement draws under the learned gold, ordering loss uses a median 24.8% of the certificate at a budget of 64 and one expected candidate and 17.8% at sixteen. Strictly increasing proxy transforms preserve the shortlist, the pick and both losses, so the certificate can be minimised over them (Equation (7)); isotonic regression computes that infimum in sample and strictly increasing perturbations of the fit approach it (Section E).

D.3 Unconditional regret and marginal error

Existing bounds use different comparator policies and error measures (Table 11).

Lemma D.10 (Cancellation form of regret on misses)**.** *Pointwise on every draw,*

$$0 \leq R^\star(Y_N^\star) - R^\star(\widehat{Y}_N) \leq \xi(\widehat{Y}_N) - \xi(Y_N^\star) \leq 2 \max_{i \leq N} |\xi(Y_i)|,$$

Table 10: **The certificate's coefficient is bounded at every width, and ordering loss uses a median 17.8 to 26.6 per cent of the certificate.** The first two rows give the supremum over $N > c$ of $c_c(N)$ in Equation (2), the $N \to \infty$ limit for $c \leq 2$ and attained at $N = c+1$ otherwise, and its value at $N = 64$. The last two give that percentage at $N = 64$ as a median over the 45 configurations, on the with-replacement draws under the learned gold, for the fixed-count exact-mass shortlist and for the inclusive one at its realised mass.

Expected candidates c	1	2	4	8	16
$\sup_{N>c} c_c(N)$	0.698	1.343	2.442	3.742	5.477
$c_c(64)$	0.692	1.339	2.388	3.739	5.477
Share used, exact mass (%)	24.8	25.6	24.8	22.3	17.8
Share used, inclusive (%)	25.8	26.6	26.0	22.8	17.9

Table 11: **The ordering certificate bounds a different quantity, from different error information, than the published Best-of-N bounds.** Objective, with its comparator, and error and coverage information of each analysis.

Analysis	Objective	Error and coverage information
KL transport (Mroueh, 2024)	Reward improvement over the reference policy	Gold sub-Gaussian scale and policy KL; sup-norm residual in a transfer bound.
Soft-BoN and BoN (Aminian et al., 2026)	Regret against an optimal policy	Tilted residual MSE for Soft-BoN; sup-norm residual error for BoN, with coverage and sampling terms.
MSE and coverage (Huang et al., 2025)	Regret against a comparator policy	Marginal residual MSE and comparator-policy coverage.
Ordering certificate (Proposition D.1 and Theorem 3.2)	Regret on hits against the best candidate drawn	Shortlist residual SD (or RMS) and exact admission loss Ξ_N.

and hence, under any draw law, $\mathbb{E}[(R^\star(Y_N^\star) - R^\star(\widehat{Y}_N))\,\mathbf{1}_{\mathcal{E}_N^c}] \leq 2\Pr(\mathcal{E}_N^c)^{1/2}(\mathbb{E}[\max_{i\leq N}\xi_i^2])^{1/2}$. *Under i.i.d. draws and a shortlist of exact mass* $1/N$, $\Pr(\mathcal{E}_N^c) = (1-1/N)^N \leq e^{-1}$, *so the right-hand side is at most* $2e^{-1/2}(\mathbb{E}[\max_{i\leq N}\xi_i^2])^{1/2}$. *The middle quantity is unchanged by* $\widehat{R} \mapsto \widehat{R}+C$, *so the same argument with* $\xi(Y_i)-C$ *in place of* $\xi(Y_i)$ *gives* $2e^{-1/2}\inf_{C\in\mathbb{R}}(\mathbb{E}[\max_{i\leq N}(\xi_i - C)^2])^{1/2}$, *which is no larger and is invariant to an additive offset between proxy and gold scores.*

Proof. The proxy selection satisfies $\widehat{R}(Y_N^\star) - \widehat{R}(\widehat{Y}_N) \leq 0$, which yields the pointwise inequalities. Cauchy–Schwarz applied to the pointwise bound times $\mathbf{1}_{\mathcal{E}_N^c}$ gives the bound on misses; under i.i.d. draws each candidate falls outside a shortlist of exact mass $1/N$ independently with probability $1-1/N$, so the whole draw misses it with probability $(1-1/N)^N$.

Corollary D.11 (Unconditional regret bound)**.** *Assume i.i.d. sampling, a shortlist of exact mass* $1/N$, *and* $\mathbb{E}_{\pi_{\mathrm{ref}}}[\xi^2] < \infty$. *Then the shortlist RMS and* $\mathbb{E}[\max_{i\leq N}\xi_i^2]$ *are finite, and unconditional regret obeys, for every* $N \geq 2$ *and* $Z_N = 1-(1-1/N)^N$,

$$\begin{aligned}\Delta J(N) &= Z_N\,\mathbb{E}\big[R^\star(Y_N^\star) - R^\star(\widehat{Y}_N) \mid \mathcal{E}_N\big] + \mathbb{E}[\Delta\,\mathbf{1}_{\mathcal{E}_N^c}] \\ &\leq Z_N\Big(\underbrace{c_1(N)\tau_{\xi,1/N}}_{\text{(I) ordering}} + \underbrace{\Xi_N}_{\text{(II) admission}}\Big) + \underbrace{2e^{-1/2}\inf_{C\in\mathbb{R}}\big(\mathbb{E}[\max_{i\leq N}(\xi_i - C)^2]\big)^{1/2}}_{\text{(III) misses}}.\end{aligned}$$

Proof. Split on the hit event. On it, Proposition D.1 gives $\Xi_N + \Lambda_N$, and Theorem D.2 bounds Λ_N by $c_1(N)\,\tau_{\xi,1/N} \leq c_1(N)\,\sigma_{\xi,1/N}$. On misses, Lemma D.10 bounds the contribution by term (III).

A fixed list of k candidates. On any given list the gold gap from its proxy maximiser to its gold-best member is at most the range of the residuals on that list, so it is at most $\sqrt{2\sum_{i\leq k}(\xi_i - \bar{\xi})^2}$, which is $\sqrt{2k}$ times the within-list spread taken with divisor k and $\sqrt{2(k-1)}$ times the spread taken with divisor $k-1$. This is a per-list inequality and holds for any list, however chosen; on the model's top-k list the proxy maximiser is the Best-of-N pick.

E CONTROLS FOR THE SPLIT

The split of Section 3 is defined against a gold score on one logged pool per prompt. This appendix varies that gold, and the scale of the proxy, at one expected candidate. The gold controls use the *with replacement, inclusive* protocol of Section 5.1 on the draws of Section C.3 and each configuration's learned gold (Table 4), and test whether admission loss still exceeds ordering loss under each perturbation. The proxy controls take no budget-N draws and read residuals on the pooled shortlist candidates. Cohort statistics are medians over the 45 configurations.

Monotone gold transforms and tie conventions. Table 12 rescores these draws after gold ranks, normal scores, and convex and concave transforms of the standardised gold. Each keeps the gold order but changes the gold gaps between candidates, and so the regret being split. At $N = 64$ the recoverable share ranges from 7.37% to 10.09%, and in every transform admission loss exceeds ordering loss in at least 42 of the 45 configurations at every budget. The fixed-count exact-mass shortlist of Section C.2, with one shared random key for the shortlist and the pick, shifts each configuration's recoverable share by a median of -0.07 to -0.12 percentage points, and lowers the largest ordering loss from 14.56% to between 14.13% and 14.25% of the shortlist residual RMS across key seeds.

Table 12: **Admission loss exceeds ordering loss in at least 42 of 45 configurations under every gold transform and tie convention.** Medians over configurations at one expected candidate under the with-replacement, inclusive protocol, with interquartile ranges below the recoverable shares and, below the admission shares, counts of configurations where admission loss is larger. S is the standardised gold. Exact-mass recoverable shares are ranges of the median over key seeds 0–5.

	Recoverable share (%)			Admission share (%)		
Gold	$N = 4$	$N = 16$	$N = 64$	$N = 4$	$N = 16$	$N = 64$
Identity gold	20.85 [18.94, 22.39]	12.88 [11.61, 13.89]	8.13 [7.23, 9.15]	65.65 45 of 45	78.43 45 of 45	86.65 45 of 45
Gold ranks	19.10 [18.41, 20.89]	12.74 [11.31, 12.98]	9.24 [8.74, 9.94]	67.31 45 of 45	78.44 45 of 45	84.15 45 of 45
Normal scores	22.11 [19.92, 23.72]	12.72 [11.80, 13.83]	7.62 [7.06, 8.27]	65.03 45 of 45	79.10 45 of 45	87.54 45 of 45
Convex, $\exp(S)$	26.53 [22.22, 30.80]	13.75 [11.33, 16.99]	7.37 [6.14, 10.50]	61.82 42 of 45	78.74 45 of 45	88.15 45 of 45
Concave, $-\exp(-S)$	17.04 [15.66, 17.87]	13.57 [12.69, 14.41]	10.09 [9.45, 10.97]	68.68 45 of 45	76.15 45 of 45	82.78 45 of 45
Exact mass, shared key	20.76–20.83	12.71–12.73	8.06–8.12	45 of 45 at every key seed		

Proxy rescaling. Strictly increasing rescalings of the proxy leave its ranking, and so the split, the pick and regret, unchanged, and change the residual scales in the certificate. On a configuration's pooled shortlist candidates the infimum of that RMS over such rescalings is the RMS of one isotonic least-squares fit, which they approach without merging any proxy scores. In sample it lies below the identity map's RMS by a median 20.41%, 34.34% and 44.12% at $N = 4, 16, 64$, in all 135 comparisons. Fitted on one half of each configuration's distinct prompt texts and evaluated on the other, in both directions, the median reduction is 19.56%, 32.74% and 40.16%, and it is positive in 133 of the 135. The residual second moment on the pooled shortlist at one expected candidate, normalised by the pooled residual variance, has cohort median 1.15, 1.97, 3.09 and 4.95 at budgets of 4, 16, 64 and 256, close to $0.73N^{0.35}$ (Figure 6). A rank-matched Gaussian surrogate at each configuration's median rank agreement grows at a similar rate from a lower level.

Figure 7 pairs each of those reductions with the two-fold held-out reduction on the same configuration.

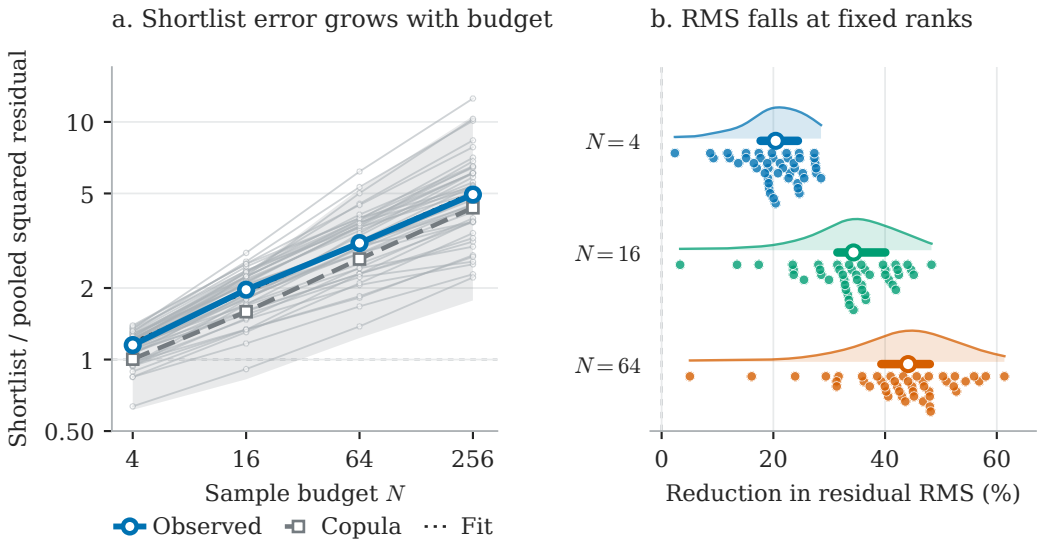


Figure 6: **Shortlist residuals grow with the budget, and rank-preserving rescaling lowers their RMS. (a)** Residual second moment on the pooled shortlist over the pooled residual variance at one expected candidate, for each configuration (grey), the median (blue), the fitted power law (dotted) and a Gaussian-copula surrogate's median and 5th–95th percentiles across configurations (squares, shading). **(b)** In-sample residual RMS reduction per configuration, from the identity map to the isotonic least-squares fit; circles mark medians and bars interquartile ranges.

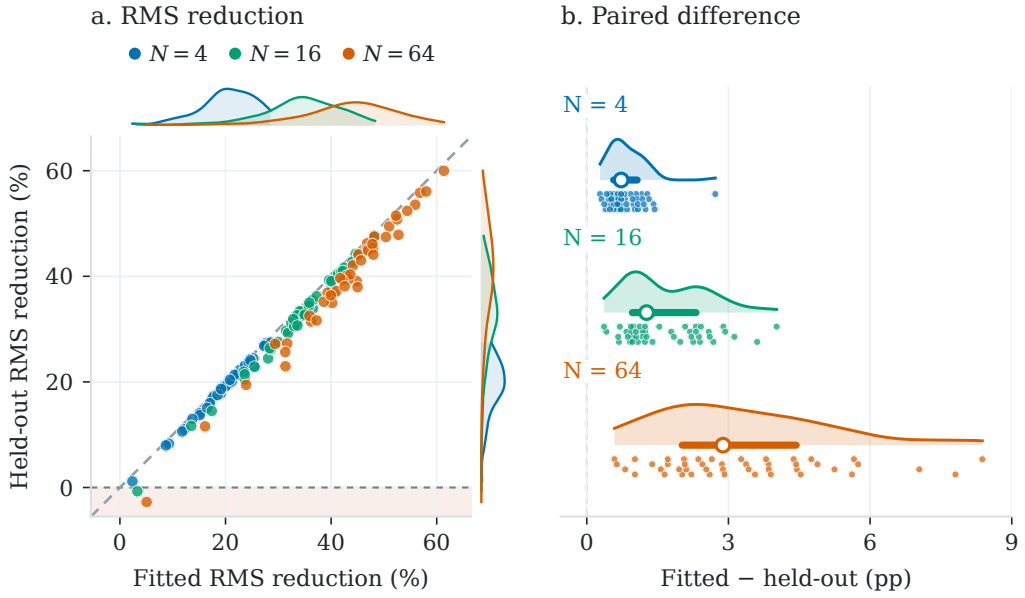


Figure 7: **Rank-preserving rescaling lowers the residual RMS on held-out prompts in 133 of 135 comparisons. (a)** Fitted (in-sample) against two-fold held-out reduction in residual RMS from the identity map, one point per configuration and budget. The diagonal marks equal reductions and the shaded band a held-out increase. **(b)** Fitted minus held-out reduction within each configuration. Open circles mark medians and bars interquartile ranges.

F RULE-CHECKABLE GOLDS

Section 5.2 rebases the width map on a gold that a rule can check, under the primary protocol. The comparisons of Sections F.1 to F.3 sit on the with-replacement law of Section C.3 with the inclusive shortlist, as exact finite-pool expectations or as its seed-7 draws, on shared draws wherever two golds or two reward models are compared, so the arms differ only in the quantity named. Section F.4 regrades the two MBPP pools of Section F.3 with only the test suite changed, Section F.5 reads whole pools and takes no draw, while Section F.6 and the two controls closing Section F.2, on unreadable candidates and on a binarised gold, use the primary protocol.

F.1 EXACT-MATCH ANSWERS ON GSM8K

The shared-draw cohort is the nine configurations, four on GSM8K, three on MATH-500 and one each on HumanEval and MT-Bench, in which ArmoRM and Skywork-V0.2 score the same drawn candidates as proxies for the RM-Mistral gold. GSM8K states a single final answer, so a candidate can be scored against the benchmark rather than a reward model: one when its final number matches the reference and zero otherwise. We apply this exact-match gold to the cohort's four GSM8K configurations, holding the pools, both proxies, the shortlist rule and the draw protocol fixed, with reference answers from the released test and train splits.

Table 14 gives the paired regret differences by exact finite-pool expectations, one row per configuration under its ID from Table 29. Under RM-Mistral every interval lies above zero. Under exact match no interval lies above zero and five lie below it. In 98.08% of bootstrap replicates for T15 and 94.46% for T42, the learned gold favours Skywork-V0.2 and exact match favours ArmoRM at all three budgets. All twelve cells of the four configurations and three budgets are positive under RM-Mistral and negative under exact match. At $N = 4$ and 16 the eight differences run from $+0.0298$ to $+0.1380$ under RM-Mistral and from -0.0304 to -0.0029 under exact match. Under exact match, admission loss exceeds ordering loss in all 24 comparisons of four configurations, two reward models and three budgets, with admission-to-ordering ratios from 2.28 to 8.87 (Table 13). Which reranker helps also changes with the gold. At a budget of 64 and one to sixteen expected candidates, Skywork-V0.2 reranking ArmoRM's shortlist recovers a median 15 to 26 per cent of the oracle ceiling under RM-Mistral, while under exact match ArmoRM reranking Skywork-V0.2's shortlist recovers 12 to 21 per cent (Figure 8b); the reverse pairings range from -9 to $+1$ per cent under the two golds.

Table 13: **Admission loss exceeds ordering loss under exact-match gold on GSM8K in all 24 comparisons.** Entries are Ξ_N/Λ_N at one expected candidate, by budget N and proxy reward model, from 400 draws per prompt. Pass is the fraction of candidates whose last number matches the reference.

		ArmoRM			Skywork-V0.2		
Configuration	Pass	4	16	64	4	16	64
T2	0.74	2.52	4.50	5.61	2.47	4.36	5.87
T13	0.82	2.94	6.06	7.09	3.41	5.54	8.81
T15	0.52	2.28	4.75	7.70	2.72	5.32	8.87
T42	0.79	2.95	6.21	8.78	3.19	4.83	8.37

Which reward model each gold prefers. The two golds rank the same two reward models by different summaries of the same pools. Each difference in this paragraph is Skywork-V0.2 minus ArmoRM. On the four GSM8K pools Skywork-V0.2 agrees better with RM-Mistral in all four, by Spearman-correlation differences of 0.0308 to 0.0661 with intervals excluding zero, while ArmoRM separates exact-match-correct candidates better in all four, by AUROC differences of -0.0134 to -0.0398 with three of the four intervals excluding zero, prompts with a single class dropped. The median differences are $+0.0448$ in Spearman correlation against RM-Mistral over the shared-draw cohort, -0.0270 in AUROC against exact match over the four GSM8K pools and -0.0419 in AUROC against unit tests over the five code pools.

Truncation, parsing and format. Three controls separate the gold from the way the answer is written. At most 0.39% of picks touch the 512-token generation cap, and excluding capped completions leaves the cohort median at −0.0079 with all twelve cells negative and five separated from zero. The strict harness marker that GSM8K uses to delimit a final answer appears in 0 of 204,800 candidates, so the harness gold cannot serve as a correctness gold on these pools. Where a candidate states an explicit final answer, with the phrase *the answer is* or a boxed answer, the last-number rule gives the same label on 99.8, 99.3, 91.5 and 99.9 per cent of such candidates. A gold that also requires such an answer turns the cohort median positive, to 0.0058. On T42, the one pool where most candidates state one, 49.8 to 63.0 per cent of ArmoRM's picks lack one across the three budgets, against 40.6 to 42.7 per cent of Skywork-V0.2's, and restricting that pool to candidates that state one keeps ArmoRM ahead in 74.9% of bootstrap replicates at all three budgets.

Outside GSM8K both golds put the same reward model ahead. On the three MATH-500 configurations the official grader of Section G agrees with the learned gold in all nine cells. Under both, Skywork-V0.2 has the lower regret, by a median 0.0221 under the grader, and no cell has more than 1.5% of its bootstrap replicates below zero. On the five code pools ArmoRM is ahead under tests, by a median regret difference of −0.0144 over fifteen pool and budget cells, and on the one pool carrying both golds the paired differences contain zero. Figure 8 shows these comparisons.

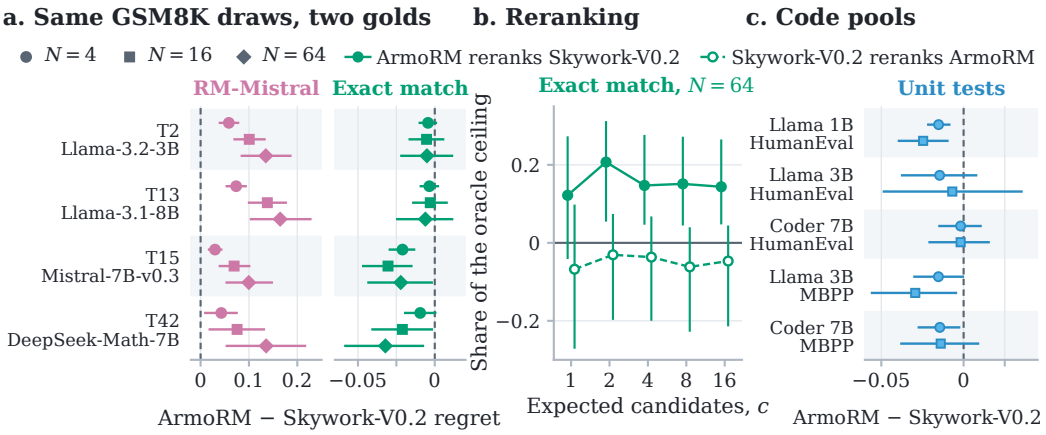


Figure 8: **Which reward model is ahead depends on the gold.** Bars are 95% paired prompt-cluster bootstrap intervals. **(a)** ArmoRM's regret minus Skywork-V0.2's on four GSM8K configurations, under the golds RM-Mistral (standardised units) and exact match (accuracy). **(b)** Median over those configurations of the share of the oracle ceiling that one reward model recovers by reranking the other's shortlist, under exact match at $N = 64$. **(c)** As (a), on five code pools under unit tests (accuracy).

Table 14: **Replacing ArmoRM with Skywork-V0.2 lowers GSM8K regret under RM-Mistral gold and raises it under exact-match gold.** Regret with ArmoRM as proxy minus regret with Skywork-V0.2 at $N = 64$, as exact finite-pool expectations on shared with-replacement draws (RM-Mistral in standardised reward units, exact match in accuracy), with 95% paired prompt-cluster bootstrap intervals (20,000 replicates). Joint reversal is the share of replicates with the difference positive under RM-Mistral and negative under exact match at all of $N = 4$, 16 and 64.

ID	N	RM-Mistral	Exact match	Joint reversal
T2	64	+0.1349 [0.0827, 0.1881]	−0.0051 [−0.0225, 0.0120]	0.6853
T13	64	+0.1646 [0.1017, 0.2295]	−0.0061 [−0.0252, 0.0121]	0.6418
T15	64	+0.0997 [0.0515, 0.1496]	−0.0221 [−0.0440, −0.0012]	0.9808
T42	64	+0.1356 [0.0518, 0.2183]	−0.0322 [−0.0590, −0.0069]	0.9446

F.2 THE WIDTH MAP UNDER A RULE-CHECKABLE GOLD ON THE WITH-REPLACEMENT DRAWS

On the 22 configurations whose gold is exact match or a unit-test suite, under the with-replacement draws and the inclusive shortlist, the recoverable share at a budget of 16 is 8.56%, 27.02%, 53.73% and 77.97% at one, two, four and eight expected candidates. On the four GSM8K pools at a budget of 64 exact match gives 7.20, 21.92, 42.95 and 64.79 per cent against 7.26, 21.77, 45.07 and 67.59 under the learned gold on the same draws, and every paired interval at that budget contains zero. At the smaller budgets four of the six paired intervals lie below zero. Exact match is lower by 1.93 and 5.02 points at a budget of 4 with one and two expected candidates and by 2.16 and 4.89 at a budget of 16 with one and eight.

Dropping unreadable candidates. A rule check and the grader of Section 6 both score a candidate with no readable answer incorrect. On the 31 configurations carrying an answer extractor, under the primary protocol at a budget of 64 with one expected candidate, the paired difference in recoverable share between the rule-checkable and the learned gold is -1.55 points under official grading and -1.99 when only answer-bearing candidates are kept. Under official grading GSM8K, whose extractor reads nearly every candidate, gives -2.17 over its 19 configurations, and MATH-500, which holds nearly all unreadable candidates, gives the smallest effect, -0.97 over its 11, which the restriction to answer-bearing candidates makes larger.

Binarising the learned gold. The binarised gold of Section 5.2 gives a gold of one to the candidates of each prompt that the configuration's learned gold ranks highest, as many as the rule check marks correct, and zero to the rest. It therefore has the rule check's scale and per-prompt pass count and differs from it only in which candidates it calls correct. On the 35 rule-checkable configurations under the primary protocol its cohort median recoverable share is 11.3 per cent against the learned gold's 10.5 at a budget of 64 and one expected candidate, 56.6 against 52.2 at four expected candidates, 90.7 against 88.6 at sixteen and 14.5 against 13.8 at a budget of 16, and the rule check sits below it by a paired median of 3.1 to 13.3 points. In each of the four cells of Table 2 the labels alone therefore move the width map further than the full replacement, whose paired medians run from -2.0 to -9.2 points, while binarising the learned gold, which changes its scale and prompt weights and also removes every gold gap among the candidates it calls correct and among the rest, moves it up by a paired median of $+0.9$ to $+4.7$. The two binary golds share their units, so the numerator and denominator of the recoverable share, the oracle ceiling $Z\Lambda_N$ and total regret $\Delta J(N)$, compare directly between them. Replacing the binarised gold by the rule check changes the oracle ceiling by a paired median of -0.0003 at a budget of 64 and 0.0001 at a budget of 16 with one expected candidate and by 0.0063 at four expected candidates, changes not separated from zero, and raises it by 0.0179 at sixteen, while total regret rises by 0.048 at a budget of 64 and 0.038 at 16. Both golds put the same mass on each prompt, so the expected best drawn gold is identical under them and the whole rise in total regret comes from the Best-of-N pick, whose gold value is a median 0.048 higher under the binarised gold, which calls correct the candidates the learned gold ranks highest. The rule check therefore lowers the recoverable share by enlarging the regret that no second stage confined to the shortlist can remove.

F.3 UNIT TESTS ON FIVE CODE POOLS

Five code pools hold 88,576 candidates graded by benchmark tests (Table 15; generation and test execution in Section K.3). Both reward models score the same candidates, gold is the binary test outcome on its original scale, and every task is kept, including those with no passing candidate. For each conditional loss we average the event-weighted loss and the event probability across tasks and divide the former by the latter. Pointwise 95% intervals come from 20,000 bootstrap replicates that resample tasks jointly across generators, budgets and threshold estimates.

Admission and ordering loss. At $N = 4$ and 16, admission loss exceeds ordering loss for both reward models in all 20 comparisons, and every paired interval for $\Xi_N - \Lambda_N$ lies above zero. From $N = 4$ to 16 the admission-loss point estimate rises in all ten pairs of pool and reward model, while ordering loss stays below 4.30 percentage points of binary regret at both budgets (Figure 9).

Table 15: **Pass rates on the five unit-tested code pools run from 0.36 to 0.88.** Pool is the number of candidates generated per task, Solvable the number of tasks with a passing candidate, and pass rate the fraction of candidates that pass.

ID	Generator	Benchmark	Tasks	Pool	Solvable	Pass rate
C1	Llama-3.2-1B	HumanEval	164	256	139	0.36
C2	Llama-3.2-3B	HumanEval	164	64	138	0.40
C3	Qwen2.5-Coder-7B	HumanEval	164	64	158	0.88
C4	Llama-3.2-3B	MBPP	200	64	154	0.44
C5	Qwen2.5-Coder-7B	MBPP	200	64	187	0.72

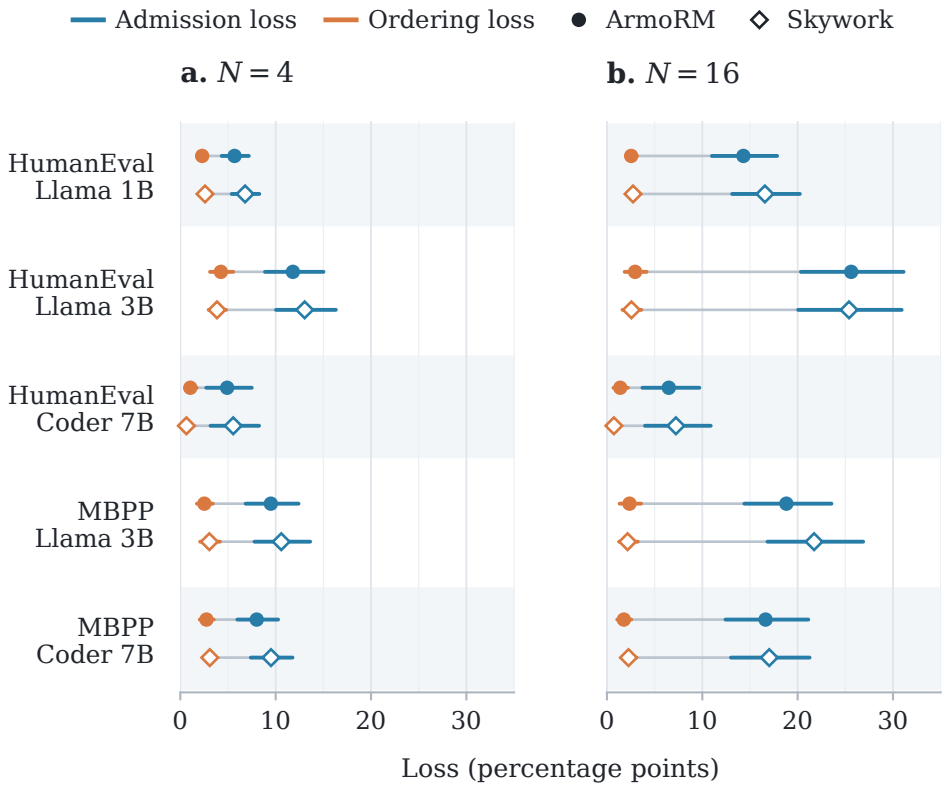


Figure 9: **Under unit tests, admission loss exceeds ordering loss for both reward models on every code pool.** Points are Ξ_N and Λ_N at one expected candidate, each reward model conditioned on the draws that hit its own shortlist. Bars are 95% task-bootstrap intervals, and Skywork is Skywork-V0.2.

The two reward models. ArmoRM has the lower unconditional regret in all ten comparisons at $N = 4$ and 16, with four paired intervals excluding zero (Figure 10a), and at $N = 16$ some event contributions favour Skywork-V0.2 in both Qwen2.5-Coder-7B pools while their signed sum favours ArmoRM (Figure 10b). At $N = 64$ the nominal shortlist of a 64-candidate pool is a single candidate, so that budget enters only the comparison of unconditional regret. Estimating each pool's threshold on the opposite half rather than the evaluated half moves the gain on draws that hit both shortlists by at most 0.45 percentage points at $N = 4$ and 16.

F.4 A WITHHELD TEST SUITE FOR MBPP

MBPP (Austin et al., 2021) supplies its three assertions inside the generation prompt, so its primary gold measures satisfaction of those examples. MBPP+ (Liu et al., 2023) extends the tests of 90 of the 200 tasks in our MBPP cohort with inputs that the prompt does not show. We grade every candidate on those 90 tasks under both suites through the same sandbox, so the test set is the only thing that

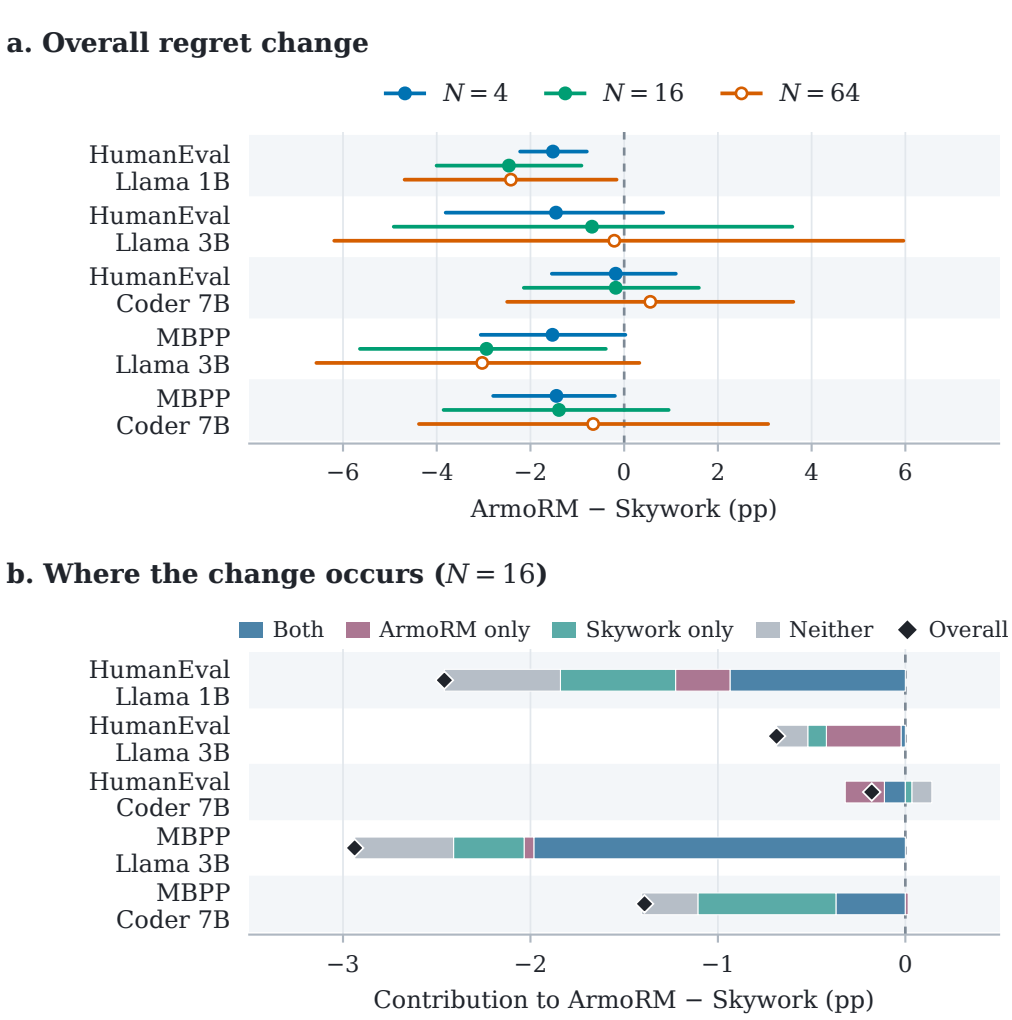


Figure 10: **Under unit tests, ArmoRM has the lower regret on every code pool at $N = 4$ and 16. (a)** ArmoRM's regret minus Skywork-V0.2's, with 95% paired task-bootstrap intervals. **(b)** The same difference at $N = 16$, split into probability-weighted contributions from draws that hit both reward models' shortlists of one expected candidate, only one or neither.

changes. MBPP+ grading lowers the pass rate from 0.50 to 0.41 on Llama-3.2-3B and from 0.79 to 0.57 on Qwen2.5-Coder-7B, and across two generators, two reward models, two budgets and two suites admission loss exceeds ordering loss in all 16 comparisons, at ratios of 2.40 to 8.67. ArmoRM has the lower unconditional regret in all 8 reward-model comparisons, 4 of them under MBPP+. Requiring candidates to pass 16 additional MBPP challenge tests, which cover 11 of the 200 tasks, changes the point estimate of ArmoRM's regret minus Skywork-V0.2's by at most 0.13 percentage points at $N = 4$, 16 and 64.

F.5 RewardBench scores and within-prompt correctness discrimination

Nine mathematics pools carry, on shared draws, five open reward models with a published RewardBench v1 score (Lambert et al., 2025): the cohort's Skywork-V2-8B, Skywork-V0.2 and ArmoRM and, from outside the cohort, GRM-gemma2-2B (Yang et al., 2024) and InternLM2-7B-Reward (Cai et al., 2024). RM-Mistral publishes no such score, and Skywork-V2-3B is scored on one of the nine only. Graded against the rule-checkable gold, the model with the highest RewardBench v1 score has the highest within-prompt AUROC on none of the nine (Table 16). The table's 90 comparisons, one per pool and pair of models, share pools and models. The model with the higher RewardBench v1 score has the higher AUROC in 36 of them, its AUROC minus the other's has median −0.0159, and

the tested quantity, the configuration-weighted mean of that difference, is -0.0038 at $p = 0.74$, so on these nine pools the published order does not reproduce the within-prompt AUROC order. The process reward model Qwen2.5-Math-PRM-7B (Zhang et al., 2025b), outside the table, leads its bold entries on T2, T13, T15 and T42, at 0.8683, 0.8552, 0.8668 and 0.9053.

Table 16: **The RewardBench v1 order of these five reward models does not match their AUROC order against a rule-checkable gold.** Each prompt's AUROC is taken over its 256 candidates and averaged over the n of 200 prompts that have both correct and incorrect candidates. Bold marks each row's largest value. A row naming several configurations is one pool they share.

		RewardBench v1 score				
Pool	n	96.4 Skywork-V2-8B	93.1 Skywork-V0.2	89.0 ArmoRM	88.4 GRM-gemma2-2B	86.6 InternLM2-7B-Reward
T2, GSM8K	160	0.7959	0.7901	0.8095	0.6698	**0.8352**
T13, GSM8K	140	0.7601	0.7627	0.7761	0.7072	**0.8127**
T15, GSM8K	192	0.8121	0.7971	0.8316	0.6865	**0.8406**
T42, GSM8K	154	0.7937	0.7721	0.8119	0.6527	**0.8926**
T10–T12, GSM8K	195	0.8331	0.8309	0.8584	0.7141	**0.8813**
T3, MATH-500	121	0.7921	0.8581	0.7701	**0.8803**	0.8001
T9, MATH-500	128	0.7494	0.8185	0.7652	**0.8568**	0.7629
T29, MATH-500	134	0.8292	0.8704	0.8236	**0.8867**	0.8252
T34, T37, MATH-500	119	0.7652	0.8347	0.7898	**0.8688**	0.7722

F.6 Gold rebasing across further pools and proxy-gold pairs

One proxy-gold pair carries 34 of the 45 configurations, so this extension tests whether the gold effect holds across pairs, pools and datasets. Seventeen pools carry up to seven reward models each on shared draws: the nine mathematics pools of Section F.5, five cohort pools on HumanEval, HumanEval+, LiveCodeBench (two) and BBH-logical rescored with the added models, and three new pools on MBPP and MMLU-Pro generated under the cohort's protocol (Section K.1). In each pair the model with the higher published RewardBench v1 score supplies the learned gold; RM-Mistral, which publishes no score, is always the gold, and the process reward model always the proxy. That gives 21 pairs, 16 of them absent from the cohort, and in each the width map is rebased from the learned gold onto the rule-checkable labels. Sixteen pools carry all seven models and the seventeenth lacks RM-Mistral, so the pools contribute $16 \times 21 + 15 = 351$ cells, which share models and pools; the tests below cluster on pairs, on pools and on datasets in turn. HumanEval and HumanEval+ are the same 164 problems under two test suites and count as one dataset cluster, which leaves seven. Every cell uses the primary protocol of Section 5.1, which takes uniform size-N subsets of the logged pool and a shortlist of size $\mathrm{Bin}(M, c/N)$ from the top of the proxy order, with proxy ties broken by one uniform key per pool and prompt shared by every proxy and both golds. The learned gold is standardised within each prompt over its pool, the rule-checkable gold is the official grader on GSM8K and MATH-500, the tests on the code pools and the answer letter on BBH-logical and MMLU-Pro, and a cell's recoverable share is a ratio of moment sums pooled over prompts, differenced between the two golds on the same prompts and the same key.

The effect is negative in every dataset: -5.73 on BBH-logical, -5.33 on MMLU-Pro, -2.72 on LiveCodeBench, -2.71 on HumanEval, -1.66 on MBPP, -1.56 on MATH-500 and -0.64 on GSM8K. The median paired difference at a budget of 64 and one expected candidate is -1.76 points over all 21 pairs and -1.58 over the 16 new ones, both inside the interval the cohort median carries, and 262 of the 351 cells are negative. The other printed cells move further: -6.95 at four expected candidates, -6.77 at sixteen and -1.94 at a budget of 16.

Each clustering level is checked by a one-way Rademacher wild cluster bootstrap (Section J.3) with every sign vector enumerated, so each p-value carries no Monte Carlo error and rests on independence across the clusters of its own level. The three checks read the same 351 cells, and none clusters on the reward models that pairs share. The configuration-weighted mean is separated from zero on pairs at $p = 2.9 \times 10^{-6}$ over all 2^{21} sign vectors, on the seventeen pools at $p = 0.0025$ over all 2^{17}, and on the seven datasets at $p = 0.015625$ over all 2^7. All seven dataset effects are negative, and that p is the smallest two-sided value seven clusters admit. Dropping any one dataset cluster leaves

six and still gives $p = 0.03125$, the smallest value at that count, with pool-level p between 0.0032 and 0.0089, so no single dataset carries the result.

On the 35-configuration cohort the gold effect's bootstrap p is 0.001953 over ten proxy-gold pairs, the smallest value that count admits, and the effect is negative in all five dataset clusters, HumanEval and HumanEval+ merged as above. Over five clusters the smallest attainable two-sided enumeration p is 0.0625, so dataset-level separation rests on the seventeen-pool extension, whose seven clusters share no problem set.

G GRADING, ANSWER EXTRACTION AND THE ZERO-CALL RULES

The accuracies of Tables 17 and 18 are on the seed-7 draws of Section C.3, 400 per prompt with replacement, under official grading, the first as medians over the configurations of each benchmark. The accuracy comparisons of Section G.1 take exact expectations over size-N subsets, Section G.2 measures extraction and grader agreement on every candidate of four fixed pools, and Section G.3 realises the size-64 subset law by Monte Carlo. A zero-call rule makes no second-stage verifier call, the cost unit of Table 4; answer extraction and equivalence checking still run.

Majority voting returns the most frequent answer in the draw (Wang et al., 2023), and Best-of-Majority restricts the draw to answers that occur often enough and returns the one with the highest reward (Di et al., 2026). We follow the experimental procedure of Di et al. (2026) with $k = 1$: answers in the N draws are grouped by the grader's equivalence rule (*Grading* below), a pool candidate drawn twice counting twice, and a group's reward is the mean reward of its trajectories while its frequency is its share of the draw. Their threshold is set from a coverage coefficient that is not observable here, so we report two thresholds, $\alpha = 0.25$ and $\alpha = 0.50$.

Best-of-N returns $\arg\max_i \widehat{r}_i$, breaking ties by the smallest index. Majority voting returns the answer of the largest group, breaking ties by the smallest first-member index. Best-of-Majority(α) takes the groups with $|G_a| \geq \lceil \alpha N \rceil$ and returns the answer of the one with the largest mean reward, then the larger $|G_a|$, then the smallest first index, falling back to the Best-of-N trajectory when no group qualifies. A group's answer is that of its first drawn member. A returned answer scores 1 when the grader accepts it against the reference and 0 otherwise, and a draw in which no trajectory yields an extracted answer scores 0 under every rule.

Grading. A MATH-500 answer is the contents of the last boxed expression in the response, taken by the boxed-answer extractor of the MATH evaluation code (Hendrycks et al., 2021); a response without one yields no extracted answer and is scored incorrect. Correctness is the PRM800K answer grader (Lightman et al., 2024), applied to the extracted answer and the reference. It accepts matching normalised strings or a symbolic difference that simplifies to zero, compares unreduced fractions as strings, and requires an integer reference to be matched exactly. Classes are the connected components, over a prompt's whole pool, of the graph that joins two distinct extracted answers when the grader accepts each as the other, so a chain of mutual acceptances forms one class; three classes, two on T9 and one on T3, hold members the grader grades differently against the reference. A GSM8K answer is the last number in the response, compared by numeric equality. The code benchmarks have no answer-class partition, so the zero-call comparisons are made on GSM8K and MATH-500 only.

On GSM8K, Best-of-Majority at $\alpha = 0.50$ has the highest median accuracy at $N = 64$, 0.8623 against 0.8586 for Best-of-N; on MATH-500 majority voting leads, 0.3545 against 0.2630, and Best-of-Majority at $\alpha = 0.50$ stays with Best-of-N because no answer reaches half of the draw in 97% of MATH-500 draws at $N = 64$, against 11% on GSM8K. At $N = 64$ the most accurate rule on each benchmark sits below a $\mathrm{pass@}N$ of 0.9742 on GSM8K and 0.5189 on MATH-500, consistent with selection plateauing below pass@N (Brown et al., 2024). Table 17 takes column medians with ArmoRM on GSM8K and Skywork-V0.2 on MATH-500, and Table 18 holds the reward model fixed within each configuration. The exact verifier of Section H uses the same grading on size-N subsets drawn without replacement.

Table 17: **Majority voting leads on MATH-500 from $N = 16$.** Medians over four GSM8K and three MATH-500 configurations under official grading. Best-of-N and Best-of-Majority use ArmoRM on GSM8K and Skywork-V0.2 on MATH-500. Fallback is the share of draws where no answer reaches $\lceil N/2 \rceil$ and Best-of-Majority at $\alpha = 0.50$ returns Best-of-N.

				Best-of-Majority			
Benchmark	N	pass@N	Best-of-N	$\alpha = 0.25$	$\alpha = 0.50$	Majority	Fallback
GSM8K	4	0.8977	0.8361	0.8262	0.8343	0.8171	0.0379
	16	0.9490	0.8533	0.8527	0.8560	0.8492	0.0946
	64	0.9742	0.8586	0.8579	0.8623	0.8565	0.1078
MATH-500	4	0.2132	0.1584	0.1781	0.1627	0.1728	0.8786
	16	0.3853	0.2280	0.2422	0.2292	0.2819	0.9614
	64	0.5189	0.2630	0.2731	0.2632	0.3545	0.9728

Table 18: **Majority voting beats every reward-model rule on all three MATH-500 configurations.** Accuracy under official grading at $N = 64$ for each configuration and reward model. BoN and BoM abbreviate Best-of-N and Best-of-Majority.

			ArmoRM			Skywork-V0.2		
Benchmark	ID	pass@N	BoN	BoM .25	BoM .50	BoN	BoM .25	BoM .50
GSM8K	T2	0.9699	0.8450	0.8411	0.8392	0.8399	0.8412	0.8387
	T13	0.9786	0.8722	0.8747	0.8855	0.8657	0.8704	0.8786
	T15	0.9407	0.7692	0.7608	0.7478	0.7475	0.7408	0.7327
	T42	0.9958	0.9054	0.8875	0.9045	0.8743	0.8768	0.8991
MATH-500	T3	0.4477	0.1612	0.1645	0.1618	0.2059	0.2073	0.2074
	T9	0.5189	0.2363	0.2586	0.2375	0.2630	0.2731	0.2632
	T29	0.7007	0.6025	0.6238	0.6151	0.6244	0.6428	0.6366
Majority voting: T2 0.8299; T13 0.8848; T15 0.7011; T42 0.8832; T3 0.2677; T9 0.3545; T29 0.6606								

G.1 Truncation and candidate-restriction controls

A candidate is truncated when its length in the generating model's own tokens reaches the pool's logged completion cap. Measured that way, the rate separates MATH-500 and LiveCodeBench from GSM8K with instruction-tuned generators (Table 19).

Table 19: **MATH-500 and LiveCodeBench truncate a large share of their candidates, and GSM8K with instruction-tuned generators almost none.** Percentage of candidates reaching the pool's logged completion cap on the 35 rule-checkable configurations; a row covering several configurations gives their range. The cap is 512 tokens, except 256 for the LiveCodeBench pool at 68.1 per cent.

Pool group	Configurations	Truncated (%)
MATH-500	11	10.7–46.4
GSM8K, instruction-tuned generators	18	≤ 2.0
GSM8K, base generator	1	53.3
BBH-logical	1	10.2
HumanEval	1	7.9
HumanEval+	1	7.7
LiveCodeBench	2	43.2–68.1

Truncation and unparseability are close to nested but not the same event. On MATH-500 only 0.6 to 5.0 per cent of truncated candidates carry a parseable answer, so reaching the cap nearly guarantees that the grader reads nothing. In the other direction truncation covers 15 to 53 per cent of the unparseable candidates on eight of the eleven configurations and 89 to 99.7 per cent on the other three. Among answer-bearing MATH-500 candidates the truncation rate is a median 0.0108, against 0.374 among unparseable ones. On GSM8K the last-number parser reads an answer from any response containing a number, truncated or not. The answer-bearing rate is 1.000 in fourteen configurations and 0.99998 in four more, so outside the base generator, whose rate is 0.860, the

official arm and the answer-bearing arm, which keeps only candidates with a parseable answer, coincide. The repository's alternative extractor, which accepts a boxed answer, an answer cue or the last inline mathematical expression, is lenient only on MATH-500, where it raises the answer-present rate, averaged over the eleven MATH-500 configurations, from 0.374 to 0.718. Elsewhere it reads only explicit answer cues. It recovers an answer from 0.0 to 16.2 per cent of candidates in eleven of the fifteen labelled GSM8K pools and from 0.9 per cent of BBH candidates, so it is read on MATH-500 only.

The control that separates truncation from extraction is a fourth gold arm, *cap-dropped*, which keeps the official grader and drops only the candidates that reached the cap. That arm and the answer-bearing arm do not agree (Table 20). The best width moves under the answer-bearing arm. For a unit, one configuration, second stage and budget, call its optimum *non-maximal-width* when the width maximising mean shortlist accuracy is not the widest on the grid, and call it *resolved* when the 2.5th percentile of the paired bootstrap difference between the accuracy at that selected width and the accuracy at the width that maximises the oracle ceiling is above zero, an exploratory screen in which the width is chosen on the same draws. The answer-bearing restriction leaves 3 non-maximal-width optima in the 140 accuracy-by-width units, none of them resolved, against 44 under official grading, and the process model's product and minimum aggregations go from non-maximal-width in 16 of 16 and 16 of 16 MATH-500 units to 0 of 16 and 1 of 16, so on the answer-bearing population the widest width on the grid is the realised optimum in 137 of the 140 units. The official gold restricted to the answer-bearing prompt set alone keeps 16 of 16 and 16 of 16 non-maximal-width, so dropping the prompts does not reproduce the change. No contrast of Table 20 changes sign under either restriction, and dropping the capped candidates moves every entry less than keeping only answer-bearing candidates does, so the length restriction does not reproduce the answer-bearing one.

Table 20: **On MATH-500, dropping the candidates that reached the generation cap changes every entry less than keeping only answer-bearing candidates does.** Medians at a budget of 64 and one call under the official grader on three candidate sets that differ only in which candidates they keep; a prompt left with fewer than N candidates is dropped. The process-model block is the MATH-500 score-only arm of Table 24, under product aggregation.

Quantity	Official	Cap-dropped	Answer-bearing
Process reward model			
Best-of-N accuracy	0.2414	0.2645	0.4161
Runners-up rule minus threshold shortlist	-0.0317	-0.0322	-0.0150
Runners-up rule minus majority voting	-0.1110	-0.1300	-0.0696
Exact verifier, eleven MATH-500 configurations			
One-call gain over the threshold shortlist	$+0.0436$	$+0.0486$	$+0.0600$

G.2 Answer extraction and equivalence checking on fixed candidate pools

The population of the two MATH-500 comparisons of Section 6 that keep every candidate is every one of the $4 \times 200 \times 256 = 204{,}800$ logged candidates of the four pools carrying process-model step scores. No candidate is filtered and no prompt is dropped in any arm, unlike the answer-bearing arm of Section G.1; the reward-model scores, step scores, draw law, shortlist rule and bootstrap seeds are those of Section 6.

Software. Math-Verify $0.9.0$ (Kydlíček, 2026) runs in the official grader's environment, so that both extractors call the same symbolic algebra library, SymPy $1.13.1$; its LaTeX-to-SymPy layer is at $1.11.0$ and its parser runtime at $4.13.2$. Every Math-Verify option is left at its version-$0.9.0$ default, in parsing and in the equivalence check, whose defaults round floats to six places at a numeric precision of 15 under strict comparison with a five-second limit. Two steps are ours. For the native arm the reference is parsed inside math delimiters, falling back to the bare string when that yields nothing, and for the extraction-only arm the first parsed value is rendered to LaTeX by SymPy's printer, or passed through when it is already a string, before the PRM800K grader reads it.

Four arms on identical candidates, and what each of them replaces. The official arm takes the last boxed expression and grades it with the PRM800K answer grader; a candidate with no boxed expression yields no extracted answer and is graded incorrect. The extraction-only arm replaces the extractor alone: Math-Verify parses the candidate under its default configuration, both the LaTeX and the plain-expression extractors with anchorless extraction enabled, first-match fallback, any-match extraction and a five-second parsing timeout, and the string it returns is graded by the same PRM800K grader under the same five-second limit. The native arm replaces the correctness checker as well: Math-Verify parses both the candidate and the reference and decides equivalence itself. Section 6 plots the native arm, and both comparisons move the same way under the extraction-only arm. When that arm selects and grades, the shortlist's lead over the call-matched random subset at a budget of 64 is -0.0157 at two calls and -0.0007 at four, against -0.0157 and -0.0015 under the native arm. Under the extraction-only arm the last-step aggregation's lead over the product is 0.0003, against -0.0041 under the native arm. The fourth arm, the repository's alternative extractor of Section G.1, is read from its stored grades rather than recomputed.

Reproducing the official arm. Re-running the official extractor and grader on the same candidate texts reproduces the stored correctness and answer-presence labels of all $204{,}800$ candidates and the official arm of every comparison exactly.

What the two extractors read, and how often they disagree. The official extractor reads a boxed answer from $57{,}490$ of the $204{,}800$ candidates, a rate of 0.2807, and Math-Verify extracts an expression from $204{,}711$, a rate of 0.9996. Against the official arm, $147{,}221$ candidates without a boxed answer gain an extracted expression, none loses one, and 126 of the $57{,}490$ both read state a different answer. Of the 800 prompt-by-pool pairs, 154 carry no parseable candidate under the official extractor and none does under Math-Verify. The official arm grades $33{,}163$ candidates correct, the native arm $78{,}409$, the extraction-only arm $71{,}977$ and the alternative extractor $44{,}914$. Against the official arm the native arm is correct on $45{,}323$ candidates it calls incorrect and incorrect on 77 it calls correct; for the extraction-only arm the two counts are $39{,}726$ and 912, and the 912 are almost all responses stating several boxed answers.

The stratified audit. A stratified sample of 160 candidates was drawn without replacement from the union of the four pools over five strata fixed before the draw: 60 from the $147{,}221$ gained candidates, 40 from the 284 changed, 40 from the $57{,}206$ agreed, 20 from the 89 on which both are silent, and none from the lost stratum, which is empty because Math-Verify extracts an expression from every candidate from which the official extractor reads a boxed answer. The strata were formed on the raw string Math-Verify returns, while the disagreement counts above use its parsed value and its own equivalence, so the changed stratum holds 284 candidates where the counts report 126. Each stratum is reweighted by its population size. For each sampled candidate we wrote a record carrying the problem and the candidate text alone, recorded the answer the adjudicator read the candidate as stating, and only then joined the record to the two extractors' outputs.

The adjudicator and its two readings. The adjudicator is a language model reading a blinded sheet: the problem and the candidate text, with no reference answer, extractor output, correctness label, stratum or arm. Every sheet is adjudicated twice, in two separate language-model readings. The first was read in an analysis session and its prompt and decoding configuration were not logged. The second is scripted: Qwen2.5-7B-Instruct[1] with greedy decoding at temperature 0, a fixed maximum length and one prompt template applied through the checkpoint's own chat template; two runs on 40 held-out sheets give byte-identical output.

The two adjudications agree on the grade of 0.9875 of the 160 candidates in this pool-level audit and of 0.9268 of the 246 distinct candidates in the returned-output audit of Section G.3. In the returned-output audit the two readings differ in grade on 18 candidates. On 16 of them the scripted reading carries a trailing unit, or in one case a Unicode root sign, into the answer string, reading 4 as *4 inches/second* and 78 as *78 dollars*, which the PRM800K grader then rejects. All 16 move a grade from correct to incorrect on returns the official extractor had already graded incorrect. On truncated candidates the scripted reading reports an expression where the first reading recorded no stated answer on 10 of 40 such cases here and 3 of 4 in the returned-output audit. The returns that extraction-only Math-Verify selects (Section G.3) were read by the scripted reader alone.

[1] Hugging Face `Qwen/Qwen2.5-7B-Instruct`, revision `a09a3545`.

String accuracy and grade agreement. String accuracy scores the string each extractor returns against the adjudicator's. Population-weighted string accuracy is 0.5325 for the official extractor and 0.5797 for Math-Verify, a paired difference of $+0.0472$ whose interval covers zero because of the gained stratum. Math-Verify's expression extractor falls back to the last expression of a truncated derivation and so returns a string on candidates that state no answer, 21 of the 60 sampled there.

The two comparisons of Section 6 are computed from grades, so grade agreement on the same 160 candidates and adjudicator verdicts was added as a post hoc statistic after string accuracy had been seen. The adjudicator's grade is one when the PRM800K grader accepts the answer the adjudicator read against the reference and zero otherwise, a candidate stating no final answer scoring zero. Population-weighted agreement with the adjudicator is 0.8321 for the official arm, 0.9878 for the native arm and 0.9758 for the extraction-only arm. Paired on the same candidates, the native arm gains $+0.1557$ over the official arm and the extraction-only arm $+0.1437$. In the gained stratum 13 of the 60 sampled candidates are graded the adjudicator's way by the native arm and not by the official one, and none the other way round; all 14 of the official arm's errors there are correct answers graded incorrect. The expressions Math-Verify extracts from unfinished derivations on truncated candidates are graded incorrect, which is the grade the official extractor's silence gives them.

G.3 Auditing the shortlist's and the random subset's outputs

Section G.2 audits the candidate pool. The call-matched comparison of Section 6 compares what the two rules *return*, a different population, and this section audits that one. Unless called native, Math-Verify in this section is the extraction-only arm of Section G.2. The adjudicator reads returns on which the Math-Verify and official grades differ, and the answer it reads is graded with the same PRM800K rule. Only those returns are re-graded and the others keep the grade both graders give them, so versions D and E below are disagreement-audit-corrected contrasts.

Realising the draw law. The call-matched comparison is computed as an exact expectation over the size-64 subset law, so no returned output is stored. Drawing that same law, $8{,}000$ draws per prompt shared between the two rules and the five call counts, reproduces every exact cohort median of shortlist minus call-matched random subset at a budget of 64, under random-size and fixed-size shortlists and both graders, to within 0.00041. Each cell is three MATH-500 configurations by two reward models by 200 prompts by $8{,}000$ draws, that is $9{,}600{,}000$ return events. The fourth MATH-500 pool carries no second-stage verifier labels and enters the aggregation comparison only, so the call-matched comparison's cohort is the three pools by two reward models of Section 6.

How often the graders disagree on a returned output. At two calls, with the official extractor selecting, the graders disagree on 0.1353 of the shortlist's returns and 0.1385 of the random subset's; with Math-Verify selecting, on 0.1661 and 0.2624. The rate over the three pools' candidates is 0.1690.

The blind reading. 270 return events, 45 from each of the six strata formed by rule and call count under the official selection, were drawn from each stratum's own return-event distribution. They carry 246 distinct candidates, a candidate returned twice counting twice, so every rate below is a return-event rate. Each sheet gives the problem and the candidate text alone, shuffled, and the adjudicator's answer was recorded against a unit identifier before any extractor output was joined to it. The adjudicator is the language model of Section G.2. Conditional on the two graders disagreeing, which is what these strata select for, the official extractor agrees with the first adjudication on 0.1302, the extraction-only Math-Verify arm on 0.8698 and the native arm on 0.9739, which are paired gains over the official extractor of $+0.7396$ and $+0.8437$. Under the scripted reading the same rates are 0.1818, 0.8182 and 0.9223, a paired gain of $+0.6364$, so the ordering is unchanged and the gap narrows by about a tenth. Of the official extractor's errors, 235 under the first reading and 221 under the scripted one, every one is a correct answer graded incorrect.

The two Math-Verify arms fail differently. Both award a grade the official extractor withholds on the same 242 returns, and the adjudicator sides with them on 235 of those under the first reading and 221 under the scripted one. The extraction-only arm also withholds a grade the official extractor awards on 28 returns, and the adjudicator sides with it on none of them; on these returns the unchanged

PRM800K grader rejects the string Math-Verify passes to it. The native arm has no such stratum, so its only errors are the seven over-awards; the extraction-only arm errs seven times in the same direction and 28 times in the other.

Five versions of the shortlist-minus-subset difference. In this pipeline the second stage is an exact verifier whose pass label is the correctness label, so the grader that scores is also the verifier that selects. The shortlist rule returns the highest-reward drawn member it passes, and the call-matched random subset returns one of its passing members, drawn uniformly. Versions A and B are the official arm and the extraction-only Math-Verify arm on the same draws, each selecting and scoring; C holds the selection under the official extractor and regrades the same returns. D substitutes, under each reading, the adjudicator's grade on the sampled disagreements of the official selection, the population the blind-reading sheets were drawn from, and E does the same on the Math-Verify selection's own disagreement returns, read by the scripted reader alone. Resolving every disagreement of the official selection in the official extractor's favour gives A exactly and in Math-Verify's favour C; D can fall outside that range, because the two rules' returns carry different disagreement compositions and the adjudicator sides with Math-Verify at a different rate on each. Table 21 gives the unaudited and the audited versions.

Table 21: **Regrading or auditing the official selection keeps the shortlist ahead at two and four calls; Math-Verify selection puts it behind at two.** Shortlist minus call-matched random subset on MATH-500 at a budget of 64, median over three configurations by two reward models on the same realised draws, with 95% prompt-cluster intervals below. Math-Verify here is the extraction-only arm; Figure 1f and Figure 3b plot the native one. A and B: each grader selects and scores; C: the official selection regraded; D and E: the adjudicator's grade substituted on sampled disagreement returns of the official or the Math-Verify selection.

Unaudited			
Calls	A official selects, scores	B Math-Verify selects, scores	C regrade only (official selects)
1	-0.0022 $[-0.0059, 0.0025]$	-0.0215 $[-0.0295, -0.0152]$	-0.0028 $[-0.0059, 0.0019]$
2	0.0164 $[0.0076, 0.0231]$	-0.0161 $[-0.0268, -0.0072]$	0.0113 $[0.0038, 0.0175]$
4	0.0416 $[0.0293, 0.0530]$	-0.0011 $[-0.0099, 0.0111]$	0.0315 $[0.0201, 0.0410]$
8	0.0585 $[0.0444, 0.0713]$	0.0127 $[0.0041, 0.0256]$	0.0455 $[0.0329, 0.0564]$
16	0.0581 $[0.0432, 0.0707]$	0.0196 $[0.0093, 0.0316]$	0.0485 $[0.0344, 0.0583]$
Disagreement-audit-corrected			
Calls	D official selects, first reading	D official selects, scripted reading	E Math-Verify selects, scripted reading
1	-0.0025 $[-0.0105, 0.0120]$	-0.0017 $[-0.0145, 0.0211]$	-0.0186 $[-0.0508, 0.0043]$
2	0.0210 $[0.0026, 0.0408]$	0.0160 $[-0.0064, 0.0426]$	-0.0126 $[-0.0450, 0.0104]$
4	0.0280 $[0.0138, 0.0399]$	0.0244 $[0.0048, 0.0446]$	0.0492 $[0.0045, 0.1085]$

The six audit strata are the two rules crossed with the call counts 1, 2 and 4, so no sheet was drawn at eight or sixteen calls and the audited block stops at four. Returning the highest-reward passer of the verified subset instead of a uniformly drawn one moves A, B and C by at most 0.00009 at one, two and four calls. Versions B to E keep the PRM800K equivalence rule and change only how the answer is read. Native Math-Verify selecting and scoring gives -0.0208, -0.0157, -0.0015, 0.0129 and 0.0210 from one call to sixteen, and native regrading of the official selection gives -0.0025, 0.0107, 0.0310, 0.0447 and 0.0484; these are the arms plotted in Figure 1f and Figure 3b. As differences of cohort medians taken along the path A, C, B, the move from A to B splits into a regrade, C minus A, and a change of verifier, B minus C. At every call count between seven tenths and nearly all of the move comes with the change of verifier, -0.0274 against -0.0051 for the regrade at two calls. The

direct reading of the Math-Verify selection below gives negative point estimates at one and two calls, with intervals that include zero, and a positive one at four. The regrade-only column is negative only at one call, where its interval covers zero, and the full Math-Verify policy itself leads from eight calls on.

The audited columns under the scripted reading. Under the scripted verdicts D stays positive wherever it was under the first and separated from zero at four calls. At two calls its per-stratum interval covers zero, while with the siding rate pooled across the six strata it stays separated, at 0.0123 against 0.0121 under the first reading. Both readings grade the same sampled returns, so the two D columns differ only through the verdicts.

The extraction-only selection read directly. The scripted reader was also run on 270 disagreement returns drawn from the Math-Verify selection's own strata, the same six rule-by-call cells with 45 events each and 263 distinct candidates. It sides with Math-Verify on 221 of the 270 events, as it does on the official selection, and again never finds a wrong answer the official extractor graded correct. Read on these returns under the scripted reading, the disagreement-audit-corrected version E, the last column of Table 21, is -0.0126 at two calls, negative in all six configuration-by-reward-model rows, and $+0.0492$ at four, positive and separated from zero. There the random subset's returns carry more Math-Verify credits, while the reader sides with Math-Verify on a smaller share of them, 0.762 against 0.875 for the shortlist.

H Realised second stages

H.1 An exact verifier

An exact verifier is one whose pass/fail label equals the binary gold. A second stage that uses it to return a correct drawn shortlist member whenever the draw holds one, and the Best-of-N pick otherwise, is the shortlist's gold oracle of Proposition 3.1. Its realised gain over the Best-of-N pick is Λ_N conditional on a hit, and unconditionally it equals the oracle ceiling $Z\Lambda_N$.

H.2 Prompt-visible assertions graded by MBPP+

MBPP supplies three assertions inside the generation prompt, and MBPP+ extends the tests of 90 of the 200 tasks in our MBPP cohort (Section F.4). On those tasks the second stage returns, on a hit, the drawn member with the highest reward score among those passing the prompt-visible assertions, and the Best-of-N pick when none passes or on a miss. Gold is the MBPP+ suite, and intervals resample tasks. Each pool holds 64 programs per task, so at a budget of 64 a draw is as large as the pool and Section 7 reports a budget of 16. Of the $5{,}760$ graded programs in each pool, 692 on Llama-3.2-3B and 1,397 on Qwen2.5-Coder-7B pass the assertions but fail MBPP+. Section H.4 compares the runners-up rule with the shortlist on these pools.

H.3 The cost of verifying a shortlist

Exhaustively checking a shortlist of exact mass c/N costs c expected calls, the fraction c/N of full-draw verification. The comparison below takes the inclusive shortlist at a budget of 64 under the size-N subset draw and indexes it by its width c; where reward scores tie, its measured cost exceeds c, a median of 9.28 expected calls at a width of four on the 64-candidate MBPP pools, which a budget of 64 draws whole. The baseline generates c candidates and verifies all of them for c calls, fewer than the shortlist spends on MBPP at each width from one to sixteen. Against it the shortlist is within noise at a width of sixteen on GSM8K, $+0.0023$, ahead at four, $+0.0167$, and within noise on MBPP at every width, from $+0.0115$ at one to -0.0118 at sixteen. On GSM8K the verifier is exact and the gold official; on MBPP the verifier is the prompt-visible assertions of Section H.2 and the gold MBPP+, at a budget of 64 rather than the 16 used there.

H.4 The runners-up rule by second-stage type

Against a call-matched random subset, which keeps the same fallback, the margin of the runners-up rule of Section 7 splits per cell into an extra-rank term $C_{k+1} - C_k$, non-negative by construction, and

a remainder comparing C_k with the random subset. At one call C_1 is that fallback's own accuracy, so by construction the remainder cannot be positive there. On GSM8K at a budget of 64 the one-call advantage of $+0.018$ is $+0.050$ of extra rank against -0.032 of remainder. The remainder is -0.0045 at two calls and $+0.0044$, $+0.0060$ and $+0.0066$ at four, eight and sixteen, against a margin of $+0.0090$ at sixteen. Resampling the 19 GSM8K configurations instead of prompt clusters, the margin over the random subset stays positive at one call and its interval contains zero from four calls on, so only the one- and two-call statements hold at both resampling levels.

The runners-up rule of Section 7 verifies the drawn candidates ranked 2 to $k+1$, for $k \leq N-1$, and keeps the Best-of-N pick otherwise, so the set it can return is the fixed list of the $k+1$ highest-scoring drawn candidates, which always holds the pick. Its guarantee holds when the second stage's pass/fail label agrees with the gold and the Best-of-N pick is kept as the fallback. Under those conditions checking the pick cannot change whether the answer is correct, so the rule's accuracy is top-$(k{+}1)$ coverage. Under an exact verifier the rule then attains that list's oracle ceiling (Section 3), since its gain $C_{k+1} - C_1$ is the expected largest gold value in the list minus that of the pick. The measurements below use the primary protocol and the official gold unless a line says otherwise. By enumeration on small pools, the rule's accuracy equals top-$(k{+}1)$ coverage under a pass/fail label, while reading the same label as a score and returning the highest-scoring runner-up, which also drops the fallback to the pick, moves it by up to 0.80. Against a fixed top-k list on the same k calls the rule's margin has no remainder, because that list's accuracy is C_k by construction. A threshold shortlist of the same expected size admits a random number of candidates and falls back on misses, so its accuracy is not C_k and the margin against it carries both terms.

An exact verifier. Over the 54 benchmark, budget and width cells of the 35 rule-checkable configurations the rule's median paired difference is positive against the threshold shortlist, the fixed top-k list, a call-matched random subset of fixed or binomial size, the Best-of-N pick and, on the mathematics pools, Best-of-Majority and zero-call majority voting, with a single exception. Table 22 prints the first three at a budget of 64 and one call. Over configurations rather than prompt clusters the GSM8K gain over the shortlist stays positive. The exception in the 54 cells is zero-call majority voting on MATH-500 at one call, -0.0077, which turns positive from two calls (Table 23).

The shortlist's one-call deficit. On GSM8K at a budget of 64 the shortlist's one-call deficit against the random subset of Section 7 is -0.0175, and -0.0258 on prompts with pass fractions below one half. It is negative in all 19 configurations, whose mean proxy tie shares run from 0.084 to 0.693. The shortlist's records share answers. At width one its pool records carry 4.005 distinct responses but 1.38 distinct answers, against 4.048 and 2.43 for as many records chosen uniformly. Keeping one copy of each exact duplicate response in a prompt's pool leaves the deficit at -0.0173. Across the 19 configurations mean within-prompt AUROC has a rank correlation of $+0.09$ with the deficit and $+0.91$ with the whole-pool advantage of the top-ranked candidate over one uniform draw.

Charging the shortlist only for the calls it needs. The threshold shortlist spends a call on the Best-of-N pick whenever the draw hits it, and under an exact verifier that check cannot change whether the returned answer is correct. Verifying the other members and keeping the pick when none passes returns a correct answer on exactly the same draws. The bill falls from K to $(K-1)_+$, whose mean is $c - Z_c$. The Best-of-N pick, the ranked runners-up and shortlist membership all follow one keyed order, proxy score and then the uniform key of Section C.2, so the pick is a member on every hit. Widened to the width c^* with $c^* - Z_{c^*} = k$, the shortlist spends the runners-up rule's k expected calls, with a largest simulated deviation of 0.0080 over the 315 configuration cells. It still trails the rule in all 54 benchmark medians and in 309 of those cells, by 0.0067 to 0.0271 at one call, against 0.0274 to 0.0780 when the shortlist is charged for the pick (Table 3).

Prompt-visible assertions as the pass/fail stage. MBPP's assertions, graded against the withheld MBPP+ suite of Section H.2, give four units at a budget of 16. The rule's accuracy rises with the calls it spends (Table 23). It beats the shortlist at one call by 0.0252, a gain that stays positive over configurations as well as over task clusters, and at two calls by 0.0198, in 4 of 4 units both times. The difference contains zero at four calls and turns negative at eight. Against a fixed-size random subset its median difference falls from $+0.0017$ at one call to -0.0096 at eight, where no unit is positive. At two calls the inclusive shortlist on these 64-candidate pools spends 3.57 expected calls against the rule's 2.00.

Table 22: **With an exact verifier the runners-up rule leads all three comparators on all six benchmarks.** Rule minus comparator accuracy at a budget of 64 and one call, median over each benchmark's rule-checkable configurations (primary protocol, official gold). The random subset verifies one uniformly chosen drawn candidate. The shortlist difference is positive in all 19 GSM8K and 11 MATH-500 configurations.

Benchmark	Threshold shortlist	Fixed top-k list	Call-matched random subset
GSM8K	+0.0346	+0.0498	+0.0182
MATH-500	+0.0436	+0.0658	+0.0252
HumanEval	+0.0426	+0.0627	+0.0240
HumanEval+	+0.0428	+0.0625	+0.0218
LiveCodeBench	+0.0274	+0.0376	+0.0128
BBH-logical	+0.0763	+0.1047	+0.0447

Table 23: **The runners-up rule's accuracy and its lead over zero-call majority voting both grow with the calls it spends.** Medians under the primary protocol, over the four MBPP units (two generators by two reward models) and the 11 MATH-500 configurations. A dash marks sixteen calls at $N = 16$, beyond the fifteen runners-up.

Calls	1	2	4	8	16
Accuracy, MBPP prompt-visible assertions graded by MBPP+, $N = 16$					
Runners-up rule	0.5578	0.5719	0.5812	0.5900	–
Minus zero-call majority voting, exact verifier on MATH-500, $N = 64$					
Runners-up rule	−0.0077	+0.0083	+0.0500	+0.0951	+0.1247

Score-only selection, an ablation that removes the fallback. The arm below scores only the candidates it is given and returns the argmax over those, so the Best-of-N pick is discarded, and Section 7 reports the same process model with the fallback restored. At one call the arm scores a single candidate, the runner-up, and returns it whatever the score; its one-call cells are a one-candidate control of discarding the pick. On the nine PRM-scored pools with two reward models each, at a budget of 64 and one call, the rule trails the threshold shortlist, the fixed top-k list and keeping the Best-of-N pick, while staying ahead of a call-matched random subset (Table 24). The deficits shrink with the calls, to -0.0015 and -0.0014 against the shortlist at four and sixteen calls on GSM8K. Across the 216 cells formed by the three official aggregations and a constant score under the official, answer-bearing and cap-dropped gold arms, the rule's cohort median trails the shortlist in 204 and leads in 12. With the fallback restored the rule leads the shortlist at one call on GSM8K (Section 7). Section G.1 gives what the answer-bearing restriction does to these contrasts on MATH-500.

Table 24: **Without the Best-of-N fallback, the runners-up rule trails the threshold shortlist on both benchmarks.** Rule minus comparator accuracy at a budget of 64 and one call, median over nine Qwen2.5-Math-PRM-7B pools each read with two reward models, under product aggregation. Brackets are 95% prompt-cluster intervals; Units counts positive shortlist differences. The random subset has binomial size with mean one.

Benchmark	Threshold shortlist	Units	Best-of-N pick	Call-matched random subset
GSM8K	−0.0097 [−0.0166, −0.0064]	0 of 10	−0.0055	+0.0476
MATH-500	−0.0317 [−0.0388, −0.0250]	0 of 8	−0.0251	+0.0681

The one-call deficits. At one call the rule and the top-k list differ by exactly one candidate, the Best-of-N pick. Verifying it cannot change the returned answer, since a pick that passes would be returned anyway, so the rule spends its call on a candidate the list cannot reach. In this ablation, returning the runner-up in place of the pick costs 0.0055 on GSM8K and 0.0251 on MATH-500, the whole of the rule's one-call deficit against the top-k list and the Best-of-N pick.

H.5 THE FALLBACK-MATCHED CONTROL

The score-only arm of Section 7 differs from the exact verifier in its fallback as well as in its signal. Under the exact verifier the runners-up rule returns the unverified Best-of-N pick when no runner-up passes, whereas the score-only arm returns the runner-up the process model scores highest and can never return the pick. This control gives the process model the same fallback and measures how much of the score-only arm's deficit against the threshold shortlist remains.

Cohort and draws. The nine pools carrying Qwen2.5-Math-PRM-7B step scores, five on GSM8K and four on MATH-500, are read with two reward models each, giving 18 configurations of 200 prompts and 256 candidates. The draws are $4{,}000$ subsets of size N drawn without replacement per prompt. The gold is the official rule check throughout. Budgets are 16 and 64, calls run over $1, 2, 4, 8$ at a budget of 16 and over $1, 2, 4, 8, 16$ at 64, and each cell is measured under all three official aggregations of the step scores, their product, their last step and their minimum.

The four arms. Each arm verifies the drawn candidates the reward model ranks 2 to $c{+}1$ and differs only in what it does with the result. The *exact verifier with the fallback* returns the highest-reward runner-up that passes the rule check and the unverified Best-of-N pick otherwise. The *score-only arm* returns the runner-up the process model scores highest, discarding the pick. The *matched arm* returns the highest-scoring runner-up when its score clears a threshold and the unverified pick otherwise. The *random-verifier control* returns a uniformly drawn member of the same runners-up set, discarding the pick like the score-only arm but ignoring the score. The comparators are the Best-of-N pick at zero calls, the threshold shortlist of mass c/N with the Best-of-N fallback on a miss, and the fixed list of the c highest-ranked candidates.

The threshold, and the half it is fitted on. The acceptance statistic is the largest process-model score among the c runners-up in the draw, and the arm accepts when it is at least τ. The grid for each configuration and aggregation is the set of quantiles of all finite step-aggregate scores of that pool, at 46 levels evenly spaced from 0 to 0.9 and 355 further levels from 0.9 to 1, deduplicated, with $-\infty$ and $+\infty$ appended. At $-\infty$ the arm always accepts and is the score-only arm, and at $+\infty$ it always falls back and is the Best-of-N pick, so on its calibration half the fitted threshold is at least as accurate as either.

The 200 prompts of each benchmark prompt list are split once into halves of 100 by a fixed permutation under seed 20260937, and every configuration on that benchmark uses the same split. On a calibration half, $\hat{\tau}$ is the grid point maximising mean accuracy, ties going to the smallest. The fit is cross-fitted. Each prompt is scored at the threshold fitted on the other half, so no prompt's label contributes to the threshold under which it is graded. The grid itself is read off the unlabelled scores of both halves. A threshold is fitted separately for each configuration, aggregation, budget and call count, so the matched arm is a calibrated control for each cell.

What holding the calibration out costs. Against the best threshold chosen on the evaluation half itself, the transferred threshold costs a median 0.13 points on GSM8K and 0.55 points on MATH-500 at a budget of 64 and one call under the product aggregation, with a largest cost over all cells of 3.19 points. The thresholds fitted on the two halves differ substantially, but the accuracy-against-threshold curve is flat near its maximum, so what transfers is the accuracy and not the cut point. On held-out prompts the matched arm accepts the runner-up at a median rate of 0.720 over the 18 configurations at a budget of 64 and one call under the product aggregation, and 0.643 under the last step.

Implementation checks. The score-only arm equals the score-only runs of Section H.4 on all 18 cells. The exact-verifier arm draws its own tie-break keys and agrees with the exact-verifier cells there to 1.0×10^{-4} in every cell mean on the seven pools that carry one at the matching reward model, and an independent implementation of the matched arm agrees to 3×10^{-8}.

Inference. Intervals are $2{,}000$ multinomial prompt-weight replicates per prompt list, with the cohort median formed inside each replicate, seeds 20260929 on GSM8K and 20261030 on MATH-500, beside a $2{,}000$-replicate configuration bootstrap and the per-configuration sign count. The threshold is fitted once on each half and held fixed across replicates, so the intervals are conditional

on the fitted thresholds. The 18 configurations sit on two datasets, one per benchmark, and two proxy–gold pairs, too few clusters for the sign enumeration of Section J.3 to reach 0.05, so the numbers here are descriptive and carry their configuration counts.

What the control returns. At one call the runners-up set holds a single candidate, so the score-only arm returns it whatever its score, and the random-verifier control returns the same answer as the score-only arm in all 108 one-call configuration cells. The matched arm still uses the score at one call, to decide whether that runner-up is accepted or the pick is kept. From two calls onwards the score also decides which runner-up the score-only arm returns, and the random-verifier control falls below it in all 378 configuration cells with two or more calls. Against the threshold shortlist the score-only arm is negative in 439 of 486 configuration cells and the matched arm in 257; at one call the count moves from 0 of 108 positive to 72 of 108. The reading rule, fixed before the run, is that the matched arm removes the deficit when its cohort median against the threshold shortlist is at or above zero at one call on both benchmarks with the configuration sign count no longer one-sidedly negative. It is met under the last step on both benchmarks and on GSM8K under all three aggregations. It is not met on MATH-500 under the product or the minimum, where the matched arm still trails the shortlist by -0.0066 and -0.0059 at a budget of 64, against -0.0317 and -0.0312 for the score-only arm. At one call and a budget of 64 the matched arm moves the difference against the shortlist from the score-only arm's -0.0097 to 0.0093 on GSM8K under the product, 0 of 10 configurations positive to 10 of 10, and from -0.0346 to 0.0029 on MATH-500 under the last step. At the widest settings it trails the shortlist on GSM8K and on MATH-500 under the last step, while the exact verifier leads the shortlist there.

H.6 Fidelity of the process reward model

Qwen2.5-Math-PRM-7B (Table 28) is run with the inference code of its model card (Qwen Team, 2025), in float16 instead of the card's bfloat16, and re-scoring 861 candidates reproduces the step scores used here bit for bit. On the same candidates float16 stays closer to float32 than bfloat16 does, with a largest step-score difference of 0.018 against 0.134. The official split on blank lines leaves 7.5% of candidates with a single step, rising to 43.3% on two pools, and on those candidates the three aggregations coincide by construction. Restricted to multi-step candidates, the product aggregation's within-prompt AUROC against correctness falls from 0.9053 to 0.8738 on the pool where it is highest, while the last-step aggregation's rises from 0.8290 to 0.8549, so the comparison between the aggregations depends on the step splitter.

I Three selection comparisons from the literature, measured on our pools under four evaluation conditions

The three measurement choices of Sections 5.1, 5.2 and 6 also apply to standard selection comparisons. We recomputed three comparisons reported in the literature on our own pools and generators under four conditions fixed before any of them was recomputed. **A**, the baseline convention, resamples with replacement and scores against the learned gold with the official grader on the whole pool. Accuracy and pass@N need a correct-or-incorrect label, so the learned gold enters A and B binarised as in Section F.2: it marks as correct the candidates of each prompt it ranks highest, as many as the rule check marks correct there, which gives the two golds the same pass fraction on every prompt. Under a uniform draw pass@N depends on that pass count alone, so it is the same under both golds at a given draw law. **B** changes the draw law alone, to size-N subsets without replacement. **C** substitutes the rule-checkable gold alone. **D** applies the subset law and the checkable gold and also restricts the candidates to answer-bearing ones wherever the official extractor can fail; the restriction removes candidates and leaves the grader unchanged. A post hoc control, **X**, keeps the subset law and the checkable gold on the whole pool, so it is D without the candidate restriction, and it is printed where it separates C from D. The cohort is the 35 rule-checkable configurations (19 GSM8K, 11 MATH-500, four code pools, one BBH), of which seven carry Qwen2.5-Math-PRM-7B step scores beside a learned gold. Inference is over the ten proxy-gold pairs (Section J.3), by a wild cluster bootstrap-t with the null imposed and all 1,024 sign vectors enumerated; with HumanEval and HumanEval+ merged the cohort spans five dataset clusters, which enter as sign counts. Table 25 is the summary.

Table 25: **Changing the gold or restricting the candidates moves three comparisons from the literature on our pools more than changing the draw law.** Median per-configuration differences, as proportions, under conditions A to D of Section I. Parentheses give the cohort size and the motivating source after each comparison and, in cells, the configurations agreeing in sign with the direction that source reports.

Comparison (cohort; source)	A baseline	B law	C gold	D all
pass@N minus Best-of-N, N=64 (35; Brown et al. 2024)	+0.1387 (35)	+0.1403 (35)	+0.1929 (35)	+0.2171 (35)
Top-1 by reward minus one uniform sample, N=64 (35; Dughmi et al. 2026)	+0.1764 (35)	+0.1801 (35)	+0.1562 (35)	+0.1060 (34)
Product minus last-step aggregation, N=8 (7; Zhang et al. 2025b)	−0.0053 (3)	−0.0054 (3)	+0.0004 (4)	+0.0066 (7)

The gap between coverage and what Best-of-N realises is larger under the rule-checkable gold. Brown et al. (2024) report that coverage keeps scaling with the budget while selection plateaus below it, beyond several hundred samples. Our budgets reach 64, so the recomputed quantity is the gap between the two and its widening from a budget of 4. On our pools the budget-64 gap between pass@N and Best-of-N accuracy is positive in all 35 configurations under every one of the four conditions. Its median rises from 0.1387 under A to 0.1929 under C on the same prompts, and to 0.2171 under D. Substituting the gold alone takes the gap to 139 per cent of its value under A, and D to 157 per cent. Restricting C to the prompts D keeps, with its law, gold and whole pool unchanged, moves the median to 0.2146, the subset law then takes it to 0.2171, and the candidate restriction leaves it there. On MATH-500 the median is 0.2311 under C, 0.3462 on D's prompts alone and 0.3629 under D. On GSM8K, where all 19 configurations keep every prompt under every condition, the gap nearly doubles, from 0.0989 under A, 19 of 19 positive, to 0.1921 under D, again 19 of 19. Between budgets of 4 and 64 under D, coverage rises by a median 0.2239 and Best-of-N by 0.0439, while the gap between them widens by a median 0.1311, sign consistent across all ten pairs and all five dataset clusters, at a pair-level $p = 0.001953$. The Best-of-N growth term alone has a pair-level $p = 0.0762$ under D, with 33 of 35 configurations positive.

The top-ranked sample's median advantage over one uniform sample stays positive and falls under D. Dughmi et al. (2026) report that ranking by the reward model raises coverage even at $k = 1$. The comparison is significant over the ten proxy-gold pairs under A, B, C and X, at p from 0.0156 to 0.0332. At a budget of 64 the advantage of the top-ranked candidate over one uniform draw is a median 0.1764 under A, positive in all 35 configurations, 0.1801 under B, 0.1562 under C and 0.1585 under X, each of them 35 of 35. Under D the median falls to 0.1060, one of the 35 configurations turns negative, and the pair-level enumeration returns $p = 0.0723$, while all ten pairs and all five dataset clusters stay positive. Restricting C to the prompts D keeps *raises* the comparison to 0.2123, where the pair-level enumeration returns 0.0156; the subset law and the candidate restriction then take it to 0.1060. On MATH-500 the median is 0.1572 under A and 0.1562 under C, 11 of 11 positive, and 0.0923 under D with one of the eleven negative; on GSM8K it is 0.3106 under A, 0.2294 under C and 0.2325 under D, all 19 positive.

All seven pools favour the product over last-step aggregation only under D. Zhang et al. (2025b, Table 13) report, for a Best-of-8 average over seven benchmarks with Qwen2.5-Math-7B-Instruct as the policy, 67.6 for the product aggregation of Qwen2.5-Math-PRM-7B against 66.1 for the last step. Our seven pools carrying both the step scores and a learned gold come from other generators, so the comparable quantity is the sign of the contrast. Its median is -0.0053 under A, with the product ahead in 3 of the 7 pools, and $+0.0066$ under D, with the product ahead in all seven. Restricting C on the three MATH-500 pools to the 147, 141 and 142 of 200 prompts that D keeps gives -0.0504, negative in all three and below C's -0.0387 on all 200 prompts; the subset law leaves it negative, and the candidate restriction turns it to $+0.0260$. The seven pools span two dataset clusters and two proxy-gold pairs, where the smallest attainable enumeration p is 0.5, so these results are descriptive.

The estimator's own error. Gao et al. (2023a) use the without-replacement estimator of Nakano et al. (2021) in place of resampling with replacement. Its size on our pools, with replacement minus without, stays in the third decimal. Under the rule-checkable gold, Best-of-N accuracy is understated by -0.00032, -0.00085 and -0.00191 at budgets of 4, 16 and 64, pass@N by -0.00075, -0.00256 and -0.00506, top-1 coverage at 64 by -0.00191 and top-2 coverage by -0.00881; under the learned gold, Best-of-64 accuracy by -0.00306 and top-2 coverage by -0.00952. Each of these differences is negative in all ten proxy-gold pairs, at a pair-level p from 0.001953 to 0.0332, and all 35 configurations are negative except at the two budget-64 cells under the checkable gold, where 29 of 35 are. Every A-to-B contrast in Table 25 is smaller than 0.004, while the same estimator choice moves the recoverable share this paper introduces by whole percentage points (Section 5.1).

J THE TIE-PRESERVING RANK-MATCHED REFERENCE

The rank-matched reference is an exploratory sensitivity analysis that asks how far the measured split sits from a split built at each prompt's measured rank agreement between proxy and gold. Every quantity here is a departure, a measured value minus the value of a reference built on the same draws from a copula whose population Spearman correlation is each prompt's measured one. All of them sit on the with-replacement draws of Section C.3 under each configuration's learned gold, so their levels differ from those of Section 4, which uses the primary protocol. For comparisons of reward models the unit of replication is the proxy-gold pair, of which this cohort has ten, and Section J.3 tests the departures over pairs and over prompt lists.

J.1 CONSTRUCTION

Definition J.1 (Tie-preserving rank-matched reference)**.** For each prompt, collapse the pool to its distinct (proxy, gold) score pairs with their multiplicities, and compute Spearman's ρ on the full pool with average ranks. Draw a Gaussian copula with latent correlation $2\sin(\pi\rho/6)$ for the distinct pairs, map its proxy and gold ranks onto the sorted proxy and gold values of the distinct pairs, and give every copy of a distinct pair the same new pair, so records that share a (proxy, gold) pair keep sharing one.

All references keep the measured arm's draws: one seed-7 stream of sampled candidate indices per configuration and budget, the replicate rule of Section C.3 and the same shortlist rule, so the arms differ only in the (proxy, gold) pairs attached to each candidate index. Undefined Spearman values are set to zero. This construction, R-dup, keeps both score distributions exactly on prompts without duplicate (proxy, gold) pairs; on prompts with repeated pairs the multiplicities move with the new pairs, so the count-weighted score distributions can change there. It targets the measured per-prompt Spearman correlation as the copula's population value rather than as the realised correlation of each reference pool. Because the shortlist depends on proxy ranks alone, hit probability and realised shortlist mass agree with the measured arm to about 0.1% in every cell.

Copulas with the same rank correlation can differ in tail dependence (Nelsen, 2006), so the departures measure distance from the Gaussian family at each prompt's rank agreement, and Section J.2 measures how far five other families sit. The recoverable-share and admission-share departures use 20 seeds, 1001 to 1020, and the fixed-list departures use 40, seed 11 and 1001 to 1039. A departure is the measured value minus the mean over that item's seeds, in percentage points for shares and in per cent of the reference for the parts of regret. Intervals use 2,000 bootstrap replicates over distinct prompt texts, with one set of weights per prompt list, shared by every configuration on that list and by all arms and budgets (Section C.3).

J.2 SIX COPULA FAMILIES AT THE SAME RANK AGREEMENT

Repeating the construction with Student-$t(3)$, Frank, Clayton, survival Clayton and Gumbel copulas, each calibrated per prompt to the measured Spearman correlation, gives Table 26. Calibration matches the per-prompt population Spearman to 7×10^{-10} over 8,928 prompts and six families; 266 prompts with negative rank agreement are rotated by 90 degrees for the three Archimedean families, eight prompts have zero agreement and one is clipped at 0.9989.

None of the six families matches the split in every quantity. Counting the quantity-cell pairs whose departure interval excludes zero over the five budget and width cells, the Gaussian reference has 25 of 34 and the other five have 33 or 34. The Gaussian is closest on ordering loss. Its departure interval contains zero at one expected candidate, -1.07% at a budget of 16 and $+0.31\%$ at 64, and at four expected candidates, $+2.00\%$ and $+0.01\%$, whereas the other five families move ordering loss at a budget of 64 and one expected candidate by at least 23 per cent in absolute size, three downwards and two upwards. Hit probability is family-invariant, with cohort-median relative departures of at most 0.16% everywhere.

Reference departures depend on the chosen copula family. At a budget of 64 and four expected candidates the recoverable-share departure is -2.84 points against the Gaussian reference, -16.02 against Student-$t(3)$, -0.28 against Frank and $+4.00$ against Clayton, so it widens, vanishes or reverses with the choice. At one expected candidate the measured value of each part of the split lies inside the range of the six seed-averaged references in 44 to 45 of the 45 configurations at budgets of 16 and 64, and the recoverable share in 34 and 36, while the sign of each departure in Table 26 changes with the family.

Table 26: **At the same per-prompt rank agreement, the size and sign of each departure depend on the copula family.** Cohort medians over the 45 configurations, with 95% prompt-cluster intervals, of the measured value minus each family's seed-averaged reference on the with-replacement draws under the learned gold; per-cent columns are relative departures. Cells counts quantity-cell pairs whose interval excludes zero, over seven quantities (Λ, Ξ, $Z\Lambda$, $Z\Xi$, $\Delta J - Z\Lambda$, regret on misses, ΔJ) at $N = 16$, $c \in \{1, 4\}$ and $N = 64$, $c \in \{1, 4, 16\}$; regret on misses at $N = 64$, $c = 16$ is not estimable, leaving 34. The recoverable share $Z\Lambda/\Delta J$ is a ratio of two counted quantities and is itself outside the count, so Frank counts 34 while its recoverable-share interval contains zero.

Family	Ordering loss (%) $N = 64$, $c = 1$	Recoverable share (pp) $N = 64$, $c = 4$	Regret ΔJ (%) $N = 64$	Cells
Gaussian	$+0.31\,[-2.82, 6.28]$	$-2.84\,[-3.71, -2.19]$	$+7.93\,[4.96, 10.88]$	25/34
Student-$t(3)$	$-32.02\,[-36.39, -29.44]$	$-16.02\,[-17.02, -15.27]$	$+11.67\,[10.21, 15.82]$	34/34
Frank	$-23.31\,[-26.71, -20.87]$	$-0.28\,[-1.02, 0.74]$	$-23.29\,[-25.37, -20.61]$	34/34
Clayton	$-26.63\,[-30.05, -24.41]$	$+4.00\,[3.38, 5.50]$	$-35.96\,[-37.42, -33.32]$	34/34
Survival Clayton	$+106.18\,[86.36, 120.66]$	$-8.92\,[-9.69, -7.73]$	$+143.34\,[131.22, 165.08]$	34/34
Gumbel	$+38.32\,[29.65, 50.60]$	$-8.94\,[-10.04, -8.38]$	$+76.21\,[67.05, 83.95]$	33/34

J.3 Cluster-robust inference for cohort comparisons

The departures of this reference are cohort medians with prompt-cluster intervals, but their units of replication are coarser, since the cohort's 45 configurations rest on 40 pools, 14 prompt lists, 15 datasets and 10 proxy-gold pairs, one of which carries 34 of the 45. Section F.6 takes the gold effect to seventeen pools on seven datasets and 21 proxy-gold pairs, 16 of these pairs absent from the cohort, where it is separated from zero at all three levels. We test the configuration-weighted mean at each coarse level on its own, by a cluster-robust interval with $t(G - 1)$ on the level's G clusters, a Rademacher wild cluster bootstrap-t with the null imposed and every sign vector enumerated, and an exact cluster sign test. HumanEval and HumanEval+ share their 164 problems and count as one dataset cluster, so on the full support the dataset level is the 14 prompt lists, with 16,384 sign vectors against 1,024 for the ten pairs. On the 35 rule-checkable configurations the merge leaves five dataset clusters, whose smallest attainable two-sided enumeration p, $2/2^5 = 0.0625$, cannot reach 0.05, so inference on that cohort is reported over the ten pairs, with the dataset clusters as sign counts. For the gold effect the dataset-level evidence is the seventeen-pool extension. The 18 configurations of the process-reward-model second stage (Sections H.4 and H.5) sit on two datasets and two proxy-gold pairs, where the smallest attainable two-sided p is 0.5. The printed interval is the wider of the one-way pair and prompt-list intervals; printing it in place of the two-way dataset-by-pair interval the design named, and merging the two HumanEval suites, are post hoc choices. With one pair holding 34 of the 45 configurations, a printed interval and its pair-level p can disagree.

Table 27 applies these tests to the reference's sign claims one quantity at a time, each a separate test with no multiplicity adjustment, and prints six departures at a budget of 64 and four of the seven top-k departures. The seven gaps of Table 1 are direct differences, fixed list minus threshold shortlist

of the same expected size; each has all ten pairs on one side and the smallest pair-level p ten pairs admit, 0.001953. The recoverable-share and admission-share departures at a budget of 4 or 16 and the three unprinted top-k departures carry pair-level p from 0.0312 to 0.5312 and prompt-list p from 0.0195 to 0.6714. The departures in the parts of regret carry pair-level p from 0.0039 to 0.9980, and each one below 0.05 has an interval covering zero.

Table 27: **Departures from the R-dup reference, tested over proxy-gold pairs and prompt lists.** Rows are departures on the with-replacement draws over the 45 configurations. Recoverable-share, admission-share and top-k departures are in percentage points, ordering loss in per cent of the reference. Median is descriptive; the interval, design and p-values refer to Mean, the configuration-weighted mean that the design tests. The interval is the wider of the 95% cluster-robust intervals over the ten proxy-gold pairs and over the 14 prompt lists, which count HumanEval and HumanEval+ as one; Design names the level printed. The two p columns are wild cluster bootstrap p-values at those two levels (Section J.3); Pairs counts pairs with a positive/negative median.

Quantity	Median	Mean	Interval	Design	p pair	p list	Pairs
Admission share, $N = 64$, $c = 1$	$+1.47$	$+1.20$	$[0.31, 2.09]$	list $t(13)$	0.0508	0.0410	9/1
Recoverable share, $N = 64$, $c = 1$	-0.66	-0.49	$[-0.95, -0.04]$	list $t(13)$	0.0234	0.0958	1/9
Ordering loss, $N = 64$, $c = 1$	$+0.31$	$+3.36$	$[-7.69, 14.41]$	pair $t(9)$	0.5586	0.3591	9/1
Recoverable share, $N = 64$, $c = 4$	-2.84	-2.66	$[-3.93, -1.39]$	list $t(13)$	0.0352	0.0464	2/8
Admission share, $N = 64$, $c = 4$	$+3.04$	$+2.81$	$[1.47, 4.15]$	list $t(13)$	0.0410	0.0432	8/2
Recoverable share, $N = 64$, $c = 16$	-2.59	-2.50	$[-3.22, -1.79]$	list $t(13)$	0.0195	0.0082	2/8
Top-4 list, $N = 16$	-2.69	-2.39	$[-3.24, -1.54]$	list $t(13)$	0.0059	0.0181	2/8
Top-8 list, $N = 16$	-1.29	-1.45	$[-1.91, -0.99]$	list $t(13)$	0.0195	0.0045	2/8
Top-8 list, $N = 64$	-3.36	-3.03	$[-4.36, -1.70]$	list $t(13)$	0.0234	0.0256	2/8
Top-16 list, $N = 64$	-2.49	-2.42	$[-3.14, -1.71]$	list $t(13)$	0.0195	0.0085	2/8

K Data, generation, scoring and configurations

K.1 Datasets

The cohort's prompts come from fifteen public datasets: GSM8K (Cobbe et al., 2021) and MATH-500 (Hendrycks et al., 2021; Lightman et al., 2024) for mathematical reasoning, HumanEval (Chen et al., 2021), HumanEval+ (Liu et al., 2023) and LiveCodeBench (Jain et al., 2025) for code generation, and No-Robots (Rajani et al., 2023) and a set of human-written instructions released by Hugging Face (Instr.-follow)[2] for instruction following. The other eight cover multi-step reasoning with BBH-logical (Suzgun et al., 2023), knowledge and science with GPQA (Rein et al., 2024) and MMLU-Pro (Wang et al., 2024b), factuality with TruthfulQA (Lin et al., 2022), dialogue and preference with MT-Bench (Zheng et al., 2023) and Arena-Pref (Chiang et al., 2024), safety with BeaverTails (Ji et al., 2023) and multilingual instructions with Aya (Singh et al., 2024). Three further pools, two on MBPP (Austin et al., 2021) and one on MMLU-Pro, were generated under the same protocol for Section F.6 alone.

K.2 Generation and reward scoring

Candidate generation. The core experiments use temperature 0.70, top-$p = 0.95$ and seed 0, and generate 256 candidates per prompt, except for T20, which uses 1,024. The output limit is 512 tokens, except for T43, which uses 256 and reaches it on 68.05% of its candidates. For each dataset, we sample up to 200 rows without replacement with NumPy seed 0 and evaluate them in their original order. HumanEval and HumanEval+ contribute all 164 of their tasks, MT-Bench's 200 sampled rows represent 73 distinct prompts, and multi-turn datasets contribute their first user turn.

Each generator receives the dataset's prompt field through its chat template with the generation prompt enabled, or the raw question if it is a base model without a template, and no answer-format instruction is added. The extraction and truncation rates of Section 6 and Section G.1 are measured under this prompt and these output limits. Generation runs in Hugging Face Transformers, truncating input at 1024 tokens and stopping at the model's end-of-sequence token, except for T20, which

[2] https://huggingface.co/datasets/HuggingFaceH4/instruction-dataset

runs in vLLM with a 4096-token context, tensor parallelism across eight GPUs and no quantisation. Seeds are set once per generation run.

Reward scoring. Each reward model receives the prompt and response in its own chat template, or in the Llama-3.2-1B-Instruct template if it has none. Scoring uses batches of eight, a 2048-token input limit, and evaluation mode with scaled dot-product attention, in bfloat16 on Ampere or newer GPUs and float16 on Volta GPUs. ArmoRM combines its 19 objective rewards into a scalar preference score through its gating model, and the other reward models return scalar logits. On T8 and T10 to T12, ArmoRM runs through the RewardBench implementation (Lambert et al., 2025) with batches of four.

Reward models and golds. Table 28 gives the Hugging Face checkpoint of each reward model and the short name used for it in the text. In the 45-configuration cohort a separately trained reward model assigns the gold score of each candidate, standardised within each prompt, and the shared-draw comparisons of two reward models in Section F.1 use RM-Mistral as the learned gold. The rule-checkable golds score agreement with the reference answer on GSM8K and MATH-500 and with the reference letter on BBH-logical, and the unit-test outcome on the code pools, each on its original 0–1 scale. Table 29 lists the generator, proxy reward model, gold and dataset of every configuration.

Table 28: **Hugging Face checkpoints of the eight reward models.** The first column gives the name used in the text, tables and figures; the first five are the proxies and golds of the 45 configurations.

Name used here	Hugging Face identifier	Reference
Skywork-V0.2	`Skywork/Skywork-Reward-Llama-3.1-8B-v0.2`	Liu et al. (2024)
Skywork-V2-8B	`Skywork/Skywork-Reward-V2-Llama-3.1-8B`	Liu et al. (2026)
Skywork-V2-3B	`Skywork/Skywork-Reward-V2-Llama-3.2-3B`	Liu et al. (2026)
ArmoRM	`RLHFlow/ArmoRM-Llama3-8B-v0.1`	Wang et al. (2024a)
RM-Mistral	`weqweasdas/RM-Mistral-7B`	Xiong et al. (2024)
GRM-gemma2-2B	`Ray2333/GRM-gemma2-2B-rewardmodel-ft`	Yang et al. (2024)
InternLM2-7B-Reward	`internlm/internlm2-7b-reward`	Cai et al. (2024)
Qwen2.5-Math-PRM-7B	`Qwen/Qwen2.5-Math-PRM-7B`, revision `0610740`	Zhang et al. (2025b)

Table 29: **Fourteen generators from six families, five reward models and fifteen datasets make up the 45 configurations.** Generators are instruction-tuned unless marked base, and Instr.-follow is the instruction set of Section K.1. Configurations marked with the same letter share a candidate pool, and configurations on the same dataset share a prompt list, with HumanEval and HumanEval+ counted as one, so the 45 configurations rest on 40 pools and 14 prompt lists.

ID	Generator	Proxy RM	Gold RM	Dataset	ID	Generator	Proxy RM	Gold RM	Dataset
T1[a]	Llama-3.2-1B	Skywork-V0.2	RM-Mistral	GSM8K	T24	Phi-3.5-mini	Skywork-V0.2	RM-Mistral	GSM8K
T2	Llama-3.2-3B	Skywork-V0.2	RM-Mistral	GSM8K	T25	Llama-3.2-1B	Skywork-V0.2	RM-Mistral	Aya
T3	Llama-3.2-1B	Skywork-V0.2	RM-Mistral	MATH-500	T26	Llama-3.2-1B	Skywork-V0.2	RM-Mistral	BBH-logical
T4	Llama-3.2-1B	Skywork-V0.2	RM-Mistral	Instr.-follow	T27	Llama-3.2-1B	Skywork-V0.2	RM-Mistral	BeaverTails
T5	Llama-3.2-3B	Skywork-V0.2	RM-Mistral	Instr.-follow	T28	Qwen2.5-7B	Skywork-V0.2	RM-Mistral	No-Robots
T6[a]	Llama-3.2-1B	RM-Mistral	Skywork-V0.2	GSM8K	T29	Qwen2.5-7B	Skywork-V0.2	RM-Mistral	MATH-500
T7	Llama-3.2-1B (base)	Skywork-V0.2	RM-Mistral	GSM8K	T30	Qwen2.5-Math-7B	Skywork-V0.2	RM-Mistral	MATH-500
T8	Llama-3.2-1B	ArmoRM	Skywork-V0.2	GSM8K	T31	Llama-3.1-8B	Skywork-V0.2	RM-Mistral	GPQA
T9	Llama-3.2-3B	Skywork-V0.2	RM-Mistral	MATH-500	T32	Llama-3.2-1B	Skywork-V0.2	RM-Mistral	HumanEval+
T10[b]	Llama-3.2-1B	Skywork-V0.2	ArmoRM	GSM8K	T33	Mistral-Nemo-12B	Skywork-V0.2	RM-Mistral	GSM8K
T11[b]	Llama-3.2-1B	ArmoRM	RM-Mistral	GSM8K	T34[d]	Llama-3.2-3B	Skywork-V2-3B	ArmoRM	MATH-500
T12[b]	Llama-3.2-1B	RM-Mistral	ArmoRM	GSM8K	T35	Llama-3.1-8B	Skywork-V0.2	Skywork-V2-8B	MATH-500
T13	Llama-3.1-8B	Skywork-V0.2	RM-Mistral	GSM8K	T36	Mistral-7B-v0.3	Skywork-V2-3B	ArmoRM	GSM8K
T14	Llama-3.1-8B	Skywork-V0.2	RM-Mistral	MATH-500	T37[d]	Llama-3.2-3B	Skywork-V0.2	Skywork-V2-8B	MATH-500
T15	Mistral-7B-v0.3	Skywork-V0.2	RM-Mistral	GSM8K	T38	Phi-3.5-mini	Skywork-V0.2	RM-Mistral	MATH-500
T16	Llama-3.2-1B	Skywork-V0.2	RM-Mistral	HumanEval	T39	Phi-3.5-mini	Skywork-V0.2	RM-Mistral	MMLU-Pro
T17	Llama-3.2-1B	Skywork-V0.2	RM-Mistral	TruthfulQA	T40	Gemma-2-9B	Skywork-V0.2	RM-Mistral	GSM8K
T18[c]	Llama-3.2-1B	Skywork-V2-8B	RM-Mistral	GSM8K	T41	Gemma-2-9B	Skywork-V0.2	RM-Mistral	MATH-500
T19[c]	Llama-3.2-1B	Skywork-V2-3B	RM-Mistral	GSM8K	T42	DeepSeek-Math-7B	Skywork-V0.2	RM-Mistral	GSM8K
T20	Llama-3.3-70B	Skywork-V0.2	RM-Mistral	GSM8K	T43	Llama-3.2-3B	Skywork-V0.2	RM-Mistral	LiveCodeBench
T21	Llama-3.2-1B	Skywork-V0.2	RM-Mistral	MT-Bench	T44	Phi-4-mini	Skywork-V0.2	RM-Mistral	GSM8K
T22	Llama-3.2-1B	Skywork-V0.2	RM-Mistral	Arena-Pref	T45	Phi-4-mini	Skywork-V0.2	RM-Mistral	MATH-500
T23	Qwen2.5-Coder-1.5B	Skywork-V0.2	RM-Mistral	LiveCodeBench					

K.3 CODE POOLS AND UNIT-TEST EXECUTION

Of the code pools of Table 15, C1 is the pool of configuration T16: 256 responses per HumanEval task from Llama-3.2-1B-Instruct under the generation settings above. For C2–C5 we evaluate Llama-3.2-3B-Instruct and Qwen2.5-Coder-7B-Instruct on all 164 HumanEval test tasks and the first 200 tasks in the MBPP full-test split (Austin et al., 2021), generating 64 responses per model and task with temperature 0.80, top-$p = 0.95$, seed 0 and a 640-token output limit. Generation used vLLM 0.6.4.post1, a 3072-token context limit, one V100, float16 and each model's chat template. ArmoRM and Skywork-V0.2 score the same responses with their own chat templates, a 2048-token input limit and batches of eight. On MBPP the scoring prompt is the task description, and the supplied tests are appended only to the generation request.

We execute the first Python code block in each response, or the full response if it contains no code block, preserving HumanEval's prompt imports and combining partial function bodies with their supplied headers when needed. A candidate passes if all primary tests finish successfully in isolated Python 3.13.12, within an 8-second CPU soft limit, a 9-second wall-time limit and a 1 GiB memory limit. All 164 HumanEval and 200 MBPP reference solutions pass this evaluator, as do the 90 MBPP+ reference solutions under the MBPP+ tests, and the deliberate failure controls are rejected. HumanEval's benchmark test function is withheld from generation, while MBPP's primary tests are supplied in the prompt, so the MBPP gold measures satisfaction of those examples. The additional MBPP challenge tests of Section F.4 are run the same way, with candidates and reward-model scores held fixed and paired intervals from 20,000 bootstrap replicates that resample tasks jointly across reward models and budgets.