ADVANCEMENT OR ACCELERATION? HOW DATA CURATION IMPROVES LLM TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

While data curation is widely credited with improving large language model training, a basic question remains unresolved: does curation expand what a model can ultimately learn, or merely reach the same solution faster? In this paper, we formalize this distinction between *advancement* (improving the attainable performance) and *acceleration* (reaching the same performance with less data). Through an analytic characterization and real-world LLM experiments, we show that advancement requires a correctable train–target gap. When the baseline training and target objectives coincide, no curation policy improves the endpoint, although curation may still accelerate training. This reframes advancement as a property of the curation task rather than the curation method. Under a common gap-centric view of data mixture and selection, we operationalize this condition as the Δ-test, an observable discrepancy between the uncurated baseline's training and target losses that is computable prior to expensive curation. Across extensive experiments, an appreciably nonzero Δ predicts persistent gains, whereas gains near $\Delta = 0$ tend to diminish as the training budget grows. Beyond individual methods, our findings recenter data curation on the task itself and point to target-set construction as a promising direction for shaping what curation can achieve.

1 INTRODUCTION

In recent years, as the cost-effectiveness of pure scaling declines, how well we use the data is drawing increasing attention. As a primary approach to this, data curation takes two main forms: *data mixture*, which balances the proportions of domain-specific data in training sets to ensure intended capabilities (Xie et al., 2023; Fan et al., 2024; Chen et al., 2025; Liu et al., 2025), and *data selection*, which chooses informative training examples that best improve performance (Li et al., 2024; Liu et al., 2024; Lin et al., 2024; Yu et al., 2024b). Remarkable gains have been reported, and pruning data has even been argued to *beat the power-law scaling* (Sorscher et al., 2022). These results suggest that data curation is a promising direction for improving models, or so it would seem.

Nevertheless, when compared carefully, the advantage of data curation sometimes becomes surprisingly fragile. Chen et al. (2025) shows that optimized mixtures do not always outperform the uniform allocation, while elaborate selection rules at scale may barely exceed, and at times underperform, random sampling (Xia et al., 2024). These observations appear contradictory because performance at a fixed training budget conflates two effects. Curation may improve the best solution the model can attain, or it may merely reduce the data needed to reach the same solution. We call the former *advancement* and the latter *acceleration*, leading to our central question: *Does data curation improve the attainable optimum, or only reach the same performance faster?*

We take a gap-centric account of this question, and demonstrate that a property of the task governs advancement: the *train–target gap* between the distribution induced by the uncurated baseline and the target we evaluate on. Data curation can advance a model only when such a gap exists and can be corrected. Without one, no allocation improves on the attainable performance, though curation may still accelerate learning. To understand this mechanism, we derive an analytic characterization in a simplified but instructive setting: perceptron learning in the teacher–student framework, where the performance of the converged model on the target under any allocation is determined by a closed system of self-consistent equations.

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.

Our theoretical analysis reveals that a single principle governs both data mixture and selection: curation yields true advancement only when it corrects a genuine train–target mismatch. Without this underlying mismatch, techniques that prioritize informative examples or rebalance domains merely accelerate learning, with their advantages vanishing as sample budget grows. Because the existence of advancement is ultimately a property of the *task* rather than the curation algorithm, it can be diagnosed in advance. We introduce the Δ-test, a discrepancy between target and baseline-training losses, as a lightweight diagnostic of advancement potential. Crucially, it is computable from a single baseline run before committing any compute to data curation.

We validate this theoretical account on real LLM training across data mixture and selection settings. The empirical results confirm our predictions: persistent gains at the largest tested budgets emerge only when a correctable gap is present. Conversely, in regimes where $\Delta \approx 0$, the benefits of curation tend to diminish with additional training, reflecting mere acceleration. This cleanly explains recent contradictory findings—such as the surprising competitiveness of uniform mixtures (Chen et al., 2025) and random selection (Xia et al., 2024)—as their empirical settings happen to lie in near-zero-gap regimes. Our findings therefore suggest that advancement potential should be diagnosed before committing to increasingly complex allocation methods. More broadly, they point to target design as an important but underexplored complement to curation for future research since it shapes the train–target gap that curation can correct, dictating the improvement it can ultimately aim for.

Our contributions are threefold:

- We distinguish two fundamental effects of data curation—*advancement*, which improves the attainable performance, and *acceleration*, which reaches the same performance faster. We show that advancement requires a correctable train–target gap.
- We propose the Δ-test, a lightweight diagnostic of advancement potential. Crucially, this indicator can be computed from a single baseline run, before committing to expensive curation.
- We validate our account on LLM training across data mixture and selection, confirming the theoretical predictions and unifying previously conflicting empirical findings under a single gap-dependent picture.

2 PRELIMINARY

Curation as Data Allocation Data curation improves models by reallocating the training budget across candidate data. We unify data mixture and selection as a single allocation operation $\boldsymbol{w} \in \mathcal{W}$. Consider a training corpus organized into K domains, where domain k provides a candidate training set $\mathcal{D}^k_{\text{train}}$ and a target set $\mathcal{D}^k_{\text{tar}}$. Training under an allocation $\boldsymbol{w}$ yields a model $\boldsymbol{\theta}^\star(\boldsymbol{w}) = \arg\min_{\boldsymbol{\theta}} \mathcal{L}_{\text{train}}(\boldsymbol{\theta}; \boldsymbol{w})$. The two paradigms are of different granularities:

- *Data mixture* ($\boldsymbol{w} \in \mathcal{W} := \{\boldsymbol{w} \in \mathbb{R}^K \mid \sum_{k=1}^K w_k = 1, w_k > 0\}$): domain k contributes a fraction w_k of the sampled training data.
- *Data selection* ($K = 1$, $\boldsymbol{w} \in \mathcal{W} := \left\{\boldsymbol{w} \in \{0,1\}^P \mid \sum_{p=1}^P w_p = fP\right\}$): A per-example keep decision retaining a subset of size fP from P total examples.

We assess $\boldsymbol{\theta}^\star(\boldsymbol{w})$ by its performance on the target, which is the distribution we care about at test time. For mixture the target is typically a domain composition $\widehat{\boldsymbol{w}}$ over the target sets $\{\mathcal{D}^k_{\text{tar}}\}$. Unlike the allocation $\boldsymbol{w}$, $\widehat{\boldsymbol{w}}$ is fixed by the task itself, reflecting how much each domain matters at test time, rather than chosen by curation. For selection, the target is simply the held-out distribution, with no domain-level composition to specify.

What Should Curation Beat? To evaluate a curation choice, we must compare it against an *uncurated baseline* $\boldsymbol{w}_{\text{baseline}}$ that performs no re-allocation. For mixture this is typically uniform allocation $\boldsymbol{w}_{\text{baseline}} = \frac{1}{K}\mathbf{1}$. For selection, it is the full retention $f = 1$ and $\boldsymbol{w}_{\text{baseline}} = \mathbf{1}$. Despite a fast-growing arsenal of curation methods (DoGE (Fan et al., 2024), TANDEM (Wang et al., 2025), RHO-1 (Lin et al., 2024), BLADE (Wang et al., 2026), among others), these simple baselines remain stubbornly hard to beat in some cases.

3 AN ANALYTIC THEORY OF DATA CURATION

To understand why these baselines are so hard to beat, we recognize that a curation choice can have two different effects: *advancement*, yielding a genuinely better attainable model, or *acceleration*, reaching the same performance faster. Distinguishing them requires knowing how the learned model behaves under a certain allocation, and how this behavior changes as the sample budget grows. We make this precise with an analytic characterization based on statistical-mechanics, and show that advancement requires a correctable train–target gap. Otherwise, curation can only accelerate.

3.1 THE TEACHER–STUDENT CURATION MODEL

We model data curation using the perceptron in a teacher–student setting (Engel & den Broeck, 2022). In the high-dimensional limit $N, P \to \infty$ with a data budget $\alpha = P/N$, the student $\boldsymbol{J}^\star(\boldsymbol{w}) = \arg\min_{\boldsymbol{J}} \mathcal{L}_{\text{train}}(\boldsymbol{J}; \boldsymbol{w})$ attains a target generalization performance $\mathcal{E}_\alpha(\boldsymbol{w})$ [1]. Tracking $\mathcal{E}_\alpha(\boldsymbol{w})$ across α allows us to separate advancement from acceleration. For any $\boldsymbol{w}$, $\mathcal{E}_\alpha(\boldsymbol{w})$ is characterized analytically via the replica method (Loureiro et al., 2021; Sorscher et al., 2022), yielding self-consistent equations that we solve numerically (derivations deferred to Appendix A.1 and A.2).

Specifically, Domain k provides training data $\mathcal{D}^k_{\text{train}} = \{(\boldsymbol{x}^k_i, y^k_i)\}_{i=1}^{P_k}$, where $\boldsymbol{x}^k_i \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_k)$ and labels $y^k_i = \text{sign}(\boldsymbol{B}_k^\top \boldsymbol{x}^k_i)$ are generated by a domain teacher $\boldsymbol{B}_k \in \mathbb{R}^N$. We instantiate the two curation paradigms as follows:

- *Data mixture:* Without loss of generality, we consider two domains $K = 2$ and let the policy $\boldsymbol{w} = (w_1, w_2)$ allocate the budget P by sampling $P_k = w_k P$ examples from domain k. The target performance is the weighted loss $\mathcal{E}(\boldsymbol{J}) = \sum_{k=1}^2 \widehat{w}_k \, \mathcal{L}^k_{\text{tar}}(\boldsymbol{J})$ under a fixed domain composition $\widehat{\boldsymbol{w}}$. The train–target gap is directly governed by the mismatch between $\boldsymbol{w}_{\text{baseline}}$ and $\widehat{\boldsymbol{w}}$.
- *Data selection:* This setting considers $K = 1$, we evaluate generalization along a distinct *target teacher* $\boldsymbol{B}_{\text{tar}}$ at an angle ϕ from the training teacher $\boldsymbol{B}$. This angle ϕ defines the train–target gap. From a candidate pool, a target-aware selection rule retains a fraction f of the samples based on a margin criterion along $\boldsymbol{B}_{\text{tar}}$, while preferentially filtering out examples whose training labels conflict with $\boldsymbol{B}_{\text{tar}}$. This rule serves as a controlled gap-correcting policy: testing it at $\phi = 0$ and $\phi > 0$ isolates the benefit of gap correction from mere target awareness. The selected data yields a student with target error $\mathcal{E}(\boldsymbol{J}) = \arccos(\rho)/\pi$, where $\rho = \frac{\boldsymbol{J}^\top \boldsymbol{B}_{\text{tar}}}{\|\boldsymbol{J}\| \|\boldsymbol{B}_{\text{tar}}\|}$ is the student–target overlap.

3.2 ADVANCEMENT AND ACCELERATION

$\mathcal{E}_\alpha(\boldsymbol{w})$ traces target performance across data budgets α, revealing two distinct effects of curation: changing the *endpoint*, i.e., the best attainable performance, or accelerating the *approach*, i.e., reducing the budget required to reach a given performance level. We refer to these effects as *advancement* and *acceleration*, respectively.

Definition 1 (Advancement and acceleration)**.** *Let $\mathcal{E}_\infty(\boldsymbol{w}) = \lim_{\alpha\to\infty} \mathcal{E}_\alpha(\boldsymbol{w})$ denote the limiting target performance. An allocation provides* advancement *if it improves the endpoint, $\mathcal{E}_\infty(\boldsymbol{w}) < \mathcal{E}_\infty(\boldsymbol{w}_{\textit{baseline}})$. For a performance level ε attainable by both allocations, let $\alpha_{\boldsymbol{w}}(\varepsilon) = \inf\{\alpha : \mathcal{E}_\alpha(\boldsymbol{w}) \le \varepsilon\}$ be the budget at which $\boldsymbol{w}$ first reaches ε. The allocation provides* acceleration *at ε if it reaches this level with fewer examples, $\alpha_{\boldsymbol{w}}(\varepsilon) < \alpha_{\boldsymbol{w}_{\textit{baseline}}}(\varepsilon)$.*

These effects are not mutually exclusive, but they differ in persistence: advancement persists as the budget grows, whereas acceleration yields transient gains that eventually vanish. The analytical characterization of these population endpoints are given in Appendix A.1 and Appendix A.2 for mixture and selection. respectively.

When is advancement possible? The distinction above raises a structural question: when can curation improve the attainable target performance, rather than merely reduce the data budget needed to reach a given performance level? A necessary condition is a *correctable train–target gap*—a

[1] Following the statistical-mechanics convention $\boldsymbol{J}$ denotes the student parameters and corresponds to the general model parameter vector $\boldsymbol{\theta}$ used above.

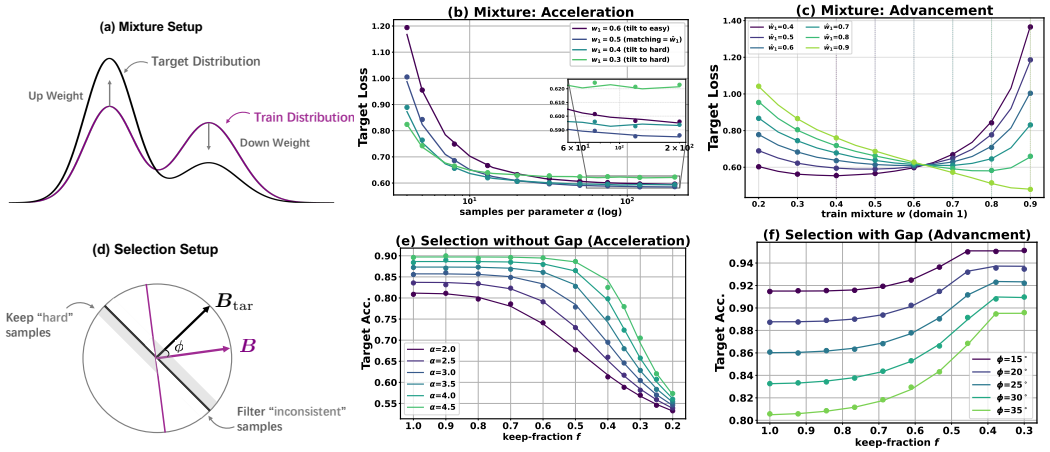


Figure 1: An analytic account of data curation. Theory (solid lines) closely matches $N = 200$ perceptron simulations (circles). *Top row—mixture.* (a) *The setup.* Two-domain training and target mixtures with compositions $\boldsymbol{w}$ and $\widehat{\boldsymbol{w}}$. (b) *Acceleration.* Tilting the training mixture toward the harder domain ($w_1 < \widehat{w}_1 = 0.5$) lowers target loss at small budgets, but the gain vanishes as α grows. (c) *Advancement.* At a large budget ($\alpha = 100$), each target loss vs. mixture curve is minimized by target matching, $\boldsymbol{w}^\star = \widehat{\boldsymbol{w}}$. Appendix A.1 shows that this holds as $\alpha \to \infty$. *Bottom row—selection.* (d) *The setup.* A selection policy favors examples near the target boundary and filters those whose labels conflict with the target. (e) *Acceleration.* At zero gap ($\phi = 0$), moderate selection preserves accuracy, but never exceeds full retention ($f = 1$). Aggressive selection degrades performance. (f) *Advancement.* With $\phi > 0$, target accuracy initially rises as f decreases: improved train–target alignment outweighs the reduction in sample size. Reaches an interior optimum $f^\star < 1$, and the gain increases with ϕ. Appendix A.2 shows that an interior selection policy remains better than full retention as $\alpha \to \infty$.

mismatch between the training distribution induced by the baseline, $\mathcal{P}_{\text{train}}(\boldsymbol{w}_{\text{baseline}})$, and the target distribution $\mathcal{P}_{\text{target}}$ that some feasible policy can reduce. Without such a mismatch, curation has nothing to correct and can at most accelerate. A correctable gap instead provides optimization headroom for advancement, although it does not guarantee improvement. Section 3.3 formalizes this condition and instantiates it for data mixture and data selection.

3.3 Results

We now instantiate the gap criterion for data mixture and selection. Figure 1 shows close agreement between the analytic predictions and finite-dimensional simulations.

Mixture: Acceleration without advancement. Under a uniform target $\widehat{\boldsymbol{w}} = \boldsymbol{w}_{\text{baseline}} = (0.5, 0.5)$, there is no train–target gap. We introduce heterogeneous domain difficulty by giving domain 2 nuisance variance orthogonal to its teacher $\boldsymbol{B}_2$, thereby lowering its signal-to-noise ratio without changing its labeling rule (Appendix A.1). Allocating more data to this harder domain ($w_1 < 0.5$) lowers the target loss at small budgets (Figure 1(b)), but the gain vanishes as α grows, and the uniform baseline attains the best endpoint. Domain difficulty improves finite-budget performance without changing the attainable endpoint: it provides acceleration, not advancement.

Mixture: Advancement by matching the target. Unlike the finite-budget effect of domain difficulty, the infinite-data endpoint is determined by alignment between the training and target mixtures. At the large budget shown in Figure 1(c), each curve is minimized at $w_1 = \widehat{w}_1$, despite heterogeneous domain difficulty. Appendix A.1 establishes this result analytically for the large-budget limit ($\alpha \to \infty$): when the training and target distributions span the same domains, target matching is globally optimal, $\boldsymbol{w}^\star = \widehat{\boldsymbol{w}}$. Consequently, if $\widehat{\boldsymbol{w}} = \boldsymbol{w}_{\text{baseline}}$, the baseline already matches the target and no mixture can provide advancement. If $\widehat{\boldsymbol{w}} \neq \boldsymbol{w}_{\text{baseline}}$, matching corrects the distribution gap and yields $\mathcal{E}_\infty(\widehat{\boldsymbol{w}}) < \mathcal{E}_\infty(\boldsymbol{w}_{\text{baseline}})$, providing genuine advancement.

Selection: A gap creates optimization headroom. The same selection policy behaves differently depending on the train–target gap. With $\phi = 0$, moderate pruning preserves accuracy, but no policy exceeds full retention over the displayed budgets (Figure 1(e)). With $\phi > 0$, target accuracy initially increases as the retained fraction f decreases (Figure 1(f)). Although fewer examples are used, the selection preferentially removes target-inconsistent examples while retaining informative examples near the target boundary. The resulting gain in train–target alignment outweighs the loss in sample size. Beyond the interior optimum $f^\star$, the reduction in the training budget dominates, causing accuracy to plateau (in the large-budget limit, $\alpha \to \infty$) or decline (at finite budgets). Appendix A.2 derives these finite-budget curves and shows that the improvement persists in the large-budget limit.

We now show that the impossibility of advancement without a gap is not an artifact of our perceptron model, but a fundamental algebraic consequence of a bi-level structure. Curation seeks an allocation $\boldsymbol{w}$ to minimize the target loss (outer problem) subject to training convergence (inner problem).

Proposition 1 (Informally)**.** *Consider the bi-level curation problem* $\min_{\boldsymbol{w}} \mathcal{L}_{\mathrm{tar}}(\boldsymbol{\theta}^\star(\boldsymbol{w}))$*, where* $\boldsymbol{\theta}^\star(\boldsymbol{w}) \in \arg\min_{\boldsymbol{\theta}} \mathcal{L}_{\mathrm{train}}(\boldsymbol{\theta}; \boldsymbol{w})$*. Suppose there is no gap, such that the population training objective (expected losses) under the uncurated baseline* $\boldsymbol{w}_{\mathrm{baseline}}$ *coincides with the population target objective. Then* $\boldsymbol{w}_{\mathrm{baseline}}$ *is a global minimizer of the outer objective: no allocation attains a lower target endpoint, and curation can at most accelerate.*

The intuition is immediate: without a gap, the target objective is exactly the training objective that the baseline already globally minimizes. We defer the formal proof to Appendix A.3, and next translate this distribution-level condition into an observable diagnostic.

4 The Δ-Test

The correctable train–target gap is a distribution-level condition, making it hard to verify directly. To bridge this, we introduce the Δ-test: an empirical signature of the gap evaluated from a single uncurated baseline run, $\boldsymbol{\theta}(\boldsymbol{w}_{\text{baseline}})$, before any curation is attempted. The test compares the baseline model's in-sample training loss with its target loss (Appendix A.4). Because data mixture and selection operate at different granularities, we define two variants. *For data mixture* over K domains, the comparison is domain-wise, allowing the diagnostic to capture domains that are over- or under-served by the baseline. Using the baseline mixture weights w_k and target weights $\widehat{w}_k$, we define

$$\Delta = \sum_{k=1}^{K} \left| \widehat{w}_k \mathcal{L}^k_{\mathrm{tar}}\big(\boldsymbol{\theta}(\boldsymbol{w}_{\mathrm{baseline}})\big) - w_k \mathcal{L}^k_{\mathrm{train}}\big(\boldsymbol{\theta}(\boldsymbol{w}_{\mathrm{baseline}})\big) \right|. \tag{1}$$

The target weights ensure that the diagnostic reflects the distribution on which performance is ultimately evaluated. *For data selection* ($K = 1$), the training and target losses are scalars:

$$\Delta = \left| \mathcal{L}_{\mathrm{tar}}\big(\boldsymbol{\theta}(\boldsymbol{w}_{\mathrm{baseline}})\big) - \mathcal{L}_{\mathrm{train}}\big(\boldsymbol{\theta}(\boldsymbol{w}_{\mathrm{baseline}})\big) \right|, \tag{2}$$

where $\boldsymbol{w}_{\mathrm{baseline}}$ denotes full retention ($f = 1$).

Designed as a lightweight operational diagnostic rather than a statistical distance, Δ indicates the *potential* for advancement. An appreciably nonzero value flags an observable train–target mismatch and hence possible advancement, provided that a feasible curation policy can correct the mismatch. Across both the mixture and selection models, the gain over the baseline increases with Δ (Figure 4). Conversely, $\Delta \approx 0$ suggests that curation will only accelerate training, although scalar losses may fail to reveal some distributional gaps. In practice, Δ can be estimated early in training using a small proxy model—after the gap has become visible but well before full convergence.

5 Experiments on Real Data

Our theory establishes a correctable train–target gap as a prerequisite for curation to advance a model. We test this in practical LLM training across data mixture and selection: does a nonzero Δ accompany persistent gains, while $\Delta \approx 0$ suggests limited headroom for consistent improvement over the baseline policy? To ensure a controlled comparison, we evaluate all baselines under a shared training and evaluation protocol and will release all code and configurations. We run each experiment five times independently. Due to limited space, the main text reports means and shows standard-deviation bands where space permits. Complete results are provided in Appendix A.6. Across all settings, the diagnostic Δ is evaluated at exactly 50% of the training budget.

Table 1: Perplexity in the no-gap ($\Delta = 0.07$) mixture setting on SlimPajama. "Average" represents the exponential of the average loss across all domains as in (Fan et al., 2024).

Methods	Arxiv	Book	C4	CommonCrawl	Github	Stackexchange	Wikipedia	Average
Uniform	10.02	39.61	49.67	42.70	5.16	12.25	20.00	$19.71_{\pm 0.17}$
RegMix	10.53	41.90	48.67	41.65	5.75	11.73	19.86	$19.80_{\pm 0.12}$
DoGE	10.96	35.49	42.42	36.81	6.70	13.97	23.92	$20.17_{\pm 0.16}$
Skill-It	10.64	38.32	47.99	40.62	5.73	12.07	20.78	$19.68_{\pm 0.16}$
Aioli	10.09	39.86	49.81	42.83	5.66	12.36	19.67	$19.77_{\pm 0.15}$
TANDEM	10.18	37.89	46.77	40.41	5.89	12.61	20.96	$19.66_{\pm 0.15}$

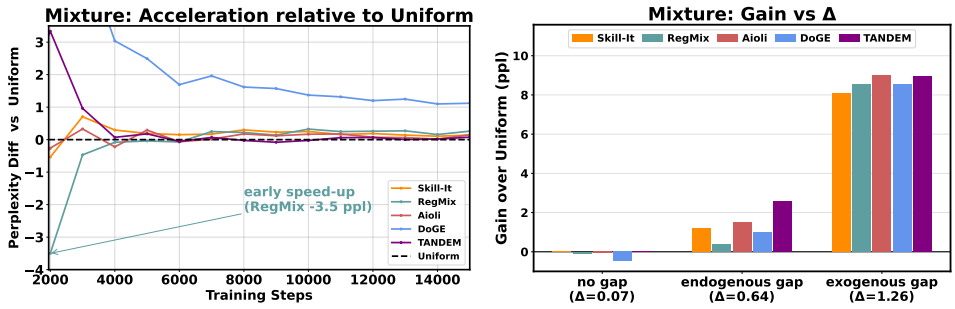


Figure 2: *Left:* With a negligible gap ($\Delta = 0.07$), the perplexity difference relative to Uniform (method − Uniform, lower is better) can be negative early but approaches zero with training. Bands show the standard deviation over five runs. *Right:* Final gain over Uniform (higher is better) is negligible in the no-gap setting and becomes substantial under endogenous and exogenous gaps.

5.1 MIXTURE

We evaluate three scenarios: a matched no-gap setting, an exogenous distribution shift, and an endogenous gap induced by limited domain data. Following Chen et al. (2025), we train 160M GPT-style LMs (Biderman et al., 2023) on a 6B sampled version of SlimPajama (Soboleva et al., 2023) across all seven domains. We compare the Uniform baseline against five representative mixture optimizers: RegMix (Liu et al., 2025), DoGE (Fan et al., 2024), Skill-It (Chen et al., 2023), Aioli (Chen et al., 2025), and TANDEM (Wang et al., 2025)–spanning mixing-law based, influence-based, and bi-level approaches. Appendix A.5 describes each method and our complete experimental protocol.

When $\Delta \approx 0$, optimized mixtures accelerate but do not advance. We first consider a setting in which the reference training distribution can match the uniformly weighted target. Although SlimPajama-6B is domain-imbalanced, in our training setting, Uniform allocation is feasible without repeatedly sampling a depleted domain, yielding a negligible train–target gap ($\Delta = 0.07$). From Table 1 we see that the final perplexities of all optimized mixtures range from 19.66 to 20.17, within experimental variation of the Uniform baseline at 19.71. Reallocating training exposure across domains does not produce a consistently better result. At intermediate checkpoints, RegMix reduces perplexity by 3.5, but this advantage vanishes as the training proceeds (Figure 2, left). This no-gap setting exhibits acceleration rather than advancement, supporting our theoretical prediction.

When $\Delta > 0$ under distribution shift, optimized mixtures can advance. An exogenous gap arises when the candidate pool and target follow different distributions. We instantiate this setting by holding out CommonCrawl as the target while training on the remaining six domains, yielding $\Delta \approx 1.26$. Although CommonCrawl is absent from the training pool, the related C4 domain provides a feasible source for partially correcting the gap. As shown in Table 2, every optimized mixture substantially outperforms Uniform, reducing target perplexity from 49.19 to 40.2–41.1. By shifting training exposure toward domains better aligned with the held-out target, curation retains a large gain through the largest feasible training budget, with no sign of the advantage shrinking. In contrast to the no-gap gains that vanish with training, this persistence provides empirical evidence of advancement rather than transient acceleration.

Table 2: Final target perplexity under endogenous ($\Delta = 0.64$) and exogenous ($\Delta = 1.26$) mixture gaps. The endogenous block reports all target domains and their "Average". The exogenous block reports the perplexity of the held-out CommonCrawl target (CC).

	Endogenous Gap								Exogenous Gap
Methods	Arxiv	Book	C4	CC.	Github	Stackexchange	Wikipedia	Average	CC.
Uniform	13.14	57.44	65.57	57.93	8.21	19.59	33.90	28.60	49.19
RegMix	13.33	54.12	63.74	56.63	8.23	20.03	33.34	28.24	40.63
DoGE	12.95	54.82	62.84	55.52	8.13	18.61	32.72	27.62	40.67
Skill-It	14.54	44.24	48.99	42.10	8.33	21.10	49.46	27.38	41.08
Aioli	12.80	51.01	61.13	53.51	8.14	18.97	32.63	27.11	40.17
TANDEM	12.60	49.76	54.19	47.24	8.35	17.98	33.51	26.02	40.24

When $\Delta > 0$ under limited data, optimized mixtures can advance. Repeated sampling of data-scarce domains produces an endogenous train–target gap: overfitting raises their target losses relative to their training losses, thereby increasing Δ. We instantiate this setting by restricting the SlimPajama pool to 300M tokens while preserving its original domain distribution. Under the fixed training budget, small domains such as ArXiv and Wikipedia are exhausted early and repeatedly sampled, yielding $\Delta = 0.64$. As shown in Table 2, all optimized mixtures outperform Uniform at the largest tested budget by reducing exposure to these overfit domains. Together with the exogenous-gap result, this provides further empirical evidence that a correctable gap can enable advancement regardless of whether it arises from distribution shift or limited data (right panel of Figure 2).

The Δ-test remains predictive across model scales. We repeat all three mixture settings at 1.1B parameters and compare TANDEM with Uniform (Table 3). At 160M and 1.1B, respectively, the no-gap gains remain negligible ($+0.05$ and $+0.11$ perplexity), whereas the exogenous-gap gains remain substantial ($+8.95$ and $+5.31$), as do the endogenous-gap gains ($+2.58$ and $+2.29$). Thus, the diagnosis obtained at 160M transfers to 1.1B, supporting the use of a compute-efficient proxy model to screen for advancement potential before full-scale training (Section 4).

Table 3: Scale study comparing Uniform and TANDEM. Gain is Uniform − TANDEM.

Scenario	Model	Uniform	TANDEM	Gain
No gap	160M	19.71	19.66	$+0.05$
$\Delta = 0.07$	1.1B	17.16	17.05	$+0.11$
Endogenous gap	160M	28.60	26.02	$+2.58$
$\Delta = 0.64$	1.1B	28.14	25.85	$+2.29$
Exogenous gap	160M	49.19	40.24	$+8.95$
$\Delta = 1.26$	1.1B	40.03	34.72	$+5.31$

5.2 SELECTION

To test whether persistent selection gains require a correctable mismatch, we use a paired design with no-gap and exogenous-gap scenarios. The two scenarios share the same candidate pool, model, selection methods, and training protocol. Only the target distribution used for selection and evaluation changes. This design isolates the role of the gap rather than merely comparing selection algorithms. We use a 5B-token OpenWebMath candidate pool (Paster et al., 2024) and TinyLlama-1.1B (Zhang et al., 2024)[2]. We report held-out target loss and accuracy on eight benchmarks spanning arithmetic, competition mathematics, word problems, table-based reasoning, and STEM knowledge. Baselines include full-pool continuous pretraining (Base-CT), random selection, three target-agnostic methods (EL2N (Paul et al., 2021), InfoBatch (Qin et al., 2024), and Token-Cleaning (Pang et al., 2025)), and two target-aware methods (RHO-1 (Lin et al., 2024) and BLADE (Wang et al., 2026)). Appendix A.5 provides the complete benchmark list, method descriptions, and experimental protocol.

When $\Delta \approx 0$, selection provides no consistent advancement. Using an additional 0.3B-token sample from OpenWebMath as the target yields a negligible train–target gap ($\Delta = 0.016$). With

[2]We choose TinyLlama over a heavily optimized SOTA model to prevent saturated capabilities from masking the intrinsic efficacy of data selection.

Table 4: Data-selection results for an in-distribution OpenWebMath target (no gap, $\Delta = 0.016$, upper) and a shifted MetaMath–Mammoth target (with gap, $\Delta = 0.549$, lower). We report accuracy on eight math benchmarks, their average, and held-out target loss. All methods operate on the same 5B-token candidate pool with a retained fraction of $f = 0.6$. Target-aware RHO-1 and BLADE improve substantially only when a gap is present.

	Retained Toks.	GSM8K	MATH	SVAMP	ASDiv	MAWPS	TAB	MQA	MMLU-STEM	Avg.	Target Loss
Selection without Gap											
Base	-	2.9	3.2	11.0	18.1	20.4	12.5	14.6	16.1	12.4	1.97
Base-CT	5B	4.3	4.3	17.1	29.4	39.4	14.5	8.3	20.4	17.21	1.70
Random	3B	4.0	3.9	15.8	29.1	37.2	13.8	9.4	17.8	16.38	1.73
EL2N	3B	3.0	3.2	15.5	25.0	31.2	14.4	8.9	18.3	14.94	1.76
InfoBatch	3B	3.8	3.3	15.8	25.8	32.9	12.5	7.1	16.5	14.71	1.76
Token Cleaning	3B	4.2	3.0	15.4	24.4	31.7	11.7	10.2	14.1	14.30	1.79
RHO-1	3B	3.2	3.1	15.3	24.6	31.4	14.5	7.1	16.7	14.49	1.78
BLADE	3B	4.5	3.7	18.1	29.9	37.9	16.5	10.4	19.3	17.54	1.73
Selection with Gap											
Base	-	2.9	3.2	11.0	18.1	20.4	12.5	14.6	16.1	12.40	1.26
Base-CT	5B	4.3	4.3	17.1	29.4	39.4	14.5	8.3	20.4	17.21	1.14
Random	3B	4.0	3.9	15.8	29.1	37.2	13.8	9.4	17.8	16.38	1.13
EL2N	3B	3.0	3.2	15.5	25.0	31.2	14.4	8.9	18.3	14.94	1.19
InfoBatch	3B	3.8	3.3	15.8	25.8	32.9	12.5	7.1	16.5	14.71	1.16
Token Cleaning	3B	4.2	3.0	15.4	24.4	31.7	11.7	10.2	14.1	14.30	1.22
RHO-1	3B	15.1	5.9	32.6	46.9	63.8	18.7	15.1	18.4	27.06	0.59
BLADE	3B	21.2	8.1	37.0	51.4	70.2	21.2	20.6	20.8	31.31	0.45

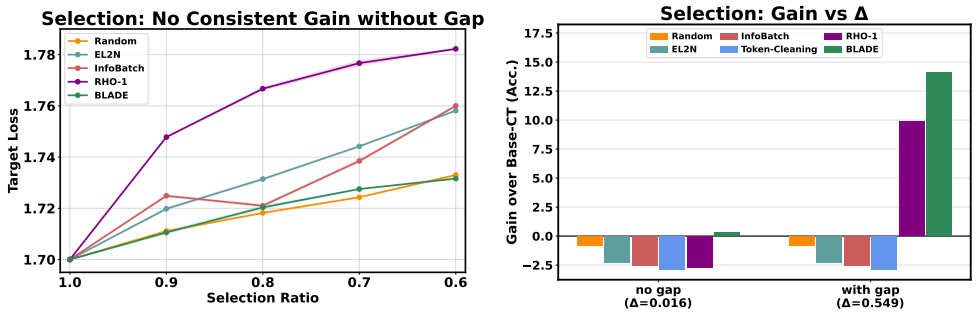


Figure 3: *Left:* Held-out target loss across retained fractions $f \in [0.6, 1.0]$ for an in-distribution OpenWebMath target ($\Delta = 0.016$). Target loss directly evaluates the distribution used to define Δ. No selector consistently improves over Base-CT and Random, and more aggressive pruning generally increases loss. *Right:* Average accuracy change relative to Base-CT in the no-gap and shifted-target settings. The comparison shows little optimization headroom in the no-gap setting, whereas substantial positive gains become possible after the target shift ($\Delta = 0.549$).

little discrepancy to correct, neither target-aware nor target-agnostic methods consistently outperform the Base-CT (17.21) and Random (16.38) baselines: their average scores range from 14.3 to 17.5, and their held-out target losses show no consistent improvement (Table 4, upper). Notably, RHO-1 and BLADE receive target data but still fail to provide consistent gains, showing that target access alone does not guarantee advancement. Figure 3 (Left) further examines the effect of the retained fraction. We report held-out target loss rather than downstream accuracy because it is measured on the target distribution used to define Δ, and thus more directly exposes the train–target gap. Varying f from 1.0 to 0.6 yields the same conclusion: full retention attains the lowest loss, no selection method consistently improves over Random, and target loss generally rises as selection becomes more aggressive. This retention sweep is consistent with the limited optimization headroom indicated by the negligible gap.

When $\Delta > 0$, selection can advance. When the target departs from the candidate pool, an exogenous gap emerges. Following Lin et al. (2024), we use a blend of MetaMath (Yu et al., 2024a) and Mammoth (Yue et al., 2024) as the target while retaining OpenWebMath as the candidate pool,[3] yielding a substantial mismatch ($\Delta = 0.549$). As shown in Table 4 (bottom), target-aware selection produces large gains: RHO-1 and BLADE increase the Base-CT average score from 17.21 to 27.06 and 31.31, respectively, whereas the target-agnostic methods EL2N, InfoBatch, and Token-Cleaning do not. Held-out target losses show the same pattern. Together with the no-gap control, this contrast provides empirical evidence that target-aware selection can achieve advancement by correcting a genuine train–target gap.

6 Related Work

Data mixture optimization. Beyond hand-tuned compositions, mixture methods learn domain weights via mixing laws (Liu et al., 2025; Chen et al., 2023; 2025) or bi-level optimization (Fan et al., 2024; Wang et al., 2025). Empirical comparisons have also found Uniform to remain a competitive baseline in several settings (Chen et al., 2025). Our analysis offers one explanation for this pattern: when the baseline training distribution is aligned with the target, Uniform is endpoint-optimal, a finite-budget advantage reflects acceleration.

Data selection and quality scores. Data selection methods generally fall into *target-agnostic* and *target-aware* paradigms. *Target-agnostic* scores rank examples by intrinsic difficulty or quality signals (Paul et al., 2021; Wang et al., 2023; Qin et al., 2024; Pang et al., 2025). Nevertheless, they have a mixed record, with reports that random selection remains highly competitive in the data-rich regime (Xia et al., 2024). Through our lens, such difficulty proxies only accelerate learning when no train–target gap exists. Conversely, *target-aware* methods can genuinely advance the model by aligning with a target. This alignment can be *implicit*, such as DCLM (Li et al., 2024) and DEITA (Liu et al., 2024) using language models whose priors encode target distributions, or *explicit*, where candidates are scored directly against a defined target (Lin et al., 2024; Yu et al., 2024b; Wang et al., 2026). These methods provide advancement by successfully reducing a genuine gap.

Solvable models of data curation. Another line of research explores data curation through solvable statistical-mechanics models, using the teacher–student perceptron and replica method (Engel & den Broeck, 2022; Loureiro et al., 2021; Sorscher et al., 2022). Closest to our setup, Dohmatob et al. (2026) derive exact scaling laws for data selection under distribution shift to characterize when "less is more". While we share this solvable-model viewpoint, our approach differs in key aspects. First, we directly formulate the classification objective rather than relying on a mean squared error (MSE) surrogate. Second, we explicitly formalize the distinction between *acceleration* and *advancement*. Finally, we translate these insights and advancement prerequisites into an observable empirical diagnostic (the Δ-test), unifying both mixture and selection under a single framework, and validated on real LLMs.

7 Conclusion

We show that the value of data curation hinges on a fundamental distinction: whether a strategy genuinely advances attainable target performance or merely accelerates training. Advancement requires a correctable mismatch between the training baseline and the target. When the baseline training and target objectives coincide, the uncurated baseline is endpoint-optimal, so curation can at most accelerate. This helps explain why uniform mixtures and random selection remain difficult to beat in data-rich, in-distribution regimes. Practically, our lightweight Δ-test diagnoses advancement potential from a baseline run before expensive optimization. A near-zero Δ suggests prioritizing scaling over curation, whereas an appreciably nonzero Δ signals potential headroom for target-aware optimization. Ultimately, our findings recenter curation on the task definition itself. Given the asymmetry between massive candidate training pools and compact, controllable target sets, constructing diverse, high-fidelity targets is not merely an evaluation afterthought, but a highly leveraged design choice in shaping LLM capabilities.

[3] This open-source blend is a subset of RHO-1's target data. The remainder has not been released.

AI USE STATEMENT

In this work, we have not used generative AI tools for any tasks with required disclosure (such as developing theoretical models, implementing core methods, or interpreting results), and all required disclosure tasks are not applicable to this work. Additionally, we used generative AI tools for tasks with recommended disclosure, specifically: creating software code for experiment management and repetitive pipeline execution, creating scientific figures, and editing the research paper to improve readability. We have reviewed all AI-assisted work. Specifically, the core methodologies were exclusively developed and implemented by the authors. the AI-generated scripts for experiment management were thoroughly tested to ensure correct execution. The generated figures were manually verified against the raw experimental data, and all AI-polished text was carefully reviewed to ensure the original scientific claims were strictly preserved. We take full responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The problem formulation, evaluation criteria, and definition of the Δ-test are provided in Sections 2 to 4. The assumptions and derivations underlying the mixture and selection models, together with the population-level optimality results, are documented in Appendix A.1, A.2 and A.3. Dataset composition, target construction, shared training and evaluation protocols are described in Appendix A.5. All experiments are repeated five times, with complete standard-deviation results reported in Appendix A.6. We cite the public implementations used for available baselines and will release our codebase.

REFERENCES

Aida Amini, Saadia Gabriel, Peter Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. Mathqa: Towards interpretable math word problem solving with operation-based formalisms. *arXiv preprint*, arXiv:1905.13319, 2019.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, and Edward Raff eta.al. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, 2023.

Mayee F. Chen, Nicholas Roberts, Kush Bhatia, Jue Wang, Ce Zhang, Frederic Sala, and Christopher Ré. Skill-it! a data-driven skills framework for understanding and training language models. In *Neural Information Processing Systems*, 2023.

Mayee F. Chen, Michael Y. Hu, Nicholas Lourie, Kyunghyun Cho, and Christopher Ré. Aioli: A unified optimization framework for language model data mixing. In *International Conference on Learning Representations*, 2025.

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

Elvis Dohmatob, Mohammad Pezeshki, and Reyhane Askari-Hemmat. Why less is more (sometimes): A theory of data curation. In *International Conference on Learning Representations*, 2026.

A. Engel and C. Van den Broeck. Statistical mechanics of learning. In *Cambridge University Press*, 2022.

Simin Fan, Matteo Pagliardini, and Martin Jaggi. Doge: Domain reweighting with generalization estimation. In *International Conference on Machine Learning*, 2024.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint*, arXiv:2009.03300, 2020.

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 *Neural Information Processing Systems*, 2021.

Rik Koncel Kedziorski, Subhro Roy, Aida Amini, Nate Kushman, and Hannaneh Hajishirzi. Mawps: A math word problem repository. In *Conference of the North American Chapter of the Association for Computational Linguistics*, 2016.

Jeffrey Li, Alex Fang, and Georgios Smyrnis et.al. Datacomp-lm: In search of the next generation of training sets for language models. In *Advances in Neural Information Processing Systems*, 2024.

Zhenghao Lin, Zhibin Gou, Yeyun Gong, Xiao Liu, Yelong Shen, Ruochen Xu, Chen Lin, Yujiu Yang, Jian Jiao, Nan Duan, and Weizhu Chen. Rho-1: Not all tokens are what you need. In *Neural Information Processing Systems*, 2024.

Qian Liu, Xiaosen Zheng, Niklas Muennighoff, Guangtao Zeng, Longxu Dou, Tianyu Pang, Jing Jiang, and Min Lin. Regmix: Data mixture as regression for language model pre-training. In *International Conference on Learning Representations*, 2025.

Wei Liu, Weihao Zeng, Keqing He, Yong Jiang, and Junxian He. What makes good data for alignment? a comprehensive study of automatic data selection in instruction tuning. In *International Conference on Learning Representations*, 2024.

Bruno Loureiro, Cedric Gerbelot, Hugo Cui, Sebastian Goldt, Florent Krzakala, Marc Mezard, and Lenka Zdeborova. Learning curves of generic features maps for realistic datasets with a teacher-student model. In *Advances in Neural Information Processing Systems*, 2021.

Pan Lu, Liang Qiu, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, Tanmay Rajpurohit, Peter Clark, and Ashwin Kalyan. Dynamic prompt learning via policy gradient for semi-structured mathematical reasoning. In *International Conference on Learning Representations*, 2023.

Shenyun Miao, Chaochun Liang, and Keh-Yih Su. A diverse corpus for evaluating and developing english math word problem solvers. In *Association for Computational Linguistics*, 2020.

Jinlong Pang, Na Di, Zhaowei Zhu, Jiaheng Wei, Hao Cheng, Chen Qian, and Yang Liu. Token cleaning: Fine-grained data selection for llm supervised fine-tuning. In *International Conference on Machine Learning*, 2025.

Keiran Paster, Marco Dos Santos, Zhangir Azerbayev, and Jimmy Ba. Openwebmath: An open dataset of high-quality mathematical web text. In *International Conference on Learning Representations*, 2024.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. Are nlp models really able to solve simple math word problems? In *Conference of the North American Chapter of the Association for Computational Linguistics*, 2021.

Mansheej Paul, Surya Ganguli, and Gintare Karolina Dziugaite. Deep learning on a data diet: Finding important examples early in training. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 20596–20607. Curran Associates, Inc., 2021.

Ziheng Qin, Kai Wang, Zangwei Zheng, Jianyang Gu, Xiangyu Peng, Zhaopan Xu, Daquan Zhou, Lei Shang, Baigui Sun, Xuansong Xie, and Yang You. Infobatch: Lossless training speed up by unbiased dynamic data pruning. In *International Conference on Learning Representations*, 2024.

Daria Soboleva, Faisal Al-Khateeb, Joel Hestness, and Jacob Robert Steeves Nolan Dey Opentensor: Robert Myers. Slimpajama: A 627b token, cleaned and deduplicated version of redpajam. *https://www.cerebras.ai/blog/slimpajama-a-627b-token-cleaned-and-deduplicated-version-of-redpajama*, 2023.

Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari S. Morcos. Beyond neural scaling laws: beating power law scaling via data pruning. In *Advances in Neural Information Processing Systems*, 2022.

Jiaxing Wang, Yong Li, Jingwei Zhuo, Xupeng Shi, Weizhong Zhang, Lixing Gong, Tong Tao, Pengzhang Liu, Yongjun Bao, and Weipeng Yan. Dynams: Dynamic margin selection for efficient deep learning. In *International Conference on Learning Representations*, 2023.

Jiaxing Wang, Deping Xiang, Jin Xu, Mingyang Yi, Guoqiang Gong, Zicheng Zhang, Haoran Li, Pengzhang Liu, Zhen Chen, Ke Zhang, Ju Fan, and Qixiang Jiang. Tandem: Bi-level data mixture optimization with twin networks. In *Neural Information Processing Systems*, 2025.

Jiaxing Wang, Deping Xiang, Jin Xu, Zirui Liu, Zicheng Zhang, Guoqiang Gong, Jun Fang, Chao Liu, Pengzhang Liu, Tongxuan Liu, Ke Zhang, and Qixia Jiang. Blade: Scalable bi-level adaptive data selection for llm training. *arXiv preprint arXiv:2606.18650*, 2026.

Tingyu Xia, Bowen Yu, Kai Dang, An Yang, Yuan Wu, Yuan Tian, Yi Chang, and Junyang Lin. Rethinking data selection at scale: Random selection is almost all you need. *arXiv preprint*, arXiv:2410.09335, 2024.

Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy Liang, Quoc V Le, Tengyu Ma, and Adams Wei Yu. Doremi: Optimizing data mixtures speeds up language model pretraining. In *Neural Information Processing Systems*, 2023.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng YU, Zhengying Liu, Yu Zhang, James Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. Metamath: Bootstrap your own mathematical questions for large language models. In *International Conference on Learning Representations*, 2024a.

Zichun Yu, Spandan Das, and Chenyan Xiong. Mates: Model-aware data selection for efficient pretraining with data influence models. In *Neural Information Processing Systems*, 2024b.

Xiang Yue, Xingwei Qu, Ge Zhang, Yao Fu, Wenhao Huang, Huan Sun, Yu Su, , and Wenhu Chen. Mammoth: Building math generalist models through hybrid instruction tuning. In *International Conference on Learning Representations*, 2024.

Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. Tinyllama: An open-source small language model. *arXiv preprint*, arXiv:2401.02385, 2024.

A Appendix

A.1 Two-Teacher Mixture with Heterogeneous Difficulty

We instantiate data mixture in a two-domain teacher–student model. A mixture $\boldsymbol{w}$ determines how the total of P training examples is allocated across the two domains, with $P_k = w_k P$. All examples are then used jointly to fit one regularized logistic-regression student. We characterize its learning curves using the multi-environment state-evolution framework of Loureiro et al. (2021).

Heterogeneous-difficulty setup. We consider two domains with orthogonal teachers $\boldsymbol{B}_1, \boldsymbol{B}_2 \in \mathbb{R}^N$ satisfying $\|\boldsymbol{B}_1\| = \|\boldsymbol{B}_2\| = \sqrt{N}$ and $\boldsymbol{B}_1^\top \boldsymbol{B}_2 = 0$. The training set contains $P = \alpha N$ examples. A mixture policy $\boldsymbol{w} = (w_1, w_2) \in \mathcal{W} := \{\boldsymbol{w} \in \mathbb{R}^K \mid \sum_{k=1}^K w_k = 1,\ w_k > 0\}$ assigns $P_k = w_k P$ examples to domain k, giving the domain-specific sample ratio $\alpha_k = P_k/N = w_k \alpha$; here $K = 2$. Examples from domain k are generated according to $\boldsymbol{x}_i^k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_k)$ and $y_i^k = \mathrm{sign}(\boldsymbol{B}_k^\top \boldsymbol{x}_i^k)$. Both domains share a student $\boldsymbol{J} \in \mathbb{R}^N$ with normalized field $h(\boldsymbol{J}, \boldsymbol{x}) = \boldsymbol{J}^\top \boldsymbol{x}/\sqrt{N}$. Define the logistic margin loss $\ell_{\log}(r) = \log(1 + e^{-r})$. The trained student is

$$\boldsymbol{J}^\star(\boldsymbol{w}) = \arg\min_{\boldsymbol{J}} \left\{ \sum_{k=1}^{2} \sum_{i=1}^{P_k} \ell_{\log}\left(y_i^k \frac{\boldsymbol{J}^\top \boldsymbol{x}_i^k}{\sqrt{N}} \right) + \frac{\lambda_{\mathrm{reg}}}{2} \|\boldsymbol{J}\|^2 \right\}, \tag{3}$$

where $\lambda_{\mathrm{reg}} > 0$ is the regularization coefficient. Domain 1 is isotropic, with $\boldsymbol{\Sigma}_1 = \mathbf{I}$. We make domain 2 harder by introducing nuisance variance orthogonal to its teacher:

$$\boldsymbol{\Sigma}_2 = \frac{\boldsymbol{B}_2 \boldsymbol{B}_2^\top}{N} + (1 + \nu^2) \left(\mathbf{I} - \frac{\boldsymbol{B}_2 \boldsymbol{B}_2^\top}{N} \right), \tag{4}$$

where $\nu \geq 0$ controls the nuisance level. Since $\boldsymbol{B}_2 \boldsymbol{B}_2^\top / N$ projects onto the teacher direction, $\boldsymbol{\Sigma}_2$ has eigenvalue 1 along $\boldsymbol{B}_2$ and eigenvalue $1 + \nu^2$ in every orthogonal direction. The nuisance therefore leaves the labeling rule and Bayes boundary unchanged while increasing the finite-sample difficulty of estimating $\boldsymbol{B}_2$.

State-evolution structure. We summarize the trained student using a finite set of order parameters. The overlap $m_k = \boldsymbol{J}^\top \boldsymbol{B}_k / N$ measures the alignment of the student with the teacher of domain k, while $q = \|\boldsymbol{J}\|^2/N$ is its normalized squared norm. We further introduce the response parameter V_k, which measures how sensitively the fitted student changes under a small perturbation to examples from domain k. To enforce the definitions of these order parameters in the saddle-point derivation, we introduce the associated dual variables $\widehat{m}_k$, $\widehat{V}_k$, and $\widehat{q}_k$, analogous to Lagrange multipliers. These are auxiliary quantities rather than additional model parameters. The resulting self-consistent equations consist of two coupled parts. The loss-side equations determine $(\widehat{m}_k, \widehat{V}_k, \widehat{q}_k)$ from (m_k, V_k, q) through the domain-specific logistic loss, while the model-side equations map the dual variables back to (m_k, V_k, q) through the domain covariances. A joint fixed point of these equations determines the asymptotic properties of the trained student.

Loss-side equations. We specialize the general state-evolution equations of Loureiro et al. (2021) to the logistic loss and the two-domain covariance structure defined above. For domain k, let Q_k denote the variance of the effective student field. The covariance structure gives $Q_1 = q$ and

$$Q_2 = q + \nu^2 (q - m_2^2). \tag{5}$$

The additional term in Q_2 arises from the component of $\boldsymbol{J}$ orthogonal to $\boldsymbol{B}_2$, whose variance is amplified by the nuisance level ν. Let $V_k^\star = 1 - m_k^2/Q_k$ denote the conditional variance of the teacher field given the student field in domain k. Specializing the loss-side saddle-point conditions yields

$$\widehat{m}_k = \frac{\alpha_k}{V_k} \mathcal{I}_m(m_k, Q_k, V_k, V_k^\star), \tag{6}$$

$$\widehat{V}_k = \alpha_k \left[\frac{1}{V_k} - \frac{\mathcal{I}_V(m_k, Q_k, V_k, V_k^\star)}{V_k^2} \right], \tag{7}$$

$$\widehat{q}_k = \frac{\alpha_k}{V_k^2} \mathcal{I}_Q(m_k, Q_k, V_k, V_k^\star), \tag{8}$$

where $\mathcal{I}_m$, $\mathcal{I}_V$, and $\mathcal{I}_Q$ are scalar Gaussian expectations induced by the logistic loss and its proximal operator. Their explicit expressions are provided below. The domain difficulty first enters these equations through Q_2. By changing the effective field distribution of domain 2, the nuisance variance modifies its conjugate parameters and, consequently, its finite-sample learning dynamics.

Model-side update and spectral decomposition. The conjugate responses define the effective precision operator

$$\mathbf{D} = \lambda_{\mathrm{reg}}\mathbf{I} + \widehat{V}_1\boldsymbol{\Sigma}_1 + \widehat{V}_2\boldsymbol{\Sigma}_2. \tag{9}$$

The model-side update requires its inverse $\mathbf{D}^{-1}$. We therefore decompose the parameter space into $\mathrm{span}(\boldsymbol{B}_1)$, $\mathrm{span}(\boldsymbol{B}_2)$, and $\mathcal{B} = \mathrm{span}(\boldsymbol{B}_1, \boldsymbol{B}_2)^{\perp}$. The $(N-2)$-dimensional subspace $\mathcal{B}$ contains directions orthogonal to both teachers and is referred to as the *bulk*.

The covariance eigenvalues in these subspaces are

	$\mathrm{span}(\boldsymbol{B}_1)$	$\mathrm{span}(\boldsymbol{B}_2)$	$\mathcal{B}$
$\boldsymbol{\Sigma}_1$	1	1	1
$\boldsymbol{\Sigma}_2$	$1+\nu^2$	1	$1+\nu^2$

. (10)

Consequently, the eigenvalues of $\mathbf{D}$ are

$$d_{\boldsymbol{B}_1} = d_{\mathrm{bulk}} = \lambda_{\mathrm{reg}} + \widehat{V}_1 + (1+\nu^2)\widehat{V}_2, \qquad d_{\boldsymbol{B}_2} = \lambda_{\mathrm{reg}} + \widehat{V}_1 + \widehat{V}_2. \tag{11}$$

The corresponding eigenvalues of $\mathbf{D}^{-1}$ are their reciprocals.

Using this decomposition, the model-side updates become

$$V_1 = \frac{1}{d_{\mathrm{bulk}}}, \qquad V_2 = \frac{1+\nu^2}{d_{\mathrm{bulk}}}, \tag{12}$$

$$m_1 = \frac{\widehat{m}_1}{d_{\boldsymbol{B}_1}}, \qquad m_2 = \frac{\widehat{m}_2}{d_{\boldsymbol{B}_2}}, \tag{13}$$

$$q = \frac{\widehat{m}_1^2}{d_{\boldsymbol{B}_1}^2} + \frac{\widehat{m}_2^2}{d_{\boldsymbol{B}_2}^2} + \frac{\widehat{q}_1 + (1+\nu^2)\widehat{q}_2}{d_{\mathrm{bulk}}^2}. \tag{14}$$

Together, the loss-side and model-side updates form a closed finite-dimensional system for any mixture policy $\boldsymbol{w}$ and sample ratio α.

Target readout. For a target example from domain k, the normalized student field can be written as

$$h_k = m_k u + s_k\sqrt{q - m_k^2}\,\eta, \qquad u, \eta \sim \mathcal{N}(0,1), \tag{15}$$

where $s_1 = 1$ and $s_2 = \sqrt{1+\nu^2}$. The first term is aligned with the teacher, while the second represents the orthogonal component of the student. The corresponding signal-to-noise ratio is $\mathrm{SNR}_k = m_k/[s_k\sqrt{q - m_k^2}]$.

Let $\widehat{\boldsymbol{w}} = (\widehat{w}_1, \widehat{w}_2) \in \mathcal{W}$ denote the domain proportions of the target. Its target loss is

$$\mathcal{L}_{\mathrm{tar}}(\boldsymbol{w}; \widehat{\boldsymbol{w}}) = \sum_{k=1}^{2} \widehat{w}_k\, \mathbb{E}_{u,\eta}\left[\ell_{\log}(\mathrm{sign}(u)h_k)\right]. \tag{16}$$

Thus, solving the closed state-evolution system determines the target loss of any training mixture $\boldsymbol{w}$ against any target mixture $\widehat{\boldsymbol{w}}$. Dependence on α, ν, and λ_{reg} is suppressed for readability.

Large-budget limit: an analytic target-matching argument. Panel (c) of Figure 1 is evaluated at a large but finite budget. We now identify its limiting optimum without extrapolating the numerical state evolution. Divide equation 3 by $P = \alpha N$. If $\|\boldsymbol{J}\|^2 = O(N)$ and λ_{reg} is fixed, the normalized ridge contribution is $O(\lambda_{\mathrm{reg}}/\alpha)$ and hence vanishes as $\alpha \to \infty$. Under convergence of the empirical domain risks, the limiting student therefore minimizes

$$\begin{aligned} \boldsymbol{J}_{\infty}^{\star}(\boldsymbol{w}) &\in \arg\min_{\boldsymbol{J}}\ \mathcal{R}_{\boldsymbol{w}}(\boldsymbol{J}), \\ \mathcal{R}_{\boldsymbol{w}}(\boldsymbol{J}) &= \sum_{k=1}^{2} w_k\, \mathcal{L}^{\mathrm{pop},k}(\boldsymbol{J}), \\ \mathcal{L}^{\mathrm{pop},k}(\boldsymbol{J}) &= \mathbb{E}_{\boldsymbol{x}\sim\mathcal{N}(\mathbf{0},\boldsymbol{\Sigma}_k)}\left[\ell_{\log}\big(\mathrm{sign}(\boldsymbol{B}_k^{\top}\boldsymbol{x})h(\boldsymbol{J},\boldsymbol{x})\big)\right]. \end{aligned} \tag{17}$$

The target population risk and the endpoint induced by a training mixture are

$$\mathcal{R}_{\widehat{\boldsymbol{w}}}(\boldsymbol{J}) = \sum_{k=1}^{2} \widehat{w}_k \mathcal{L}^{\mathrm{pop},k}(\boldsymbol{J}), \qquad \mathcal{E}_\infty(\boldsymbol{w}; \widehat{\boldsymbol{w}}) = \mathcal{R}_{\widehat{\boldsymbol{w}}}(\boldsymbol{J}^\star_\infty(\boldsymbol{w}))\,. \tag{18}$$

Choosing $\boldsymbol{w} = \widehat{\boldsymbol{w}}$ makes the limiting training objective exactly the target objective. Consequently,

$$\mathcal{E}_\infty(\widehat{\boldsymbol{w}}; \widehat{\boldsymbol{w}}) \le \mathcal{E}_\infty(\boldsymbol{w}; \widehat{\boldsymbol{w}}), \qquad \forall\, \boldsymbol{w} \in \mathcal{W}, \tag{19}$$

so target matching is a global optimum. This argument is analytic and does not depend on the nuisance level ν. The *location in mixture space* of the matching optimum is therefore independent of difficulty, although the population minimizer and attained target loss generally still depend on ν through Σ_2.

If the target risk has a unique minimizer and the baseline induces a different population minimizer, the inequality in equation 19 is strict. These conditions hold in the nondegenerate two-teacher setting considered here: the Gaussian covariances are full rank, the mixed logistic population risk is strictly convex at its finite optimum, and distinct teachers do not share a stationary point of their individual risks. Two regimes follow:

- *No gap* ($\widehat{\boldsymbol{w}} = \boldsymbol{w}_{\text{baseline}}$): matching returns the baseline, so $\mathcal{E}_\infty(\boldsymbol{w}; \widehat{\boldsymbol{w}}) \ge \mathcal{E}_\infty(\boldsymbol{w}_{\text{baseline}}; \widehat{\boldsymbol{w}})$ for all $\boldsymbol{w}$ and no reallocation lowers the endpoint—acceleration only.
- *With a correctable composition gap* ($\widehat{\boldsymbol{w}} \neq \boldsymbol{w}_{\text{baseline}}$): the baseline minimizes a differently weighted population risk. Under the nondegeneracy conditions above, the two weightings induce different minimizers, and hence $\mathcal{E}_\infty(\widehat{\boldsymbol{w}}; \widehat{\boldsymbol{w}}) < \mathcal{E}_\infty(\boldsymbol{w}_{\text{baseline}}; \widehat{\boldsymbol{w}})$—advancement.

Thus the numerical curve at $\alpha = 100$ in panel (c) approximates an exactly target-matched population optimum; it is not the sole evidence for that endpoint.

Consequences for acceleration and advancement. At finite sample ratios, the unequal field variances in Equation (5) cause the two domains to be learned at different rates. Allocating more data to the harder domain can therefore reduce target loss at small α, as in Figure 1(b). When the baseline already matches the target, this advantage vanishes with increasing budget and is acceleration. When the target composition differs from the baseline, equation 19 instead gives a strictly better population endpoint under the stated nondegeneracy conditions, which is the advancement effect in panel (c).

Logistic integrals. For completeness, we specify the Gaussian expectations in Equations (6) to (8). Let $\omega = \sqrt{Q}\,\xi$ with $\xi \sim \mathcal{N}(0,1)$, $\omega_\star = (m/\sqrt{Q})\xi$, and

$$a(\xi) = \operatorname{erf}\left(\frac{(m/\sqrt{Q})\xi}{\sqrt{2V^\star}}\right). \tag{20}$$

Define

$$\eta^\pm = \operatorname{prox}_{V\ell_{\log}(\pm\cdot)}(\omega) = \arg\min_r \left\{\frac{(r-\omega)^2}{2V} + \ell_{\log}(\pm r)\right\}. \tag{21}$$

The required integrals are

$$\mathcal{I}_m = \frac{1}{\sqrt{2\pi V^\star}} \mathbb{E}_\xi \left[e^{-\frac{\omega_\star^2}{2V^\star}} (\eta^+ - \eta^-)\right], \tag{22}$$

$$\mathcal{I}_V = \frac{1}{2}\mathbb{E}_\xi \left[\frac{1+a(\xi)}{1/V + \frac{1}{4}\operatorname{sech}^2(\eta^+/2)} + \frac{1-a(\xi)}{1/V + \frac{1}{4}\operatorname{sech}^2(\eta^-/2)}\right], \tag{23}$$

$$\mathcal{I}_Q = \frac{1}{2}\mathbb{E}_\xi \left[(1+a(\xi))(\eta^+ - \omega)^2 + (1-a(\xi))(\eta^- - \omega)^2\right]. \tag{24}$$

Reduction checks and numerical solution. When $\nu = 0$, the two domains have identical covariance, $d_{\boldsymbol{B}_1} = d_{\boldsymbol{B}_2} = d_{\text{bulk}} = \lambda_{\text{reg}} + \widehat{V}_1 + \widehat{V}_2$, and $V_1 = V_2$. The system then reduces to the isotropic two-teacher logistic state evolution. We solve the fixed-point equations using a nonuniform composite quadrature for the Gaussian expectations. At large sample ratios, $V^\star$ approaches zero and $a(\xi)$ becomes sharply concentrated, causing fixed Gauss–Hermite grids to under-resolve the integrands. Refining the grid around the shrinking transition layer avoids this numerical artifact. The isotropic reduction at $\nu = 0$ and agreement with finite-dimensional simulations provide independent checks of the solution.

A.2 Selection under Matched and Shifted Targets

Selection starts from $P = \alpha N$ candidate examples and retains a fraction f, leaving a training budget $f\alpha N$. We first consider a matched target and then extend the model to a shifted target, where label consistency provides an additional selection signal. We write the target-error learning curve as $\mathcal{E}_\alpha(f;\phi)$, where ϕ is the angle between the training and target teachers; dependence on the fixed $\lambda_{\rm sel}$ in the shifted-target policy is suppressed.

A.2.1 Matched target: classical margin selection

Let $\boldsymbol{B} \in \mathbb{R}^N$, $\|\boldsymbol{B}\| = \sqrt{N}$, be both the training and target teacher. The $P = \alpha N$ candidates are generated by $\boldsymbol{x}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $y_i = \mathrm{sign}(\boldsymbol{B}^\top \boldsymbol{x}_i)$. Writing the teacher margin as $z = \boldsymbol{B}^\top \boldsymbol{x}/\sqrt{N}$ (and $z_i = \boldsymbol{B}^\top \boldsymbol{x}_i/\sqrt{N}$ for candidate i), margin selection retains $|z| < \gamma_f$, where

$$f = \int_{|z|<\gamma_f} \varphi(z)\,\mathrm{d}z \tag{25}$$

is the retained fraction and φ is the standard Gaussian density. The selected set is $S_f = \{i : |z_i| < \gamma_f\}$. We use the spherical max-margin student

$$\boldsymbol{J}^\star \in \arg\max_{\|\boldsymbol{J}\|=\sqrt{N}} \min_{i\in S_f} \frac{y_i \boldsymbol{J}^\top \boldsymbol{x}_i}{\sqrt{N}}, \qquad \kappa = \min_{i\in S_f} \frac{y_i (\boldsymbol{J}^\star)^\top \boldsymbol{x}_i}{\sqrt{N}}. \tag{26}$$

This has the same direction as the usual minimum-norm hard-margin solution; the spherical form makes the normalization used by the replica equations explicit.

Rather than repeating the replica calculation, we use the arbitrary-margin single-teacher equations of Sorscher et al. (2022, Eqs. 66–67); for a truncated Gaussian margin distribution, they reduce to their Eqs. 68–69. The retained examples have sample ratio $f\alpha$ and conditional margin density

$$p_f(z) = \frac{\varphi(z)\mathbf{1}\{|z| < \gamma_f\}}{f}. \tag{27}$$

In the high-dimensional limit, the empirical selection threshold and retained margin distribution concentrate at γ_f and p_f, respectively. The cited equations depend on the product of the sample ratio and margin density. Substituting $(f\alpha, p_f)$ therefore gives $f\alpha p_f(z) = \alpha\varphi(z)\mathbf{1}\{|z| < \gamma_f\}$, exactly the exposure induced by selecting from the candidate set. Thus the normalized student–teacher overlap $R = (\boldsymbol{J}^\star)^\top \boldsymbol{B}/N$ and maximum stability κ satisfy

$$R = 2f\alpha \int_{-\infty}^{\kappa} \mathrm{d}t \int_0^\infty \mathrm{d}z\, p_f(z) \frac{e^{-\frac{(t-Rz)^2}{2(1-R^2)}}}{\sqrt{2\pi(1-R^2)}} (\kappa - t) \frac{z - Rt}{1-R^2}, \tag{28}$$

$$1 - R^2 = 2f\alpha \int_{-\infty}^{\kappa} \mathrm{d}t \int_0^\infty \mathrm{d}z\, p_f(z) \frac{e^{-\frac{(t-Rz)^2}{2(1-R^2)}}}{\sqrt{2\pi(1-R^2)}} (\kappa - t)^2. \tag{29}$$

The factor of 2 uses the symmetry between positive and negative teacher margins, allowing the z-integral to be restricted to $z > 0$. The target error is $\mathcal{E}_\alpha(f;0) = \arccos(R)/\pi$. We solve Equations (28) and (29) by adaptive quadrature; $f = 1$ recovers the standard unpruned perceptron equations. These curves give the matched-target selection result in Figure 1(e).

A.2.2 Shifted target: consistency-aware selection

We now let the target teacher differ from the training teacher. Applying a target-margin rule alone is insufficient: labels still come from the training teacher, so examples near the target boundary may carry labels that conflict with the target. We therefore model the additional signal used by target-aware selectors through a consistency filter. This remains selection rather than label correction: inconsistent examples are removed, and no label is changed.

Setup and selection rule. Let the training and target teachers be $\boldsymbol{B}$ and $\boldsymbol{B}_{\rm tar}$, with $\|\boldsymbol{B}\| = \|\boldsymbol{B}_{\rm tar}\| = \sqrt{N}$ and

$$R_0 = \frac{\boldsymbol{B}^\top \boldsymbol{B}_{\rm tar}}{N} = \cos\phi. \tag{30}$$

We take $0 < \phi < \pi/2$ in this shifted-target analysis; the matched endpoint $\phi = 0$ is handled separately above. For an example $\boldsymbol{x}$, define

$$u = \frac{\boldsymbol{B}^\top \boldsymbol{x}}{\sqrt{N}}, \qquad z = \frac{\boldsymbol{B}_{\mathrm{tar}}^\top \boldsymbol{x}}{\sqrt{N}}, \qquad y = \mathrm{sign}(u), \tag{31}$$

so (u, z) is jointly standard Gaussian with correlation R_0. Let $A = \{\mathrm{sign}(u) = \mathrm{sign}(z)\}$ denote label consistency. We rank candidates by the single target-aware score

$$q_{\lambda_{\mathrm{sel}}}(u, z) = |z| + \lambda_{\mathrm{sel}} \mathbf{1}_{A^c}, \qquad \lambda_{\mathrm{sel}} > 0. \tag{32}$$

When $R_0 = 1$, A holds almost surely and this score reduces to the matched-target margin rule above. We retain the lowest-scoring fraction f. Equivalently, for a threshold τ_f, a consistent example is retained when $|z| < \tau_f$, whereas an inconsistent one is retained only when $|z| < (\tau_f - \lambda_{\mathrm{sel}})_+$, where $(r)_+ = \max\{r, 0\}$. The threshold is determined by

$$f = 2\int_0^{\tau_f} \varphi(z)\Phi\left(\frac{R_0 z}{\sqrt{1-R_0^2}}\right) \mathrm{d}z + 2\int_0^{(\tau_f - \lambda_{\mathrm{sel}})_+} \varphi(z)\left[1 - \Phi\left(\frac{R_0 z}{\sqrt{1-R_0^2}}\right)\right] \mathrm{d}z, \tag{33}$$

where φ and Φ are the standard Gaussian density and cdf. Unlike a lexicographic "consistency first, margin second" rule, this score changes the selected distribution continuously as f varies. The two terms nevertheless retain distinct roles: consistency makes the gap correctable, whereas margin controls finite-budget informativeness. We write $s_f(u, z) = \mathbf{1}\{q_{\lambda_{\mathrm{sel}}}(u, z) < \tau_f\}$ for the resulting selection indicator. Panel (f) uses $\lambda_{\mathrm{sel}} = 0.75$.

Applied to the $P = \alpha N$ candidates, s_f retains fP examples and hence uses sample ratio $f\alpha$, as in the matched-target case.

Conditional field and truncated moments. The selected student is again the spherical max-margin solution, with $\|\boldsymbol{J}\| = \sqrt{N}$; let κ denote its maximum stability. Hence $R = \boldsymbol{J}^\top \boldsymbol{B}/N$ and $\rho = \boldsymbol{J}^\top \boldsymbol{B}_{\mathrm{tar}}/N$ are normalized cosine overlaps. For $t = \boldsymbol{J}^\top \boldsymbol{x}/\sqrt{N}$, Gaussian conditioning gives

$$t \mid (u, z) \sim \mathcal{N}(au + bz, \Sigma^2), \tag{34}$$

where

$$a = \frac{R - \rho R_0}{1 - R_0^2}, \qquad b = \frac{\rho - R R_0}{1 - R_0^2}, \qquad \Sigma^2 = 1 - \frac{R^2 + \rho^2 - 2R\rho R_0}{1 - R_0^2}. \tag{35}$$

Let $h = yt$, $\mu = au + bz$, and

$$\eta = \frac{\kappa - y\mu}{\Sigma}, \quad \mathcal{I}_0 = \Phi(\eta), \quad \mathcal{I}_1 = \Sigma[\eta\Phi(\eta) + \varphi(\eta)], \tag{36}$$

$$\mathcal{I}_2 = \Sigma^2[(1 + \eta^2)\Phi(\eta) + \eta\varphi(\eta)]. \tag{37}$$

These are the truncated Gaussian moments generated by the active hard-margin constraints. For any $F(u, z)$, denote expectation under the normalized selected distribution by

$$\langle F \rangle_f = \frac{\mathbb{E}[F(u, z) s_f(u, z)]}{f}. \tag{38}$$

Finite-budget saddle-point equations. The replica-symmetric, max-margin limit gives the following closed system for (κ, R, ρ):

$$f\alpha \langle \mathcal{I}_2 \rangle_f = \Sigma^2, \tag{39}$$

$$f\alpha \left[\frac{\langle y(u - R_0 z)\mathcal{I}_1 \rangle_f}{1 - R_0^2} + a\langle \mathcal{I}_0 \rangle_f\right] = a, \tag{40}$$

$$f\alpha \left[\frac{\langle y(z - R_0 u)\mathcal{I}_1 \rangle_f}{1 - R_0^2} + b\langle \mathcal{I}_0 \rangle_f\right] = b. \tag{41}$$

The target error is $\mathcal{E}_\alpha(f; \phi) = \arccos(\rho)/\pi$. These equations generate the curves in Figure 1(f); numerical quadrature and finite-dimensional hard-margin simulations use the same candidate set and policy.

An interior point on the score path. The score in equation 32 itself contains an explicit interior policy for every finite $\lambda_{\mathrm{sel}} > 0$. Consider the threshold $\tau_f = \lambda_{\mathrm{sel}}$. A consistent example is then retained when $|z| < \lambda_{\mathrm{sel}}$, whereas an inconsistent example would have to satisfy $|z| < 0$ and is therefore discarded up to a measure-zero boundary. The corresponding retained fraction is

$$f_{\lambda_{\mathrm{sel}}} = \Pr[A, |z| < \lambda_{\mathrm{sel}}] = 2\int_0^{\lambda_{\mathrm{sel}}} \varphi(z)\Phi\left(\frac{R_0 z}{\sqrt{1-R_0^2}}\right)\mathrm{d}z, \qquad 0 < f_{\lambda_{\mathrm{sel}}} < 1. \tag{42}$$

Thus $f_{\lambda_{\mathrm{sel}}}$ is an interior point on the same λ_{sel}-parameterized policy path used in panel (f), rather than a certificate supplied by a separate consistency-only rule. More generally, $f \le f_{\lambda_{\mathrm{sel}}}$ corresponds to $\tau_f \le \lambda_{\mathrm{sel}}$ and contains no inconsistent examples, while $f > f_{\lambda_{\mathrm{sel}}}$ corresponds to $\tau_f > \lambda_{\mathrm{sel}}$ and retains a positive-measure set of inconsistent examples near the target boundary.

Large-budget endpoint of the score path. At $f = f_{\lambda_{\mathrm{sel}}}$, multiplying each retained input by its training label y gives signed fields $yu = |u| > 0$ and $yz = |z| > 0$. In the $(\boldsymbol{B}, \boldsymbol{B}_{\mathrm{tar}})$ plane, the local support is therefore a wedge bounded by the two teacher decision boundaries. The truncation $|z| < \lambda_{\mathrm{sel}}$ changes its radial extent but not its tangent cone at the intersection of the boundaries. As the sample ratio grows, the active constraints approach both sides of this wedge. Within this isotropic model, the direction maximizing the angular margin to the two sides is their bisector; an orthogonal component cannot improve the margin because it carries no systematic label signal. This support-geometry characterization gives

$$\frac{\boldsymbol{J}_\infty}{\|\boldsymbol{J}_\infty\|} = \frac{\boldsymbol{B} + \boldsymbol{B}_{\mathrm{tar}}}{\|\boldsymbol{B} + \boldsymbol{B}_{\mathrm{tar}}\|}, \qquad \rho_\infty = \cos\frac{\phi}{2}. \tag{43}$$

By contrast, full retention preserves all labels from $\boldsymbol{B}$, so its large-sample solution aligns with $\boldsymbol{B}$ and has target overlap $\cos\phi$. Consequently,

$$\mathcal{E}_\infty(f_{\lambda_{\mathrm{sel}}}; \phi) = \frac{\phi}{2\pi} < \frac{\phi}{\pi} = \mathcal{E}_\infty(1; \phi). \tag{44}$$

Thus the family contains a fixed interior policy that strictly beats full retention at the endpoint. The particular value $\lambda_{\mathrm{sel}} = 0.75$ is not needed for existence: every $\lambda_{\mathrm{sel}} > 0$ defines a nontrivial $f_{\lambda_{\mathrm{sel}}}$ through equation 42. Consistency supplies the endpoint shift, whereas the margin term locates that shift on the continuous selection path and controls which consistent constraints are retained.

The same argument clarifies the two sides of the threshold. For any fixed $0 < f \le f_{\lambda_{\mathrm{sel}}}$, the selected support has the same agreement wedge near the origin and hence the same bisector endpoint. Once $f > f_{\lambda_{\mathrm{sel}}}$, the cutoff $|z| < (\tau_f - \lambda_{\mathrm{sel}})$ admits inconsistent examples in an open neighborhood of the intersection. Any separating direction with a component orthogonal to $\boldsymbol{B}$ then violates some training-label constraint in the population support; the asymptotic direction therefore returns to $\boldsymbol{B}$. Thus, within this idealized model, the support-level prediction is

$$\mathcal{E}_\infty(f; \phi) = \begin{cases} \phi/(2\pi), & 0 < f \le f_{\lambda_{\mathrm{sel}}}, \\ \phi/\pi, & f_{\lambda_{\mathrm{sel}}} < f \le 1. \end{cases} \tag{45}$$

Finite candidate sets smooth this sharp support-level transition.

Why the finite-budget curve has an interior optimum. The finite-budget tradeoff is now explicit. At $f = 1$ the learner uses the full candidate set and approaches the training teacher. Near $f = f_{\lambda_{\mathrm{sel}}} < 1$, the score has removed the inconsistent constraints and the learner approaches the improved bisector endpoint in equation 44. If f is reduced too far at fixed α, however, the retained training ratio $f\alpha$ vanishes and target accuracy returns to chance. Continuity of the saddle equations then implies that, for all sufficiently large but finite α, a maximum is attained at some

$$f^\star(\alpha) \in (0, 1) \tag{46}$$

and exceeds the value at $f = 1$. The optimum lies near the point where the last inconsistent constraints disappear; for the parameters of panel (f), this is $f_{\lambda_{\mathrm{sel}}}$. As $\alpha \to \infty$, the small-f sample-efficiency penalty disappears for every fixed $f > 0$, so the smooth finite-budget peak need not converge to a unique optimizer. Instead it can broaden into the improved plateau in equation 45. What persists is the advancement: the fixed interior policy $f = f_{\lambda_{\mathrm{sel}}}$ remains strictly better than full retention in the large-budget limit.

A.3 Formalizing Proposition 1: Bi-level Optimality under No Gap

We now formalize Proposition 1 for mixture and selection. For mixture, the reference policy is the target-matching allocation $\widehat{\boldsymbol{w}}$. When $\widehat{\boldsymbol{w}} = \boldsymbol{w}_{\text{baseline}}$, the resulting optimality statement is exactly the no-gap conclusion of Proposition 1. For selection, the corresponding baseline is the uniform soft allocation $\bar{w}_i = f$, whose operational counterpart is uniform random hard selection. The connection holds in the large-sample limit.

Domain-level mixture. We begin with data mixture, where the allocation variable assigns training exposure across domains. Throughout this section the losses are *population* quantities, written $\mathcal{L}^{\text{pop}}$ to distinguish them from the empirical, seen-data losses $\mathcal{L}$ measured by the Δ-test (Section 4).

Let x denote a data example (including its label, when applicable), and let $\ell(\boldsymbol{\theta}; x)$ be the loss of model $\boldsymbol{\theta}$ on that example. For domain k, define

$$\mathcal{L}^{\text{pop},k}_{\text{train}}(\boldsymbol{\theta}) := \mathbb{E}_{x\sim\mathcal{P}^k_{\text{train}}}[\ell(\boldsymbol{\theta}; x)], \qquad \mathcal{L}^{\text{pop},k}_{\text{tar}}(\boldsymbol{\theta}) := \mathbb{E}_{x\sim\mathcal{P}^k_{\text{target}}}[\ell(\boldsymbol{\theta}; x)].$$

A mixture policy $\boldsymbol{w}$ can be realized either by sampling $P_k = w_k P$ unit-weight examples from each domain, or by using w_k directly as weighting coefficients on the domain losses. For an additive per-example loss, the empirical risk under the sampling-based implementation satisfies

$$\begin{aligned}\frac{1}{P}\sum_{k=1}^{K}\sum_{i=1}^{P_k}\ell(\boldsymbol{\theta}; x_i^k) &= \sum_{k=1}^{K}\frac{P_k}{P}\left[\frac{1}{P_k}\sum_{i=1}^{P_k}\ell(\boldsymbol{\theta}; x_i^k)\right] \\ &\xrightarrow[P\to\infty]{} \sum_{k=1}^{K} w_k\,\mathbb{E}_{x\sim\mathcal{P}^k_{\text{train}}}[\ell(\boldsymbol{\theta}; x)] \\ &= \sum_{k=1}^{K} w_k \mathcal{L}^{\text{pop},k}_{\text{train}}(\boldsymbol{\theta}).\end{aligned} \tag{47}$$

Assuming i.i.d. sampling within each domain and uniform convergence of the empirical domain risks to their population counterparts, the sampling-based objective converges uniformly to the loss-weighted population objective. Thus, the two implementations induce the same limiting population problem. At finite P, they may differ because $w_k P$ must be rounded and the empirical domain averages fluctuate, but these differences vanish as $P_k \to \infty$ for every k. This domain-level equivalence is distinct from the example-level soft relaxation introduced below. We therefore state the bi-level mixture result directly in the loss-weighted population form.

Proposition 2 (Target-matching optimality for mixture)**.** *Consider the bi-level mixture problem*

$$\boldsymbol{w}^\star \in \arg\min_{\boldsymbol{w}\in\mathcal{W}} \sum_{k=1}^{K}\widehat{w}_k\mathcal{L}^{\text{pop},k}_{\text{tar}}\big(\boldsymbol{\theta}^\star(\boldsymbol{w})\big), \quad \textit{s.t.} \quad \boldsymbol{\theta}^\star(\boldsymbol{w}) \in \arg\min_{\boldsymbol{\theta}} \sum_{k=1}^{K} w_k\mathcal{L}^{\text{pop},k}_{\text{train}}(\boldsymbol{\theta}), \tag{48}$$

where $\widehat{\boldsymbol{w}} \in \mathcal{W}$ is the target domain composition. Suppose $\mathcal{P}^k_{\text{train}} = \mathcal{P}^k_{\text{target}}$ and hence $\mathcal{L}^{\text{pop},k}_{\text{train}} = \mathcal{L}^{\text{pop},k}_{\text{tar}} =: \mathcal{L}^{\text{pop},k}$ for every k. Then the target-matching allocation $\boldsymbol{w} = \widehat{\boldsymbol{w}}$ is a global minimizer of equation 48. In particular, if the target weights are uniform, then $\boldsymbol{w} = \frac{1}{K}\mathbf{1}$ is globally optimal.

Proof. Under the no-gap condition, the training and target population losses coincide within every domain. When $\boldsymbol{w} = \widehat{\boldsymbol{w}}$, the inner problem becomes

$$\boldsymbol{\theta}^\star(\widehat{\boldsymbol{w}}) \in \arg\min_{\boldsymbol{\theta}} \sum_{k=1}^{K}\widehat{w}_k\mathcal{L}^{\text{pop},k}(\boldsymbol{\theta}), \tag{49}$$

which is exactly the outer target objective viewed as a function of $\boldsymbol{\theta}$. Therefore, its value at $\boldsymbol{\theta}^\star(\widehat{\boldsymbol{w}})$ cannot exceed its value at the model $\boldsymbol{\theta}^\star(\boldsymbol{w})$ induced by any other mixture $\boldsymbol{w} \in \mathcal{W}$:

$$\sum_{k=1}^{K}\widehat{w}_k\mathcal{L}^{\text{pop},k}\big(\boldsymbol{\theta}^\star(\widehat{\boldsymbol{w}})\big) \le \sum_{k=1}^{K}\widehat{w}_k\mathcal{L}^{\text{pop},k}\big(\boldsymbol{\theta}^\star(\boldsymbol{w})\big). \tag{50}$$

Thus, $\widehat{\boldsymbol{w}}$ is a global minimizer of the outer objective. □

Setting $\widehat{\boldsymbol{w}} = \boldsymbol{w}_{\text{baseline}}$ thus recovers the mixture instantiation of Proposition 1.

The assumption $\mathcal{P}^k_{\text{train}} = \mathcal{P}^k_{\text{target}}$ is a distribution-level condition. It implies equality of the corresponding population losses but does not assert equality between individual training and target examples. This distinction becomes important for data selection, where exposure is allocated at the level of individual sample rather than domains.

Example-level selection. We next express selection as a per-example bi-level problem. Let $\mathcal{D}_{\text{train}} = \{x_i\}_{i=1}^P$ be the candidate pool. A hard selection policy is represented exactly by binary indicators in

$$\mathcal{W}_f^{\text{hard}} = \left\{ \boldsymbol{w} \in \{0,1\}^P : \sum_{i=1}^P w_i = fP \right\}. \tag{51}$$

For the population-level argument, we consider the continuous relaxation

$$\mathcal{W}_f^{\text{soft}} = \left\{ \boldsymbol{w} \in [0,1]^P : \sum_{i=1}^P w_i = fP \right\}. \tag{52}$$

The relaxed bi-level problem is:

$$\min_{\boldsymbol{w} \in \mathcal{W}_f^{\text{soft}}} \mathcal{L}^{\text{pop}}_{\text{tar}}\big(\boldsymbol{\theta}^\star(\boldsymbol{w})\big) \quad \text{subject to} \quad \boldsymbol{\theta}^\star(\boldsymbol{w}) \in \arg\min_{\boldsymbol{\theta}} \frac{1}{fP} \sum_{i=1}^P w_i \ell(\boldsymbol{\theta}; x_i). \tag{53}$$

Here, selection reweights individual examples in the finite candidate pool, whereas evaluation is performed on the target distribution as a whole. We therefore define the no-gap condition as $\mathcal{P}_{\text{train}} = \mathcal{P}_{\text{target}}$. The inner objective is a weighted loss over the selected examples, while the outer objective is the target population loss. Under uniform selection, the former converges to the latter as $P \to \infty$, which motivates the asymptotic result below.

Proposition 3 (Asymptotic optimality of uniform selection)**.** *Suppose the candidate and target examples are drawn from the same distribution, $\mathcal{P}_{\text{train}} = \mathcal{P}_{\text{target}}$. Assume that empirical risk minimization is consistent over the model class (e.g., via uniform convergence). Then the uniform soft allocation $\bar{w}_i = f$ is a global minimizer of equation 53 in the large-sample limit.*

Proof. Under the no-gap condition, the population training and target objectives coincide:

$$\mathcal{L}^{\text{pop}}_{\text{tar}}(\boldsymbol{\theta}) = \mathbb{E}_{x \sim \mathcal{P}_{\text{target}}}[\ell(\boldsymbol{\theta}; x)] = \mathbb{E}_{x \sim \mathcal{P}_{\text{train}}}[\ell(\boldsymbol{\theta}; x)]. \tag{54}$$

For the uniform allocation $\bar{w}_i = f$, the inner objective becomes

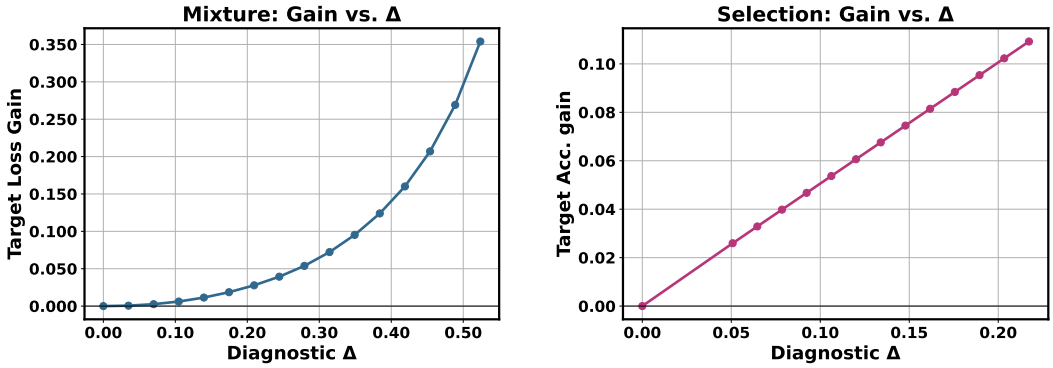
$$\frac{1}{fP} \sum_{i=1}^P \bar{w}_i \ell(\boldsymbol{\theta}; x_i) = \frac{1}{P} \sum_{i=1}^P \ell(\boldsymbol{\theta}; x_i). \tag{55}$$

As $P \to \infty$, consistency of empirical risk minimization implies that $\boldsymbol{\theta}^\star(\bar{\boldsymbol{w}})$ approaches the set of population target minimizers. Hence, in the large-sample limit, for every admissible allocation $\boldsymbol{w} \in \mathcal{W}_f^{\text{soft}}$,

$$\mathcal{L}^{\text{pop}}_{\text{tar}}\big(\boldsymbol{\theta}^\star(\bar{\boldsymbol{w}})\big) \le \mathcal{L}^{\text{pop}}_{\text{tar}}\big(\boldsymbol{\theta}^\star(\boldsymbol{w})\big). \tag{56}$$

Therefore, no alternative selection allocation achieves a better limiting target endpoint than uniform selection. □

The allocation $\bar{w}_i = f$ is an analytical reference rather than a literal subset: it reproduces the empirical risk of the full candidate pool while satisfying the same total weight constraint. Its hard-selection analogue is uniform random selection without replacement, under which every example has marginal inclusion probability f. Conditional on the candidate pool, the resulting empirical objective is an unbiased estimator of the soft-allocation objective in equation 55 and concentrates around it as $P \to \infty$. Moreover, $\mathcal{W}_f^{\text{hard}} \subset \mathcal{W}_f^{\text{soft}}$, so the result also rules out a lower limiting target loss from any hard selection policy. Thus, uniform random selection is the fixed-retention, asymptotic counterpart of the full-retention reference policy in Proposition 1: both induce the baseline distribution in the population limit. This correspondence does not imply finite-sample equivalence between the soft allocation and any particular random subset.



(a) (b)

Figure 4: **Curation gain versus the Δ diagnostic in the solvable models.** Each point varies the parameter that generates the train–target gap and reports the improvement of the optimized policy over the uncurated baseline. (a) Mixture: target-composition sweep. Gain is the reduction in target logistic loss. (b) Selection: teacher-angle sweep. Gain is the reduction in target classification error.

Connection to Proposition 1. The two results instantiate the same no-gap principle at different granularities. For mixture, target matching is optimal directly at the population level; setting $\widehat{\boldsymbol{w}} = \boldsymbol{w}_{\text{baseline}}$ recovers the mixture case of Proposition 1. For selection, full retention is the population reference, and uniform random hard selection approaches the same endpoint asymptotically. Thus, no gap rules out advancement in both cases, while still allowing finite-budget acceleration.

Relation to the empirical Δ-test. The propositions above are stated in terms of population losses $\mathcal{L}^{\text{pop}}$, whereas the Δ-test of Section 4 is computed from the empirical, seen-data losses $\mathcal{L}$ of a single baseline run. The two are connected by the same large-sample limit used in the proofs: when the baseline visits each example about once in the data-rich regime, the empirical training loss converges to its population counterpart, so the no-gap condition $\mathcal{P}_{\text{train}} = \mathcal{P}_{\text{target}}$ typically manifests as a vanishing empirical loss gap $\Delta \approx 0$, and Propositions 2 and 3 predict that no allocation can advance the model. This correspondence breaks precisely in the endogenous regime, where repetition on under-resourced domains drives the empirical training loss below the population loss: the seen-data Δ then becomes nonzero and detects a gap that a purely population statement would miss (Section A.4). The Δ-test is thus a useful observable, finite-sample signature of the population condition these propositions characterize, but it is not equivalent to that condition.

A.4 Measuring and Interpreting Δ

Measurement on seen training data. We evaluate $\mathcal{L}_{\text{train}}$ on the examples used to train the uncurated baseline, rather than on a held-out sample from the training distribution. This distinction matters for an endogenous gap: when a domain is under-resourced, repeated exposure can reduce its loss on the seen examples without producing the same improvement on the target distribution. A held-out training split would not reveal this repetition-induced discrepancy. Because Δ compares a finite collection of losses rather than distributions directly, it remains an operational diagnostic rather than a complete statistical test: distinct distributions can yield similar losses, and finite-sample fluctuations can produce a small nonzero value.

Relation to curation gain. Figure 4 evaluates the diagnostic in the two solvable models studied above. In mixture, varying the target composition changes Δ while the train and target distributions remain matched within each domain. The gain from optimizing the mixture is zero at $\Delta = 0$ and increases as the composition gap grows. In selection, increasing the teacher angle ϕ enlarges the candidate–target mismatch, and the improvement achieved by target-aware selection over full retention increases correspondingly. Thus, despite their different allocation units and performance metrics, both models exhibit the same qualitative relation: a larger gap exposed by the uncurated baseline provides greater headroom for curation.

A.5 Real-Data Experimental Details

This section collects the implementation details. Across settings, we evaluate all baselines within a shared protocol and report five independent runs. The diagnostic Δ is measured at 50% training budget.

A.5.1 Data-mixture experiments

Setup and evaluation. Following Chen et al. (2025), we use a 6B-token subset of SlimPajama (Soboleva et al., 2023), comprising ArXiv, Books, CommonCrawl, C4, GitHub, StackExchange, and Wikipedia. In the no-gap setting, both training and target weight all seven domains uniformly. In the exogenous-gap setting, CommonCrawl is held out as the target and the remaining six domains form the candidate training pool. The endogenous-gap setting retains the uniformly weighted target but restricts the available corpus to 300M tokens while preserving its original domain proportions, causing small domains to be repeatedly sampled. We report held-out perplexity for each target domain. For a multi-domain target, "Average" is the exponential of the mean domain loss, following Fan et al. (2024). for the exogenous-gap setting, the metric is CommonCrawl perplexity.

Shared training protocol. The main experiments train 160M-parameter GPT-style models (Biderman et al., 2023) for 40K steps. As summarized in Table 5. All three gap settings use the same batch size, context length, schedule, and number of steps, so the comparison changes the available data or target rather than the base training protocol. Mixture optimization and final-model training use the same model-training hyperparameters unless stated otherwise.

Table 5: Shared configuration for the data-mixture experiments.

Configuration	160M	1.1B
Training steps	40K	40K
Batch size	8	8
Context length	2048	2048
Peak learning rate	5×10^{-4}	2×10^{-4}
Warmup steps	–	500
Weight decay	0.01	0.01
Gradient clipping	1.0	1.0

Baselines and implementation. All methods use the shared protocol above. Uniform assigns equal exposure to the candidate domains and serves as the baseline policy. We compare it with RegMix (Liu et al., 2025), which fits global data mixing laws and predict the performance on other mixtures; DoGE (Fan et al., 2024), which updates domain weights using their estimated influence on the target objective; Skill-It (Chen et al., 2023) utilize the inter-domain dependencies to establish a local data mixing law that predicts the per-domain performance; Aioli (Chen et al., 2025) improves Skill-It by dynamically updating the relation matrix along model training; and TANDEM (Wang et al., 2025), which optimizes the mixture through a bi-level objective with twinned networks. We fit the RegMix mixing law using 128 independently trained 1M-parameter proxy models. For RegMix, DoGE, Skill-It, and Aioli, we adapt the public implementations to our common data interface and training pipeline.[4] Method-specific choices otherwise follow the corresponding papers and released implementations. The degree of smoothing recommended in the original Skill-It and Aioli settings can yield nearly uniform allocations in our experiments, we adjust only their smoothing controls for each gap setting, using less smoothing in the endogenous- and exogenous-gap settings. The learned allocation is used to train the final model with the shared configuration above. Implementation and configuration will be released with the complete codebase.

[4]Public implementations: RegMix,`https://github.com/sail-sg/regmix`; DoGE, `https://github.com/Olivia-fsm/DoGE`; and Skill-It and Aioli, `https://github.com/HazyResearch/aioli`.

A.5.2 DATA-SELECTION EXPERIMENTS

Setup and target construction. We use TinyLlama-1.1B (Zhang et al., 2024) as the base model and draw a 5B-token candidate pool from OpenWebMath (Paster et al., 2024). The in-distribution target is a held-out 0.3B-token OpenWebMath subset. The shifted target blends MetaMath (Yu et al., 2024a) and Mammoth (Yue et al., 2024). The main comparison retains 60% of the candidate pool. Figure 3 additionally sweeps the retained fraction from 0.6 to 1.0.

Shared training protocol. All selection methods use the configuration in Table 6. They share the same base model, candidate pool, retained-token budget, and training protocol, so that only the selection rule and target distribution vary.

Table 6: Shared configuration for the data-selection experiments.

Configuration	TinyLlama-1.1B
Candidate-training steps	5000
Batch size	512
Context length	2048
Learning rate	8×10^{-5}
Learning-rate schedule	Constant
Weight decay	0.01
Gradient clipping	1.0
Main retained fraction	0.6

Baselines and implementation. The original pretrained checkpoint is denoted Base, while Base-CT continually pre-trains it on the full candidate pool and serves as the full-retention baseline. Random controls for the reduced token budget by drawing an equally sized subset. We compare three target-agnostic selectors—EL2N (Paul et al., 2021), which prioritizes examples with large prediction error; InfoBatch (Qin et al., 2024), which approximately preserves the expected full-data gradient; and Token-Cleaning (Pang et al., 2025), which applies a noisy-label cleaning criterion—with two target-aware selectors: RHO-1 (Lin et al., 2024), which ranks tokens by excess loss relative to the target distribution, and BLADE (Wang et al., 2026), which performs bi-level optimization for large-scale data selection. For EL2N, InfoBatch, and Token-Cleaning, we adapt their public implementations only as needed for our common data interface and training pipeline.[5] For RHO-1 and BLADE, for which no public implementations are available, we implement the methods following their published descriptions. Method-specific settings otherwise follow the corresponding papers or released implementations, while every selected model is trained with the shared configuration in Table 6.

Evaluation. We report held-out target loss, which is evaluated on the distribution used to define Δ and therefore reflects the train–target gap more directly than downstream accuracy. We additionally report accuracy on eight complementary mathematics benchmarks: GSM8K (Cobbe et al., 2021), MATH (Hendrycks et al., 2021), SVAMP (Patel et al., 2021), ASDiv (Miao et al., 2020), MAWPS (Kedziorski et al., 2016), TabMWP (Lu et al., 2023) for tabular reasoning, MathQA (Amini et al., 2019) for operation-based multiple-choice reasoning and MMLU-STEM (Hendrycks et al., 2020). This evaluation encompasses diverse question types like multiple-choice and open-ended questions.

A.6 COMPARISONS WITH STANDARD DEVIATION

Each real-world LLM training experiment is run 5 times independently. The results, including the standard deviation, are reported in Table Table 7, Table Table 8 and Table Table 9.

[5]Public implementations: EL2N, `https://github.com/mansheej/data_diet`; InfoBatch, `https://github.com/NUS-HPC-AI-Lab/InfoBatch`; and Token-Cleaning, `https://github.com/UCSC-REAL/TokenCleaning`.

Table 7: Comparison of data mixture optimization methods in the no-gap regime ($\Delta = 0.07$). The averaged target perplexity and standard deviation over 5 independent runs are reported.

Methods	Arxiv	Book	C4	CommonCrawl	Github	Stackexchange	Wikipedia	Average
Uniform	10.02±0.11	39.61±0.25	49.67±0.41	42.70±0.38	5.16±0.06	12.25±0.09	20.00±0.22	19.71±0.17
RegMix	10.53±0.34	41.90±1.32	48.67±1.85	41.65±1.59	5.75±0.24	11.73±0.73	19.86±1.19	19.80±0.12
DoGE	10.96±0.22	35.49±0.19	42.42±0.26	36.81±0.28	6.70±0.10	13.97±0.17	23.92±0.30	20.17±0.16
Skill-It	10.64±0.17	38.32±0.21	47.99±0.37	40.62±0.45	5.73±0.07	12.07±0.13	20.78±0.20	19.68±0.16
Aioli	10.09±0.13	39.86±0.35	49.81±0.54	42.83±0.52	5.66±0.04	12.36±0.06	19.67±0.21	19.77±0.15
TANDEM	10.18±0.12	37.89±0.32	46.77±0.40	40.41±0.37	5.89±0.05	12.61±0.12	20.96±0.21	19.66±0.15

Table 8: Comparison of data mixture optimization methods under endogenous ($\Delta = 0.64$)and exogenous ($\Delta = 1.26$) train–target gaps. The averaged target perplexity and standard deviation over 5 independent runs are reported.

	Endogenous Gap								Exogenous Gap
Methods	Arxiv	Book	C4	CommonCrawl	Github	Stackexchange	Wikipedia	Average	CommonCrawl
Uniform	13.14±0.04	57.44±0.73	65.57±0.56	57.93±0.54	8.21±0.04	19.59±0.03	33.90±0.20	28.60±0.11	49.19±0.28
RegMix	13.33±0.07	54.12±0.23	63.74±0.67	56.63±0.70	8.23±0.08	20.03±0.11	33.34±0.19	28.24±0.19	40.63±0.90
DoGE	12.95±0.05	54.82±1.28	62.84±0.63	55.52±0.61	8.13±0.05	18.61±0.08	32.72±0.17	27.62±0.22	40.67±0.44
Skill-It	14.54±0.13	44.24±0.26	48.99±0.31	42.10±0.32	8.33±0.10	21.10±0.19	49.46±0.54	27.38±0.22	41.08±0.17
Aoili	12.80±0.14	51.01±0.37	61.13±0.31	53.51±0.34	8.14±0.05	18.97±0.14	32.63±0.13	27.11±0.16	40.17±0.64
TANDEM	12.60±0.12	49.76±1.56	54.19±0.76	47.24±0.90	8.35±0.05	17.98±0.11	33.51±0.41	26.02±0.22	40.24±0.38

Table 9: Comparison of data selection methods with an in-distribution OpenWebMath target (no gap, $\Delta = 0.016$, upper) and the shifted MetaMath–Mammoth reasoning target (with gap, $\Delta = 0.549$, lower). Results on 8 downstream tasks and the averaged accuracy are reported. Each experiment is carried out 5 times.

	Train Toks.	GSM8K	MATH	SVAMP	ASDiv	MAWPS	TAB	MQA	MMLU-STEM	Avg.
Selection without Gap										
Base	-	2.9	3.2	11.0	18.1	20.4	12.5	14.6	16.1	12.4
Base-CT	5B	4.3±$_{0.5}$	4.3±$_{0.2}$	17.1±$_{0.7}$	29.4±$_{0.6}$	39.4±$_{1.2}$	14.5±$_{2.0}$	8.3±$_{0.6}$	20.4±$_{0.8}$	17.21±$_{0.4}$
Random	3B	4.0±$_{0.3}$	3.9±$_{1.0}$	15.8±$_{1.0}$	29.1±$_{2.3}$	37.2±$_{1.6}$	13.8±$_{0.7}$	9.4±$_{1.1}$	17.8±$_{2.2}$	16.38±$_{0.1}$
EL2N	3B	3.0±$_{0.3}$	3.2±$_{0.5}$	15.5±$_{0.9}$	25.0±$_{1.2}$	31.2±$_{1.1}$	14.4±$_{1.7}$	8.9±$_{1.6}$	18.3±$_{1.2}$	14.94±$_{0.4}$
InfoBatch	3B	3.8±$_{0.4}$	3.3±$_{0.6}$	15.8±$_{1.5}$	25.8±$_{1.5}$	32.9±$_{2.3}$	12.5±$_{1.3}$	7.1±$_{0.9}$	16.5±$_{2.3}$	14.71±$_{0.7}$
Token Cleaning	3B	4.2±$_{0.6}$	3.0±$_{0.9}$	15.4±$_{0.7}$	24.4±$_{0.6}$	31.7±$_{1.0}$	11.7±$_{0.7}$	10.2±$_{1.2}$	14.1±$_{0.9}$	14.3±$_{0.5}$
RHO-1	3B	3.2±$_{0.2}$	3.1±$_{1.3}$	15.3±$_{0.5}$	24.6±$_{0.7}$	31.4±$_{0.5}$	14.5±$_{1.9}$	7.1±$_{2.3}$	16.7±$_{2.1}$	14.49±$_{0.3}$
BLADE	3B	4.5±$_{0.3}$	3.7±$_{0.4}$	18.1±$_{0.6}$	29.9±$_{1.7}$	37.9±$_{1.7}$	16.5±$_{2.1}$	10.4±$_{1.5}$	19.3±$_{2.1}$	17.54±$_{0.8}$
Selection with Gap										
Base	-	2.9	3.2	11.0	18.1	20.4	12.5	14.6	16.1	12.4
Base-CT	5B	4.3±$_{0.5}$	4.3±$_{0.2}$	17.1±$_{0.7}$	29.4±$_{0.6}$	39.4±$_{1.2}$	14.5±$_{2.0}$	8.3±$_{0.6}$	20.4±$_{0.8}$	17.21±$_{0.4}$
Random	3B	4.0±$_{0.3}$	3.9±$_{1.0}$	15.8±$_{1.0}$	29.1±$_{2.3}$	37.2±$_{1.6}$	13.8±$_{0.7}$	9.4±$_{1.1}$	17.8±$_{2.2}$	16.38±$_{0.1}$
EL2N	3B	3.0±$_{0.3}$	3.2±$_{0.5}$	15.5±$_{0.9}$	25.0±$_{1.2}$	31.2±$_{1.1}$	14.4±$_{1.7}$	8.9±$_{1.6}$	18.3±$_{1.2}$	14.94±$_{0.4}$
InfoBatch	3B	3.8±$_{0.4}$	3.3±$_{0.6}$	15.8±$_{1.5}$	25.8±$_{1.5}$	32.9±$_{2.3}$	12.5±$_{1.3}$	7.1±$_{0.9}$	16.5±$_{2.3}$	14.71±$_{0.7}$
Token Cleaning	3B	4.2±$_{0.6}$	3.0±$_{0.9}$	15.4±$_{0.7}$	24.4±$_{0.6}$	31.7±$_{1.0}$	11.7±$_{0.7}$	10.2±$_{1.2}$	14.1±$_{0.9}$	14.3±$_{0.5}$
RHO-1	3B	15.1±$_{1.1}$	5.9±$_{0.6}$	32.6±$_{1.3}$	46.9±$_{0.7}$	63.8±$_{0.5}$	18.7±$_{2.1}$	15.1±$_{1.0}$	18.4±$_{2.5}$	27.06±$_{0.4}$
BLADE	3B	21.2±$_{0.8}$	8.1±$_{1.3}$	37.0±$_{1.3}$	51.4±$_{0.4}$	70.2±$_{1.1}$	21.2±$_{0.2}$	20.6±$_{1.7}$	20.8±$_{0.9}$	31.31±$_{0.6}$

A.7 Limitations and Future Work

While our analytic characterization and empirical results provide a unified, gap-centric account of data curation, several challenges remain open for future research. The limitations of this paper are as follows: (1) Due to limited computational resources, our real-world experiments evaluate models up to 1.1 billion parameters. (2) Our analytic characterization relies on static high-dimensional limits that abstract away LLM training dynamics and data curriculum order. While our theory cleanly distinguishes finite-budget acceleration from genuine advancement, extending these solvable models to capture precise convergence speeds or detailed data consumption trajectories presents an interesting direction for future investigation.

A.8 Notation

Table 10: Summary of the notations used throughout this paper. Variables only used in theoretical analysis are grayed for better readability.

Symbol	Meaning
Data and dimensions	
N	input dimension.
P	number of training examples.
$\alpha = P/N$	total sample budget (samples per dimension). In high-dim limit $N, P \to \infty$, α fixed.
$\alpha_k = w_k \alpha$	sample budget contributed by domain k.
$\boldsymbol{x} \in \mathbb{R}^N$	input data.
$y = \mathrm{sign}(\boldsymbol{B} \cdot \boldsymbol{x})$	label generated by a teacher.
K	number of domains.
$\mathcal{N}(z; 0, 1) = e^{-z^2/2}/\sqrt{2\pi}$	standard Gaussian density.
Teachers, student, model	.
$\boldsymbol{B}_k$	teacher vector of domain k ($\|\boldsymbol{B}_k\| = \sqrt{N}$).
$\boldsymbol{B}_{\text{tar}}$	target teacher.
ϕ	teacher angle (exogenous gap) between $\boldsymbol{B}_k$ and the target.
$\boldsymbol{J}$	student weights = model parameters (replica symbol).
$\boldsymbol{\theta}$	model parameters ($\boldsymbol{\theta} \equiv \boldsymbol{J}$).
ρ	student–target overlap $\boldsymbol{J} \cdot \boldsymbol{B}_{\text{tar}}/N$ in Appendix A.2.
Curation controls	
$\boldsymbol{w}$	curation weights: mixture ratios w_k ($\sum_k w_k = 1$) or per-example keep weights.
$\widehat{\boldsymbol{w}}$	target mixture ratio.
f	retained fraction of the candidate set.
λ_{reg}	regularization coefficient in the mixture model.
λ_{sel}	disagreement penalty in the selection policy.
ν^2	difficulty: nuisance variance orthogonal to a teacher (Appendix A.1)
Order parameters (state evolution / replica)	
$m_k = \boldsymbol{J} \cdot \boldsymbol{B}_k/N$	student–teacher overlap for domain k.
$q = \|\boldsymbol{J}\|^2/N$	student self-overlap.
V, V_k	response term. V_k per domain in the multi-environment case.
$V_k^\star = 1 - m_k^2/Q_k$	teacher-conditional variance in the loss integrals ($Q_1 = q$, $Q_2 = q + \nu^2(q - m_2^2)$).
κ	margin constraint parameter (hard-margin solver).
R	student–training-teacher overlap; in the matched-target case it is also the target overlap.
$\hat{m}, \hat{V}, \hat{q}$	conjugate (hat) overlaps in the SE fixed point.
Performance and diagnostic	
$\mathcal{E}_\alpha(\boldsymbol{w})$	target performance at sample budget α under allocation $\boldsymbol{w}$; for selection we write $\mathcal{E}_\alpha(f; \phi)$.
ε	fixed target-performance level used to define acceleration.
Δ	empirical train–target diagnostic on the baseline model (Section 4)