When to Prune? The Importance of Timing in Data Efficiency Training

Vinicius Yuiti Fukase, Heitor Gama, Barbara Bueno, Lucas Libanio,
Anna Helena Reali Costa, and Artur Jordao
Escola Politécnica, Universidade de São Paulo, Brazil

Abstract. Data pruning techniques aim to reduce training costs while maintaining model predictive performance. However, existing approaches typically apply data pruning heuristics from initialization, neglecting the decisive influence of early epochs on model generalization. This oversight ignores the **critical learning periods**, a phase where data density determines the final generalization capacity of the model. In this work, we address a fundamental question: *Is there a suitable moment to start pruning? If so, how can we identify it?* To confirm and answer the previous questions, we introduce a novel strategy that aligns data reduction with the intrinsic learning dynamics. Unlike state-of-the-art methods that exhibit instability–often re-introducing pruned samples to recover model generalization–our method ensures a stable, monotonic reduction by respecting the critical period. We validate our approach in a broad range of architectures on standard benchmarks. Empirical results demonstrate our approach achieves near-lossless performance, limiting the mean accuracy drop to just **0.09%** (vs. **0.26%** for top-performance methods), while maintaining equivalent training efficiency.

Keywords: Deep Learning · Data Pruning · Critical Periods · Green AI

1 Introduction

Deep learning transforms our modern society by achieving unprecedented performance in complex tasks, such as visual pattern recognition and natural language processing [23, 3]. This success depends on scaling neural network architectures to billions of parameters and training them on massive datasets [43, 23, 3]. However, such trajectory incurs substantial computational costs, restricting deployment in resource-limited settings. As energy consumption and carbon footprints intensify, the community increasingly prioritizes Green AI, seeking methods to align high-performance modeling with environmental sustainability [10, 26].

To address the previous challenges, researchers focus on reducing training costs [19, 26]. In this direction, strategies such as dataset distillation [41, 5], offer potential solutions by reducing training size through synthetic samples; however they frequently incur significant computational overheads or fail to match the accuracy of the full dataset [29]. In contrast, data pruning targets a more efficient balance [30, 42], aiming to achieve lossless generalization with minimal extra cost. By identifying and discarding samples that provide negligible learning

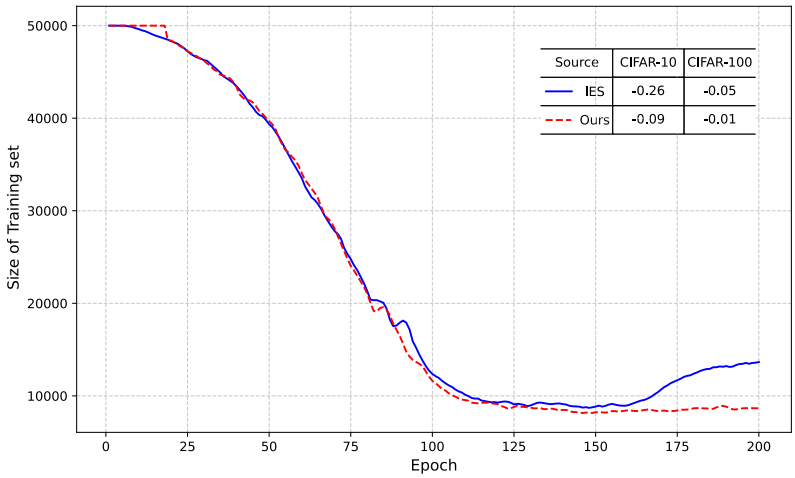

Fig. 1. Impact of data pruning timing on training dynamics. The curves show the training set size evolution for a modern state-of-the-art data pruning method, IES [42] (blue) vs. our method (red) on ResNet18 architecture. Note that IES re-introduces samples in late epochs (e.g., 160-200) to recover predictive performance. The table (top-right) reports the **mean accuracy drop** compared to the full baseline (lower value is better) on stardard benchamarks in data pruning. Our method achieves near-lossless accuracy (**-0.09** vs. -0.26 for IES on CIFAR-10) by preserving data during the early phase (i.e., the critical learning period [16]). We observe the same behavior when applying our method on other state-of-the-art data pruning techniques, such as InfoBatch [30] and random pruning [27].

information, data pruning lowers the computational budget while maintaining the final model quality.

Despite promoting positive results, existing state-of-the-art data pruning methods typically apply pruning heuristics statically, often ignoring the critical early phases of learning dynamics [30, 42]. In particular, these strategies frequently fail to account for the dynamic nature of learning, often removing samples that appear redundant or unimportant simply because the model did not learn them. Consequently, the model lacks vital information required to shape its early optimization path [30, 42]. Figure 1 corroborates this behavior on ResNet18 architecture on CIFAR-10. The figure reveals an instability in the Instance-dependent Early Stopping (IES) [42] method (blue line): in later epochs, it re-introduces previously pruned samples—increasing the training set size again—in an attempt to recover lost predictive performance. Note that this behavior occurs at different epochs (e.g., 91, 160-200).

The previous behavior in training dynamics raises a natural question for the design of an effective data pruning:

When is the suitable moment to start data pruning without compromising learning effectiveness?

Determining this effective onset is crucial; removing data too early risks discarding data before the model captures essential patterns, while pruning too late diminishes the computational gains. Therefore, identifying the boundary between the need of full-data training and the safety of data reduction becomes the central challenge for existing data pruning methods.

To answer the previous issue, we examine the phenomenon of critical learning periods [18, 1]. Prior research confirms that critical periods manifest early in the training process, beyond which numerous training recipes yield minimal to no additional advantage [1]. These initial epochs determine the final connectivity patterns of the model and its capacity to generalize from diverse sources [17]. Hence, information loss during this period becomes irreversible. It turns out that the model cannot recover the *forgotten* patterns even if training continues indefinitely [12]. Despite this established importance, no prior work considers the concept of critical periods to the domain of data pruning, leaving a significant gap between learning theory and efficiency optimization. Figure 1 highlights this behavior, our approach (red dashed line) maintains a consistently compact dataset only after the critical learning phase, avoiding this inefficient *re-learning*. The table on the top-right in Figure 1 summarizes this advantage: while IES suffers a significant mean accuracy drop (e.g., -0.26 on CIFAR-10), our method achieves near-lossless performance (-0.09), providing superior results while preserving training efficiency.

In this work, we fill the previous gap by integrating critical period into data pruning. Specifically, we apply data pruning only after idenfitying the conclusion of this critical phase. For this purpose, we employ a generalization estimation technique to assess how neural networks generalize to unseen data during the initial training phase [11]. We select this methodology specifically for its simplicity and versatility; crucially, it remains agnostic to the underlying dataset, model architecture, and optimization strategy. By relying on such a model- and hyperparameters- agnostic metric, we ensure that our identification of the critical learning period is invariant across diverse training regimes, avoiding the pitfalls of hyperparameter-specific tuning.

Research Statement and Contributions. To sum up, our work has the following research statement. *Once we establish the end of the critical period, we determine the suitable moment to initiate data pruning. By restricting data pruning to this safe zone, the model retains essential information from the start, removing the necessity to re-assimilate unlearned data that is inherent to heuristic approaches.*

Our strategy enhances overall training efficacy and results in a stable optimization trajectory. In summary, our key contributions are:

i) We are the first to investigate the effective onset for data pruning, shifting the focus from what to prune to when to prune, moving beyond the limitation of initiating data pruning before the model captures essential data patterns.

ii) We propose a strategy that explicitly aligns data reduction with critical learning periods, ensuring effective information learning before starting to remove data.
iii) We outperform state-of-the-art methods on standard benchmarks, achieving comparable training time speed but superior stability and higher accuracy. Furthermore, by simply coupling our orthogonal approach to existing techniques (i.e. dynamic random pruning) we push these performance boundaries even further, demonstrating that preserving critical learning periods is a fundamental key to data efficiency.
iv) We achieve significant reductions in training time, energy consumption, and CO_2 emissions by leveraging these strategies. Specifically, experiments demonstrate reductions in training time of popular architectures by up to 42%.

Beyond these practical benefits, our work clarifies essential training dynamics, establishing a new foundation for research into data-efficient learning. To promote reproducibility, we release our code at github.com/ViniFukase/WhenToPrune.

2 Related Work

Data Pruning. Data pruning strategies reduce computational overhead by identifying and retaining a smaller, representative subset of the training data that preserves the predictive performance. These methods predominantly fall into two categories: importance-based and optimization-based. Importance-based approaches assign a scalar relevance score to individual samples, filtering out those redundant or trivial [13, 35, 6]. In contrast, optimization-based techniques select subsets that minimize the distance to the distribution of the full dataset or gradient characteristics [21, 9, 36, 20]. While effective, these methods often incur significant computational costs themselves. Recent work by Okanovic et al. [27] challenges this complexity, demonstrating that well-calibrated random pruning strategies rival sophisticated techniques, highlighting the efficacy of simpler and scalable solutions.

Several pruning frameworks emerge to address specific inefficiencies in standard training. Yuan et al. [42] propose *Instance-dependent Early Stopping* (IES), a method that targets redundant computations at the sample level. Unlike conventional early stopping, IES monitors the second-order differences of individual sample losses. Once this metric stabilizes around zero—indicating the model *mastered* the instance—IES halts backpropagation for that specific sample. It is important to mention that this heuristic allows samples to return to the active training set if they satisfy the condition again. Empirical results show this granular approach reduces backpropagation volume by 10–50% while maintaining or marginally improving accuracy.

To mitigate the potential gradient bias inherent in selective pruning strategies, Qin et al. [30] introduce *InfoBatch*, a framework for unbiased dynamic data pruning. InfoBatch operates by pruning less informative samples based on their loss distribution and subsequently rescaling the gradients of the retained samples to approximate the original full-batch gradient. Also, they achieve lossless

acceleration across diverse tasks, including classification and semantic segmentation. Notably, it demonstrates a 40% cost reduction on standard benchmarks like CIFAR, and up to 10× acceleration for large-scale instruction fine-tuning when combined with additional mechanims.

In contrast to these state-of-the-art data pruning methods, we reconsider the timing of data reduction. While immediate pruning risks discarding critical information during sensitive early learning phases, we explicitly delay any heuristic application until the end of the critical learning period.

Critical Periods. The training dynamics of neural networks reveal that early stages of learning play a decisive role in determining model performance [1, 12, 22]. These stages, named critical periods, represent phases where regularization techniques, such as weight decay and data augmentation, have the most significant impact on generalization [12, 1]. Once these periods pass, persistently applying regularization yields diminishing returns, adding computational overhead without comparable benefits [17].

Recent studies emphasize the importance of understanding and leveraging critical learning periods. For example, Golatkar et al. [12] and Achille et al. [1] suggest that reducing or even removing regularization after initial learning phases lead to more effective training. Recent works explore the role of critical learning periods beyond traditional deep learning. For example, Yan et al. [37, 39, 38] investigate how leveraging critical periods enhance robustness against adversarial attacks and optimize client selection strategies. From the lens of combining sources of information, Kleinman et al. [17] observe that critical periods also impair the ability to synergistically merge data across multiple sources. Although these studies provide insights into critical periods phenomena, mainly from a theoretical perspective, none suggest a systematic method for identifying them in the context of data pruning. Our work fills this gap, representing the first effort to integrate these periods into data pruning strategies.

Importantly, identifying and acting upon critical periods offers a promising direction for reducing computational cost and improving training efficiency [10]. In practical terms, this enables halting training recipes at critical points, achieving comparable or improved accuracy while significantly reducing training time.

3 Preliminaries and Proposed Method

Preliminaries. Let $\mathcal{D} = \{z_i\}_{i=1}^N = \{(x_i, y_i)\}_{i=1}^N$ denote a training dataset consisting of N samples, where x_i represents the input and y_i the target label. The standard goal when learning a deep learning model is to minimize the empirical risk over $\mathcal{D}$ by optimizing the parameters θ of a network $\mathcal{F}(x;\theta)$:

$$\min_{\theta} \mathcal{L}(\mathcal{D};\theta) = \frac{1}{N}\sum_{i=1}^{N} \ell(\mathcal{F}(x_i;\theta), y_i), \tag{1}$$

where $\ell(\cdot)$ is a sample-wise loss function (e.g., Cross-Entropy). From this optimization perspective, data pruning aims to identify a subset $\mathcal{S} \subset \mathcal{D}$, $|\mathcal{S}| \ll N$

such that training $\mathcal{F}(\cdot;\theta)$ on $\mathcal{S}$ maintains predictive ability (i.e., accuracy) comparable to the full dataset $\mathcal{D}$, while significantly reducing the total training duration.

Static Pruning. In this setting, a data pruning algorithm assigns a scoring function $H(z)$ to each sample prior to training. Then, a probability $P(z;H) \in \{0,1\}$ determines pruning decisions. For example, Toneva et al. [34] define the pruning rule as:

$$P(z;H) = \mathbb{1}(H(z) < \bar{H}), \tag{2}$$

where $\bar{H}$ is a pre-defined threshold and $\mathbb{1}(\cdot)$ is the indicator function. The method constructs a static subset $\mathcal{S}$ discarding samples where $P(z;H) = 1$ before optimization (i.e., Eq. 1) begins.

Dynamic Pruning. Unlike static approaches, dynamic pruning selects data adaptively throughout the training process. Here, the importance score H_t varies with the training step t, reflecting the temporal status of the learning process. The probability becomes step-dependent, $P_t(z) = P(z;H_t)$, forming a dynamically evolving subset $\mathcal{S}_t$ at each epoch. Crucially, dynamic pruning leverages the full dataset $\mathcal{D}$ (i.e., subsets) over the course of training and results in a significantly lower gradient expectation bias compared to static methods.

Proposed Method. Standard data pruning (static or dynamic) methods typically initiate data removal from the very first epoch ($t = 0$). While efficient, this strategy neglects the critical learning period [1], a sensitive phase where the model establishes foundational features. Discarding data during this period often leads to irreversible information loss as indicates the behavior of blue curve in Figure 1. To address this, we require a robust numerical indicator to identify effectively *when* this critical phase concludes, ensuring we avoid pruning prematurely.

To this end, we adopt Layer Rotation, originally proposed by Carbonnelle et al. [4] and adapted by Fukase et al. [11], to estimate generalization capability within the critical period. This metric tracks the cosine distance between the weight vector and its initial state. In the context of critical periods, this evolution serves as a proxy for training stability: significant rotation implies the model is actively acquiring essential patterns to generalize, while stabilization signals the conclusion of the critical phase.

Following the refinement by Mason-Williams et al. [24], rather than computing similarity in a layer-wise manner, we concatenate all weights (parameters) of the model $\mathcal{F}$ into a single linearized vector. We denote the initial parameter vector as θ^0 (random initialization) and the vector at training step t as θ^t ($t > 0$).

Using the previous definition, we estimate the Layer Rotation at step t as the cosine distance between the current weights and the initialization:

$$d_{cos} = 1 - \frac{\boldsymbol{\theta^0} \cdot \boldsymbol{\theta^t}}{\|\boldsymbol{\theta^0}\|\|\boldsymbol{\theta^t}\|}. \tag{3}$$

Aligned with Carbonnelle et al. [4], we observe a consistent pattern: larger layer rotations (i.e., increased distance from initialization) reliably predict enhanced generalization performance. Consequently, we employ this metric to define the

boundary of the critical period. This identification enables a strategy of delaying data pruning until the model exhibits sufficient learning stability. Figure 1 highlights the divergent behaviors in late-stage training (e.g. epochs 160-200), where our method successfully avoids the re-introduction of pruned samples; thus corroborating the learning effectiveness.

Algorithm 1 summarizes our strategy. The core mechanism relies on a dynamic conditional check at each epoch: calculating the Layer Rotation d_{cos} to assess model stability. Overall, during the initial phase, when the indicator suggests the model remains within the critical learning period, the algorithm strictly enforces training on the full dataset $\mathcal{D}$, protecting the formation of essential representations. Once the metric surpasses the critical boundary, our strategy transitions to a resource-efficient mode, activating a data pruning technique to reduce the training subset $\mathcal{S}_t$.

Crucially, our method is agnostic to the data pruning technique. Line 9 of Algorithm 1 illustrates it, allowing the integration of any data pruning technique (static or dynamic) once the critical period ends. Furthermore, the identification process simplifies the overall algorithm: upon crossing the critical boundary, the method abandons further checks, allowing training to proceed solely under the chosen pruning regime. For the sake of simplicity, we avoid introduce it details in Algorithm 1. Finally, the metric itself is highly computationally efficient, requiring only the calculation of the cosine distance. This renders the cost of finding the critical period negligible in comparison to the overall optimization process.

Algorithm 1 Overview of the Proposed Method

Require: Model $\mathcal{F}$, Total Epochs T, Dataset $\mathcal{D}$
1: Initialize $\mathcal{F}$ with parameters θ^0
2: Initialize active training data $\mathcal{D}_{active} \leftarrow \mathcal{D}$
3: **for** $t \leftarrow 1$ to T **do**
4: Train $\mathcal{F}$ on $\mathcal{D}_{active}$
5: Update parameters to θ^t (e.g., Eq. 1 or other optimization strategy)
6: Calculate Layer Rotation $d_{cos}(\theta^0, \theta^t)$ using Eq. 3
7: **if** Critical Period Identified (based on d_{cos}) **then**
8: ▷ Switch to resource-efficient mode
9: Apply data pruning to obtain subset $\mathcal{S}_t \subset \mathcal{D}$
10: $\mathcal{D}_{active} \leftarrow \mathcal{S}_t$
11: **else**
12: ▷ Critical Learning Period: Maintain full data
13: $\mathcal{D}_{active} \leftarrow \mathcal{D}$
14: **end if**
15: **end for**
16: **return** Trained Model $\mathcal{F}$

4 Experiments

Experimental Setup. We conduct all experiments using a standard training protocol of 200 epochs and SGD optimizer [12, 18]. The learning rate follows an exponential decay schedule, initializing at 0.1 and applying a multiplicative decay factor of 0.96 at each epoch.

To ensure comprehensive validation, we evaluate our method across diverse architectures, including Residual Networks (ResNets) [14] of varying depths, VGG-16 [33], and DenseNet-121 [15]. We train these models on the standard CIFAR-10 and CIFAR-100 benchmarks. We select these specific settings (model $\times$ dataset configurations) as they represent established common practices in the context of critical periods and data pruning research [30, 12, 17].

Throughout this study, the term **baseline** denotes when the model trains on the full dataset using standard practices, without any pruning intervention or awareness of critical periods. For a fair comparison, we implement the training setup as outlined in the original baseline (paper) configuration [42, 30]. This approach guarantees that any observed performance differences arise directly from our strategy and not from variations in hyperparameters or optimization schedules.

To assess the trade-off between computational efficiency and generalization performance, we introduce two primary metrics:

Normalized Training Cost ($\mathcal{C}_{norm}$). This metric quantifies the computational demand of the training process relative to the baseline. It accounts for dynamic changes in dataset size, particularly the resource savings achieved after applying data pruning. We define the normalized cost as the ratio of the total samples processed by a data pruning method to those processed by the baseline:

$$\mathcal{C}_{norm} = \frac{\sum_{t=1}^{T} |\mathcal{S}_t|}{T \times |\mathcal{D}|}, \tag{4}$$

where $|\mathcal{S}_t|$ is the size of the subset at epoch t, $|\mathcal{D}|$ is the size of the full dataset, and T is the total number of epochs. A lower $\mathcal{C}_{norm}$ indicates greater efficiency.

Accuracy Delta (Δ_{Acc}). This metric measures the impact of data pruning on model generalization. It calculates the difference in accuracy between the pruned model ($\mathcal{A}_{pruned}$) and the baseline model ($\mathcal{A}_{baseline}$):

$$\Delta_{Acc} = \mathcal{A}_{pruned} - \mathcal{A}_{baseline}. \tag{5}$$

A Δ_{Acc} close to zero (or positive) indicates that the method preserves (or improves) model generalization (i.e., accuracy), while a negative value quantifies the degradation caused by removing training data. Finally, we emphasize the use of accuracy as our primary evaluation metric, as it remains the standard benchmark within state-of-the-art data pruning literature [42, 30, 40].

Comparison with State-of-the-Art methods. Existing data pruning methods provide positive results in reducing computational costs during the training of deep models [30, 6]. However, to the best of our knowledge, no method currently takes into account the critical period phenomenon. In this experiment,

we investigate the behavior of applying data pruning *only after* our method identifies the conclusion of critical periods and compare the results with state-of-the-art data pruning methods.

Comparison with IES Method. To demonstrate the efficacy of our strategy, we leverage the Instance-dependent Early Stopping (IES) [42] technique as a testbed. By integrating our critical period identification with IES, we isolate the impact of delaying data removal until achieving the learning stability. This combination allows us to assess whether respecting the critical period mitigate the risks associated with early, heuristic-based pruning while preserving the efficiency gains inherent to the IES approach.

To this end, we evaluate the impact of combining our critical period strategy with the IES technique [42] across six distinct architectures. Table 1 summarizes the results and demonstrates that applying pruning only after the critical period (*Ours*) allows maintaining accuracy close to or even higher than that obtained with pruning throughout the entire training (*Original*, IES [42]). For example, for the ResNet18 architecture on CIFAR-10, our approach achieves 94.89% accuracy, surpassing the *Original* IES [42] scenario (94.69%) and coming very close to the *Baseline* (95.04%). Notably, in cases like VGG16 and DenseNet121, our approach even surpasses the baseline, suggesting better data allocation by providing more samples at times of higher learning rates. Such results reinforce that initial phases represent the most delicate periods for training [12].

Figure 1 highlights the divergent behaviors in late-stage training (epochs 160–200), where our method successfully avoids the re-introduction of previously pruned samples. The original IES brings samples back into the training set to recover lost accuracy, a symptom of pruning too early. In contrast, our approach maintains a consistently compact dataset throughout this phase, avoiding this inefficient re-learning behavior precisely because it respects the critical period before initiating reduction.

Comparison with InfoBatch Method. We perform a similar analysis with the InfoBatch technique [30], testing on ResNet18, ResNet50, and ResNet101 architectures. The results in Table 2 substantiate our earlier conclusions regarding IES. Our approach consistently positions itself as an effective balance

Table 1. Accuracy comparison (Acc) for IES [42] (i.e., using IES as the data pruning technique in Line 9 of Algorithm 1). Original denotes the standard IES method, Ours refers to IES applied only after the critical period, and Baseline represents training on the full dataset without pruning.

Bold and underline indicate the best and second-best results, respectively.

	Dataset	CIFAR-10			CIFAR-100		
	Architecture	ResNet18	ResNet50	VGG16	ResNet34	ResNet101	DenseNet121
IES	Original	0.9469	0.9464	0.9312	0.7747	<u>0.7799</u>	<u>0.7912</u>
	Ours	<u>0.9489</u>	<u>0.9474</u>	**0.9331**	**0.7754**	0.7788	**0.7930**
	Baseline	**0.9504**	**0.9488**	<u>0.9329</u>	<u>0.7752</u>	**0.7815**	0.7907

Table 2. Accuracy comparison (Acc) for InfoBatch [30] (i.e., using InfoBatch as the data pruning technique in Line 9 of Algorithm 1). Original denotes the standard InfoBatch method, Ours refers to InfoBatch applied only after the critical period, and Baseline represents training on the full dataset without pruning.

Bold and underline indicate the best and second-best results, respectively.

	Dataset	CIFAR-10			CIFAR-100		
	Architecture	ResNet18	ResNet50	ResNet101	ResNet18	ResNet50	ResNet101
Info Batch	Original	0.9484	0.9400	0.9438	0.7796	<u>0.7701</u>	0.7788
	Ours	<u>0.9501</u>	<u>0.9404</u>	<u>0.9451</u>	<u>0.7830</u>	0.7687	<u>0.7830</u>
	Baseline	**0.9534**	**0.9495**	**0.9513**	**0.7866**	**0.7777**	**0.7941**

point, with intermediate performance between the *Original*, InfoBatch [30], and the *Baseline*. This highlights that respecting the critical period is fundamental to preserving the predictive capacity when applying data pruning techniques. Specifically, our method consistently outperforms the original InfoBatch configuration across all tested scenarios. For example, on CIFAR-10 on ResNet18, our approach increases accuracy to **95.01%**, surpassing the original InfoBatch result of 94.84% and narrowing the gap to the baseline (95.34%). Similarly, on CIFAR-100, we observe consistent gains, such as improving ResNet18 performance from 77.96% to **78.30%**. Our approach consistently positions itself as an effective balance point, delivering intermediate performance between the *Original*, InfoBatch [30], and the *Baseline*. This highlights that respecting the critical period is fundamental to preserve the predictive capacity of the model when applying data pruning techniques.

Apart from the comparison with IES and InfoBatch, we also assess our method with other state-of-the-art methods. Table 3 introduces the results and confirms that our method outperforms all static pruning techniques across both datasets and pruning ratios, consistently demonstrating lower accuracy degradation. Against dynamic methods, our approach remains highly competitive, matching the performance of the state-of-the-art InfoBatch method [30]. For example, on CIFAR-10 with a 50% pruning ratio, our method incurs a minimal accuracy drop of just 0.3 percentage points (pp). Moreover, on CIFAR-100 with a 30% pruning ratio, our approach yields a notable accuracy gain of 0.2 pp, highlighting its robustness and consistency across methods and datasets. These results underscore the critical role of timing in data pruning: by delaying reduction mechanisms until after the critical period concludes, we enable the model to leverage the full dataset during its most effective learning phase, thereby optimizing performance while minimizing trade-offs.

The previous experiments confirms our research statement that current state-of-the-art methods typically neglect critical learning periods. By successfully addressing this oversight, we fill a significant gap in the literature, demonstrating that respecting these initial phases is essential for maximizing data pruning efficiency without compromising model accuracy.

Table 3. Comparison of accuracy delta (ΔAcc) with state-of-the-art data pruning methods. We report results on **ResNet18** as it represents the most common architecture in data pruning literature. Bold, underline, ↑ and ↓ mean the best results, second-best results, accuracy improvement and accuracy decrease, respectively.

	Dataset	CIFAR-10			CIFAR-100		
	Pruning Ratio %	30	50	70	30	50	70
Static	Random	↓1.0	↓2.3	↓5.4	↓4.4	↓6.1	↓8.5
	CD [2]	↓0.6	↓1.3	↓4.8	↓4.0	↓5.9	↓7.9
	K-Center [32]	↓0.9	↓1.7	↓4.7	↓4.1	↓6.0	↓8.0
	Least Confidence [7]	↓0.6	↓1.1	↓5.3	↓4.0	↓5.9	↓8.4
	Margin [7]	↓0.7	↓1.3	↓4.7	↓4.2	↓6.0	↓8.0
	Forgetting [34]	↓0.9	↓1.5	↓3.9	↓2.9	↓5.1	↓8.3
	GraNd-4 [28]	↓<u>0.3</u>	↓1.0	↓4.4	↓3.6	↓6.8	↓9.4
	DeepFool [8]	↓0.5	↓1.5	↓5.6	↓4.0	↓5.0	↓6.4
	Craig [25]	↓0.8	↓3.3	↓7.2	↓3.8	↓6.3	↓8.5
	Glister [16]	↓0.4	↓1.6	↓4.7	↓3.6	↓5.0	↓7.8
	EL2N-20 [34]	↓<u>0.3</u>	↓0.5	↓3.7	↓1.0	↓6.1	-
	DP [40]	↓0.7	↓1.8	↓4.8	↓1.0	↓5.1	-
Dynamic	Random [27]	↓0.8	↓1.1	↓2.6	↓0.9	↓2.9	-
	ϵ-greedy [31]	↓0.4	↓0.7	↓<u>1.5</u>	↓1.8	↓3.4	-
	UCB [31]	↓<u>0.3</u>	↓0.9	↓1.7	↓0.9	↓2.9	-
	InfoBatch [30]	↑**0.0**	↓<u>0.5</u>	↓**0.9**	↑<u>0.0</u>	↓**0.1**	↓**1.7**
	Random+Ours	↑**0.0**	↓**0.3**	↓1.6	↑**0.2**	↓<u>1.4</u>	↓<u>1.8</u>

Effectiveness on Random Data Pruning. Recent study confirms that random data pruning is surprisingly effective, often matching or outperforming more complex selection criteria when calibrated correctly [27]. Building on this insight, we aim to verify the performance of our method when coupling with this data pruning strategy, isolating the specific contribution of our timing strategy when applied to this stochastic baseline, further clarifying how respecting critical periods enhances even the most elementary pruning mechanisms. By applying our timing-aware approach to random selection, we seek to demonstrate that respecting the critical learning period enhances even the simplest data pruning heuristics. For this purpose, we analyze its performance on ResNet18 trained on CIFAR-10, applying random data pruning at the standard ratios of 30%, 50%, and 70%. Crucially, we implement a **dynamic random pruning** strategy [27]: at each epoch, we select a new random subset of samples, ensuring that the specific data points vary throughout the training process. Our objective is to demonstrate that the method remains robust even when the data selection mechanism is stochastic.

Table 4 presents the Accuracy Delta and time savings achieved when applying our method. The results confirm that our approach improves accuracy while maintaining competitive reductions in computational costs, regardless of the pruning ratio. Specifically, at a 70% pruning rate, we increase accuracy by 1.14

Table 4. Model predictive performance comparison on ResNet18/CIFAR-10 with random pruning. Δ_{Acc} (pp) denotes the percentage point difference from the baseline (higher is better).

Method	Pruning Ratio	Δ_{Acc} (pp)	$\mathcal{C}_{norm}$
Random	30%	-0.88	0.727
Ours + Random	30%	-0.56	0.754
Random	50%	-0.30	0.542
Ours + Random	50%	-0.24	0.579
Random	70%	-2.05	0.355
Ours + Random	70%	-0.91	0.409

(pp) while incurring only a marginal 5.39% increase in training time compared to standard random pruning, primarily due to the removal of samples from epoch zero. It is interesting to note that even with the marginal increase in training time compared to random pruning (due to the full-data initial phase), this approach remains significantly faster than top-performance methods such as IES [42] and InfoBatch [30], offering a superior balance of speed and accuracy.

Computational Benefits and GreenAI. Existing works establish that modern deep learning models emit high levels of carbon dioxide (CO_2) due to their substantial processing capacity and energy requirements during training [19, 10, 26]. Our strategy achieves significant efficiency by belonging to the data pruning domain, inherently lowering the cost of training hence reducing CO_2 emissions.

Specifically, applying our method with a pruning ratio of 50% (after the critical period) results in an average reduction of **33.79%** in resource consumption compared to the full dataset. This economy translates into a significant decrease in CO_2 emissions associated with training. To measure the environmental impacts, we estimate them using the Machine Learning Impact Calculator [19]. For popular architectures, we achieve comparable results in both CO_2 savings and financial costs. To summarize, our efforts yield significant advancements in Green AI by effectively lowering the carbon footprint and enhancing the financial accessibility of deep learning models without compromising generalization.

5 Conclusions

In this work, we address a fundamental limitation in existing data pruning strategies: the uniform application of reduction heuristics that neglect the decisive influence of early training epochs. By formalizing the concept of Critical Learning Periods within the pruning domain, we demonstrate that the timing of data removal is just as critical as the selection criteria itself. Our method leverages a generalization estimator to pinpoint a potential end of this critical phase. By doing so, it ensures that the model establishes robust foundational features before discarding any data. Importantly, this approach resolves the instability in state-of-the-art methods like IES and InfoBatch, eliminating the need to re-introduce

pruned samples to recover model predictive performance and guaranteeing a stable, monotonic reduction in training complexity.

Empirically, our strategy sets a new benchmark for efficient training, achieving near-lossless generalization (i.e., accuracy) across diverse architectures on standard benchmarks. By delaying pruning until the critical period closes, we not only preserve the predictive capacity of the model—limiting accuracy drops to a negligible 0.09%—but also unlock significant computational savings, reducing resource consumption by over 33%. These results directly contribute towards Green AI, offering a practical, scalable solution that lowers the carbon footprint of deep learning without compromising the quality of the final model. We hope these findings encourage future research to prioritize temporal dynamics in the design of efficient optimization algorithms.

While our results confirm the efficacy of critical periods in standard data pruning settings, several open questions remain. A promising avenue for future research is extending this framework to large-scale foundation models, such as Large Language Models (LLMs) and multimodal architectures. Given the astronomical costs associated with training these systems, identifying their critical learning periods could yield disproportionately large efficiency gains. Furthermore, while Layer Rotation demonstrates as a effective metric to identify critical periods, investigating alternative generalization metrics could refine the identification process. Exploring metrics that capture more granular dynamics of training might allow for even earlier, yet safe, pruning interventions, pushing the boundaries of data efficiency further.

Acknowledgments

This study was financed, in part, by the São Paulo Research Foundation (FAPESP), Brasil. Process Number #2023/11163-0. This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001. The authors would like to thank grant #402734/2023-8, National Council for Scientific and Technological Development (CNPq). Anna H. Reali Costa would like to thank grant #312360/2023-1 CNPq.

References

1. Achille, A., Rovere, M., Soatto, S.: Critical learning periods in deep networks. In: International Conference on Learning Representations (2019)
2. Agarwal, S., Arora, H., Anand, S., Arora, C.: Contextual diversity for active learning. In: European Conference on Computer Vision (2020)
3. Bengio, Y., et al.: International AI safety report (2025)
4. Carbonnelle, S., Vleeschouwer, C.D.: Layer rotation: a surprisingly simple indicator of generalization in deep networks? In: International Conference on Machine Learning (2019)
5. Cazenavette, G., Wang, T., Torralba, A., Efros, A.A., Zhu, J.Y.: Dataset distillation by matching training trajectories. Computer Vision and Pattern Recognition (2022)

6. Choi, H., Ki, N., Chung, H.W.: BWS: best window selection based on sample scores for data pruning across broad ranges. In: International Conference on Machine Learning (2024)
7. Coleman, C., Yeh, C., Mussmann, S., Mirzasoleiman, B., Bailis, P., Liang, P., Leskovec, J., Zaharia, M.: Selection via proxy: Efficient data selection for deep learning. International Conference on Learning Representations (2019)
8. Ducoffe, M., Precioso, F.: Adversarial active learning for deep networks: a margin based approach. International Conference on Machine Learning (2018)
9. Engstrom, L., Feldmann, A., Madry, A.: Dsdm: Model-aware dataset selection with datamodels. In: International Conference on Machine Learning (2024)
10. Faiz, A., Kaneda, S., Wang, R., Osi, R.C., Sharma, P., Chen, F., Jiang, L.: Llmcarbon: Modeling the end-to-end carbon footprint of large language models. In: International Conference on Learning Representations (2024)
11. Fukase, V.Y., Gama, H., Bueno, B., Libanio, L., Reali Costa, A.H., Jordao, A.: One period to rule them all: Identifying critical learning periods in deep networks. Procedia Computer Science (2025)
12. Golatkar, A., Achille, A., Soatto, S.: Time matters in regularizing deep networks: Weight decay and data augmentation affect early learning dynamics, matter little near convergence. In: Neural Information Processing Systems (2019)
13. Han, X., Simig, D., Mihaylov, T., Tsvetkov, Y., Celikyilmaz, A., Wang, T.: Understanding in-context learning via supportive pretraining data. In: Association for Computational Linguistics (2023)
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Computer Vision and Pattern Recognition (2016)
15. Huang, G., Liu, Z., van der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Computer Vision and Pattern Recognition (2017)
16. Killamsetty, K., Sivasubramanian, D., Ramakrishnan, G., Iyer, R.: Glister: Generalization based data subset selection for efficient and robust learning. In: Proceedings of the AAAI conference on artificial intelligence (2021)
17. Kleinman, M., Achille, A., Soatto, S.: Critical learning periods for multisensory integration in deep networks. In: Computer Vision and Pattern Recognition (2023)
18. Kleinman, M., Achille, A., Soatto, S.: Critical learning periods emerge even in deep linear networks. In: International Conference on Learning Representations (2024)
19. Lacoste, A., Luccioni, A., Schmidt, V., Dandres, T.: Quantifying the carbon emissions of machine learning. In: Neural Information Processing Systems (2019)
20. Li, Z., Wu, T., Tan, J., Zhang, M., Wang, J., Lin, D.: IDIV: Intrinsic decomposition for arbitrary number of input views and illuminations. In: International Conference on Learning Representations (2025)
21. Mahabadi, S., Trajanovski, S.: Core-sets for fair and diverse data summarization. In: Neural Information Processing Systems (2023)
22. Maini, P., Mozer, M.C., Sedghi, H., Lipton, Z.C., Kolter, J.Z., Zhang, C.: Can neural network memorization be localized? In: International Conference on Machine Learning (2023)
23. Maslej, N., et al.: Artificial intelligence index report (2025)
24. Mason-Williams, G., Dahlqvist, F.: What makes a good prune? maximal unstructured pruning for maximal cosine similarity. In: International Conference on Learning Representations (2024)
25. Mirzasoleiman, B., Bilmes, J., Leskovec, J.: Coresets for data-efficient training of machine learning models. In: International Conference on Machine Learning (2020)

26. Morrison, J., Na, C., Fernandez, J., Dettmers, T., Strubell, E., Dodge, J.: Holistically evaluating the environmental impact of creating language models. In: International Conference on Learning Representations (2025)
27. Okanovic, P., Waleffe, R., Mageirakos, V., Nikolakakis, K.E., Karbasi, A., Kalogerias, D.S., Gürel, N.M., Rekatsinas, T.: Repeated random sampling for minimizing the time-to-accuracy of learning. In: International Conference on Learning Representations (2024)
28. Paul, M., Ganguli, S., Dziugaite, G.K.: Deep learning on a data diet: Finding important examples early in training. Advances in neural information processing systems (2021)
29. Pons, I., Stern, G.B., Costa, A.H.R., Jordao, A.: Enhancing distilled datasets via natural data mixing. In: Conference on Graphics, Patterns and Images (2025)
30. Qin, Z., Wang, K., Zheng, Z., Gu, J., Peng, X., Xu, Z., Zhou, D., Shang, L., Sun, B., Xie, X., You, Y.: Infobatch: Lossless training speed up by unbiased dynamic data pruning. In: International Conference on Learning Representations (2024)
31. Raju, R.S., Daruwalla, K., Lipasti, M.: Accelerating deep learning with dynamic data pruning. arXiv (2021)
32. Sener, O., Savarese, S.: Active learning for convolutional neural networks: A core-set approach. International Conference on Learning Representations (2018)
33. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. Computer Vision and Pattern Recognition (2014)
34. Toneva, M., Sordoni, A., Combes, R.T.d., Trischler, A., Bengio, Y., Gordon, G.J.: An empirical study of example forgetting during deep neural network learning. International Conference on Learning Representations (2019)
35. Xia, M., Malladi, S., Gururangan, S., Arora, S., Chen, D.: LESS: selecting influential data for targeted instruction tuning. In: International Conference on Machine Learning (2024)
36. Xiao, G., Tang, J., Zuo, J., junxian guo, Yang, S., Tang, H., Fu, Y., Han, S.: Duoattention: Efficient long-context LLM inference with retrieval and streaming heads. In: International Conference on Learning Representations (2025)
37. Yan, G., Wang, H., Li, J.: Seizing critical learning periods in federated learning. In: Association for the Advancement of Artificial Intelligence (2022)
38. Yan, G., Wang, H., Yuan, X., Li, J.: Criticalfl: A critical learning periods augmented client selection framework for efficient federated learning. In: Knowledge Discovery and Data Mining (2023)
39. Yan, G., Wang, H., Yuan, X., Li, J.: Defl: Defending against model poisoning attacks in federated learning via critical learning periods awareness. In: Association for the Advancement of Artificial Intelligence (2023)
40. Yang, S., Xie, Z., Peng, H., Xu, M., Sun, M., Li, P.: Dataset pruning: Reducing training data by examining generalization influence. International Conference on Learning Representations (2022)
41. Yu, R., Liu, S., Wang, X.: Dataset distillation: A comprehensive review. IEEE Transactions on Pattern Analysis and Machine Intelligence (2023)
42. Yuan, S., Lin, R., Feng, L., Han, B., Liu, T.: Instance-dependent early stopping. International Conference on Learning Representations (2025)
43. Zhong, Z., Zheng, L., Kang, G., Li, S., Yang, Y.: Random erasing data augmentation. In: Association for the Advancement of Artificial Intelligence (2020)