NOT EVERY SUCCESS IS WORTH IMITATING: RETHINKING DATA SELECTION FOR REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Rejection-sampling supervised fine-tuning (RS-SFT) improves a language model by training on its own generated responses that reach the correct answer. Despite its simplicity and low cost, RS-SFT is commonly regarded as a weaker alternative to reinforcement learning (RL) for reasoning. We identify an overlooked limitation in its data selection: **a response that eventually succeeds is not necessarily a desirable trajectory to imitate**. Standard outcome-based selection checks only the final answer and can therefore admit responses that make failed attempts before reaching the correct answer or continue generating meaningless text after already answering. Under iterative self-training, these undesirable behaviors can be learned, regenerated, and selected again; in our experiments, post-answer degeneration among selected responses increases from 5.2% to 34.7%. Motivated by this observation, we propose **Imitation-Aware Trajectory Selection (IMIT)**, a lightweight selection method that retains a correct response only if it contains exactly one final answer and terminates immediately after it. IMIT changes neither the training objective nor the training budget. On Qwen3-1.7B-Base and MATH-500, IMIT improves first-answer pass@1 over standard RS-SFT by 2.12 percentage points in one-round training and by 4.37 points under iterative self-training. Controlled experiments show that these gains arise primarily from which responses are selected, rather than from the reduced number of training problems. Despite its more selective training data, IMIT preserves solution diversity close to that of the base model and reaches 89.0% first-answer pass@32, compared with 86.4% for GRPO, although GRPO achieves substantially higher single-sample accuracy. Our results show that outcome correctness alone is insufficient for selecting self-generated responses for imitation, and suggest that the selection criterion is a critical component of effective self-training.

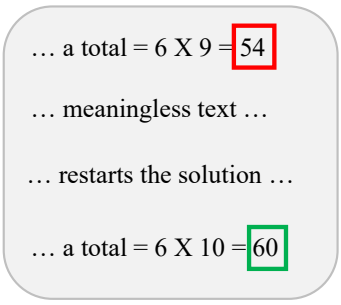
1 INTRODUCTION

Rejection-sampling supervised fine-tuning (RS-SFT) retrains a language model on solutions generated by the model itself (Zelikman et al., 2022; Yuan et al., 2023; Singh et al., 2024). For each training problem, RS-SFT samples multiple candidate solutions, retains those whose final answers are verified as correct, and fine-tunes the model on the retained responses. This simple generate–select-train procedure requires neither human-written reasoning traces nor a stronger teacher model while using self-generated solutions that naturally match the model's own distribution. As a result, RS-SFT can turn solutions that the model finds only occasionally into solutions that it produces more reliably.

A single round of RS-SFT, however, provides supervision only for problems that the current model solves at least once. ReST-style methods therefore iterate generation, selection, and fine-tuning (Gulcehre et al., 2023; Singh et al., 2024). After one round of training, the updated model may solve problems that were previously out of reach, thereby providing additional training data for the next round. Singh et al. (2024) cast this process as an expectation–maximization procedure that alternates between generation with filtering (E-step) and fine-tuning (M-step). This expansion is pronounced in our experiments: among 6,488 training problems, the fraction with at least one correct solution grows from 48.4% in the initial generation pool to 76.1% after four rounds of self-training. **Iteration thus progressively expands the set of problems from which the model can learn.**

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.

(a) A retry inside one sample



First answer grading: Incorrect
Last answer grading: Correct

(b) Coverage with **k** samples



Figure 1: **Outcome-only selection admits retries, and last-answer grading rewards them.** (a) An output of iterative RS-SFT on a MATH-500 problem (reference answer 60): the model first answers 54, generates meaningless text, retries the solution, and eventually answers 60. Last-answer grading counts the response as correct, whereas first-answer grading does not. (b) First-answer pass@k on MATH-500. GRPO achieves higher accuracy at small k, whereas iterative IMIT reaches the highest coverage at $k = 32$.

The selection rule, however, verifies only the outcome: if the final answer of a response matches the reference, the entire response becomes training data. Prior work has largely focused on *what outcome-based selection discards*, for example by improving the coverage or allocation of self-generated training data (Tong et al., 2024; Koh et al., 2025). We instead consider the complementary question: *what undesirable responses does it admit?* Because outcome verification inspects only the final answer, it can accept an *incorrect attempt* $\rightarrow$ *retry* $\rightarrow$ *correct answer* trajectory, as well as a response that continues generating after already producing the correct answer. SFT then learns the entire response, positively reinforcing these undesirable behaviors together with the eventual success. Thus, ***outcome correctness does not necessarily imply suitability for imitation***.

Motivated by this distinction, we propose ***Imitation-Aware Trajectory Selection (IMIT)***, which augments outcome correctness with a lightweight criterion for selecting trajectories for supervised imitation. Specifically, IMIT retains a correct response only if it contains exactly one final answer and terminates immediately after it. IMIT does not attempt to verify the correctness of individual reasoning steps; instead, it prevents SFT from imitating two behaviors that outcome-only selection ignores: repeated answer attempts and continued generation after an answer. Importantly, IMIT changes only the selection rule, leaving the training objective and budget unchanged. This simple constraint becomes increasingly important under iterative self-training, where behaviors admitted in one round are learned and regenerated in the next, creating a feedback loop that can progressively amplify undesirable generation patterns.

Despite being more selective, IMIT preserves solution diversity. Among correct solutions from eight samples per problem, we find that for GRPO, 65% of solution pairs are near-duplicates, with only about 1.6 effectively distinct solutions per problem on average, as measured by the Vendi score (Friedman & Dieng, 2023). For IMIT, solution diversity remains close to that of the base model, with only 5-8% near-duplicate pairs and a Vendi score of 2.2-2.3. Consistent with this preserved diversity, iterative IMIT reaches 89.0% first-answer pass@32, compared with 86.4% for GRPO, despite GRPO's substantially higher single-sample accuracy. This result is consistent with recent findings that RL can improve single-sample accuracy while narrowing the space of generated solutions (Yue et al., 2025). Taken together, our results show that **IMIT improves SFT by being selective about which successful trajectories to imitate, without substantially reducing the diversity of the resulting solution space**.

2 Related Work

Self-training for reasoning. STaR (Zelikman et al., 2022), RFT (Yuan et al., 2023), ReST (Gulcehre et al., 2023), RAFT (Dong et al., 2023), and ReST^{EM} (Singh et al., 2024) train models on their own generated solutions selected according to their outcomes, often over multiple rounds. ReST and ReST^{EM}, in particular, establish iterative generate–select–train frameworks in which updated models generate new training data for subsequent rounds. Recent work has also contrasted SFT with RL, reporting stronger memorization in SFT and better generalization in RL (Chu et al., 2025). We study a different component of self-training: *which successful responses should be selected for imitation*. Rather than modifying the training objective or the iterative self-training procedure, we revisit the outcome-based selection criterion used to construct the training data.

Data selection and allocation. Prior work improves self-training data by allocating more generation to difficult problems (Tong et al., 2024; Koh et al., 2025), filtering responses or tokens according to what the model has learned (Malladi et al., 2026; Jia et al., 2026), studying the effect of data repetition (Muennighoff et al., 2023; Kopiczko et al., 2026), or exploiting incorrect responses (Xueyun et al., 2026; Chen et al., 2025; Li et al., 2025). Much of this work concerns how to obtain more useful supervision from generated data, including responses that outcome-based selection would otherwise discard. We consider the complementary problem: *a response can pass the outcome check and still be undesirable to imitate*. This motivates distinguishing outcome correctness from suitability for supervised imitation.

Limits of outcome verification. A correct final answer does not guarantee correct intermediate reasoning (Lightman et al., 2024), and outcome-based rewards can be optimized in unintended ways (Shao et al., 2025). High-temperature sampling can produce incoherent text (Holtzman et al., 2020), while repeated training on model-generated data can progressively distort the generated distribution (Shumailov et al., 2024). We study how these issues interact in iterative RS-SFT: outcome verification can admit successful responses containing failed attempts followed by retries or continued generation after an answer, and SFT then positively reinforces the entire response. Our focus is therefore different from verifying the correctness of individual reasoning steps. IMIT addresses this problem at the data-selection stage without requiring step-level verification.

Evaluation and diversity of reasoning models. Mathematical reasoning is conventionally evaluated using the final `\boxed{}` answer (Hendrycks et al., 2021), with performance often summarized by pass@k (Chen et al., 2021). Such evaluation can obscure multiple attempts within a single response: an earlier failed attempt can still be counted as correct if the last answer matches the reference. We therefore complement last-answer evaluation with first-answer evaluation to measure whether the model reaches the correct answer on its initial commitment. Whether RL expands or narrows the diversity of reasoning remains debated (Yue et al., 2025; Liu et al., 2025), and UCPO (Lochab et al., 2026) modifies GRPO to preserve diversity among correct responses. Using near-duplicate rates and the Vendi score (Friedman & Dieng, 2023), we find that IMIT preserves solution diversity close to that of the base model despite applying a more selective criterion. Appendix B provides a more detailed discussion.

3 Method

3.1 Problem Setup and Notation

Let $\mathcal{D} = \{(x_i, a_i^\star)\}_{i=1}^N$ denote the set of training problems, where x_i is a problem and $a_i^\star$ is its reference answer. At round r, the generator π_r samples K responses per problem at temperature τ under a fixed few-shot prompt $p(x_i)$:

$$\mathcal{R}_r = \big\{(x_i, y_{i,j}) : y_{i,j} \sim \pi_r(\cdot \mid p(x_i)),\ j = 1, \ldots, K\big\}. \tag{1}$$

We define two functions that extract an answer from a response y: $\mathrm{last}(y)$ returns the content of the *last* `\boxed{}` in y, whereas $\mathrm{first}(y)$ returns the content of the *first* one. A verifier $V(a, a^\star) \in \{0, 1\}$ determines whether an answer a is mathematically equivalent to the reference answer $a^\star$, and $c(y) \in \{0, 1\}$ indicates whether the response terminates on its own rather than reaching the length limit.

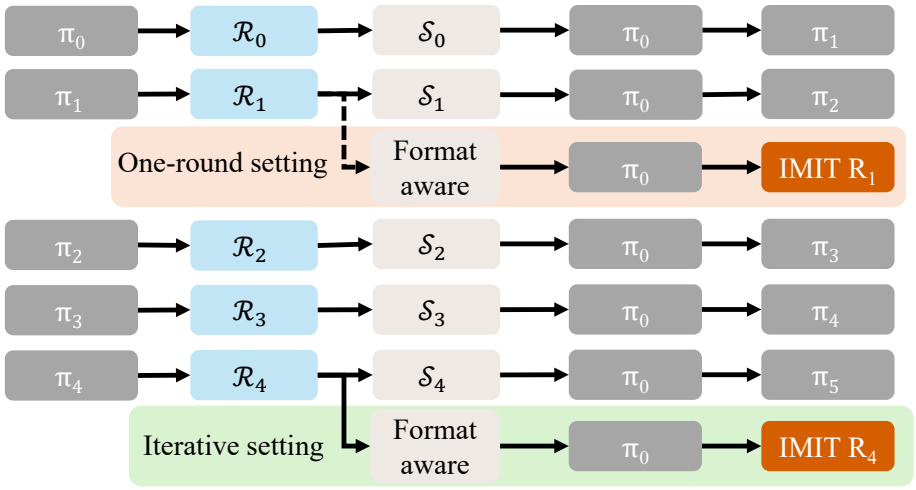


Figure 2: **Experimental pipeline.** Each round samples $K{=}8$ responses per problem from π_r to form the pool $\mathcal{R}_r$, applies outcome-only selection to obtain $\mathcal{S}_r$, and retrains from π_0 to produce π_{r+1}. For controlled comparison, IMIT is applied to the same pools $\mathcal{R}_1$ and $\mathcal{R}_4$ to construct the one-round and iterative variants, respectively. The RS-SFT and IMIT models use the same candidate pools, initialization, training recipe, and budget, so each comparison differs only in the selection rule.

3.2 RS-SFT with Outcome-Only Selection

Standard RS-SFT accepts a response based on its final answer:

$$\mathrm{acc}_{\mathrm{out}}(y) = c(y) \cdot V\big(\mathrm{last}(y), a^\star\big). \tag{2}$$

For each problem, one distinct response is chosen among the accepted responses to form the selected set $\mathcal{S}_r$. The choice is deterministic, so the same candidate set always yields the same response. Problems without an accepted response are excluded from $\mathcal{S}_r$.

The model is trained on the responses in $\mathcal{S}_r$ using the standard cross-entropy objective. We mask the prompt and supervise only the positions $T(y)$ of the response tokens and the terminating token:

$$\mathcal{L}(\theta) = -\frac{1}{Z} \sum_{(x,y)\in\mathcal{B}} \sum_{t\in T(y)} \log \pi_\theta\big(y_t \mid p(x), y_{<t}\big), \qquad Z = \sum_{(x,y)\in\mathcal{B}} |T(y)|, \tag{3}$$

where $\mathcal{B}$ denotes the minibatch of a single update.

Importantly, once a response passes outcome verification, SFT provides positive supervision to the *entire* response. Outcome-only selection, however, inspects only $\mathrm{last}(y)$ and therefore does not distinguish whether the response reached that answer directly or through earlier failed attempts. For example, a response that writes `\boxed{498}`, retries the solution, and ends with `\boxed{492}` becomes a positive training target whenever 492 is the correct answer. The failed attempt and retry are then imitated together with the eventual success.

3.3 Iterative Self-Training

A single round of RS-SFT obtains learning signal only from problems that π_r has already solved at least once. Iterative self-training widens this range by letting the model trained on the selected set generate the pool for the next round. Following ReST$^{\mathrm{EM}}$ (Singh et al., 2024), we retrain from the base model π_0 at every round:

$$\pi_{r+1} = \mathrm{SFT}\big(\pi_0; \mathcal{S}_r\big), \qquad \mathcal{S}_r = \mathrm{Select}\big(\mathcal{R}_r\big). \tag{4}$$

In this loop, behavior admitted by Select is learned by the next model, can reappear in the pool generated by that model, and may be selected again. Behaviors that outcome-only verification fails

to filter can therefore become *self-reinforcing* across rounds. In our iterative lineage ($\tau = 1.2$), the share of selected responses in which the answer is followed by meaningless text grows from 5.2% to 34.7% over $r = 0, \ldots, 4$.

We study two settings: the *one-round setting* uses $\mathcal{R}_1$, generated by π_1 after one round of training, and the *iterative setting* uses $\mathcal{R}_4$, generated by π_4 after four rounds of training. In both settings, models are trained from π_0 on the selected data, and the compared methods differ only in their selection rule.

3.4 Imitation-Aware Trajectory Selection

Outcome correctness indicates whether a response eventually reaches the correct answer, but does not determine whether the entire response is appropriate for imitation. To address this limitation, we propose IMIT (*Imitation-Aware Trajectory Selection*), which augments outcome-based selection with a lightweight trajectory-level condition. Specifically, IMIT excludes two observable behaviors that would otherwise be positively reinforced by SFT: repeated answer attempts and continued generation after an answer.

We define the trajectory condition $\phi(y)$ as

$$\phi(y) = \mathbf{1}\big[n_{\mathrm{box}}(y) = 1 \ \wedge \ \mathrm{suffix}(y) \in \mathcal{C}\big], \tag{5}$$

where $n_{\mathrm{box}}(y)$ denotes the number of `\boxed` markers in y, and $\mathrm{suffix}(y)$ denotes the text following the closing brace of the answer marker. $\mathcal{C}$ is the set of admissible endings, consisting of an optional period, an optional closing math delimiter (`$`, `$$`, `\]`, or `\)`), and another optional period, with arbitrary whitespace in between. Thus, $\phi(y) = 1$ if and only if the response contains exactly one answer and terminates immediately after it.

IMIT combines outcome correctness with the trajectory condition:

$$\mathrm{acc}_{\mathrm{IMIT}}(y) = \underbrace{\mathrm{acc}_{\mathrm{out}}(y)}_{\text{outcome correctness}} \cdot \underbrace{\phi(y)}_{\text{imitation suitability}}. \tag{6}$$

The first term determines whether the response eventually succeeds, whereas the second determines whether its observable trajectory satisfies our minimal criterion for imitation. All subsequent steps—per-problem selection, the training objective, and the training budget—remain identical to standard RS-SFT. Since a response satisfying $\phi(y) = 1$ contains a single answer, $\mathrm{first}(y) = \mathrm{last}(y)$.

Scope of IMIT. IMIT is deliberately a minimal trajectory-level filter rather than a step-level verifier. It removes responses containing multiple answer attempts ($n_{\mathrm{box}} \geq 2$) and responses that continue generating after an answer, but does not verify the correctness of the reasoning preceding that answer. Consequently, incorrect reasoning or pre-answer degeneration may still pass the criterion. According to our degeneration detector, pre-answer degeneration occurs in 3.2% and 20.9% of the responses selected by IMIT from $\mathcal{R}_1$ and $\mathcal{R}_4$, respectively. We deliberately avoid additional models or heuristic detectors to keep the selection rule lightweight.

Algorithm 1 in Appendix A.1 summarizes the complete IMIT selection procedure.

3.5 First-Answer Evaluation

Standard evaluation scores a response by $V(\mathrm{last}(y), a^\star)$. Under this rule, a response that attempts several answers within a single sample counts as correct as long as its last attempt is correct. Thus, pass@1 can reward multiple attempts within a single sample rather than measuring whether the model succeeds on its first commitment.

We therefore additionally report *first-answer* accuracy, which scores the answer that the model commits to first:

$$\mathrm{score}_{\mathrm{first}}(y) = c(y) \cdot V\big(\mathrm{first}(y), a^\star\big). \tag{7}$$

The two rules coincide for responses containing a single answer and differ only when a response writes multiple answers. We compute pass@k from $n = 32$ samples per problem using the standard unbiased estimator (Chen et al., 2021).

First-answer evaluation has its own limitation: when a solution legitimately places an intermediate result in `\boxed{}`, the first-answer rule may count it as incorrect. We therefore treat neither rule

as ground truth and report both. Before evaluation, we pre-specified first-answer accuracy as the primary metric for comparing selection rules.

4 Experimental Setup

Model and data. All models are trained from Qwen3-1.7B-Base (Yang et al., 2025). We use the official MATH training split (Hendrycks et al., 2021), divided into 6,488 training problems, 1,000 development problems, and four few-shot demonstrations (Appendix C). For evaluation, we use all 500 problems in MATH-500. None of the evaluation problems overlaps with the training, development, or demonstration sets after normalizing the problem text. Generation and evaluation use the same 4-shot prompt without a chat template.

Generation pools. We construct the generation pools using the iterative procedure in Section 3.3. At each round, π_r generates $K = 8$ responses per training problem using temperature $\tau = 1.2$, top-p 0.95, and a maximum of 2,048 tokens. The model for the next round is retrained from the base model using the selected responses, with a learning rate of 10^{-5} for 256 updates and 32 responses per update.

We evaluate selection methods on two pools: $\mathcal{R}_1$ for the one-round setting and $\mathcal{R}_4$ for the iterative setting. Each pool contains 51,904 responses. Among them, 15,198 responses in $\mathcal{R}_1$ and 20,770 responses in $\mathcal{R}_4$ have correct final answers. For the iterative setting, all selection methods are compared on the same 4,072 problems having at least one correct, non-degenerate response (Appendix C).

Baselines. Our primary baseline is standard RS-SFT using the outcome-only selection rule in Section 3.2, corresponding to the selection strategy used by STaR, ReST$^{\text{EM}}$, and RFT (Zelikman et al., 2022; Singh et al., 2024; Yuan et al., 2023). In the one-round setting, standard RS-SFT selects 4,213 training problems. In the iterative setting, it is trained from the base model using the selection from $\mathcal{R}_4$, following the ReST$^{\text{EM}}$-style iterative procedure.

IMIT applies the trajectory-selection rule in Section 3.4 to exactly the same generation pools. It selects 2,660 problems in the one-round setting and 2,727 problems in the iterative setting. Thus, the comparison between standard RS-SFT and IMIT uses the same candidate responses and differs only in the selection rule.

We additionally compare with TailSFT (Malladi et al., 2026), which filters responses according to their learning progress, and with the untrained base model. TailSFT is trained using a separate generation pool with a single seed. We also report GRPO (Shao et al., 2024) and UCPO (Lochab et al., 2026) as online RL reference methods. They use the same base model, prompt, and verifier, but their rollout budgets are not matched to those of the SFT methods. We therefore do not treat the RL results as controlled comparisons.

Training. Standard RS-SFT and IMIT differ only in their selection rules. Both are initialized from the same base model and trained with full-parameter updates using the token-averaged cross-entropy objective in Eq. (3). We use AdamW with $\beta = (0.9, 0.999)$, $\epsilon = 10^{-8}$, and no weight decay. The learning rate is 3×10^{-6}, with 31 warmup updates followed by cosine decay, and gradients are clipped to a norm of 1.0. Each update contains 64 responses in the one-round setting and 32 responses in the iterative setting.

Evaluation. For each MATH-500 problem, we generate $n = 32$ responses using temperature 1.0, top-p 0.95, and a maximum of 2,048 tokens. We use a verifier based on `math-verify` and evaluate each response using both the last-answer and first-answer rules defined in Section 3.5. Responses for which the verifier exceeds its time limit are counted as incorrect; this occurs for at most three of the 16,000 responses generated by each model. We report pass@k for $k \in \{1, 8, 32\}$ using the unbiased estimator.

Solution diversity. We measure solution diversity using the first eight responses generated for each problem, each truncated at its first `\boxed{}`, and report the mean pairwise cosine similarity of bge-m3 embeddings (Chen et al., 2024), the Vendi score (Friedman & Dieng, 2023), and the

Table 1: **Evaluation on MATH-500** (32 samples at T=1.0; two seeds averaged except TailSFT, GRPO, and UCPO). All accuracies use first-answer grading, which grades the first `\boxed{}`. Tokens: mean generated tokens per sample. Online RL rows use their own rollout budgets and are references, not matched controls. †Same trainer and budget as IMIT. §Correct-only SFT from the base model on the fifth-round selection.

Method	pass@1	pass@8	pass@32	Tokens
Base (Qwen3-1.7B-Base)	24.05	60.02	79.00	310
RS-SFT†	30.60	65.28	82.50	384
TailSFT	28.71	64.38	81.20	337
Iterative RS-SFT§	35.86	71.14	85.00	656
GRPO	**58.24**	**78.40**	86.40	665
UCPO	47.69	72.03	81.00	1036
IMIT (one round)	32.73	66.77	82.70	**246**
IMIT (iterative)	40.23	76.60	**89.00**	430

near-duplicate rate among correct solutions on the 131 problems for which every method generates at least two correct, non-degenerate solutions (Appendix C.1).

5 Results

5.1 Main Results

Table 1 reports the results on MATH-500. In the one-round setting, IMIT improves first-answer pass@1 from 30.60% to 32.73%, a gain of 2.12 percentage points (pp). The improvement becomes larger in the iterative setting, where first-answer pass@1 increases from 35.86% to 40.23% (+4.37 pp). This improvement is consistent across sampling budgets: first-answer pass@8 and pass@32 increase by 5.46 pp and 4.00 pp.

The larger gain in the iterative setting is consistent with the self-reinforcing behavior described in Section 3.3. As undesirable trajectories become more frequent across rounds, IMIT removes a larger fraction of responses that outcome-only selection would otherwise admit. This difference is also reflected in generation behavior. The average number of `\boxed{}` markers per output decreases from 1.42 for RS-SFT to 1.01 with IMIT in the one-round setting and to 0.99 in the iterative setting. The proportion of degenerate outputs decreases from 5.4% to 0.6% in the one-round setting and from 37.9% to 15.6% in the iterative setting. Mean output length decreases by 36% (384 → 246 tokens) and 34% (656 → 430 tokens), respectively.

5.2 Isolating the Effect of Trajectory Selection

Controlling for the reduced problem set. As discussed in Appendix C, IMIT selects fewer training problems than outcome-only RS-SFT. Under the same training budget, the remaining problems are therefore repeated more often. The improvement could thus arise either from the selected trajectories or simply from the changed problem set.

To separate these effects, we construct RS-SFT controls in which outcome-only selection is restricted to exactly the same problem set used by IMIT. Each control therefore matches IMIT in its problem set, training budget, and number of repetitions, while differing only in which responses are selected for imitation. Under this control, IMIT improves first-answer pass@1 by 2.03 pp in the one-round setting and by 3.86 pp in the iterative setting (Appendix D.3). By comparison, restricting the problem set itself changes pass@1 by only 0.09 pp and 0.51 pp, respectively, neither of which is distinguishable from zero. Thus, the improvement comes primarily from *which trajectories are selected for imitation*, rather than from the reduced set of training problems.

5.3 What Last-Answer Evaluation Hides

For standard RS-SFT, last-answer pass@1 exceeds first-answer pass@1 by 2.1 pp in the one-round setting and by 6.2 pp in the iterative setting (42.02% vs. 35.86% in the latter; Appendix D.1). For IMIT, the gap is only 0.3–0.6 pp, and for GRPO and UCPO it is zero. Thus, a substantial part of the apparent gain of iterative RS-SFT under last-answer evaluation comes from responses that reach the correct answer only after an earlier answer attempt. IMIT largely removes such retries from the trajectories used for imitation and improves first-answer accuracy instead.

The same distinction appears at larger k. In the one-round setting, the last-answer pass@32 of IMIT is 3.10 pp lower than that of RS-SFT, whereas their first-answer pass@32 is nearly identical (+0.20 pp for IMIT).

Among the 16,000 responses generated by iterative RS-SFT, 1,174 (7.3%) have an incorrect first answer but a correct last answer, compared with 524 (3.3%) for one-round RS-SFT. A detailed categorization of these retries is provided in Appendix D.8.

First-answer evaluation can itself misclassify a legitimate intermediate `\boxed{}` expression as an answer attempt. Even under a conservative upper bound on this effect, however, the first-answer pass@1 improvement of IMIT remains at least 3.9 pp in the iterative setting and 1.0 pp in the one-round setting. Appendix D.9 provides the detailed analysis.

5.4 Why the Effect Is Larger under Iterative Self-Training

To understand why IMIT provides a larger improvement under iterative self-training, Table 7 in Appendix D.6 summarizes how the generation pools and selected responses evolve across rounds.

As rounds progress, the fraction of problems with at least one correct response increases from 48.4% to 76.1%, showing that iterative self-training expands its coverage as intended. At the same time, however, the fraction of correct responses satisfying the IMIT trajectory criterion decreases from 36.3% to 26.1%, while the fraction containing two or more answers increases from 14.3% to 24.4%. Because outcome-only selection accepts these responses as long as their final answers are correct, this shift is directly reflected in the selected training data. The fraction of selected responses containing meaningless text after an answer increases from 5.2% to 34.7%, while their mean length increases 3.7-fold, from 788 to 2,879 characters.

Two observations are important. First, these behaviors are already present before iterative training: even in the initial pool generated by the base model, 17.0% of the responses chosen by outcome-only selection contain two or more answers. Iterative self-training therefore amplifies an existing behavior rather than creating it. Second, although the fraction of correct responses satisfying the IMIT criterion decreases across rounds, their absolute number increases from 3,021 to 5,424 because the total number of correct responses grows more rapidly. Iteration thus produces two simultaneous effects: outcome-only selection admits an increasing proportion of trajectories that IMIT would reject, while the number of trajectories satisfying the IMIT criterion also increases. These two effects help explain why IMIT provides a larger gain in the iterative setting.

5.5 Solution Diversity and Coverage

Despite its more selective training data, IMIT does not substantially reduce solution diversity relative to standard RS-SFT. In the iterative setting, IMIT also reaches 89.0% first-answer pass@32, compared with 86.4% for GRPO, although GRPO achieves substantially higher pass@1. Detailed diversity analyses and qualitative examples are provided in Appendix D.7.

6 Conclusion and Limitations

6.1 Conclusion

This paper revisited a basic assumption in rejection-sampling SFT: **a response that eventually succeeds is not necessarily a desirable trajectory to imitate**. Outcome-only selection accepts a response whenever its final answer is correct, even when the response contains failed attempts, retries,

or unnecessary continuation after an answer. Because SFT learns the entire accepted response, these behaviors are positively reinforced and can become self-reinforcing under iterative self-training.

Motivated by this observation, we proposed **Imitation-Aware Trajectory Selection (IMIT)**, which supplements outcome correctness with a lightweight trajectory-level criterion. IMIT retains correct responses containing a single final answer and terminating immediately after it, without modifying the SFT objective or training budget. On MATH-500, IMIT improves first-answer pass@1 over outcome-only RS-SFT by 2.12 pp in the one-round setting and by 4.37 pp in the iterative setting. Controlled experiments show that these gains arise primarily from which trajectories are selected for imitation rather than from the reduced number of training problems.

Our analysis also reveals an evaluation artifact in iterative self-training. Last-answer evaluation rewards responses that reach the correct answer only after an earlier failed attempt, thereby overstating the improvement in first-attempt success. First-answer evaluation substantially reduces this gap. At the same time, despite selecting a considerably smaller subset of correct responses, IMIT preserves solution diversity close to that of the base model. GRPO achieves substantially higher single-sample accuracy, whereas iterative IMIT reaches 89.0% first-answer pass@32 compared with 86.4% for GRPO in our experiments.

Taken together, our results suggest that *outcome correctness and suitability for imitation should be treated as distinct criteria in self-training*. IMIT provides a minimal realization of this principle: rather than changing how SFT learns, it changes which successful trajectories SFT is asked to imitate. We further recommend reporting first-answer evaluation alongside conventional last-answer evaluation when analyzing reasoning self-training.

6.2 Limitations

Scope of experiments. Our experiments use a single model, Qwen3-1.7B-Base, and focus on mathematical reasoning with MATH-500. Whether the observed amplification of undesirable trajectories and the gains from IMIT generalize to larger models, other model families, and other reasoning benchmarks remains to be established.

Scope of IMIT. IMIT is a minimal trajectory-level filter, not a verifier of reasoning correctness. Its current criterion detects repeated answer attempts and continued generation after an answer, but does not verify the reasoning preceding the final answer. Incorrect reasoning or pre-answer degeneration can therefore still be selected. In addition, IMIT excludes problems for which no correct response satisfies its trajectory criterion. Although Section 5.2 shows that the reduced problem set has little effect in our experiments, this loss of coverage may become more important when trajectories satisfying the criterion are rare.

Evaluation rules. First-answer evaluation can underestimate performance when a model legitimately places an intermediate result in `\boxed{}`. Our conservative analysis in Section 5.3 shows that the main conclusions remain under an upper bound on this effect, but neither first- nor last-answer evaluation should be regarded as ground truth. In addition, the improvements reported in our main experiments are based primarily on sampling-based evaluation. Under greedy decoding, neither setting shows a statistically significant difference, and the one-round result points slightly in the opposite direction.

Scope of comparisons and diversity analysis. The rollout budgets of GRPO and UCPO are not matched to those of the SFT methods. The higher pass@32 of iterative IMIT relative to GRPO should therefore be interpreted as an exploratory observation rather than a controlled comparison. In addition, our diversity metrics are based on embedding similarity between generated solutions. They quantify diversity in solution text but do not directly establish diversity in underlying reasoning strategies.

Future work. The distinction between outcome correctness and imitation suitability suggests several extensions. A natural next step is to develop richer trajectory-level criteria that detect undesirable behavior before the final answer while remaining substantially cheaper than step-level verification. The same principle may also inform RL reward design, where outcome quality and trajectory quality can be modeled separately.

AI USE STATEMENT

We used large language model (LLM) assistants in this work. LLMs were used to implement and run parts of the experimental code (data selection, training, and evaluation scripts), to translate and edit drafts written by the authors, to create figures, and to format references. The authors reviewed all LLM-assisted content and take full responsibility for the paper.

REPRODUCIBILITY STATEMENT

All experiments use the public Qwen3-1.7B-Base model and the public MATH and MATH-500 benchmarks. Section 4 lists the generation, training, and evaluation hyperparameters; Appendix C describes the data split, the training budget, the problem set of the iterative setting, and the degeneration detector; Algorithm 1 specifies the full selection procedure.

REFERENCES

Huayu Chen, Kaiwen Zheng, Qinsheng Zhang, Ganqu Cui, Lifan Yuan, Yin Cui, Haotian Ye, Tsung-Yi Lin, Ming-Yu Liu, Jun Zhu, and Haoxiang Wang. NFT: Bridging supervised learning and reinforcement learning in math reasoning. *arXiv preprint arXiv:2505.18116*, 2025. URL https://arxiv.org/abs/2505.18116v3. Version 3, revised March 2026.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. BGE M3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In *Findings of the Association for Computational Linguistics: ACL 2024*, 2024. URL https://arxiv.org/abs/2402.03216.

Mark Chen, Jerry Tworek, Heewoo Jun, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021. URL https://arxiv.org/abs/2107.03374.

Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V. Le, Sergey Levine, and Yi Ma. SFT memorizes, RL generalizes: A comparative study of foundation model post-training. *arXiv preprint arXiv:2501.17161*, 2025. URL https://arxiv.org/abs/2501.17161v2. Version 2.

Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, Kashun Shum, and Tong Zhang. RAFT: Reward ranked finetuning for generative foundation model alignment. *Transactions on Machine Learning Research*, 2023. URL https://arxiv.org/abs/2304.06767v4.

Dan Friedman and Adji Bousso Dieng. The Vendi score: A diversity evaluation metric for machine learning. *Transactions on Machine Learning Research*, 2023. URL https://arxiv.org/abs/2210.02410.

Caglar Gulcehre, Tom Le Paine, Srivatsan Srinivasan, et al. Reinforced self-training (ReST) for language modeling. *arXiv preprint arXiv:2308.08998*, 2023. URL https://arxiv.org/abs/2308.08998v2. Version 2.

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*, volume 34, 2021. URL https://arxiv.org/abs/2103.03874.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=rygGQyrFvH.

Yaning Jia, Chunhui Zhang, Wenxuan Xu, Xingjian Diao, Xiaoyuan Wang, and Soroush Vosoughi. Which tokens should SFT actually learn? a token-trimming perspective on mathematical reasoning. *arXiv preprint arXiv:2609.09707*, 2026. URL https://arxiv.org/abs/2609.09707v1. Version 1.

Reiss Koh, Wonbeen Oh, Jaein Jang, MinHyung Lee, Hyeongjin Kim, Ah Kim, Joonkee Kim, Junghyun Lee, Taehyeon Kim, and Se-Young Yun. AdaSTaR: Adaptive data sampling for training self-taught reasoners. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/840abfadd04c967feaa2a49aba94a32d-Abstract-Conference.html.

Dawid J. Kopiczko, Sagar Vaze, Tijmen Blankevoort, and Yuki M. Asano. Data repetition beats data scaling in long-CoT supervised fine-tuning. *arXiv preprint arXiv:2602.11149*, 2026. URL https://arxiv.org/abs/2602.11149v2. Version 2, revised August 2026.

Mengqi Li, Lei Zhao, Anthony Man-Cho So, Ruoyu Sun, and Xiao Li. A model can help itself: Reward-free self-training for LLM reasoning. *arXiv preprint arXiv:2510.18814*, 2025. URL https://arxiv.org/abs/2510.18814v4. Version 4, revised July 2026.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri 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*, 2024. URL https://openreview.net/forum?id=v8L0pN6EOi.

Mingjie Liu, Shizhe Diao, Ximing Lu, Jian Hu, Xin Dong, Yejin Choi, Jan Kautz, and Yi Dong. ProRL: Prolonged reinforcement learning expands reasoning boundaries in large language models. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/1a22b912945fb7c0bdd079e792b31b6f-Abstract-Conference.html.

Anamika Lochab, Bolian Li, and Ruqi Zhang. Uniform-correct policy optimization: Breaking RLVR's indifference to diversity. *arXiv preprint arXiv:2605.00365*, 2026. URL https://arxiv.org/abs/2605.00365v1. Version 1.

Sadhika Malladi, Samy Jelassi, Dylan Foster, Jordan T. Ash, and Akshay Krishnamurthy. TailSFT: Filtered fine-tuning improves post-training performance. *arXiv preprint arXiv:2608.25756*, 2026. URL https://arxiv.org/abs/2608.25756v1. Version 1.

Niklas Muennighoff, Alexander M. Rush, Boaz Barak, Teven Le Scao, Aleksandra Piktus, Nouamane Tazi, Sampo Pyysalo, Thomas Wolf, and Colin Raffel. Scaling data-constrained language models. *arXiv preprint arXiv:2305.16264*, 2023. URL https://arxiv.org/abs/2305.16264v5. Version 5, revised June 2025.

Rulin Shao, Shuyue Stella Li, Rui Xin, et al. Spurious rewards: Rethinking training signals in RLVR. *arXiv preprint arXiv:2506.10947*, 2025. URL https://arxiv.org/abs/2506.10947v2. Version 2, revised February 2026.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. URL https://arxiv.org/abs/2402.03300.

Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson, and Yarin Gal. AI models collapse when trained on recursively generated data. *Nature*, 631(8022):755–759, 2024. URL https://doi.org/10.1038/s41586-024-07566-y.

Avi Singh, John D. Co-Reyes, Rishabh Agarwal, et al. Beyond human data: Scaling self-training for problem-solving with language models. *Transactions on Machine Learning Research*, 2024. URL https://arxiv.org/abs/2312.06585.

Yuxuan Tong, Xiwen Zhang, Rui Wang, Ruidong Wu, and Junxian He. DART-Math: Difficulty-aware rejection tuning for mathematical problem-solving. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/0ef1afa0daa888d695dcd5e9513bafa3-Abstract-Conference.html.

Tian Xueyun, MingHua Ma, Bingbing Xu, Nuoyan Lyu, Wei Li, Heng Dong, Zheng Chu, Yuanzhuo Wang, and Huawei Shen. Learning from mistakes: Negative reasoning samples enhance out-of-domain generalization. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 29689–29711. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.1370. URL https://aclanthology.org/2026.acl-long.1370/.

An Yang, Anfeng Li, Baosong Yang, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

Zheng Yuan, Hongyi Yuan, Chengpeng Li, Guanting Dong, Keming Lu, Chuanqi Tan, Chang Zhou, and Jingren Zhou. Scaling relationship on learning mathematical reasoning with large language models. *arXiv preprint arXiv:2308.01825*, 2023. URL https://arxiv.org/abs/2308.01825v2. Version 2.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Yang Yue, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in LLMs beyond the base model? In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://arxiv.org/abs/2504.13837v5.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. STaR: Bootstrapping reasoning with reasoning. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://arxiv.org/abs/2203.14465.

A Additional Method Details

A.1 IMIT Selection Algorithm

Algorithm 1 summarizes the complete selection procedure used by IMIT. Given a response pool $\mathcal{R}_r$ generated by π_r, IMIT first applies the same outcome verification as standard RS-SFT and then applies the trajectory condition $\phi(y)$ defined in Section 3.4. For each problem, one response is selected from the remaining candidates using the same deterministic selection rule as the RS-SFT baseline. Thus, IMIT changes only which successful trajectories are admitted for supervised imitation.

Algorithm 1 IMIT: Imitation-Aware Trajectory Selection

Require: Response pool $\mathcal{R}_r$ and reference answers $\{a_i^\star\}_{i=1}^N$
Ensure: Selected training set $\mathcal{S}_r^{\mathrm{IMIT}}$

$\mathcal{S}_r^{\mathrm{IMIT}} \leftarrow \emptyset$
for each problem x_i **do**
  $\mathcal{A}_i \leftarrow \emptyset$
  for each response $y \in \mathcal{R}_r(x_i)$ **do**
    $o(y) \leftarrow c(y)\, V\big(\mathrm{last}(y), a_i^\star\big)$
    $\phi(y) \leftarrow \mathbf{1}\big[n_{\mathrm{box}}(y) = 1 \wedge \mathrm{suffix}(y) \in \mathcal{C}\big]$
    if $o(y)\phi(y) = 1$ **then**
      $\mathcal{A}_i \leftarrow \mathcal{A}_i \cup \{y\}$
    end if
  end for
  if $\mathcal{A}_i \neq \emptyset$ **then**
    Select one $y_i \in \mathcal{A}_i$ using the fixed deterministic rule
    $\mathcal{S}_r^{\mathrm{IMIT}} \leftarrow \mathcal{S}_r^{\mathrm{IMIT}} \cup \{(x_i, y_i)\}$
  end if
end for
return $\mathcal{S}_r^{\mathrm{IMIT}}$

B Extended Related Work

Learning to reason from self-generated data. Training on self-generated solutions that reach the correct answer is a widely used approach for improving reasoning. STaR (Zelikman et al., 2022) repeatedly trains a model on rationales that lead to correct answers. RFT (Yuan et al., 2023) uses rejection-sampling fine-tuning to collect multiple distinct correct solutions for each problem. ReST (Gulcehre et al., 2023) and RAFT (Dong et al., 2023) generalize this generate–filter–train paradigm by selecting generated outputs according to a reward. ReST$^{\mathrm{EM}}$ (Singh et al., 2024) formulates iterative self-training as an expectation–maximization procedure that alternates between generation with selection and training, and applies this procedure to mathematical reasoning and code generation while retraining from the base model at each round. Recent work comparing SFT with RL further reports that SFT tends to fit or memorize training examples more strongly, whereas RL can exhibit better generalization (Chu et al., 2025).

Our work is complementary to these approaches. We do not modify the self-training loop or the SFT objective; instead, we revisit the criterion used to decide which successful responses become supervised targets. Specifically, we distinguish whether a response eventually reaches the correct outcome from whether its entire trajectory is suitable for imitation.

Data selection and allocation. Prior work has studied both which self-generated data should be used and how generation should be allocated. DART-Math (Tong et al., 2024) observes that correctness filtering biases the resulting training data toward easier problems and allocates more generation to difficult ones. AdaSTaR (Koh et al., 2025) adapts problem-level sampling according to training history and difficulty. TailSFT (Malladi et al., 2026) selects responses according to learning progress, while TrimSFT (Jia et al., 2026) performs selection at the token level. Other studies examine

the role of repeated data under data-constrained training (Muennighoff et al., 2023) and in long chain-of-thought SFT (Kopiczko et al., 2026).

Another line of work extracts learning signal from responses that would otherwise be rejected. GLOW (Xueyun et al., 2026) uses incorrect reasoning during training to improve out-of-distribution generalization, NFT (Chen et al., 2025) treats incorrect responses as an implicit negative policy to connect supervised learning with policy optimization, and SePT (Li et al., 2025) alternates between generation and training without external rewards. These approaches largely improve how supervision is obtained or allocated from generated data, including responses that outcome-based selection would otherwise discard. We study the complementary case: *a response can pass the outcome check and still be undesirable to imitate*. IMIT therefore operates within the set of successful responses and further selects them according to a lightweight trajectory-level criterion.

Limits of outcome verification. A correct final answer does not guarantee that the intermediate reasoning is correct. Process-supervision methods therefore evaluate reasoning at the step level rather than relying only on the final outcome (Lightman et al., 2024). Outcome-based rewards can also be optimized in unintended ways, and unexpected improvements have been observed even under incorrect or random rewards (Shao et al., 2025). Separately, degeneration in generated text has been studied extensively: high-temperature sampling can produce incoherent text (Holtzman et al., 2020), while repeated training on model-generated data can progressively distort the generated distribution (Shumailov et al., 2024).

Our setting differs from step-level verification. We do not attempt to determine whether every reasoning step is correct. Instead, we identify a specific failure mode of outcome-only selection in iterative RS-SFT: responses containing failed answer attempts, retries, or continued generation after an answer can pass outcome verification and are then imitated in their entirety. These behaviors can reappear in subsequent generation pools and be selected again. IMIT addresses this failure mode at the data-selection stage without requiring an additional process verifier or changing the training objective.

Evaluation and diversity of reasoning models. Mathematical reasoning is commonly evaluated by extracting the final `\boxed{}` answer (Hendrycks et al., 2021), while performance over multiple samples is often summarized by pass@k (Chen et al., 2021). Recent work has also examined how RL affects the diversity and coverage of reasoning. Some studies report higher accuracy at small sampling budgets but narrower coverage at large k after RL (Yue et al., 2025), whereas others find that sufficiently prolonged RL can expand the set of discovered reasoning behaviors (Liu et al., 2025). UCPO (Lochab et al., 2026) observes that GRPO does not explicitly reward diversity among equally correct responses and modifies the objective to encourage diversity. The Vendi score (Friedman & Dieng, 2023) provides a complementary measure by estimating the effective number of distinct samples from their pairwise similarities.

Our analysis contributes two observations. First, last-answer evaluation can reward within-sample retries, allowing a response with an incorrect earlier answer to be counted as correct when its final answer matches the reference. We therefore additionally report first-answer evaluation. Second, despite selecting a substantially smaller subset of successful training trajectories, IMIT preserves solution diversity close to that of standard RS-SFT and the base model. The comparison with RL is exploratory, since the rollout budgets are not matched, but it provides additional evidence that improved trajectory selection need not collapse the diversity of successful solutions.

Relation to RL. GRPO (Shao et al., 2024) learns from relative rewards over on-policy responses and can assign both positive and negative advantages within a group. RS-SFT instead discards rejected responses and applies positive token-level supervision to the entire accepted response. Thus, once a response passes outcome verification, any failed attempt, retry, or unnecessary continuation within it is also reinforced. IMIT addresses this difference at the data-selection stage, without modifying the SFT objective.

C Experimental Details

Data split. We use the official MATH training split (Hendrycks et al., 2021), which contains 7,500 problems. We exclude four problems whose reference answers cannot be extracted, two duplicates within the training split, and two problems that overlap with the test split. The remaining 7,492 problems are divided into 6,488 training problems, 1,000 development problems, and four few-shot demonstrations.

Effect on the selected training set. In $\mathcal{R}_1$, 20.3% of the 4,213 responses chosen by standard selection contain two or more answers, whereas none selected by IMIT do; the mean response length decreases from 1,060 to 559 characters. Overall, 34.1% of the correct responses in $\mathcal{R}_1$ (5,181 of 15,198) and 26.1% in $\mathcal{R}_4$ (5,424 of 20,770) satisfy the IMIT criterion. Consequently, fewer problems remain for training: in $\mathcal{R}_1$, 2,660 of the 4,213 problems covered by standard selection (63.1%) remain. Under a fixed training budget, these problems are therefore repeated more often. Section 5.2 controls for this effect by comparing IMIT with standard selection restricted to the same problem set.

Training budget. All SFT models, including the RS-SFT baselines and the matched-problem controls in Appendix D.3, are trained for 309 updates, that is, on 19,776 responses in the one-round setting and 9,888 responses in the iterative setting. Each selected problem is therefore seen about 4.7 times by RS-SFT and 7.4 times by IMIT in the one-round setting, and about 2.4 and 3.6 times in the iterative setting.

Problem set of the iterative setting. In $\mathcal{R}_4$, 4,937 problems have at least one correct response, and 4,072 of them have at least one correct response that the degeneration detector below does not flag. All methods in the iterative setting are trained on problems from these 4,072: Iterative RS-SFT applies outcome-only selection to all of them, and IMIT selects the 2,727 problems that also have a correct response satisfying ϕ.

Degeneration detector. We flag a response as degenerate when some window of 40 consecutive tokens contains more than 60% tokens outside a reference vocabulary. Tokens are English words of at least three letters or single non-Latin characters, and the vocabulary consists of the words that appear at least twice in the MATH training problems and the few-shot demonstrations. The detector flags none of the 12,500 human-written solutions in the MATH training and test splits. Across windows of 30–50 tokens and thresholds of 0.5–0.7, the share of degenerate responses in the fifth-round selection ranges from 45.8% to 49.8% (48.1% at the default setting), and at most 6 of the 12,500 human-written solutions are flagged.

C.1 Diversity Metrics

We measure solution diversity using the first eight responses generated for each problem. Each response is truncated at its first `\boxed{}` so that retries or text generated after an answer do not artificially increase the measured diversity. We embed the truncated responses using bge-m3 (Chen et al., 2024) and report three metrics: mean pairwise cosine similarity, the Vendi score (Friedman & Dieng, 2023), and the near-duplicate rate. The Vendi score estimates the effective number of distinct solutions among the eight responses, while the near-duplicate rate is the fraction of response pairs with cosine similarity of at least 0.95.

Our primary diversity analysis considers correct solutions, since diversity among incorrect responses does not necessarily indicate a broader space of successful reasoning strategies. To ensure a common comparison set across models, we evaluate all methods on the 131 problems for which every method generates at least two correct, non-degenerate solutions.

C.2 Prompt

All generation and evaluation use the same raw 4-shot prompt without a chat template, shown below. The prompt consists of an instruction, four demonstrations from the MATH training split, and the target problem, separated by blank lines; `{problem}` is replaced with the problem text, and generation stops at the string `"\n\nProblem:"`. The four demonstrations are excluded from

the training problems and cover four different subjects (number theory, algebra, geometry, and precalculus). Each demonstration solution contains exactly one `\boxed{}` and ends immediately after it, so all four satisfy $\phi(y) = 1$. Multiple answers and text after the answer therefore do not originate from the prompt.

```
Solve the following mathematics problems. Show your reasoning and put the final answer in
    \boxed{} .

Problem: Express .$\overline{28}$ as a common fraction.

Solution: If $x=.\overline{28}$, then $100x=28.\overline{28}$. Notice that we can eliminate
    the repeating decimal by subtracting $.\overline{28}$ from $28.\overline{28}$. We have
    $100x-x=99x=28$, so $x=\frac{28}{99}$. The repeating decimal can be expressed as the
    fraction $\boxed{\frac{28}{99}}$.

Problem: Suppose that $\alpha$ is inversely proportional to $\beta$.  If $\alpha = -3$ when
    $\beta = -6$, find $\alpha$ when $\beta = 8$.  Express your answer as a fraction.

Solution: Since $\alpha$ is inversely proportional to $\beta$, by definition $\alpha\beta =
    k$ for some constant $k$.  Plugging in, we see that $(-3)\cdot (-6) = k$, so $k = 18$.
     So when $\beta = 8$, we have that $8\alpha = 18$, or $\alpha = \boxed{\frac{9}{4}}$.

Problem: A $5 \times 8$ rectangle can be rolled to form two different cylinders with
    different maximum volumes. What is the ratio of the larger volume to the smaller
    volume? Express your answer as a common fraction.

Solution: Holding the rectangle vertically, we can form a cylinder with height 8 and base
    circumference of 5.  Let this cylinder have volume $V_A$ and radius $r_A$; we have
    $2\pi r_A = 5$ so $r_A = \frac{5}{2\pi}$ and $V_A = \pi r_A ^2 h = \pi
    \left(\frac{5}{2\pi}\right)^2 (8) = \frac{50}{\pi}$.

Holding the rectangle horizontally, we can form a cylinder with height 5 and base
    circumference of 8.  Similarly, let this cylinder have volume $V_B$ and radius $r_B$;
    we have $2\pi r_B = 8$ so $r_B = \frac{4}{\pi}$ and $V_B = \pi r_B^2 h = \pi
    \left(\frac{4}{\pi}\right)^2 (5) = \frac{80}{\pi}$.

Hence, the ratio of the larger volume to the smaller volume is
    $\frac{80/\pi}{50/\pi}=\boxed{\frac{8}{5}}$.

Problem: Compute $\cos \left( \arcsin \frac{2}{3} \right).$

Solution: Consider a right triangle where the opposite side is 2 and the hypotenuse is 3.

[asy]
unitsize (1 cm);

draw((0,0)--(sqrt(5),0)--(sqrt(5),2)--cycle);

label("$\sqrt{5}$", (sqrt(5)/2,0), S);
label("$3$", (sqrt(5)/2,1), NW);
label("$2$", (sqrt(5),1), E);
label("$\theta$", (0.7,0.3));
[/asy]

Then $\sin \theta = \frac{2}{3},$ so $\theta = \arcsin \frac{2}{3}.$  By Pythagoras, the
    adjacent side is $\sqrt{5},$ so $\cos \theta = \boxed{\frac{\sqrt{5}}{3}}.$

Problem: {problem}

Solution:
```

D Additional Experimental Results and Analysis

D.1 Last-Answer and First-Answer Accuracy

Table 2: **Last-answer and first-answer accuracy on MATH-500.** Same models and evaluation as Table 1. *Last answer* grades the last `\boxed{}` (the standard rule); *first answer* grades the first `\boxed{}`.

	Last answer		First answer	
Method	pass@1	pass@32	pass@1	pass@32
Base (Qwen3-1.7B-Base)	24.55	80.00	24.05	79.00
RS-SFT	32.66	86.10	30.60	82.50
TailSFT	30.65	84.00	28.71	81.20
Iterative RS-SFT	42.02	89.00	35.86	85.00
GRPO	58.24	86.40	58.24	86.40
UCPO	47.69	81.00	47.69	81.00
IMIT (one round)	33.05	83.00	32.73	82.70
IMIT (iterative)	40.86	89.40	40.23	89.00

D.2 Statistical Tests

Table 3: **Paired differences on MATH-500** (IMIT minus the RS-SFT row trained on the same pool; two seeds averaged; 95% CIs from a paired question bootstrap with 10,000 resamples). *Pre-specified primary metric; both primary differences remain significant after Holm correction over the two settings. Other rows are secondary and uncorrected.

Rule	Metric	One round	Iterative
First answer	pass@1*	$+2.12$ $[+1.50, +2.76]$	$+4.37$ $[+3.63, +5.13]$
	pass@8	$+1.50$ $[+0.41, +2.61]$	$+5.46$ $[+4.17, +6.81]$
	pass@32	$+0.20$ $[-1.90, +2.30]$	$+4.00$ $[+2.10, +6.00]$
Last answer	pass@1	$+0.39$ $[-0.26, +1.05]$	-1.15 $[-1.86, -0.44]$
	pass@8	-2.42 $[-3.51, -1.33]$	-0.97 $[-2.01, +0.05]$
	pass@32	-3.10 $[-5.20, -1.10]$	$+0.40$ $[-1.30, +2.10]$

D.3 Matched-Problem Controls

Table 4: **Matched-problem controls on MATH-500.** *Same problems* applies outcome-only selection to exactly the problem set selected by IMIT, with the same training budget and number of repetitions. Two seeds averaged. Paired question bootstrap (95% CI), first-answer pass@1: IMIT − same-problem control $+2.03$ $[+1.44, +2.64]$ (one round) and $+3.86$ $[+3.17, +4.55]$ (iterative); same-problem control − RS-SFT $+0.09$ $[-0.44, +0.62]$ and $+0.51$ $[-0.05, +1.06]$. Last-answer pass@1, IMIT − same-problem control: $+0.89$ $[+0.30, +1.48]$ and -0.50 $[-1.15, +0.14]$.

		First answer				Last answer	
Method	Problems	pass@1	pass@8	pass@32	L5 pass@1	pass@1	Tokens
One-round setting ($\mathcal{R}_1$)							
RS-SFT	4,213	30.60	65.28	82.50	8.96	32.66	384
RS-SFT, same problems	2,660	30.69	65.36	82.50	9.03	32.16	345
IMIT (one round)	2,660	32.73	66.77	82.70	9.57	33.05	246
Iterative setting ($\mathcal{R}_4$)							
Iterative RS-SFT	4,072	35.86	71.14	85.00	11.94	42.02	656
Iterative RS-SFT, same problems	2,727	36.37	72.29	86.20	12.30	41.37	654
IMIT (iterative)	2,727	40.23	76.60	89.00	15.79	40.86	430

On the matched problem set, the one-round improvement also appears under last-answer evaluation (+0.89 pp). This suggests that the slight decrease in the standard score caused by restricting the problem set partly explains the near tie under last-answer evaluation discussed in Section 5.3.

D.4 Accuracy by Difficulty Level

Table 5: **First-answer** pass@1 **on MATH-500 by difficulty level.** Numbers of problems per level in parentheses. These splits were not pre-specified.

Method	L1 (43)	L2 (90)	L3 (105)	L4 (128)	L5 (134)
Base (Qwen3-1.7B-Base)	51.49	43.21	26.93	16.94	6.89
RS-SFT	61.41	52.90	35.37	23.33	8.96
TailSFT	57.27	49.48	33.45	21.92	8.37
Iterative RS-SFT	66.02	59.77	42.90	28.19	11.94
GRPO	87.43	80.31	73.48	54.47	25.70
UCPO	81.69	68.33	55.33	44.87	19.64
IMIT (one round)	63.12	56.27	38.33	25.61	9.57
IMIT (iterative)	70.82	63.06	47.11	33.86	15.79

D.5 Output Behavior

Table 6: **Output behavior on MATH-500** (32 samples per problem; two seeds averaged). Boxes: mean number of `\boxed` markers per output. $\geq$2 boxes and ϕ=1: share of outputs with two or more answers and share satisfying the condition in Eq. (5). Degenerate: share flagged by the degeneration detector (Appendix C).

Method	Tokens	Boxes	$\geq$2 boxes (%)	ϕ=1 (%)	Degenerate (%)
RS-SFT	384	1.42	23.6	30.5	5.35
RS-SFT, same problems	345	1.28	17.1	43.0	3.67
IMIT (one round)	246	1.01	1.7	86.6	0.64
Iterative RS-SFT	656	1.42	33.5	19.7	37.94
Iterative RS-SFT, same problems	654	1.36	29.6	26.5	38.55
IMIT (iterative)	430	0.99	2.9	80.8	15.64

D.6 Round-by-Round Statistics

Table 7: Round-by-round statistics of the generation pools and of the data chosen by standard selection. Degenerate: responses flagged by the degeneration detector anywhere; Degen. after answer: flagged only after the first `\boxed{}`.

		Correct responses		Selected data			
Pool	Problems solved	Format-compliant	$\geq$2 answers	Degenerate	Degen. after answer	$\geq$2 answers	Mean length (chars)
$\mathcal{R}_0$ (base)	48.4%	36.3%	14.3%	5.9%	5.2%	17.0%	788
$\mathcal{R}_1$	64.9%	34.1%	16.8%	11.3%	9.4%	20.3%	1,060
$\mathcal{R}_2$	69.9%	30.0%	18.9%	22.8%	17.8%	22.7%	1,534
$\mathcal{R}_3$	72.9%	27.8%	22.7%	35.9%	27.4%	26.4%	2,206
$\mathcal{R}_4$	76.1%	26.1%	24.4%	48.1%	34.7%	27.6%	2,879

D.7 Solution Diversity and Coverage

Table 8: **Diversity of the first eight samples per MATH-500 question** (T=1.0, top-p 0.95). Each sample is truncated at its first `\boxed{}`, so text after the answer is ignored. Similarity is the cosine between bge-m3 embeddings (Chen et al., 2024); Vendi is the effective number of distinct samples among eight (1 = all identical, 8 = all orthogonal; Friedman & Dieng, 2023). *Correct solutions* compares samples that are first-answer correct and non-degenerate, on the 131 questions where every model produces at least two such samples; near-duplicate is the share of pairs with cosine ≥ 0.95. All eight includes incorrect samples, so distinct answers also reflect errors. Two seeds are averaged except TailSFT, GRPO, and UCPO. Paired question bootstrap (95%): IMIT (iterative) − GRPO correct-solution cosine -0.103 near-duplicate -57.4 pp; IMIT − its matched SFT baseline cosine $+0.004$(one round) and -0.001 (iterative).

			All eight samples			Correct solutions		
Method	pass@1	pass@32	Vendi ↑	Cosine ↓	Answers	Cosine ↓	Near-dup. (%) ↓	4-gram Jacc. ↓
Base (Qwen3-1.7B-Base)	24.05	79.00	2.40	0.767	5.80	0.841	4.2	0.033
RS-SFT	30.60	82.50	2.36	0.773	5.43	0.842	4.7	0.033
TailSFT	28.71	81.20	2.37	0.772	5.64	0.846	4.4	0.036
Iterative RS-SFT	35.86	85.00	2.31	0.779	4.87	0.854	7.8	0.050
GRPO	58.24	86.40	1.55	0.894	2.04	0.957	65.3	0.302
UCPO	47.69	81.00	1.77	0.851	1.58	0.956	68.4	0.257
IMIT (one round)	32.73	82.70	2.28	0.784	5.34	0.846	4.8	0.035
IMIT (iterative)	40.23	89.00	2.23	0.791	4.74	0.853	7.9	0.051

Table 8 reports solution diversity among eight samples per problem. The RL models exhibit substantially higher concentration among their correct solutions. For GRPO, 65.3% of correct-solution pairs are near-duplicates, with a Vendi score of 1.55; UCPO shows a similar pattern, with 68.4% near-duplicates and a Vendi score of 1.77. In contrast, the near-duplicate rate and Vendi score are 4.2% and 2.40 for the base model, 4.7% and 2.36 for RS-SFT, and 4.8–7.9% and 2.23–2.28 for IMIT. In the iterative setting, IMIT has a 0.103 lower mean cosine similarity among correct solutions and a 57.4 pp lower near-duplicate rate than GRPO. Importantly, solution diversity remains similar between IMIT and standard RS-SFT. Thus, despite selecting a substantially smaller subset of training trajectories, IMIT does not substantially reduce the observed diversity of successful solutions.

The difference in solution diversity is accompanied by a crossover in pass@k. In the iterative setting, IMIT reaches 89.0% first-answer pass@32, compared with 86.4% for GRPO. GRPO, however, achieves substantially higher pass@1 (58.24% vs. 40.23%) and slightly higher pass@8 (78.40% vs. 76.60%). This comparison is exploratory: it was not pre-specified, GRPO was evaluated with a single seed, and the RL and SFT rollout budgets are not matched. Nevertheless, the observed pattern is consistent with recent evidence that RL can improve single-sample accuracy while concentrating the distribution over successful reasoning trajectories (Yue et al., 2025).

Qualitative example. Consider a problem asking for the distance between $(2, -6)$ and $(-4, 3)$, for which both GRPO and IMIT answer all eight samples correctly. Nearly all GRPO responses begin with the same sentence, "To find the distance between the points $(2, -6)$ and $(-4, 3)$, we use the distance formula," and proceed through essentially the same sequence of steps. In contrast, the IMIT responses exhibit greater variation: one first states the distance formula, another proceeds directly to the computation, and another assigns the coordinates to variables before computing the distance. For this problem, the mean pairwise cosine similarity among correct solutions is 0.983 for GRPO and 0.887 for IMIT; all GRPO pairs are near-duplicates, compared with 7% for IMIT.

D.8 Analysis of Within-Sample Retries

Among the 16,000 responses generated by iterative RS-SFT on MATH-500, 1,174 (7.3%) have an incorrect first answer but a correct last answer. These responses account for the majority of the gap between first- and last-answer evaluation. Based on the text between the two answers, we divide them into three categories:

- 30% contain meaningless text between the two answers;
- 63% develop a new solution of at least 400 characters without meaningless text; and
- 7% contain at most 400 characters between the two answers.

Figure 1(a) shows a representative example of the first category. On a handshake problem, the model first gives the incorrect answer $\binom{10}{2} = 45$, generates a sequence of words unrelated to the problem, and then re-derives the solution from scratch to obtain the correct answer, 60. Last-answer evaluation counts this response as correct, as if the model had reached the correct answer on its first attempt.

Such responses are less frequent for one-round RS-SFT: 524 responses (3.3%) have an incorrect first answer and a correct last answer, and only 5% of these retries contain meaningless text. In contrast, this fraction rises to 30% for iterative RS-SFT. This pattern is consistent with the round-wise analysis in Section 5.4, indicating that iterative self-training particularly amplifies retries involving degenerate generation.

D.9 Limitations of First-Answer Evaluation

First-answer evaluation can underestimate performance when a `\boxed{}` expression denotes a legitimate intermediate result rather than an answer attempt. For example, one response states "$\boxed{5-\sqrt{3}}$ is the radical conjugate of $5+\sqrt{3}$, so their product is $\boxed{22}$." The first boxed expression is an intermediate object in the solution, but first-answer evaluation nevertheless counts the response as incorrect.

To estimate the largest possible effect of such cases, we consider all responses in which the span between the first and last answers contains at most 400 characters. Under the conservative assumption that every such response contains a legitimate intermediate answer, first-answer evaluation underestimates RS-SFT by at most 0.5 pp in the iterative setting (84 of 16,000 responses) and 1.1 pp in the one-round setting (168 of 16,000 responses).

IMIT, which generates approximately one answer marker per response, is minimally affected by this ambiguity. Even under this conservative upper bound, its first-answer pass@1 improvement over RS-SFT remains at least approximately 3.9 pp in the iterative setting and 1.0 pp in the one-round setting. Thus, the main comparison between outcome-only selection and IMIT does not depend on treating every first `\boxed{}` expression as an actual answer attempt.