BEYOND WHAT TO SELECT: PLUG-AND-PLAY OSCILLATORY DATA-VOLUME SCHEDULING FOR EFFICIENT MODEL TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Data selection accelerates training by prioritizing informative training samples, yet existing methods primarily optimize *which samples to select* while keeping *how much data to use* fixed throughout training. As a result, the temporal role of data volume remains largely underexplored, even in dynamic selection. In this work, we find that the instantaneous data volume directly shapes subset-based optimization dynamics, revealing a regularization–fidelity trade-off: lower ratios induce stronger stochastic regularization while reducing computation, whereas higher ratios provide more faithful and stable updates. Therefore, fixing data volume throughout training constrains optimization and can lead to suboptimal performance under a given training budget, motivating a rethink of data selection beyond fixed-ratio training. Building on this insight, we propose PODS, a **P**lug-and-play **O**scillatory **D**ata-volume **S**cheduling framework that alternates low- and high-ratio phases while preserving a prescribed cumulative data budget. Given a target ratio, the schedule is analytically determined without dataset-specific tuning and can be directly integrated with existing static or dynamic selectors. Experimental results across diverse vision and language tasks show that PODS consistently extends the efficiency frontier of fixed-ratio selection, reducing ImageNet-1k training cost by 40% while improving accuracy by 0.4%, and achieving approximately $2\times$ faster LLM instruction tuning without performance degradation.

1 INTRODUCTION

Recently, deep learning has made significant progress, driven in part by increasingly large and complex models trained on ever-growing datasets Kaplan et al. (2020); Falk et al. (2026). While such scaling has improved model capability, it has also led to rapidly rising training costs, raising barriers to large-scale model development and adoption Xia et al. (2023); Naharas et al. (2026); Ye et al. (2023). More importantly, large-scale datasets often contain substantial redundancy Yang et al. (2026), such that training on large datasets incurs considerable computation without proportional performance gains. To this end, data selection offers a promising approach to improving data efficiency by selecting more informative samples, thereby accelerating training without sacrificing performance.

Existing data selection methods can be broadly categorized as static Yang et al. (2023a; 2024); Mindermann et al. (2022); Cho et al. (2025); Xia et al. (2023) or dynamic Qin et al. (2023); Raju et al. (2021); He et al. (2024b). Static methods construct a fixed subset before training, whereas dynamic methods adapt the selected subset as the model state evolves. Despite this distinction, both paradigms predominantly focus on **which samples to select**, typically through increasingly sophisticated importance or difficulty criteria. In contrast, **how much data to select** at each training stage is usually fixed in advance by a prescribed target selection ratio. Consequently, even dynamic selection is generally dynamic in subset composition while fixing the data volume throughout training.

This fixed-volume formulation imposes a hidden rigidity on data selection by implicitly assuming that the same amount of data is equally appropriate across all optimization stages. However, data volume is not merely a computational knob: as our analysis shows, varying it also changes the stochasticity and fidelity of subset-based optimization. Smaller subsets reduce computation and induce stronger stochasticity, which can act as an implicit regularizer, whereas larger subsets provide

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.

more faithful and stable optimization signals at a higher computational cost. Consequently, different ratios induce distinct yet complementary optimization behaviors throughout training. Exploiting this complementarity could break the fixed efficiency-optimization trade-off of conventional data selection and enable greater training acceleration, presenting a promising yet less explored direction. This naturally raises a question: *beyond deciding what to select under a fixed ratio, can we dynamically allocate data volume over training to exploit the complementary benefits of low- and high-ratio optimization regimes?*

To answer this question, we propose **PODS**, a Plug-and-play Oscillatory Data-volume Scheduling framework that treats the selection ratio as a time-varying control variable rather than a fixed budget parameter. The key idea is to replace a single fixed-ratio regime with structured alternation between complementary low- and high-ratio phases while preserving the prescribed average data budget. We show that low-ratio phases reduce computation while exploiting stronger stochastic regularization, whereas intermittent high-ratio phases suppress approximation uncertainty and restore higher-fidelity optimization. Therefore, rather than introducing another sample-scoring rule, PODS operates as a complementary scheduling layer that determines *how much* data to use at each stage, while the underlying selector determines *which* samples constitute the subset. Given a target selection ratio, the scheduling parameters are analytically determined without dataset-specific tuning or auxiliary optimization, resulting in negligible overhead. Meanwhile, we pair PODS with a lightweight loss-based hard-mining strategy, and its decoupled mechanism allows it to be readily integrated with existing static or dynamic selectors without modifying their selection rules, highlighting broad plug-and-play compatibility.

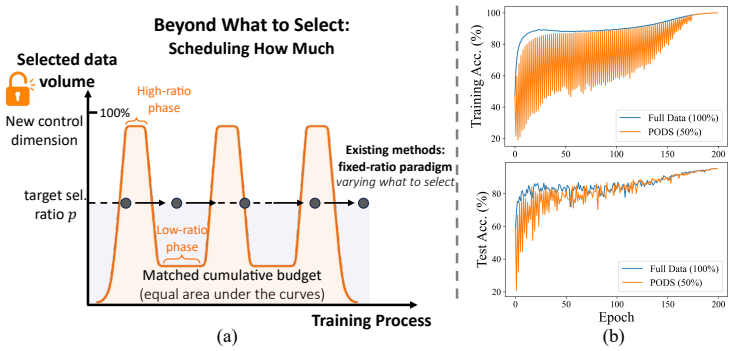

Figure 1: (a) **Oscillatory data-volume scheduling.** Existing data selection methods primarily focus on what to select while keeping the selected data volume fixed. In contrast, PODS introduces data volume as a time-varying control dimension, scheduling how much to select over training under a prescribed cumulative budget. (b) **Training dynamics**. Oscillatory scheduling induces distinct optimization behaviors aligned with low- and high-ratio phases, while test performance progressively improves at only 50% of the full-data training cost.

Extensive experiments on diverse datasets, architectures, and tasks demonstrate the broad practical applicability of PODS for data-efficient learning. In standard, fine-grained, long-tailed, and out-of-distribution recognition settings, PODS consistently preserves or improves full-data performance while substantially reducing training cost, e.g., reducing training cost by 40% while improving accuracy by 0.4% on ImageNet-1k Deng et al. (2009). The benefits further extend beyond recognition to object detection and large-scale LLM fine-tuning; for example, PODS achieves nearly 2× training speedup on Qwen2.5-7B without performance degradation. Notably, these gains hold across heterogeneous architectures, including ResNet He et al. (2016) classifiers, YOLOv8 Yaseen (2024) and RT-DETR Zhao et al. (2024b) detectors, and Qwen2.5/3 Qwen et al. (2025); Yang et al. (2025a) and LLaMA Touvron et al. (2023) language models. Moreover, PODS can be integrated with existing selection methods and further improves their performance, demonstrating strong plug-and-play compatibility. The broad applicability comes at negligible additional cost, adding only 5.4 seconds of overhead over the entire ImageNet-1k training process.

The contributions can be summarized as follows: **(1)** We introduce dynamic data volume as a new temporal dimension of data selection, moving beyond the conventional view that data efficiency is determined solely by *which* samples are selected. **(2)** We establish a theoretical connection between data volume and optimization dynamics, showing that the selection ratio modulates a curvature-weighted stochastic correction and induces a fundamental regularization–fidelity trade-off. **(3)** Building on this insight, we propose PODS, a principled and lightweight oscillatory scheduling framework that alternates complementary low- and high-ratio regimes under a prescribed cumulative data budget for the first time. **(4)** Extensive experiments across 21 vision and language benchmarks

demonstrate that dynamic data-volume scheduling consistently pushes the efficiency frontier beyond fixed-ratio selection across diverse tasks, architectures, and selection methods.

2 Related Work

Existing data selection methods can be broadly categorized into static and dynamic approaches.

Static Data Selection Static data selection constructs a fixed subset according to predefined sample-importance criteria, which can be broadly grouped into geometry-based Xia et al. (2023); Zheng et al. (2023), uncertainty-based Yang et al. (2024); Feldman & Zhang (2020), and optimization-based approaches Yang et al. (2023a); Killamsetty et al. (2021). *Geometry-based approaches* Welling (2009) select representative samples in feature space through clustering, e.g., extending geometric set cover to distribution coverage Zheng et al. (2023), selecting samples near the median score Xia et al. (2023), leveraging neighborhood relationships Maharana et al. (2023), and other geometric coverage under uncertainty metrics Ramalingam et al. (2023); Sener & Savarese (2017). *Difficulty- or uncertainty-based methods* instead evaluate sample importance using prediction errors Paul et al. (2021); Cho et al. (2025), geometric sparsity in feature space Yang et al. (2025c), forgetting events Toneva et al. (2018), generalization loss Mindermann et al. (2022); Tan et al. (2024), and semantic representativeness Yang et al. (2024). Moreover, *optimization-based approaches* formulate selection as a subset search or policy optimization problem. Representative strategies include influence estimation Yang et al. (2023a), bi-level optimization Killamsetty et al. (2021), temporal scoring Zhang et al. (2024), and reinforcement learning Yang et al. (2025b). To further enhance diversity or coverage, some methods incorporate facility location objectives Yang et al. (2023b), submodularity formulations Nohyun et al. (2023); Iyer et al. (2021), or gradient matching criteria Mirzasoleiman et al. (2020).

Dynamic Data Selection Dynamic data selection instead updates the selected subset during training according to the model's evolving state Okanovic et al. (2023); Hassan et al. (2025); Zhou et al. (2026); Wang et al. (2019); Yang et al. (2026). To avoid incurring significant overhead in online training, existing dynamic methods typically rely on lightweight scoring or sampling strategies while keeping the selection ratio fixed. For example, the work Raju et al. (2021) introduces Upper Confidence Bound (UCB) and ϵ-greedy metrics, leveraging uncertainty sampling and moving averages to score and select informative samples dynamically. RCAP Hassan et al. (2025) proposes a class-aware, probabilistic sampling method that prioritizes difficult samples at a class-wise level. Similarly, Infobatch Qin et al. (2023) introduces a soft pruning mechanism to prune less informative samples according to the evolving loss distribution. Dynamic selection has also been extended beyond standard supervised learning, including contrastive self-supervised learning Joshi & Mirzasoleiman (2023), Taylor-approximation-based batch selection Wang et al. (2024a), and joint data selection and augmentation Yang et al. (2025c). Despite their different mechanisms and settings, these methods primarily adapt subset constitution as training evolves, while the selected data volume remains prescribed by a fixed target ratio. Our work introduces data volume as an orthogonal temporal control dimension, enabling the selection ratio itself to vary throughout training and complementing both static and dynamic selection criteria.

3 The Proposed Method

3.1 Problem Formulation: Dynamic Selection Beyond Fixed Ratios

Let $\mathcal{D}$ denote the full training dataset of size N, θ the model parameters, and T the total number of training epochs. We represent data selection at stage t by a subset $S_t \subseteq \mathcal{D}$, whose size is determined by an instantaneous selection ratio $p_t \in (0, 1]$: $|S_t| = \lfloor p_t N \rfloor$. The sequence $p_t{}_{t=1}^{T}$ therefore specifies how the data volume evolves throughout training, while the identities of samples in S_t can be determined by any underlying selection criterion. To ensure a fair comparison with conventional fixed-ratio selection using target ratio p, we constrain the cumulative data usage by:

$$\frac{1}{T}\sum_{t=1}^{T} p_t N \le pN \implies \frac{1}{T}\sum_{t=1}^{T} p_t \le p, \tag{1}$$

Thus, dynamic scheduling redistributes the same prescribed data budget over time without increasing the total number of optimized samples.

3.2 Data Volume as an Implicit Regularizer

We study how the selection ratio affects training dynamics by characterizing subset-based optimization as a stage-wise sample average approximation (SAA) Kleywegt et al. (2002); Wang et al. (2021) of the full empirical objective. Let $L(\theta) = \frac{1}{N}\sum_{i=1}^{N} \ell_i(\theta)$ denote the full-data empirical risk. At stage t, the selected subset S_t, with $|S_t| = \lfloor p_t N \rfloor$, induces the sample-average objective $L_{S_t}(\theta) = 1/|S_t| \sum_{i \in S_t} \ell_i(\theta)$. The selection ratio p_t therefore governs both computational cost and approximation fidelity: larger ratios generally provide a more faithful approximation to the full empirical objective, whereas smaller ratios amplify subset-induced stochasticity.

To characterize this ratio-induced stochasticity while isolating the effect of data volume from subset composition, we consider uniform sampling without replacement and define the finite-population gradient covariance as $C(\theta) = \frac{1}{N-1}\sum_{i=1}^{N} \left(g_i(\theta) - g(\theta)\right)\left(g_i(\theta) - g(\theta)\right)^\top$, where $g_i(\theta) = \nabla \ell_i(\theta)$ is the individual-sample gradient and $g(\theta) = \nabla L(\theta)$ the full-data gradient. We then describe how the selection ratio p_t enters the local optimization dynamics.

Proposition 3.1. ***(Ratio-dependent curvature effect)****. Assume that L is twice differentiable with a Lipschitz-continuous Hessian Li (2019). Consider the local update $\theta_t^+ = \theta_t - \eta_t \widehat{g}_t(\theta_t)$. Then,*

$$\mathbb{E}\left[L\left(\theta_t^+\right) \mid \theta_t\right] = L(\theta_t) - \eta_t \left\|\nabla L(\theta_t)\right\|^2 + \frac{\eta_t^2}{2} \nabla L(\theta_t)^\top H_t \nabla L(\theta_t) + \frac{\eta_t^2}{2N} \lambda(p_t) \operatorname{Tr}\left(H_t C_t\right) + O\left(\eta_t^3\right), \tag{2}$$

where $H_t = \nabla^2 L(\theta_t), C_t = C(\theta_t)$, and $\lambda(p_t) = \frac{1-p_t}{p_t}$.

The proof is provided in Appendix A. The additional term $R(p_t, \theta_t) = \frac{\eta_t^2}{2N} \lambda(p_t) \operatorname{Tr}(H_t C_t)$ arises from subset-induced gradient variability. Since the subset gradient remains unbiased w.r.t. the full-data gradient, p_t does not alter the expected first-order descent direction but instead modulates the second-order correction through $\lambda(p_t) = (1 - p_t)/p_t$. As p_t decreases, this ratio-dependent correction is amplified, revealing an implicit regularization effect induced by reduced data volume. However, this does not imply that smaller ratios are always preferable: persistently training on a very small subset can make optimization increasingly noisy and eventually hurt generalization. Thus, the selection ratio governs a fundamental *regularization–fidelity trade-off*: lower ratios favor stronger stochastic regularization and lower computational cost, whereas higher ratios provide more faithful and stable optimization signals.

This trade-off suggests that no single ratio needs to dominate the entire training process. Instead, data volume can be allocated temporally so that low-ratio training provides efficiency and stronger stochastic regularization, while occasional high-ratio training restores optimization fidelity. Crucially, these complementary regimes can be combined under the same cumulative data budget, enabling higher-fidelity refinement without sacrificing training acceleration.

3.3 Oscillatory Data Selection

Overview. Following the above principle, `PODS` formulates the selection ratio p_t as a time-varying control variable and constructs a periodic schedule that alternates between low- and high-ratio phases under a fixed cumulative budget. The scheduler determines *how much* data is used at each stage, while sample identities are determined independently by the underlying selection criterion. This separation allows the data-volume schedule to operate independently of how sample importance is defined. Moreover, `PODS` uses lightweight loss-based hard mining as the default criterion to prioritize currently under-optimized samples, while the scheduling mechanism itself remains selector-agnostic.

Selection Scheduling. The regularization-fidelity trade-off motivates periodically exposing optimization to both low- and high-ratio regimes. Given a target selection ratio p, we construct each period as $\underbrace{p_{\text{low}}, \ldots, p_{\text{low}}}_{k \text{ epochs}}, p_{\text{high}}$, with period length $\tau = k + 1$. Low-ratio epochs strengthen the ratio-induced stochastic effect while reducing sample usage, whereas the subsequent high-ratio epoch suppresses approximation uncertainty and provides higher-fidelity refinement toward the full-data objective.

To ensure that the cumulative sample usage does not exceed that of fixed-ratio training with target p, each complete period satisfies

$$\frac{k \cdot p_{\text{low}} + p_{\text{high}}}{k+1} \leq p. \tag{3}$$

We parameterize the schedule using a fixed boundary margin ϵ and set

$$p_{\text{high}} = 1 - \epsilon. \tag{4}$$

For $\epsilon < p < p_{\text{high}}$, we jointly determine k and p_{low} by seeking the shortest feasible period while maintaining $p_{\text{low}} \geq \epsilon$. For a given k, exact budget matching gives

$$p_{\text{low}} = \frac{(k+1)p - p_{\text{high}}}{k} = p - \frac{p_{\text{high}} - p}{k}. \tag{5}$$

The boundary constraint therefore requires

$$k \geq \frac{p_{\text{high}} - p}{p - \epsilon}. \tag{6}$$

Accordingly, the schedule is determined by

$$k^* = \left\lceil \frac{p_{\text{high}} - p}{p - \epsilon} \right\rceil, \quad p^*_{\text{low}} = \frac{(k^*+1)p - p_{\text{high}}}{k^*}. \tag{7}$$

Since p_{low} increases monotonically with k, the smallest feasible k^* simultaneously minimizes the interval between high-ratio refinement epochs while maximizing the separation between the low- and high-ratio regimes under the budget and boundary constraint. Each complete period exactly matches the target ratio p. If training terminates within an incomplete period, only low-ratio epochs remain, so the realized cumulative budget can only decrease. In practice, unless otherwise stated, we fix $\epsilon = 0.05$; hence, once p is specified, the entire schedule is determined analytically without dataset-, model-, or selector-specific tuning. As shown in Fig. 2, it empirically verifies the intended temporal modulation predicted by the ratio-dependent factor $\lambda(p_t)$. Low-ratio phases amplify the ratio-induced stochastic effect, whereas high-ratio phases attenuate it and restore higher-fidelity optimization. This shows that oscillatory scheduling actively redistributes the strength of subset-induced stochastic regularization over training.

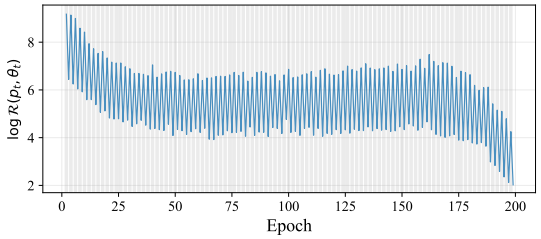

Figure 2: Evolution of the curvature-weighted correction $\mathcal{R}(p_t, \theta_t)$ under a 50% target selection ratio. Gray regions denote low-ratio phases; $\mathcal{R}$ increases during low-ratio phases and decreases during high-ratio phases.

Sample Selection. Given the scheduled ratio $p_t \in \{p^*_{\text{low}}, p_{\text{high}}\}$ specified by the schedule in Eq. (5) and (7), we use lightweight loss-based hard mining as the default selector. At t, let $m_t = \lfloor p_t N \rfloor$. We select the top-$p_t$ fraction of samples $S_t = \text{TopK}_{m_t}\left(\{\ell_i(\theta_t)\}_{i=1}^N\right)$. The selector determines the composition of S_t, whereas PODS independently controls its cardinality over training. Alternative selection metrics can be plugged into the same oscillatory schedule without changing the ratio-control mechanism, enabling PODS to complement existing static and dynamic data selection methods.

4 Experiment

4.1 Experiment Settings

Datasets and Architectures. We evaluate PODS across a wide range of datasets, architectures, and learning paradigms to examine its effectiveness and scalability. For image classification, we use coarse-grained benchmarks including CIFAR-10/100 Krizhevsky et al. (2009), Tiny-ImageNet Chrabaszcz et al. (2017), and ImageNet-1k Deng et al. (2009), and fine-grained benchmarks including FGVC-Aircraft Maji et al. (2013), Oxford-IIIT Pets Parkhi et al. (2012), Stanford Cars Krause et al. (2013), and Oxford Flowers Nilsback & Zisserman (2008). We further evaluate more challenging recognition settings, including long-tailed recognition on ImageNet-LT and

Table 1: Accuracy (%) comparison with the state-of-the-arts. All methods are trained with ResNet-18 on CIFAR-10/100 and ResNet-50 on Tiny-ImageNet. Random* denotes per-epoch random selection under the same selection ratio. Some results are taken from Qin et al. (2023); Cho et al. (2025).

Method	CIFAR-10			CIFAR-100			Tiny-ImageNet		
	30%	50%	70%	30%	50%	70%	30%	50%	70%
Full Dataset		95.6			78.2			45.0	
Random	90.2	92.3	93.9	69.7	72.1	73.8	29.8	37.2	42.2
Herding Welling (2009)	80.1	88.0	92.2	69.6	71.8	73.1	29.4	31.6	39.8
EL2N Paul et al. (2021)	91.6	95.0	95.2	69.5	72.1	77.2	26.6	37.1	44.0
GraNd Paul et al. (2021)	91.2	94.6	95.3	68.8	71.4	74.6	29.7	36.3	43.2
Glister Killamsetty et al. (2021)	90.9	94.0	95.2	70.4	73.2	76.6	30.1	39.5	43.9
Forgetting Toneva et al. (2018)	91.7	94.1	94.7	69.9	73.1	75.3	28.7	33.0	41.2
Moderate-DS Xia et al. (2023)	91.5	94.1	95.2	70.2	73.4	77.3	30.6	38.2	42.8
CLIP-Sel Yang et al. (2024)	91.9	94.5	95.0	70.8	73.7	77.0	31.7	40.0	46.0
Self-sup. proto. Sorscher et al. (2022)	91.0	94.0	95.2	70.0	71.7	76.8	27.7	37.9	43.4
MoSo Tan et al. (2024)	91.1	94.2	95.3	70.9	73.6	77.5	31.2	38.5	43.4
DP Yang et al. (2023a)	90.8	93.8	94.9	-	73.1	77.2	-	-	-
RL-Selector Yang et al. (2025b)	91.8	-	95.4	71.1	-	77.6	31.1	-	44.5
DUAL Cho et al. (2025)	91.8	95.4	95.3	66.4	74.6	77.4	-	-	-
Random*	93.0	94.5	94.8	74.4	75.3	77.3	41.5	42.8	43.1
UCB Raju et al. (2021)	93.9	94.7	95.3	-	75.3	77.3	-	-	-
ϵ-Greedy Raju et al. (2021)	94.1	94.9	95.2	-	74.8	76.4	-	-	-
Dyn. Sel. Aug. Yang et al. (2025c)	94.6	95.1	95.5	75.9	77.6	78.6	41.5	45.1	48.5
InfoBatch Qin et al. (2023)	94.7	95.1	95.6	76.5	78.1	78.2	42.2	43.2	43.8
PODS	**95.1**	**95.4**	**95.9**	**78.4**	**79.1**	**79.4**	**44.5**	**46.9**	**49.0**

Table 2: Results on ImageNet-1k at a 60% cumulative selection ratio using ResNet-50 on an 8-A100 server. We report wall-clock training time (h) and total GPU hours. Glister and CG-Score are not reported due to high computational and memory costs Xia et al. (2023).

Method	Herding	EL2N	GraNd	Forgetting	RL-Selector	SSP	Moderate	CLIP-Sel	DUAL	UCB	Infobatch	Ours	Whole Dataset
Acc. (%)	71.1	72.3	71.0	72.5	73.4	70.0	73.1	73.2	75.5	75.8	76.5	**76.8**	76.4
Time (h)	10.5	10.5	10.5	10.5	10.5	10.5	10.5	10.5	10.5	10.5	10.5	10.5	17.5
Overhead (h)	>17.5	>17.5	>17.5	>17.5	>17.5	>24.0	>17.5	>1.6	>11.6	0.03	0.0028	**0.0015**	0.0
Total GPU Hours	>224.0	>224.0	>224.0	>224.0	>224.0	>276.0	>224.0	>96.8	>176.8	**84.0**	**84.0**	**84.0**	140.0

Places-LT Liu et al. (2019), and OOD generalization on ImageNet-A/O Hendrycks et al. (2021b), ImageNet-R Hendrycks et al. (2021a), and ImageNet-Hard Taesiri et al. (2024). Beyond classification, we apply PODS on MS-COCO Lin et al. (2014) for object detection with YOLOv8 Yaseen (2024) and RT-DETR Zhao et al. (2024b). We also extend its scalability to large-scale LLM instruction tuning by fine-tuning Qwen 2.5/3 Qwen et al. (2025) and LLaMA 3 series models Touvron et al. (2023), covering both general instruction benchmarks such as MMLU Hendrycks et al. (2020), BBH Suzgun et al. (2022), and reasoning-oriented benchmarks including AIME24 Zhang & Math-AI (2024), Olympiad He et al. (2024a), GPQA-Diamond Rein et al. (2023), and MMLU-Pro Wang et al. (2024b). Due to the limited space, detailed implementations are provided in Appendix D.

Baselines. We compare our method with 17 state-of-the-art baselines, including (1) Herding Welling (2009), (2) EL2N Paul et al. (2021), (3) GraNd Paul et al. (2021), (4) Glister Killamsetty et al. (2021), (5) Forgetting Toneva et al. (2018), (6) Moderate-DS Xia et al. (2023), (7) CLIP-Sel Yang et al. (2024), (8) Self-sup. Prototypes Sorscher et al. (2022), (9) MoSo Tan et al. (2024), (10) DP Yang et al. (2023a), (11) RL-Selector Yang et al. (2025b), (12) DUAL Cho et al. (2025), (13) Random*, (14) Dyn. Sel. Aug. Yang et al. (2025c), (15) UCB Raju et al. (2021), (16) ϵ-Greedy Raju et al. (2021), and (17) InfoBatch Qin et al. (2023).

4.2 Comparison with The State-of-the-arts

Performance Comparison. We compare PODS with representative static and dynamic data selection methods on CIFAR-10/100 and Tiny-ImageNet. For PODS, the reported ratio denotes the cumulative data budget. As shown in Table 1, PODS consistently achieves the best performance across datasets and selection ratios. At a 30% selection ratio, PODS outperforms the leading baseline by 1.9% on CIFAR-100 and 2.3% on Tiny-ImageNet. Notably, on CIFAR-100, PODS is the only one that surpasses full-dataset training using only 30% of the data budget.

At larger ratios, PODS further improves upon the full-dataset baseline while accelerating training. These results indicate that dynamically scheduling data volume can improve predictive performance beyond what is achieved by conventional fixed-ratio selection at the same or larger training cost.

Table 3: Long-tailed classification accuracy (%) on ImageNet-LT and Places-LT at a 50% cumulative sample budget. Cost denotes wall-clock training time on a 4-V100 server.

Method	ImageNet-LT				Places-LT				Cost
	Many	Med.	Few	Acc.	Many	Med.	Few	Acc.	
Full Dataset	**43.2**	35.1	18.5	35.6	**44.7**	37.0	25.3	35.9	35.2h
PODS	43.0	**35.5**	**20.8**	**36.3**	43.0	**40.8**	**29.4**	**39.2**	**16.2h**

Table 4: Object detection mAP (%) on MS-COCO using YOLOv8 and RT-DETR.

Method	70%	80%	90%	Full
YOLOv8	38.5	39.1	**39.8**	39.6
RT-DETR	55.9	56.6	**57.6**	57.5

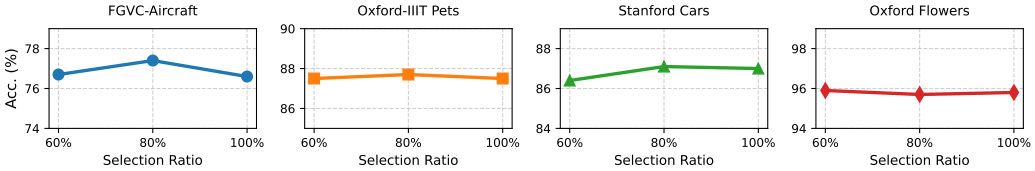

Figure 3: Results on four fine-grained recognition benchmarks using ResNet-50.

Efficiency Comparison. Following Qin et al. (2023), we evaluate accuracy and efficiency on large-scale ImageNet-1k. As shown in Table 2, PODS achieves the best overall accuracy-efficiency trade-off, delivering the highest accuracy while incurring nearly zero additional overhead. Compared with full-data training, PODS improves accuracy by 0.4% and reduces the total cost from 140 to 84 GPU hours, corresponding to a 40% reduction in training cost. The advantage is particularly important at scale. Although many static selection methods reduce the number of training samples, they often require expensive preprocessing, pretraining, or scoring stages, which, in several cases, are even higher than those of full-data training. In contrast, PODS avoids additional pretraining or complex metric computation: it only **adds 5.4 seconds of overhead** over the entire 90-epoch ImageNet-1k training process. Additional results across different selection ratios on ImageNet-1k are reported in Figure 7 in Appendix E, where PODS reduces cost by up to 50%. Together, these results demonstrate that PODS offers a practical accuracy-efficiency trade-off for large-scale training.

4.3 Generalization and Compatibility

Generalization across Diverse Data Distributions. While most existing methods generalize well under specific data distributions, we systematically evaluate PODS's effectiveness beyond standard balanced recognition under substantially different data distributions, including fine-grained categories, severe class imbalance, and distribution shifts. On four *fine-grained benchmarks*, i.e., FGVC-Aircraft, Oxford-IIIT Pets, Stanford Cars, and Oxford Flowers, as shown in Fig. 3, PODS consistently preserves or surpasses full-data performance under reduced sample budgets, showing that oscillatory data-volume scheduling achieves lossless acceleration even when recognition depends on subtle inter-class differences. Moreover, the same advantage holds under severe class imbalance: with only a 50% selection ratio, PODS surpasses full-data training on both *long-tailed recognition benchmarks*, i.e., ImageNet-LT and Places-LT, including a 2.3% improvement in few-shot accuracy on ImageNet-LT. Under *distribution shifts*, models trained with PODS not only reduce training costs but also maintain or even improve generalization in OOD settings using ImageNet-R/O/Hard. As shown in Fig. 4, PODS can accelerate training while maintaining competitive generalization across ratios. Together, these results demonstrate that the superior performance of PODS persists across diverse data distributions, highlighting its broad applicability.

Generalization across Tasks and Models. Most existing data selection methods are developed and evaluated primarily for image classification tasks. PODS extends data-efficient training substantially beyond this setting: its scheduling principle is independent of specific training objectives and model architectures, allowing the same framework to transfer directly across diverse learning scenarios. On MS-COCO object detection, as shown in Table 4, the same scheduling framework achieves competitive or improved performance under reduced training budgets with both YOLOv8 and RT-DETR. These results demonstrate that oscillatory data-volume scheduling transfers directly to modern detection architectures without task-specific redesign.

Beyond vision tasks, we further evaluate PODS on large-scale LLM instruction tuning. Specifically, following Wang et al. (2024a), we fine-tune Qwen-2.5-7B-Instruct and LLaMA-3.2-3B using instruc-

Table 5: LLM instruction-tuning results on BBH and MMLU using Qwen-2.5-7B-Instruct and LLaMA-3.2-3B. Cost is wall-clock GPU hours on an 8-A100 server.

Model	Method	Selection Ratio	BBH	MMLU	Avg.	Δ Avg.	Cost (h)	Speedup
Qwen-2.5-7B-Instruct	Full Data	100%	66.2	74.0	70.1	–	32.8	–
	PODS	50%	66.2	74.0	70.1	0.0	16.9	1.9×
LLaMA-3.2-3B	Full Data	100%	48.4	55.9	52.2	–	17.6	–
	PODS	50%	48.4	56.0	52.2	0.0	9.0	2.0×

Table 6: Reasoning-oriented LLM fine-tuning results using DAPO-MATH-17K Yu et al. (2025). Full-data and PODS use the same number of optimization steps.

Model	Method	Selection Ratio	Mathematical Reasoning		General Capability		Avg.
			AIME24	Olympiad	GPQA-Diamond	MMLU-Pro	
Qwen3-4B	Full Data	100%	50.0	65.7	35.8	61.1	53.2
	PODS	50%	50.0	**66.0**	**35.9**	**61.5**	**53.4**
Qwen3-8B	Full Data	100%	53.3	64.5	36.4	65.8	55.0
	PODS	50%	53.3	**65.1**	**37.4**	**66.4**	**55.6**

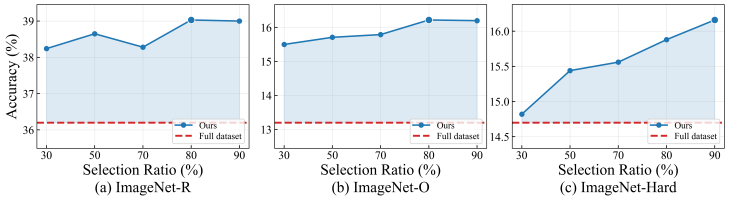

Figure 4: OOD generalization on ImageNet-R/O/Hard. We report AUPR (%) on ImageNet-O and accuracy (%) on the others.

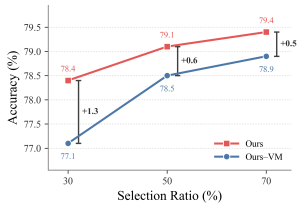

Figure 5: Effect of ratio-induced stochasticity.

tion mixtures from FLAN V2 Longpre et al. (2023), CoT Wei et al. (2022), DOLLY Conover et al. (2023), and OPEN ASSISTANT Köpf et al. (2023), and evaluate on MMLU and BBH. As shown in Table 5, at only a 50% selection ratio, PODS matches full-data performance while reducing LLM SFT costs from 32.8 to 16.9 hours for Qwen-2.5-7B and from 17.6 to 9.0 hours for LLaMA-3.2-3B, yielding **approximately** 2× **acceleration without performance degradation**. We additionally examine reasoning-oriented fine-tuning on Qwen3-4B/8B using DAPO-MATH-17K, where PODS is trained under the same optimization horizon as full-data training. Across AIME24, OlympiadBench, GPQA-Diamond, and MMLU-Pro, PODS consistently achieves the strongest average performance, showing that its advantage is not merely high-efficiency training with fewer optimization steps.

Together, these results validate that PODS generalizes well across fundamentally different tasks, objectives, and model families without changing the underlying scheduling mechanism. This task- and architecture-agnostic design makes PODS a practical plug-and-play scheduling layer that can be integrated into existing training pipelines to accelerate training without sacrificing performance or requiring task-specific redesign.

Compatibility with Existing Selectors. Since PODS decouples data-volume scheduling from the underlying sample-scoring criterion, it can be directly applied on top of existing selection methods without modifying their underlying criteria. We validate this property by applying PODS to representative static and dynamic selectors, including Glister, Moderate, UCB, ϵ-Greedy, and InfoBatch. As shown in Table 7, PODS consistently improves these methods across ratios, with substantial gains of 7.0% and 6.4% for Moderate and Glister at 30%, and 3.6% and 3.7% for UCB and ϵ-Greedy at 50%. These consistent gains show that the effectiveness of PODS is not tied to a particular scoring criterion; instead, **data-volume scheduling provides an orthogonal and reusable control dimension that complements how samples are selected**. Notably, this property enables PODS to systematically enhance diverse existing selectors in a plug-and-play manner with negligible additional overhead.

Table 7: Integration with existing static and dynamic data selection methods. "PODS+X" denotes applying PODS as a data-volume scheduling layer on top of selector X.

Selection	Moderate		Glister		UCB		ϵ-Greedy		InfoBatch	
Ratio	Base	PODS+	Base	PODS+	Base	PODS+	Base	PODS+	Base	PODS+
30%	70.2	**77.2** (+7.0)	70.4	**76.8** (+6.4)	74.0	**77.7** (+3.7)	73.9	**77.3** (+3.4)	76.5	**78.5** (+2.0)
50%	73.4	**78.4** (+5.0)	73.2	**78.7** (+5.5)	75.3	**78.9** (+3.6)	74.8	**78.5** (+3.7)	78.1	**78.8** (+0.7)
70%	77.3	**79.2** (+1.9)	76.6	**79.1** (+2.5)	77.3	**79.3** (+2.0)	76.4	**78.7** (+2.3)	78.2	**79.3** (+1.1)

4.4 Mechanism Analysis and Ablation Study

Effect of Ratio-Induced Stochasticity. Oscillatory data-volume scheduling induces a pronounced phase-dependent stochasticity predicted by our analysis. On CIFAR-10/100, the gradient variance in low-ratio phases reaches 2.0/2.6$\times$ that of high-ratio phases, directly exhibiting the ratio-dependent modulation characterized by $\lambda(p_t)$ in Sec. 3.2.

More importantly, this elevated stochasticity is a functional component of PODS, rather than a by-product of subsampling. We construct PODS-VarMatch to selectively suppress gradient dispersion in low-ratio phases while preserving the first-order mean gradient. Let $\bar{g}_t$ denote the estimated mean gradient, and V_{low} and V_{high} the corresponding gradient variances in the two phases. We transform the low-phase gradient as $\widetilde{g}_t = \bar{g}_t + \sqrt{V_{\text{high}}/V_{\text{low}}}\,(g_t - \bar{g}_t)$, which preserves the first-order mean gradient while reducing the centered gradient dispersion. As shown in Figure 5, suppressing low-phase stochasticity consistently degrades accuracy by 0.5–1.3% across all settings, with larger drops at lower selection ratios where the stochastic effect is strongest. These results identify phase-dependent stochasticity as a key optimization mechanism behind PODS: varying data volume actively regulates gradient dispersion, and the stronger stochasticity induced by low-ratio phases contributes directly to the resulting performance gains.

Effect of Scheduling and Sample Selection. We further disentangle the effect of oscillatory data-volume scheduling and the default loss-based hard mining on Tiny-ImageNet using ResNet-50. As shown in Table 8, each component improves performance independently, while combining them consistently yields the strongest results across selection ratios. This result separates two distinct design dimensions in PODS: oscillatory scheduling controls *how much* data is used over training, whereas hard mining determines *which* samples constitute the selected subset. Importantly, the benefit of scheduling is not tied to the default loss-based criterion. Combined with the consistent gains obtained when PODS is integrated with diverse existing selectors in Table 7, these results establish dynamic data volume as an independent and broadly effective optimization dimension beyond conventional sample scoring. They further suggest that data-efficient learning can benefit not only from improving which samples are selected, but also from explicitly controlling how data volume is allocated over time.

Table 8: Effect of oscillatory scheduling and hard mining.

Osc. Sch.	Hard Min.	30%	50%	70%
		41.5	42.8	43.1
✓		42.0	45.9	47.8
	✓	41.7	45.5	46.1
✓	✓	**44.5**	**46.9**	**49.0**

5 Conclusion

We present PODS, a plug-and-play oscillatory data-volume scheduling framework that extends data selection beyond deciding which samples to select to explicitly scheduling how much data to use over training. Guided by the regularization–fidelity trade-off revealed in our analysis, PODS alternates low-ratio regularization with high-ratio refinement under a prescribed cumulative budget, achieving substantial training acceleration with negligible scheduling overhead. Its effectiveness consistently transfers across diverse tasks, architectures, and existing selectors, demonstrating strong generality and plug-and-play compatibility. To the best of our knowledge, this is the first work to explicitly formulate data volume as a time-varying control dimension in data selection. We hope this work provides the community with a broader view of data selection beyond sample importance and establishes data volume as a complementary temporal design space for data-efficient learning.

Ethics Statement

All datasets used in this work are publicly available benchmarks, including CIFAR-10/100, Tiny-ImageNet, ImageNet-1K, FGVC-Aircraft, Oxford-IIIT Pets, Stanford Cars, Oxford Flowers, ImageNet-LT, Places-LT, ImageNet-A, ImageNet-R, ImageNet-O, ImageNet-Hard, MS-COCO, MMLU, BBH, DAPO-17K, AIME24, and Olympiad. These datasets were obtained from their original public sources and used according to their respective licenses and terms of use. No new data involving human subjects or personally identifiable information were collected or introduced in this work.

Reproducibility statement

We provide detailed implementation settings, hyperparameters, and training protocols for all experiments in the appendix. Complete proofs and additional analyses for the theoretical results are also included. Dataset preprocessing and evaluation protocols follow standard benchmarks and are documented where applicable. We will release the source code.

AI use statement

Generative AI tools were used primarily for language polishing, structural refinement, document organization, and LaTeX assistance. All AI-assisted material was carefully reviewed and verified by the authors, who take full responsibility for the paper's arguments, claims, proofs, citations, results, and final presentation.

References

Yeseul Cho, Baekrok Shin, Changmin Kang, and Chulhee Yun. Lightweight dataset pruning without full training via example difficulty and prediction uncertainty. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=9rLxi2cnZC.

Patryk Chrabaszcz, Ilya Loshchilov, and Frank Hutter. A downsampled variant of imagenet as an alternative to the cifar datasets. *arXiv preprint arXiv:1707.08819*, Aug 2017.

Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. Free dolly: Introducing the world's first truly open instructiontuned llm. 2023.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Sophia Falk, Nicholas Kluge Corrêa, Sasha Luccioni, Lisa Biber-Freudenberger, and Aimee van Wynsberghe. From computation to environmental cost the resource burden of artificial intelligence. *Communications Earth & Environment*, 7(1):397, 2026.

Vitaly Feldman and Chiyuan Zhang. What neural networks memorize and why: Discovering the long tail via influence estimation. *Advances in Neural Information Processing Systems*, 33:2881–2891, 2020.

Atif Hassan, Swanand Khare, and Jiaul H Paik. Rcap: Robust, class-aware, probabilistic dynamic dataset pruning. In *The 41st Conference on Uncertainty in Artificial Intelligence*, 2025.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems, 2024a. URL https://arxiv.org/abs/2402.14008.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 770–778, 2016.

Muyang He, Shuo Yang, Tiejun Huang, and Bo Zhao. Large-scale dataset pruning with dynamic uncertainty. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7713–7722, 2024b.

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, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 8340–8349, 2021a.

Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 15262–15271, 2021b.

Rishabh Iyer, Ninad Khargoankar, Jeff Bilmes, and Himanshu Asanani. Submodular combinatorial information measures with applications in machine learning. In *Algorithmic Learning Theory*, pp. 722–754. PMLR, 2021.

Siddharth Joshi and Baharan Mirzasoleiman. Data-efficient contrastive self-supervised learning: Most beneficial examples for supervised learning contribute the least. In *International conference on machine learning*, pp. 15356–15370. PMLR, 2023.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models, 2020. URL `https://arxiv.org/abs/2001.08361`.

Krishnateja Killamsetty, Durga Sivasubramanian, Ganesh Ramakrishnan, and Rishabh Iyer. Glister: Generalization based data subset selection for efficient and robust learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 8110–8118, 2021.

Anton J Kleywegt, Alexander Shapiro, and Tito Homem-de Mello. The sample average approximation method for stochastic discrete optimization. *SIAM Journal on optimization*, 12(2):479–502, 2002.

Andreas Köpf, Yannic Kilcher, Dimitri Von Rütte, Sotiris Anagnostidis, Zhi Rui Tam, Keith Stevens, Abdullah Barhoum, Duc Nguyen, Oliver Stanley, Richárd Nagyfi, et al. Openassistant conversations-democratizing large language model alignment. *Advances in neural information processing systems*, 36:47669–47681, 2023.

Jonathan Krause, Jia Deng, Michael Stark, and Li Fei-Fei. Collecting a large-scale dataset of fine-grained cars. 2013.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Zhize Li. Ssrgd: Simple stochastic recursive gradient descent for escaping saddle points. *Advances in Neural Information Processing Systems*, 32, 2019.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2537–2546, 2019.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Barret Zoph, Jason Wei, et al. The flan collection: Designing data and methods for effective instruction tuning. In *International conference on machine learning*, pp. 22631–22648. PMLR, 2023.

Adyasha Maharana, Prateek Yadav, and Mohit Bansal. D2 pruning: Message passing for balancing diversity and difficulty in data pruning. *arXiv preprint arXiv:2310.07931*, 2023.

Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

Sören Mindermann, Jan Brauner, Muhammed Razzak, Mrinank Sharma, Andreas Kirsch, Winnie Xu, Benedikt Höltgen, Aidan N. Gomez, Adrien Morisot, Sebastian Farquhar, and Yarin Gal. Prioritized training on points that are learnable, worth learning, and not yet learnt, 2022. URL https://arxiv.org/abs/2206.07137.

Baharan Mirzasoleiman, Jeff Bilmes, and Jure Leskovec. Coresets for data-efficient training of machine learning models. In *International Conference on Machine Learning*, pp. 6950–6960. PMLR, 2020.

Nilay Naharas, Dang Nguyen, Neslihan Bulut, Mohammadhossein Bateni, Vahab Mirrokni, and Baharan Mirzasoleiman. Data selection for fine-tuning vision language models via cross modal alignment trajectories. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=ZptT82VZ4E.

Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, pp. 722–729. IEEE, 2008.

Ki Nohyun, Hoyong Choi, and Hye Won Chung. Data valuation without training of a model. In *The Eleventh International Conference on Learning Representations*, 2023.

Patrik Okanovic, Roger Waleffe, Vasilis Mageirakos, Konstantinos E. Nikolakakis, Amin Karbasi, Dionysis Kalogerias, Nezihe Merve Gürel, and Theodoros Rekatsinas. Repeated random sampling for minimizing the time-to-accuracy of learning, 2023. URL https://arxiv.org/abs/2305.18424.

Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pp. 3498–3505. IEEE, 2012.

Mansheej Paul, Surya Ganguli, and Gintare Karolina Dziugaite. Deep learning on a data diet: Finding important examples early in training. *Advances in Neural Information Processing Systems*, 34: 20596–20607, 2021.

Ziheng Qin, Kai Wang, Zangwei Zheng, Jianyang Gu, Xiangyu Peng, Zhaopan Xu, Daquan Zhou, Lei Shang, Baigui Sun, Xuansong Xie, et al. Infobatch: Lossless training speed up by unbiased dynamic data pruning. *arXiv preprint arXiv:2303.04947*, 2023.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

Ravi S Raju, Kyle Daruwalla, and Mikko Lipasti. Accelerating deep learning with dynamic data pruning. *arXiv preprint arXiv:2111.12621*, 2021.

Srikumar Ramalingam, Pranjal Awasthi, and Sanjiv Kumar. A weighted k-center algorithm for data subset selection. *arXiv preprint arXiv:2312.10602*, 2023.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. Gpqa: A graduate-level google-proof q&a benchmark, 2023. URL https://arxiv.org/abs/2311.12022.

Ranjan Sapkota and Manoj Karkee. Ultralytics yolo evolution: An overview of yolo26, yolo11, yolov8 and yolov5 object detectors for computer vision and pattern recognition. *arXiv preprint arXiv:2510.09653*, 2025.

Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489*, 2017.

Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari S. Morcos. Beyond neural scaling laws: beating power law scaling via data pruning. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V. Le, Ed H. Chi, Denny Zhou, and Jason Wei. Challenging big-bench tasks and whether chain-of-thought can solve them, 2022. URL https://arxiv.org/abs/2210.09261.

Mohammad Reza Taesiri, Giang Nguyen, Sarra Habchi, Cor-Paul Bezemer, and Anh Nguyen. Imagenet-hard: The hardest images remaining from a study of the power of zoom and spatial biases in image classification. *Advances in Neural Information Processing Systems*, 36, 2024.

Haoru Tan, Sitong Wu, Fei Du, Yukang Chen, Zhibin Wang, Fan Wang, and Xiaojuan Qi. Data pruning via moving-one-sample-out. *Advances in Neural Information Processing Systems*, 36, 2024.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J Gordon. An empirical study of example forgetting during deep neural network learning. *arXiv preprint arXiv:1812.05159*, 2018.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Jiachen T. Wang, Tong Wu, Dawn Song, Prateek Mittal, and Ruoxi Jia. GREATS: Online selection of high-quality data for LLM training in every iteration. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024a.

Tian-Zuo Wang, Sheng-Jun Huang, and Zhi-Hua Zhou. Towards identifying causal relation between instances and labels. In *Proceedings of the 19th SIAM International Conference on Data Mining (SDM)*, pp. 289–297, 2019.

Tian-Zuo Wang, Tian Qin, and Zhi-Hua Zhou. Estimating possible causal effects with latent variables via adjustment. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 36308–36335, 2023.

Yafei Wang, Bo Pan, Wei Tu, Peng Liu, Bei Jiang, Chao Gao, Wei Lu, Shangling Jui, and Linglong Kong. Sample average approximation for stochastic optimization with dependent data: Performance guarantees and tractability, 2021. URL https://arxiv.org/abs/2112.05368.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark, 2024b. URL https://arxiv.org/abs/2406.01574.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Max Welling. Herding dynamical weights to learn. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 1121–1128, 2009.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. Less: Selecting influential data for targeted instruction tuning. *arXiv preprint arXiv:2402.04333*, 2024.

Xiaobo Xia, Jiale Liu, Jun Yu, Xu Shen, Bo Han, and Tongliang Liu. Moderate coreset: A universal method of data selection for real-world data-efficient deep learning. In *The Eleventh International Conference on Learning Representations*, 2023.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025a. URL https://arxiv.org/abs/2505.09388.

Shuo Yang, Zeke Xie, Hanyu Peng, Min Xu, Mingming Sun, and Ping Li. Dataset pruning: Reducing training data by examining generalization influence. In *International Conference on Learning Representations*, 2023a.

Suorong Yang, Peng Ye, Wanli Ouyang, Dongzhan Zhou, and Furao Shen. A clip-powered framework for robust and generalizable data selection. *arXiv preprint arXiv:2410.11215*, 2024.

Suorong Yang, Peijia Li, Furao Shen, and Jian Zhao. Rl-selector: Reinforcement learning-guided data selection via redundancy assessment. *arXiv preprint arXiv:2506.21037*, 2025b.

Suorong Yang, Peng Ye, Furao Shen, and Dongzhan Zhou. When dynamic data selection meets data augmentation: Achieving enhanced training acceleration. *arXiv preprint arXiv:2505.03809*, 2025c.

Suorong Yang, Fangjian Su, Hai Gan, Ziqi Ye, Jie Li, Baile Xu, Furao Shen, and Soujanya Poria. Data agent: Learning to select data via end-to-end dynamic optimization, 2026. URL https://arxiv.org/abs/2603.07433.

Yu Yang, Hao Kang, and Baharan Mirzasoleiman. Towards sustainable learning: Coresets for data-efficient deep learning. In *International Conference on Machine Learning*, pp. 39314–39330. PMLR, 2023b.

Muhammad Yaseen. What is yolov8: An in-depth exploration of the internal features of the next-generation object detector, 2024. URL https://arxiv.org/abs/2408.15857.

Peng Ye, Tong He, Shengji Tang, Baopu Li, Tao Chen, Lei Bai, and Wanli Ouyang. Stimulative training++: Go beyond the performance limits of residual networks, 2023. URL https://arxiv.org/abs/2305.02507.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.

Xin Zhang, Jiawei Du, Yunsong Li, Weiying Xie, and Joey Tianyi Zhou. Spanning training progress: Temporal dual-depth scoring (tdds) for enhanced dataset pruning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26223–26232, 2024.

Yifan Zhang and Team Math-AI. American invitational mathematics examination (aime) 2024, 2024.

Yian Zhao, Wenyu Lv, Shangliang Xu, Jinman Wei, Guanzhong Wang, Qingqing Dang, Yi Liu, and Jie Chen. Detrs beat yolos on real-time object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16965–16974, 2024a.

Yian Zhao, Wenyu Lv, Shangliang Xu, Jinman Wei, Guanzhong Wang, Qingqing Dang, Yi Liu, and Jie Chen. Detrs beat yolos on real-time object detection, 2024b. URL https://arxiv.org/abs/2304.08069.

Haizhong Zheng, Rui Liu, Fan Lai, and Atul Prakash. Coverage-centric coreset selection for high pruning rates. In *The Eleventh International Conference on Learning Representations*, 2023.

Yuzhe Zhou, Zhenglin Hua, Haiyun Guo, and Yuheng Jia. Rethinking representativeness and diversity in dynamic data selection. *arXiv preprint arXiv:2603.04981*, 2026.

A PROOF OF PROPOSITION 3.1

Lemma A.1 (Finite-Population Moments of the Subset Gradient)**.** *Let $\mathcal{S}_t$ be sampled uniformly without replacement from $\mathcal{D}$, with $|\mathcal{S}_t| = m_t = p_t N$. Conditioned on the current parameters θ_t, define $g_{t,i} = \nabla_\theta \ell(\theta_t; z_i), g_t = \frac{1}{N}\sum_{i=1}^N g_{t,i}$ and $\widehat{g}_t = \frac{1}{m_t}\sum_{i\in\mathcal{S}_t} g_{t,i}$. Further, define the finite-population gradient covariance as*

$$C_t = \frac{1}{N-1}\sum_{i=1}^{N}(g_{t,i} - g_t)(g_{t,i} - g_t)^\top. \tag{8}$$

Then, $\mathbb{E}[\widehat{g}_t \mid \theta_t] = g_t$, and

$$\mathrm{Cov}[\widehat{g}_t \mid \theta_t] = \frac{N - m_t}{N m_t} C_t = \frac{1 - p_t}{p_t N} C_t. \tag{9}$$

Proof. For each sample $i \in \{1, \dots, N\}$, introduce the inclusion indicator

$$I_i = \mathbb{1}\{i \in \mathcal{S}_t\}.$$

Since $\mathcal{S}_t$ is sampled uniformly from all subsets of size m_t,

$$\mathbb{E}[I_i] = \frac{m_t}{N}.$$

The subset gradient can be written as

$$\widehat{g}_t = \frac{1}{m_t}\sum_{i=1}^{N} I_i g_{t,i}.$$

Therefore,

$$\begin{aligned}\mathbb{E}[\widehat{g}_t \mid \theta_t] &= \frac{1}{m_t}\sum_{i=1}^{N}\mathbb{E}[I_i] g_{t,i} \\ &= \frac{1}{m_t}\sum_{i=1}^{N}\frac{m_t}{N} g_{t,i} \\ &= \frac{1}{N}\sum_{i=1}^{N} g_{t,i} = g_t.\end{aligned} \tag{10}$$

We next derive the covariance. Define the centered per-sample gradients.

$$\delta_{t,i} = g_{t,i} - g_t.$$

By definition,

$$\sum_{i=1}^{N}\delta_{t,i} = 0,$$

and

$$\widehat{g}_t - g_t = \frac{1}{m_t}\sum_{i=1}^{N} I_i \delta_{t,i}.$$

For distinct indices $i \neq j$, uniform sampling without replacement gives

$$\mathbb{E}[I_i I_j] = \frac{m_t(m_t - 1)}{N(N-1)}.$$

Hence,

$$\begin{aligned}\mathrm{Cov}[\widehat{g}_t \mid \theta_t] &= \mathbb{E}\big[(\widehat{g}_t - g_t)(\widehat{g}_t - g_t)^\top \mid \theta_t\big] \\ &= \frac{1}{m_t^2}\mathbb{E}\left[\sum_{i=1}^{N}\sum_{j=1}^{N} I_i I_j \delta_{t,i}\delta_{t,j}^\top\right] \\ &= \frac{1}{m_t^2}\left[\frac{m_t}{N}\sum_{i=1}^{N}\delta_{t,i}\delta_{t,i}^\top + \frac{m_t(m_t-1)}{N(N-1)}\sum_{i\neq j}\delta_{t,i}\delta_{t,j}^\top\right].\end{aligned} \tag{11}$$

Since $\sum_{i=1}^{N} \delta_{t,i} = 0$, we have

$$\sum_{i \neq j} \delta_{t,i}\delta_{t,j}^{\top} = \left(\sum_{i=1}^{N} \delta_{t,i}\right)\left(\sum_{j=1}^{N} \delta_{t,j}\right)^{\top} - \sum_{i=1}^{N} \delta_{t,i}\delta_{t,i}^{\top}$$
$$= -\sum_{i=1}^{N} \delta_{t,i}\delta_{t,i}^{\top}. \tag{12}$$

Substituting this identity into Eq. (11) yields

$$\mathrm{Cov}[\widehat{g}_t \mid \theta_t] = \frac{1}{m_t^2}\left[\frac{m_t}{N} - \frac{m_t(m_t-1)}{N(N-1)}\right]\sum_{i=1}^{N} \delta_{t,i}\delta_{t,i}^{\top}$$
$$= \frac{N-m_t}{m_t N(N-1)}\sum_{i=1}^{N} \delta_{t,i}\delta_{t,i}^{\top}$$
$$= \frac{N-m_t}{Nm_t} C_t. \tag{13}$$

Finally, using $m_t = p_t N$, we obtain

$$\mathrm{Cov}[\widehat{g}_t \mid \theta_t] = \frac{1-p_t}{p_t N} C_t. \tag{14}$$

□

Proof of Proposition 3.1

Proof. Condition on the current parameter θ_t, and consider the local update $\theta_t^{+} = \theta_t - \eta_t \widehat{g}_t$ based on SAA. Let$H_t = \nabla^2 L(\theta_t)$. The second-order Taylor expansion of L around the current parameter state θ_t gives:

$$L\left(\theta_t^{+}\right) = L(\theta_t) - \eta_t g_t^{\top}\widehat{g}_t + \frac{\eta_t^2}{2}\widehat{g}_t^{\top} H_t \widehat{g}_t + r_t, \tag{15}$$

where the remainder satisfies $|r_t| \leq \frac{\rho\eta_t^3}{6}\left\|\hat{g}_t\right\|^3$, and ρ is the Lipschitz constant of the Hessian.

For the second-order term, we utilize the standard trace identity for the expectation of quadratic forms: $\mathbb{E}\left[x^T A x\right] = \mathbb{E}[x]^T A \mathbb{E}[x] + \mathrm{Tr}(A\,\mathrm{Cov}(x))$. Applying this identity, we obtain

$$\mathbb{E}\left[\widehat{g}_t^{\top} H_t \widehat{g}_t \mid \theta_t\right] = g_t^{\top} H_t g_t + \mathrm{Tr}\left(H_t\,\mathrm{Cov}\left[\widehat{g}_t \mid \theta_t\right]\right)$$
$$= g_t^{\top} H_t g_t + \frac{1-p_t}{p_t N}\,\mathrm{Tr}\left(H_t C_t\right), \tag{16}$$

where the second equality follows from Lemma A.1. Substituting these results into the Taylor expansion yields

$$\mathbb{E}\left[L\left(\theta_t^{+}\right) \mid \theta_t\right] = L(\theta_t) - \eta_t \left\|\nabla L(\theta_t)\right\|^2 + \frac{\eta_t^2}{2}\nabla L(\theta_t)^{\top} H_t \nabla L(\theta_t)$$
$$+ \frac{\eta_t^2}{2N}\frac{1-p_t}{p_t}\,\mathrm{Tr}\left(H_t C_t\right) + \mathbb{E}[r_t \mid \theta_t]. \tag{17}$$

It remains to characterize the remainder. Since the dataset is finite, for any fixed θ_t, $G_t = \max_{1\leq i\leq N}\|g_{t,i}\|$ is finite. Moreover, because $\hat{g}_t$ is an average of the selected per-sample gradients, $\|\hat{g}_t\| \leq G_t$. Consequently, $|\mathbb{E}[r_t \mid \theta_t]| \leq \frac{\rho\eta_t^3}{6} G_t^3 = O(\eta_t^3)$. We therefore obtain

$$\mathbb{E}\big[L(\theta_t^{+}) \mid \theta_t\big] = L(\theta_t) - \eta_t\|g_t\|^2 + \frac{\eta_t^2}{2} g_t^{\top} H_t g_t + \frac{\eta_t^2}{2N}\frac{1-p_t}{p_t}\,\mathrm{Tr}(H_t C_t) + O(\eta_t^3). \tag{18}$$

This completes the proof. □

Remark 1 (Regularization-Fidelity Trade-off). Proposition 3.1 reveals that the selection ratio simultaneously controls two complementary effects through the common factor $\lambda(p_t) = \frac{1-p_t}{p_t}$. Specifically, $\lambda(p_t)$ scales both the ratio-dependent curvature correction $R(p_t, \theta_t)$ and the covariance of the sample-average gradient $\mathrm{Cov}(\widehat{g}_t)$. Consequently, decreasing the selection ratio simultaneously strengthens the curvature-weighted stochastic effect while increasing the approximation uncertainty of the subset objective.

Therefore, as widely observed by the empirical results, a very small ratio throughout training is generally undesirable, as the stronger stochastic effect is accompanied by persistently low-fidelity objective approximations. Instead, a more desirable strategy is to periodically recover approximation fidelity using larger ratios while retaining the stronger stochastic effect accumulated during preceding low-ratio stages.

B Further Analysis of the Ratio-Induced Regularization Effect

Recall that the ratio-dependent second-order correction in Proposition 3.1 is

$$\mathcal{R}(p_t, \theta_t) = \frac{\eta_t^2}{2N} \lambda(p_t) G_t, \tag{19}$$

where

$$\lambda(p_t) = \frac{1 - p_t}{p_t}, \qquad G_t = \mathrm{Tr}(H_t C_t), \tag{20}$$

and $C_t = C(\theta_t)$ denotes the finite-population covariance of the per-sample gradients. We first establish conditions under which this correction admits a non-negative regularization interpretation, and then examine the empirical behavior of G_t during training.

Corollary B.1 (Non-Negativity of the Correction)**.** *Following Yang et al. (2023a), where the Hessian is positive semidefinite by assumption, the geometry-dependent term satisfies*

$$G_t = \mathrm{Tr}(H_t C_t) \geq 0. \tag{21}$$

Consequently, for any $p_t \in (0, 1]$,

$$\mathcal{R}(p_t, \theta_t) \geq 0. \tag{22}$$

Moreover,

$$\frac{\partial \mathcal{R}(p_t, \theta_t)}{\partial p_t} = -\frac{\eta_t^2}{2N p_t^2} G_t \leq 0. \tag{23}$$

Therefore, within locally non-negative-curvature regions, the ratio-induced regularization effect increases monotonically as the selection ratio decreases.

Proof. By definition, C_t is a covariance matrix and hence satisfies $C_t \succeq 0$. Under the assumption, we also have $H_t \succeq 0$. It follows that

$$C_t^{1/2} H_t C_t^{1/2} \succeq 0. \tag{24}$$

Using the cyclic invariance of the trace,

$$\mathrm{Tr}(H_t C_t) = \mathrm{Tr}\left(C_t^{1/2} H_t C_t^{1/2}\right) \geq 0. \tag{25}$$

Because

$$\frac{1 - p_t}{p_t} \geq 0 \tag{26}$$

for every $p_t \in (0, 1]$, Eq. (19) implies

$$\mathcal{R}(p_t, \theta_t) \geq 0. \tag{27}$$

Finally,

$$\frac{\partial \lambda(p_t)}{\partial p_t} = -\frac{1}{p_t^2}. \tag{28}$$

which directly yields the stated monotonicity result. □

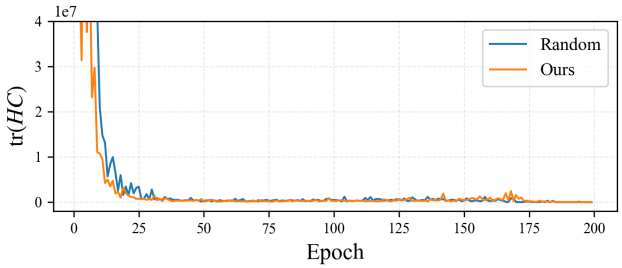

Figure 6: Evolution of the term $\mathrm{Tr}(H(\theta_t)C(\theta_t))$ under random sampling and loss-based hard mining during oscillatory training on CIFAR-10 with ResNet-50.

Empirical Behavior of the Geometry Term. We empirically track the geometry-dependent term $G_t = \mathrm{Tr}(H(\theta_t)C(\theta_t))$ under both uniform random sampling and the loss-based hard-mining selector used by our method. As shown in Figure 6, G_t remains predominantly non-negative throughout training, providing empirical support for the local regularization interpretation in Corollary B.1. In both cases, G_t is relatively large during early training and gradually decreases as optimization progresses.

The two selectors also exhibit broadly similar global trends, which suggests that the selector influences the detailed evolution of the geometry term without fundamentally changing its overall behavior. Accordingly, G_t can be viewed as a geometry-dependent envelope shaped by the optimization trajectory, while the prescribed factor $\lambda(p_t)$ provides explicit and controllable temporal modulation of the resulting correction. These observations clarify the complementary roles of sample selection and data-volume scheduling. The selector determines which examples contribute to the update and may consequently reshape G_t, whereas the oscillatory schedule controls how strongly this geometry-dependent effect is expressed through $\lambda(p_t)$. **The ratio dimension therefore provides an additional mechanism for balancing stochastic regularization and optimization fidelity, even when paired with an informative sample-selection strategy.**

C Algorithm Pipeline

Algorithm 1 Oscillatory Data Selection

Require: Full training dataset $\mathcal{D}$, target selection ratio $p \in (0, 1)$, boundary margin ϵ, the batch size b_s, the number of epochs T, loss function $\mathcal{L}$.

Set $p_{high} \leftarrow 1 - \epsilon$
Compute k^* and p_{low} according to Eq. (7).
Set oscillation period $\tau \leftarrow k^* + 1$
for Epoch $t = 1$ **to** T **do**
  if $t \bmod \tau = 0$ **then**
    $p_t \leftarrow p_{high}$ {High-ratio phase}
  else
    $p_t \leftarrow p^*_{low}$ {Low-ratio phases}
  end if
  Set subset size $m_t \leftarrow \lfloor p_t N \rfloor$
  Select the top-p_t fraction of samples according to the loss scores $S_t \leftarrow \mathrm{TopK}_{m_t}\left(\{s_i^{(t)}\}_{i=1}^N\right)$.
  Set $n_t \leftarrow \lceil m_t / b_s \rceil$
  for each mini-batch $\mathcal{B}_j \subset \mathcal{S}_t$, $j = 1$ **to** n_t **do**
    Compute sample-wise losses $\ell_i(\theta)$ for $i \in \mathcal{B}_j$ and update corresponding scores
    Compute $g_j \leftarrow \frac{1}{|\mathcal{B}_j|} \sum_{i \in \mathcal{B}_j} \nabla_\theta \ell_i(\theta)$
    Update $\theta^+ \leftarrow \theta - \eta_t g_j$
  end for
end for
return Trained parameters θ

D Implementation Details

Image Classification. Following prior data-selection protocols Qin et al. (2023); Yang et al. (2024), we train classification models with standard random cropping and horizontal flipping. Unless otherwise specified, we use SGD/LARS with a batch size of 128, momentum 0.9, weight decay 5×10^{-4}, and a OneCycle learning-rate schedule with cosine annealing. For ImageNet-1k, we use LARS with a batch size of 1024 and a maximum learning rate of 6.4.

Dynamic Selection Baselines and Fair Comparison. We apply the annealing strategy introduced in Qin et al. (2023) to all dynamic data-selection baselines for a consistent training protocol. Because InfoBatch performs soft pruning and therefore processes a varying number of samples over training, we match its computational budget to PODS using the total number of forward passes. For Yang et al. (2025c), we report results under the same standard augmentation pipeline used by the other classification methods. All methods are otherwise trained with matched model architectures, optimization settings, and target selection ratios unless explicitly stated.

Object Detection. For object detection, we follow the default training configurations of the corresponding YOLOv8 Sapkota & Karkee (2025) and RT-DETR Zhao et al. (2024a) implementations, modifying only the data-selection procedure. We keep the detector architecture, optimizer, augmentation pipeline, and training schedule unchanged from their respective default settings.

LLM Fine-tuning. For language-model experiments, we perform parameter-efficient fine-tuning with LoRA following Xia et al. (2024). LoRA adapters are applied to the attention projection modules with rank $r = 8$, scaling factor $\alpha = 64$, and dropout 0.1. We use a maximum sequence length of 2048 tokens, a learning rate of 2×10^{-5}, bfloat16 precision, and gradient checkpointing. All compared data-selection methods use the same fine-tuning configuration and differ only in the samples exposed to optimization.

E More Results on ImageNet-1k

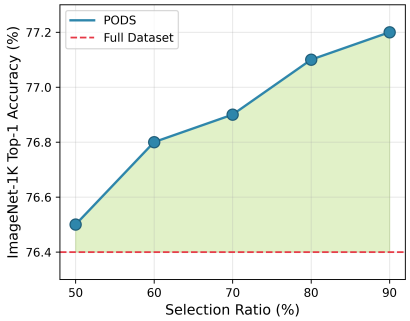

Figure 7: ImageNet-1K Top-1 accuracy of PODS across selection ratios, compared with the full-dataset baseline (76.4, red dashed line). The green-shaded region and outlined markers indicate ratios at which PODS achieves lossless training acceleration.

To further evaluate the performance of PODS on large-scale ImageNet-1k across selection ratios, as shown in Figure 7, we provide accuracy across 50%-90%. Notably, even with a 50% selection ratio, PODS exceeds the full-dataset baseline, bringing a 50% reduction in training costs. When selection ratios increase, the accuracy improves further, reaching 77.2 at a 90% ratio, consistently outperforming the full-dataset performance.

F More Evaluations under Step-Matched Settings

The selected-data training naturally lowers the overall training cost, with the computational budget scaling approximately with the selection ratio. Here, we additionally consider a **step-matched setting**, where each selection ratio is trained for more epochs so that **the total number of optimization steps matches that of full-data training.**

Table 9: Results (mAP, %) on step-matched settings on MS-COCO Lin et al. (2014) using YOLOv8 Yaseen (2024) and RT-DETR Zhao et al. (2024b). For each selection ratio, we train on the selected data with the same number of optimization steps as full-dataset training.

Model	70%	80%	90%	Full Dataset
YOLOv8	**40.3**	40.1	40.0	39.6
RT-DETR	56.3	57.5	**58.1**	57.5

As shown in Table 9, the proposed method further enhances the generalization by a large margin. These results suggest that the benefit of the oscillation selection mechanism is not merely a byproduct of training acceleration or fewer optimization steps. Instead, even when the optimization budget is matched, the proposed method continues to improve the training dynamics and final generalization performance.

As a result, the results indicate that our method has the potential to serve as a lightweight training plug-in that improves standard training performance by dynamically modulating the optimization–generalization trade-off over the course of learning.

G Further Analysis on MMLU

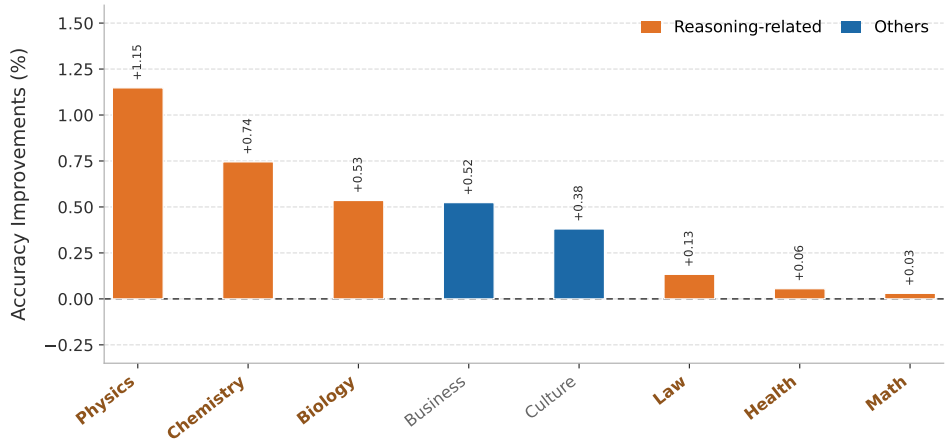

Figure 8: Further analysis of the subject-level improvements on MMLU using Qwen-2.5-7B-Instruct with a selection ratio of 50%.

As shown in Table 5, we show that PODS achieves comparable performance with 2× training acceleration on MMLU. Here, we further conduct an analysis on MMLU. As shown in Figure 8, we visualize the subject groups with positive accuracy improvements over the full-data baseline. Although PODS uses only 50% of the training steps, it still achieves substantial gains on several MMLU subject groups. Interestingly, most of the improved groups are related to structured reasoning or domain-specific inference, such as Physics, Chemistry, Biology, Law, Health, and Math. The largest gains appear in scientific reasoning categories, including Physics, Chemistry, and Biology. Even under a reduced-step setting, PODS can preserve overall performance and may particularly benefit reasoning-relevant or domain-specific training signals under efficient LLM fine-tuning.

H Cross-Optimizer Generalization.

We further evaluate PODS with Adam, LARS, LAMB, and SGD. As shown in Table 10, the gains remain consistent across optimizers. At a 50% selection ratio, PODS matches or slightly exceeds full-data training across optimizers. This indicates that the oscillatory mechanism is not tied to a particular optimizer or to SGD-specific dynamics.

Table 10: Cross-optimizer accuracy (%) on CIFAR-100 using ResNet-18.

Optimizer	30%	40%	50%	Full dataset
Adam	73.5	73.6	**73.9**	73.7
LARS	78.2	78.8	**79.5**	**79.5**
LAMB	75.6	76.0	**76.7**	76.6
SGD	78.4	78.9	**79.4**	78.2

I Additional Results on ImageNet-A

Table 11: Performance on ImageNet-A under different data selection ratios. Results are reported in accuracy (%). **Bold** indicates performance exceeding full-data training.

Selection Ratio (%)	30	50	70	80	90	Full
ImageNet-A Acc. (%)	2.13	2.55	3.04	**3.11**	**3.35**	3.10

J Evaluation of Oscillation Margin ϵ

Table 12: Effect of different ϵ values on C-100 using R-18.

ϵ	30%	50%	70%
0.01	78.5	79.3	79.4
0.05	78.4	79.1	79.4
0.1	78.2	79.0	79.3
0.2	78.1	78.7	79.0

PODS uses a single boundary margin ϵ to parameterize the admissible scheduling range, while the remaining schedule parameters are analytically determined from the target budget p. To assess its sensitivity, we vary ϵ while keeping p fixed. As shown in Table 12, PODS maintains stable performance across a broad range of ϵ, indicating limited sensitivity to this hyperparameter. We observe a modest performance degradation as ϵ increases. Increasing ϵ contracts the admissible ratio range and generally reduces the separation between the low- and high-ratio regimes, thereby weakening the oscillation amplitude induced by $\lambda(p_t)$, which is consistent with the regularization-fidelity interpretation in Sec. 3.2.

K Optimization Dynamics in LLM Fine-tuning

As shown in Figure 9, PODS exhibits a staged optimization trajectory. In the early stage, the loss curves are close, indicating that reduced data exposure does not destabilize optimization. During the middle stage, PODS reaches lower training loss earlier, suggesting that oscillatory data-volume scheduling helps the model focus on more informative samples and improves optimization efficiency. In the late stages, PODS maintains comparable convergence behavior with the full-data baseline. This indicates that our method reshapes the data exposure trajectory and training dynamics rather than merely reducing training cost.

L Discussion and Future Work

In Eq. (7), we provide a tuning-free parameter setting for our selection trajectory. However, if the target selection ratio is extremely low (e.g., 1%), the effective training budget may not satisfy the expected selection ratio. In fact, most data selection methods employ selection ratios above 20% Qin et al. (2023); Toneva et al. (2018); Xia et al. (2023); Yang et al. (2025c; 2023a). This is because aggressively pruning beyond this threshold severely degrades the model performance. Future work

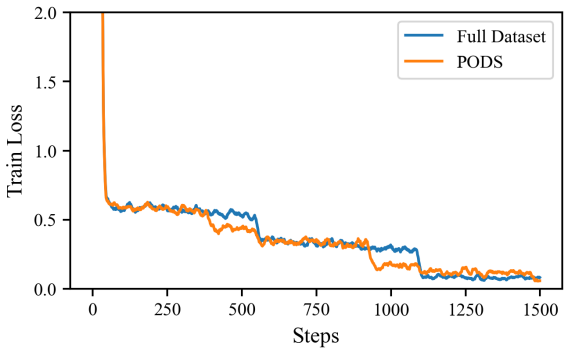

Figure 9: Training loss trajectories under step-matched LLM fine-tuning using Qwen3-8B on DAPO-MATH-17K Yu et al. (2025). PODS uses a 50% selection ratio with the same number of steps as full-data training.

could investigate lossless training acceleration under such ultra-low selection regimes. One possible direction is to relax the margin constraint $p_{low} < p$ and adjust p_{high} and k according to Eq. (3).

Our work focuses on data-volume scheduling with a target selection ratio, rather than determining the optimal selection ratio for each dataset. However, determining the optimal selection ratio remains an open problem in data selection research, as it depends on multiple factors, including dataset complexity, model capacity, label noise, and task difficulty. Future work may explore this direction, which is of importance for both theoretical and practical machine learning.

Similar to many data selection approaches Qin et al. (2023); Mindermann et al. (2022); Killamsetty et al. (2021); Tan et al. (2024); He et al. (2024b), our method mainly works on standard datasets. However, when substantial noise exists, such as label noise or data corruption, the high-ratio phases may inevitably introduce noisy samples. A promising future direction is to combine data-volume scheduling with noise-aware sample scoring or explicit noise detection, allowing the model to restore data coverage while avoiding corrupted or mislabeled samples Wang et al. (2023).

`PODS` is an initial step toward improving training efficiency from the data-volume dimension, beyond "what to select". Currently, we use an offline oscillatory scheduling strategy, where the ratio trajectory is determined before training from the target selection ratio. Although effective and lightweight, this approach offers promising future directions, e.g., adaptive online scheduling techniques to achieve enhanced data-efficient learning.