LOSS ALONE IS NOT ENOUGH: STABILITY AND GEOMETRY IN DYNAMIC SAMPLE REWEIGHTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Dynamic sample reweighting can emphasize informative examples during training, and several recent methods use the per-sample loss to determine these weights. Per-sample loss indicates which samples are difficult, but it does not reveal how emphasizing those samples changes the geometry of the optimization update. We show that this distinction can matter even on simple strongly convex problems. For a broad class of loss-only reweighting schemes, upweighting higher-loss examples can lead to an arbitrarily worse local convergence rate than gradient descent with uniform sample weights. Moreover, for exponential loss-based reweighting, increasing the step size past a threshold produces a stable two-cycle (a persistent oscillation) even when uniform-weight gradient descent remains stable. To address this limitation, we propose **Geometry-Aware Reweighting (GAR)**, which chooses weights using both loss and gradient information. For minibatch SGD, GAR satisfies the standard upper bound of uniform-weight SGD minus an explicit nonnegative term, and it avoids the instability in our hard example. We also derive practical extensions to momentum and AdamW. Experiments on 120M–300M language models show that GAR achieves lower evaluation log-perplexity than both uniform-weight AdamW and the loss-based LinUpper reweighting (Sow et al., 2025).

1 INTRODUCTION

Dynamic sample reweighting aims to spend more optimization effort on examples that are more useful at the current stage of training. Classical importance-sampling methods choose nonuniform sampling probabilities using smoothness, gradient norms, side information, or variance estimates, typically with inverse-probability correction so that the stochastic gradient remains unbiased (Needell et al., 2014; Zhao & Zhang, 2015; Alain et al., 2015; Gopal, 2016; Borsos et al., 2018; Katharopoulos & Fleuret, 2018; El Hanchi et al., 2022). A different line of work deliberately changes the update by emphasizing selected examples based on online training signals. A particularly simple choice is the current per-example loss, as in loss-based batch selection and Selective Backprop (Loshchilov & Hutter, 2015; Jiang et al., 2019), Ordered SGD (Kawaguchi & Lu, 2020), exponential tilting and KL-based distributionally robust reweighting (RGD) (Li et al., 2023; Kumar et al., 2024), and the recent LinUpper method for language-model pretraining (Sow et al., 2025). Other approaches instead use additional information, such as losses and gradients calculated from a holdout dataset or gradient/optimizer geometry (Mindermann et al., 2022; Ren et al., 2018; Wang et al., 2024; 2026).

This contrast raises a central question: *what information is needed to decide which examples to emphasize?* In particular, *is per-example loss alone sufficient?* Loss is attractive because it is cheap and immediately available, but it only indicates which samples are difficult; it does not reveal how emphasizing them changes the optimization update. Two high-loss examples may have gradients that reinforce each other, cancel, or point in very different directions. A reweighting scheme that only sees losses cannot distinguish these cases.

We show that this missing geometric information can fundamentally change optimization dynamics, even on simple strongly convex problems. We consider a broad class of loss-only reweighting schemes and construct a well-conditioned problem on which loss-only reweighting can be arbitrarily slower than uniform gradient descent or even unstable. Motivated by this limitation, we propose

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.

Geometry-Aware Reweighting (GAR), which combines loss information with gradient geometry. The contributions below make these statements precise:

- We first sharpen LinUpper's interpolation-based minibatch bound (Sow et al., 2025), improving its stated leading term from $O(T^{-1/2})$ to $O(T^{-1})$ and removing its per-sample weight cap. Without interpolation, we show that an additional gradient-deviation term appears in the convergence bound and can be up to b^2 larger than under uniform weighting when the batch size is b (Proposition 1 and Theorem 1).
- We next analyze a broad class of loss-only weighting schemes and show that such a gradient-deviation term can harm convergence, even in the full-batch case. We show that the Jacobian of the reweighted iteration at the ERM optimum contains an additional curvature term arising from the gradient covariance across samples. On a simple strongly convex example, this term can make local convergence arbitrarily slower than uniform gradient descent or make the ERM optimum repelling (Proposition 2 and Theorem 2). For exponential weighting, we further show the transition from global contraction to the emergence of an attracting two-cycle (Theorem 3).
- In order to remedy this problem, we propose Geometry-Aware Reweighting (GAR), a dynamic sample reweighting scheme that uses both losses and gradient geometry. For minibatch SGD, GAR matches the standard uniform-SGD upper bound, minus an explicit nonnegative margin (Theorem 4). A local Jacobian analysis for GAR shows that the gradient covariance term enters the GAR Jacobian through a nonexpansive factor rather than as added curvature (Proposition 3).
- We derive momentum and AdamW variants of GAR (Section 6) and evaluate them on 120M–300M language models (Section 7). GAR improves average evaluation log-perplexity over both uniform AdamW and LinUpper (Sow et al., 2025) at all three model sizes.

1.1 Related Work

Loss-based methods include biased importance sampling, Ordered SGD, KL-DRO/RGD, TERM, and LinUpper (Katharopoulos & Fleuret, 2017; Kawaguchi & Lu, 2020; Kumar et al., 2024; Li et al., 2023; Sow et al., 2025). Our work is particularly inspired by LinUpper (Sow et al., 2025), which dynamically assigns sample weights from their current losses and analyzes how such loss-based reweighting helps convergence under interpolation. We extend their analysis by improving its leading term and also removing the upper limit on per-sample weights. More importantly, we study what happens beyond interpolation, where an additional gradient-deviation term appears and, as our later stability analysis shows, loss-based reweighting can slow or destabilize optimization.

Methods such as meta-reweighting, GREATS, and OPUS use gradient information together with a separate clean validation set or target-domain proxy to guide sample selection or weighting (Ren et al., 2018; Wang et al., 2024; 2026). These methods therefore operate in a different setting from our loss-only schemes and GAR, which use only information from the current training minibatch. Another related work is Balles et al. (2023), which uses last-layer gradient inner products to select a sparse, weighted subset that approximates the uniform minibatch gradient. Their objective is to reduce backward computation while preserving the minibatch update, whereas GAR selects weights without a sparsity constraint, balancing per-sample loss gaps against the norm of the resulting update. Relatedly, An et al. (2021) show that resampling and reweighting can induce different stochastic-gradient dynamics even when they correspond to the same target objective. Appendix A gives additional comparisons with prior work.

2 Problem Setup

We consider the finite-sum objective, with optimization variable $\boldsymbol{x}$,

$$f(\boldsymbol{x}) = \frac{1}{n}\sum_{i=1}^{n} f_i(\boldsymbol{x}),$$

with its minimizer $\boldsymbol{x}^\star$. We analyze reweighted preconditioned SGD on this objective. At iteration t, draw $B_t = (i_{t,1}, \ldots, i_{t,b})$ independently and uniformly with replacement and set

$$\boldsymbol{G}_t = [\nabla f_{i_{t,1}}(\boldsymbol{x}_t), \ldots, \nabla f_{i_{t,b}}(\boldsymbol{x}_t)].$$

Given a fixed positive-definite preconditioner $\boldsymbol{P} \succ 0$, the reweighted update is

$$\boldsymbol{x}_{t+1} = \boldsymbol{x}_t - \eta \boldsymbol{P}^{-1} \boldsymbol{G}_t \boldsymbol{w}_t, \qquad \boldsymbol{w}_t \in \Delta_b := \{\boldsymbol{w} \in \mathbb{R}^b : \boldsymbol{w} \geq \boldsymbol{0}, \boldsymbol{1}^\top \boldsymbol{w} = 1\}. \tag{1}$$

Setting $\boldsymbol{P} = \boldsymbol{I}$ recovers the standard Euclidean reweighted SGD update. The weights may depend on the entire sampled minibatch, so $\boldsymbol{G}_t \boldsymbol{w}_t$ need not be an unbiased gradient estimator. We write $\bar{\boldsymbol{x}}_T = T^{-1} \sum_{t=0}^{T-1} \boldsymbol{x}_t$ and let $\mathbb{E}_t[\cdot]$ denote expectation conditional on the history before drawing B_t, and let $\mathbb{E}[\cdot]$ denote total expectation.

We use the associated norms $\|\boldsymbol{x}\|_{\boldsymbol{P}}^2 = \boldsymbol{x}^\top \boldsymbol{P} \boldsymbol{x}$ and $\|\boldsymbol{g}\|_{\boldsymbol{P}^{-1}}^2 = \boldsymbol{g}^\top \boldsymbol{P}^{-1} \boldsymbol{g}$. The main theory keeps $\boldsymbol{P}$ fixed; Section 6 discusses how the same construction is adapted to momentum and AdamW.

We analyze (1) under the following assumptions.

Assumption 1 (Convexity and smoothness)**.** *Each f_i is convex and L-smooth in $\|\cdot\|_{\boldsymbol{P}}$: $\|\nabla f_i(\boldsymbol{x}) - \nabla f_i(\boldsymbol{y})\|_{\boldsymbol{P}^{-1}} \leq L \|\boldsymbol{x} - \boldsymbol{y}\|_{\boldsymbol{P}}$.*

Assumption 2 (Average bounded gradient variation)**.** *For every $\boldsymbol{x}$,*

$$\frac{1}{n} \sum_{i=1}^{n} \|\nabla f_i(\boldsymbol{x}) - \nabla f(\boldsymbol{x})\|_{\boldsymbol{P}^{-1}}^2 \leq \sigma^2.$$

Assumption 1 requires that each component function f_i is convex, and its gradient does not change too rapidly, and Assumption 2 controls the average component-gradient variation. Both assumptions are standard in the literature.

Assumption 3 (Interpolation)**.** $\nabla f_i(\boldsymbol{x}^\star) = 0$ *for every i.*

The interpolation condition commonly appears in analyses of SGD for overparameterized finite-sum problems (Ma et al., 2018; Vaswani et al., 2019; Gower et al., 2021) and is also used in the LinUpper convergence analysis (Sow et al., 2025). In this paper, Assumption 3 is used only in Proposition 1.

3 Convergence of Reweighting With and Without Interpolation

We first study arbitrary within-minibatch reweighting under interpolation and sharpen an earlier result of Sow et al. (2025). Define the *component loss gap*

$$d_{t,j} := f_{i_{t,j}}(\boldsymbol{x}_t) - f_{i_{t,j}}(\boldsymbol{x}^\star).$$

Under interpolation, $\nabla f_i(\boldsymbol{x}^\star) = \boldsymbol{0}$, so smoothness and convexity imply (Lemma 1)

$$\|\nabla f_i(\boldsymbol{x})\|_{\boldsymbol{P}^{-1}}^2 \leq 2L(f_i(\boldsymbol{x}) - f_i(\boldsymbol{x}^\star)).$$

By convexity of the squared norm and Jensen's inequality, this also controls the squared norm of any weighted gradient by the corresponding weighted loss gaps. Applying this inequality to the one-step distance recursion yields the following convergence result.

Proposition 1 (Reweighted SGD under interpolation)**.** *Suppose Assumptions 1 and 3 hold and $0 < \eta \leq \frac{1}{2L}$. For arbitrary, possibly minibatch-dependent $\boldsymbol{w}_t \in \Delta_b$,*

$$\mathbb{E}[f(\bar{\boldsymbol{x}}_T) - f(\boldsymbol{x}^\star)] \leq \frac{\|\boldsymbol{x}_0 - \boldsymbol{x}^\star\|_{\boldsymbol{P}}^2}{\eta T} + \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}\left[\sum_{j=1}^{b} \left(\frac{1}{b} - w_{t,j}\right) d_{t,j}\right],$$

where $\bar{\boldsymbol{x}}_T = T^{-1} \sum_{t=0}^{T-1} \boldsymbol{x}_t$.

Note that we did not use Assumption 2 here. The bound in Proposition 1 is the standard $O(T^{-1})$ optimization term plus a reweighting correction. The correction term suggests that choosing higher $w_{t,j}$ for samples with larger loss gap $d_{t,j}$ can reduce the upper bound. Uniform weights cancel the correction, recovering the standard $O(T^{-1})$ SGD rate under interpolation (Vaswani et al., 2019; Gower et al., 2021). This baseline bound sharpens the stated minibatch-SGD bound of Sow et al. (2025, Appendix C.3, Theorem 3(i)): its leading term improves from $O(T^{-1/2})$ to $O(T^{-1})$, the cap $w_{t,j} \leq 2/b$ is removed, and any fixed preconditioner is allowed. Appendix B gives the proof.

We now study arbitrary within-minibatch reweighting without interpolation, where an additional term appears. Define the *gradient deviation term* at t as

$$\mathcal{V}_t := \left\| \sum_{j=1}^{b} w_{t,j} (\nabla f_{i_{t,j}}(\boldsymbol{x}_t) - \nabla f(\boldsymbol{x}_t)) \right\|_{\boldsymbol{P}^{-1}}^2.$$

Theorem 1 (Reweighted SGD without interpolation)**.** *Under Assumption 1, for* $0 < \eta \leq \frac{1}{4L}$ *and arbitrary, possibly minibatch-dependent* $\boldsymbol{w}_t \in \Delta_b$*,*

$$\mathbb{E}[f(\bar{\boldsymbol{x}}_T) - f(\boldsymbol{x}^\star)] \leq \frac{\|\boldsymbol{x}_0 - \boldsymbol{x}^\star\|_{\boldsymbol{P}}^2}{\eta T} + \frac{2\eta}{T}\sum_{t=0}^{T-1}\mathbb{E}[\mathcal{V}_t] + \frac{2}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\sum_{j=1}^{b}\left(\frac{1}{b} - w_{t,j}\right)d_{t,j}\right]. \quad (2)$$

The final correction term captures how the weights shift mass relative to uniform weighting, while the $\mathcal{V}_t$ term is an additional contribution from component gradient deviations that loss values alone do not control. The proof bounds the squared weighted-gradient norm by the full-gradient term plus $\mathcal{V}_t$ and then telescopes; Appendix B gives the details.

It is important that $\mathcal{V}_t$ is not simply the variance of an unbiased minibatch estimator. The weights are selected after observing B_t, so the weighted deviations need not cancel. Also note that Theorem 1 does not require Assumption 2. Under Assumption 2, for uniform weights $\boldsymbol{u} = \frac{1}{b}\mathbf{1}$, independent sampling with replacement gives

$$\mathbb{E}_t[\mathcal{V}_t] = \mathbb{E}_t\left[\|\boldsymbol{G}_t\boldsymbol{u} - \nabla f(\boldsymbol{x}_t)\|_{\boldsymbol{P}^{-1}}^2\right] = \frac{1}{bn}\sum\nolimits_{i=1}^{n}\|\nabla f_i(\boldsymbol{x}_t) - \nabla f(\boldsymbol{x}_t)\|_{\boldsymbol{P}^{-1}}^2 \leq \frac{\sigma^2}{b}.$$

Thus, under Assumption 2, Theorem 1 recovers the usual uniform-weight minibatch-SGD guarantee

$$\mathbb{E}[f(\bar{\boldsymbol{x}}_T) - f(\boldsymbol{x}^\star)] \leq \frac{\|\boldsymbol{x}_0 - \boldsymbol{x}^\star\|_{\boldsymbol{P}}^2}{\eta T} + \frac{2\eta\sigma^2}{b},$$

which gives the standard $\mathcal{O}(1/\sqrt{bT})$ stochastic scaling after tuning η.

Arbitrary dynamic weights behave differently under the same average-variance assumption. Jensen's inequality gives only

$$\mathbb{E}_t[\mathcal{V}_t] \leq b\sigma^2.$$

This $b\sigma^2$ scaling is tight in the worst case under Assumption 2. Since the assumption controls deviations only on average, dynamic weights can correlate with rare large deviations that uniform averaging suppresses. Appendix B gives the proof and tightness calculation. Thus, the gradient deviation term can be up to b^2 times the uniform scale. Although the tightness proof deliberately emphasizes an outlier, Section 4 shows that the same geometric issue persists for the broad class of loss-only weighting maps studied there and can cause arbitrarily slow convergence or instability even in deterministic full-batch optimization.

4 Why Loss-Only Reweighting Can Be Slow or Unstable

Theorem 1 shows that data-dependent reweighting can amplify gradient deviations that uniform averaging suppresses. We now show that this is not only a stochastic-variance issue: even in full-batch optimization, a broad class of loss-only reweighting schemes can change the local geometry of an otherwise well-conditioned problem.

We specialize to full-batch reweighting, $b = n$ and $B_t = [n]$. Let $\Phi : \mathbb{R}^n \to \Delta_n$ be a loss-only weighting map, which takes the scores (calculated based on loss values) of n samples and produces a weight vector. We assume that Φ is permutation-equivariant and shift-invariant:

$$\Phi(\boldsymbol{Q}\boldsymbol{s}) = \boldsymbol{Q}\Phi(\boldsymbol{s}), \qquad \Phi(\boldsymbol{s} + c\mathbf{1}) = \Phi(\boldsymbol{s}) \quad (3)$$

for every score vector $\boldsymbol{s} \in \mathbb{R}^n$ calculated from per-sample losses, permutation matrix $\boldsymbol{Q}$, and scalar $c \in \mathbb{R}$. These conditions encode symmetry across examples and dependence only on relative score differences. A canonical example is the exponential weighting (Li et al., 2023; Kumar et al., 2024):

$$\Phi_\tau(\boldsymbol{s}) = \mathrm{softmax}(\boldsymbol{s}/\tau), \qquad \tau > 0. \quad (4)$$

Methods that additionally use validation losses or gradient information lie outside this loss-only class.

4.1 How loss-only weighting changes local curvature

For the local analysis, define the loss-gap vector $\boldsymbol{d}(\boldsymbol{x}) := (f_1(\boldsymbol{x}) - f_1(\boldsymbol{x}^\star), \ldots, f_n(\boldsymbol{x}) - f_n(\boldsymbol{x}^\star))$ and consider the full-batch update with weights $\Phi(\boldsymbol{d}(\boldsymbol{x}))$. Because $\boldsymbol{d}(\boldsymbol{x}^\star) = \mathbf{0}$, permutation equivariance gives $\Phi(\boldsymbol{d}(\boldsymbol{x}^\star)) = \frac{1}{n}\mathbf{1}$. Therefore, $\boldsymbol{G}(\boldsymbol{x}^\star)\Phi(\boldsymbol{d}(\boldsymbol{x}^\star)) = \frac{1}{n}\sum_{i=1}^n \nabla f_i(\boldsymbol{x}^\star) = \nabla f(\boldsymbol{x}^\star) = \mathbf{0}$. Hence the ERM optimum remains a fixed point of the reweighted dynamics. Let

$$\boldsymbol{g}_i^\star := \nabla f_i(\boldsymbol{x}^\star), \qquad \boldsymbol{H}_\star := \nabla^2 f(\boldsymbol{x}^\star), \qquad \boldsymbol{C}_\star := \frac{1}{n}\sum_{i=1}^n \boldsymbol{g}_i^\star (\boldsymbol{g}_i^\star)^\top, \tag{5}$$

and write $\bar{\boldsymbol{H}}_\star = \boldsymbol{P}^{-1/2}\boldsymbol{H}_\star\boldsymbol{P}^{-1/2}$ and $\bar{\boldsymbol{C}}_\star = \boldsymbol{P}^{-1/2}\boldsymbol{C}_\star\boldsymbol{P}^{-1/2}$. Since $\nabla f(\boldsymbol{x}^\star) = 0$, $\boldsymbol{C}_\star$ is the *component-gradient covariance* at the ERM optimum.

Proposition 2 (Local Jacobian of loss-only reweighting)**.** *Suppose $n \geq 2$. Assume that Φ satisfies (3) and is differentiable at $\mathbf{0}$. Then, there is a unique scalar $\gamma_\Phi \in \mathbb{R}$ such that, with $\Pi_\perp = \boldsymbol{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$,*

$$D\Phi(\mathbf{0}) = \frac{\gamma_\Phi}{n}\Pi_\perp.$$

Let $\boldsymbol{z} = \boldsymbol{P}^{1/2}(\boldsymbol{x} - \boldsymbol{x}^\star)$. If each f_i is twice continuously differentiable near the stationary point $\boldsymbol{x}^\star$, the Jacobian at $\boldsymbol{z} = \mathbf{0}$ of the induced full-batch update map $\boldsymbol{x} \mapsto \boldsymbol{x} - \eta\boldsymbol{P}^{-1}\boldsymbol{G}(\boldsymbol{x})\Phi(\boldsymbol{d}(\boldsymbol{x}))$ is

$$\boldsymbol{J}_\Phi = \boldsymbol{I} - \eta(\bar{\boldsymbol{H}}_\star + \gamma_\Phi \bar{\boldsymbol{C}}_\star). \tag{6}$$

Here γ_Φ quantifies how the weights change to first order with relative loss differences; uniform weighting has $\gamma_\Phi = 0$, and $\gamma_\Phi > 0$ means that, to first order around equal scores, Φ gives larger weights to samples with larger scores. Exponential weighting (4) has $\gamma_\Phi = 1/\tau$. For $\gamma_\Phi > 0$, (6) shows that component-gradient disagreement contributes the positive-semidefinite curvature term $\gamma_\Phi\bar{\boldsymbol{C}}_\star$, which can shrink the stable step-size range and slow down local convergence. The next subsection makes this degradation arbitrarily large on a well-conditioned hard instance.

Connection to TERM. For exponential weighting, the same covariance term appears in the Hessian of the tilted objective studied by TERM (Li et al., 2023). TERM derives it as objective curvature, whereas Proposition 2 shows that it appears in the Jacobian of the reweighted iteration at the ERM optimum for any differentiable map satisfying (3).

4.2 A hard instance for local slowdown and instability

For fixed $\boldsymbol{P} \succ 0$, whiten coordinates by $\boldsymbol{z} = \boldsymbol{P}^{1/2}(\boldsymbol{x} - \boldsymbol{x}^\star)$ and define

$$f_s(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{z}^\top \begin{pmatrix} L/2 & 0 \\ 0 & L \end{pmatrix}\boldsymbol{z} + s\sigma z_2 + \frac{\sigma^2}{2L}, \qquad s \in \{-1, +1\}. \tag{7}$$

Both components are L-smooth and $(L/2)$-strongly convex, while their average has condition number two and average squared component-gradient variation σ^2. At $\boldsymbol{x}^\star$, both losses equal $c = \sigma^2/(2L)$. With $\boldsymbol{\ell}(\boldsymbol{x}) := (f_{-1}(\boldsymbol{x}), f_{+1}(\boldsymbol{x}))$ and $\boldsymbol{d}(\boldsymbol{x}) = (f_{-1}(\boldsymbol{x}) - f_{-1}(\boldsymbol{x}^*), f_{+1}(\boldsymbol{x}) - f_{+1}(\boldsymbol{x}^*))$, we have $\boldsymbol{\ell}(\boldsymbol{x}) = \boldsymbol{d}(\boldsymbol{x}) + c\mathbf{1}$; hence shift invariance gives $\Phi(\boldsymbol{\ell}(\boldsymbol{x})) = \Phi(\boldsymbol{d}(\boldsymbol{x}))$. Also, $\Phi(\boldsymbol{\ell}(\boldsymbol{x}))$ assigns uniform weights at $\boldsymbol{x}^\star$, so the ERM optimum remains a fixed point. Proposition 2 therefore applies exactly to this instance, yielding the following result.

Theorem 2 (Loss-only reweighting can degrade local optimization)**.** *Fix a differentiable weighting map Φ satisfying (3), with sensitivity γ_Φ from Proposition 2, and set $\boldsymbol{w}(\boldsymbol{x}) = \Phi(\boldsymbol{\ell}(\boldsymbol{x}))$ on (7). If $\gamma_\Phi > 0$, then even with the optimal scalar step size, the linearized dynamics of (1) around $\boldsymbol{x}^\star$ require*

$$\Theta\left(\left(1 + \frac{\gamma_\Phi\sigma^2}{L}\right)\log\frac{1}{\epsilon}\right)$$

iterations to contract the slowest eigendirection by a factor ϵ, while uniform GD needs $\Theta(\log(1/\epsilon))$ iterations. If $\gamma_\Phi < 0$ and $\sigma^2 > L/|\gamma_\Phi|$, the ERM optimum is locally repelling for every $\eta > 0$.

Indeed, the two local eigenvalues are

$$1 - \eta L/2, \qquad 1 - \eta(L + \gamma_\Phi\sigma^2). \tag{8}$$

Thus $\gamma_\Phi > 0$ requires $\eta < 2/(L + \gamma_\Phi \sigma^2)$ for stability, whereas uniform GD is optimally tuned by $\eta = 4/(3L)$ and contracts by $1/3$. Crucially, σ^2 appears even in *deterministic full-batch* optimization: it measures component-gradient disagreement, not sampling noise. Increasing σ changes only affine component terms in (7), leaving the Hessians, smoothness/strong-convexity constants, and average condition number two unchanged; hence the slowdown relative to uniform full-batch GD can be arbitrarily large. It is also worth noting that Theorem 2 applies to both choices of score vector $\boldsymbol{s}$: current loss values $\boldsymbol{\ell}(\boldsymbol{x})$ and loss gaps $\boldsymbol{d}(\boldsymbol{x})$.

4.3 Exponential reweighting: a sharp global transition

Consider applying the exponential scheme (4) to our hard instance (7). Writing $\boldsymbol{z}^+$ for the next iterate obtained from the current iterate $\boldsymbol{z}$, the exact reweighted preconditioned SGD dynamics are

$$z_1^+ = (1 - \eta L/2) z_1, \qquad z_2^+ = (1 - \eta L) z_2 - \eta\sigma \tanh(\sigma z_2/\tau). \tag{9}$$

This lets us strengthen the local slowdown above to a global statement.

Theorem 3 (Global transition for exponential loss weighting)**.** *For (9), the ERM optimum $\boldsymbol{x}^\star$ is the unique fixed point. Let $\eta_c = \frac{2}{L+\sigma^2/\tau}$. If $0 < \eta < \eta_c$, the map is globally contractive with factor*

$$\rho(\eta) = \max\{|1 - \eta L/2|, |1 - \eta(L + \sigma^2/\tau)|\}. \tag{10}$$

Under the optimal step size, the worst-case number of iterations needed to reduce the distance to $x^\star$ by a factor ϵ is $\Theta((1 + \frac{\sigma^2}{L\tau}) \log \frac{1}{\epsilon})$. If $\eta_c < \eta < \frac{2}{L}$, there is a unique nonzero symmetric attracting two-cycle; when also $\eta \geq \frac{1}{L}$, every initialization with $z_{2,0} \neq 0$ converges to the two-cycle.

Appendix C proves the theorem. Here, η_c is a genuine global stability boundary: crossing it destabilizes the ERM optimum and creates a persistent attracting oscillation. Smaller τ lowers this threshold. For comparison, the uniform GD contracts by $1/3$ under its optimal step size $\eta = 4/(3L)$. In contrast, with the same η, exponential loss weighting converges to the two-cycle from almost every initialization whenever $\sigma^2/(L\tau) > 1/2$, even in deterministic full-batch optimization. The two-cycle phenomenon is not specific to exponential reweighting: Appendix C extends it to a broader class of monotone weighting maps, including cases with $\gamma_\Phi = 0$.

5 Geometry-Aware Reweighting

The preceding sections show that without interpolation, reweighting incurs an additional gradient-deviation term, and that in full-batch dynamics component-gradient covariance can appear as additional local curvature. We now propose *Geometry-Aware Reweighting (GAR)*, which incorporates gradient geometry directly into weight selection by accounting for the norm of the weighted gradient.

5.1 The GAR weight-selection problem

Recall $d_{t,j} = f_{i_{t,j}}(\boldsymbol{x}_t) - f_{i_{t,j}}(\boldsymbol{x}^\star)$ and let $\boldsymbol{d}_t = (d_{t,1}, \ldots, d_{t,b})$. Conditional on $\boldsymbol{x}_t$ and B_t, convexity and exact distance expansion give, for any $\boldsymbol{w} \in \Delta_b$,

$$\|\boldsymbol{x}_t - \eta \boldsymbol{P}^{-1} \boldsymbol{G}_t \boldsymbol{w} - \boldsymbol{x}^\star\|_{\boldsymbol{P}}^2 \leq \|\boldsymbol{x}_t - \boldsymbol{x}^\star\|_{\boldsymbol{P}}^2 - 2\eta \boldsymbol{d}_t^\top \boldsymbol{w} + \eta^2 \|\boldsymbol{G}_t \boldsymbol{w}\|_{\boldsymbol{P}^{-1}}^2. \tag{11}$$

Rather than optimizing only the linear loss-gap term, GAR minimizes the complete weight-dependent part of this bound, with entropy regularization $h(\boldsymbol{w}) := \sum_j w_j \log w_j$:

$$\boldsymbol{w}_t^{\mathrm{GAR}} \in \arg\min_{\boldsymbol{w} \in \Delta_b} Q_t(\boldsymbol{w}), \qquad Q_t(\boldsymbol{w}) := -\boldsymbol{d}_t^\top \boldsymbol{w} + \frac{\eta}{2} \|\boldsymbol{G}_t \boldsymbol{w}\|_{\boldsymbol{P}^{-1}}^2 + \tau h(\boldsymbol{w}). \tag{12}$$

The quadratic term distinguishes gradients that cancel from gradients that reinforce. For $\tau > 0$, the minimizer is unique and satisfies

$$\boldsymbol{w}_t^{\mathrm{GAR}} = \mathrm{softmax}\big((\boldsymbol{d}_t - \eta \boldsymbol{G}_t^\top \boldsymbol{P}^{-1} \boldsymbol{G}_t \boldsymbol{w}_t^{\mathrm{GAR}})/\tau\big). \tag{13}$$

Algorithm 1 summarizes the generic within-minibatch procedure. The theoretical update (12) uses $(\boldsymbol{s}, \boldsymbol{R}, \alpha) = (\boldsymbol{d}_t, \boldsymbol{P}, \eta)$; Section 6 gives the corresponding momentum and AdamW instantiations.

Algorithm 1 Geometry-Aware Reweighting (GAR) within a minibatch

Require: score vector $\boldsymbol{s} \in \mathbb{R}^b$, gradient matrix $\boldsymbol{G}_t$, selection metric $\boldsymbol{R} \succ 0$, geometry coefficient $\alpha > 0$, temperature $\tau > 0$

1: $\boldsymbol{w}_t \leftarrow \arg\min_{\boldsymbol{w}\in\Delta_b} \left\{-\boldsymbol{s}^\top \boldsymbol{w} + \frac{\alpha}{2}\|\boldsymbol{G}_t\boldsymbol{w}\|^2_{\boldsymbol{R}^{-1}} + \tau h(\boldsymbol{w})\right\}$

2: $\widehat{\boldsymbol{w}}_t \leftarrow \mathrm{stopgrad}(\boldsymbol{w}_t)$

3: Apply the base optimizer to $\sum_j \widehat{w}_{t,j} f_{i_{t,j}}(\boldsymbol{x}_t)$

5.2 Global guarantee: the uniform bound minus a margin

Let $\boldsymbol{u} = \frac{1}{b}\mathbf{1}$ be the uniform weights. Because $\boldsymbol{u}$ is feasible in (12), GAR can be compared directly with the standard uniform-minibatch SGD bound.

Theorem 4 (Uniform-comparator convergence)**.** *Suppose Assumptions 1 and 2 hold, $\tau \geq 0$, and $0 < \eta \leq \frac{1}{2L}$. Define $\Gamma_t := Q_t(\boldsymbol{u}) - Q_t(\boldsymbol{w}_t^{\mathrm{GAR}}) + \tau\,\mathrm{KL}(\boldsymbol{w}_t^{\mathrm{GAR}}\|\boldsymbol{u}) \geq 0$. Then,*

$$\mathbb{E}[f(\bar{\boldsymbol{x}}_T) - f(\boldsymbol{x}^\star)] \leq \frac{\|\boldsymbol{x}_0 - \boldsymbol{x}^\star\|^2_{\boldsymbol{P}}}{\eta T} + \frac{\eta\sigma^2}{b} - \frac{2}{T}\sum\nolimits_{t=0}^{T-1} \mathbb{E}[\Gamma_t]. \tag{14}$$

The first two terms on the right are the usual upper bound for uniform minibatch SGD under the same assumptions; GAR gets this bound *minus* an explicit nonnegative margin. Importantly, $\boldsymbol{G}_t\boldsymbol{w}_t^{\mathrm{GAR}}$ is generally a biased estimator of the full gradient because its weights depend on the sampled minibatch. Nevertheless, since the uniform weights remain feasible in the GAR weight-selection problem, the comparator argument gives a convergence upper bound no worse than that of unbiased uniform minibatch SGD, and improves it by the nonnegative margin above. Thus GAR can use nonuniform, data-dependent weights without paying the worst-case b^2 amplification in Theorem 1.

Under μ-strong convexity, Corollary 2 in Appendix D gives the corresponding standard uniform-weight SGD bound, again improved by a discounted nonnegative GAR margin.

5.3 Local stability of GAR

Theorem 4 is a general convergence result under the assumptions of Section 2. We next study the local full-batch dynamics of GAR, using the Hessian $\bar{\boldsymbol{H}}_\star$ and gradient-covariance matrix $\bar{\boldsymbol{C}}_\star$ defined in (5).

Proposition 3 (Component-gradient covariance enters through a nonexpansive factor)**.** *Suppose each f_i is twice continuously differentiable near a stationary point $\boldsymbol{x}^\star$ and $\tau > 0$. Then exact full-batch GAR has uniform weights at $\boldsymbol{x}^\star$, and in the coordinates $\boldsymbol{z} = \boldsymbol{P}^{1/2}(\boldsymbol{x} - \boldsymbol{x}^\star)$ the Jacobian at $\boldsymbol{z} = \mathbf{0}$ of the update map induced by GAR is*

$$\boldsymbol{J}_{\mathrm{GAR}} = \left(\boldsymbol{I} + \frac{\eta}{\tau}\bar{\boldsymbol{C}}_\star\right)^{-1}(\boldsymbol{I} - \eta\bar{\boldsymbol{H}}_\star), \quad \textit{and for any } \boldsymbol{v}, \quad \|\boldsymbol{J}_{\mathrm{GAR}}\boldsymbol{v}\|_2 \leq \|(\boldsymbol{I} - \eta\bar{\boldsymbol{H}}_\star)\boldsymbol{v}\|_2. \tag{15}$$

Proposition 2 gives $\boldsymbol{J}_\Phi = \boldsymbol{I} - \eta(\bar{\boldsymbol{H}}_\star + \gamma_\Phi\bar{\boldsymbol{C}}_\star)$ for weights $\Phi(\boldsymbol{d}(\boldsymbol{x}))$ from any differentiable, permutation-equivariant, and shift-invariant weighting map. GAR does not belong to that loss-only class because its weights also depend on the gradient Gram matrix $\boldsymbol{G}^\top\boldsymbol{P}^{-1}\boldsymbol{G}$. That extra information changes the covariance contribution from explicit added curvature to the nonexpansive factor in (15).

For exponential loss weighting with temperature τ, the comparison is especially direct:

$$\boldsymbol{J}_{\exp} = \boldsymbol{I} - \eta\left(\bar{\boldsymbol{H}}_\star + \frac{1}{\tau}\bar{\boldsymbol{C}}_\star\right), \qquad \boldsymbol{J}_{\mathrm{GAR}} = \left(\boldsymbol{I} + \frac{\eta}{\tau}\bar{\boldsymbol{C}}_\star\right)^{-1}(\boldsymbol{I} - \eta\bar{\boldsymbol{H}}_\star).$$

Why GAR stabilizes exponential weighting: an implicit-step interpretation. The Jacobian comparison above admits a simple optimization interpretation. Exponential loss weighting takes an explicit gradient step on a local log-sum-exp model, whereas GAR takes the corresponding implicit (proximal) step. This explains why component-gradient covariance appears as added curvature for exponential weighting but through a nonexpansive inverse factor for GAR. Fix the current minibatch, write $\boldsymbol{g}_{t,j} = \nabla f_{i_{t,j}}(\boldsymbol{x}_t)$, and define:

$$\mathcal{M}_t(\boldsymbol{y}) := \tau\log\left[\frac{1}{b}\sum\nolimits_{j=1}^{b}\exp\left(\frac{d_{t,j} + \langle\boldsymbol{g}_{t,j}, \boldsymbol{y} - \boldsymbol{x}_t\rangle}{\tau}\right)\right]. \tag{16}$$

Let $\boldsymbol{w}_t^{\exp} = \mathrm{softmax}(\boldsymbol{d}_t/\tau)$, and let $\boldsymbol{w}_t^{\mathrm{GAR}}$ solve (12); define the corresponding next iterates by $\boldsymbol{x}_{t+1}^{\exp} := \boldsymbol{x}_t - \eta \boldsymbol{P}^{-1}\boldsymbol{G}_t\boldsymbol{w}_t^{\exp}$ and $\boldsymbol{x}_{t+1}^{\mathrm{GAR}} := \boldsymbol{x}_t - \eta \boldsymbol{P}^{-1}\boldsymbol{G}_t\boldsymbol{w}_t^{\mathrm{GAR}}$.

Proposition 4 (Explicit versus implicit exponential reweighting)**.**

$$\begin{aligned} \boldsymbol{x}_{t+1}^{\exp} &= \boldsymbol{x}_t - \eta \boldsymbol{P}^{-1}\nabla \mathcal{M}_t(\boldsymbol{x}_t), \\ \boldsymbol{x}_{t+1}^{\mathrm{GAR}} &= \mathrm{prox}_{\eta \mathcal{M}_t}^{\boldsymbol{P}}(\boldsymbol{x}_t) = \boldsymbol{x}_t - \eta \boldsymbol{P}^{-1}\nabla \mathcal{M}_t(\boldsymbol{x}_{t+1}^{\mathrm{GAR}}). \end{aligned} \tag{17}$$

Here, $\mathrm{prox}_{\eta\mathcal{M}}^{\boldsymbol{P}}(\boldsymbol{x}) := \arg\min_{\boldsymbol{y}}\{\mathcal{M}(\boldsymbol{y}) + \frac{1}{2\eta}\|\boldsymbol{y}-\boldsymbol{x}\|_{\boldsymbol{P}}^2\}$ *is the proximal map (Parikh & Boyd, 2014).*

Thus, exponential weighting evaluates the gradient of the current linearized log-sum-exp objective at $\boldsymbol{x}_t$, whereas GAR satisfies the corresponding implicit relation at $\boldsymbol{x}_{t+1}^{\mathrm{GAR}}$. In the full-batch local setting, linearizing these relations at $\boldsymbol{x}^\star$ explains Proposition 3: the covariance term appears as added curvature for exponential weighting but yields the nonexpansive factor $(\boldsymbol{I} + \eta\bar{\boldsymbol{C}}_\star/\tau)^{-1}$ for GAR.

5.4 Returning to the hard instance: GAR remains globally stable

Corollary 1 (Global stability on the hard instance)**.** *On (7), using the same notation* $\boldsymbol{z}^+$ *for the next iterate, exact full-batch GAR satisfies*

$$z_1^+ = (1-\eta L/2)z_1, \qquad z_2^+ + \eta\sigma\tanh(\sigma z_2^+/\tau) = (1-\eta L)z_2. \tag{18}$$

For $0 < \eta < 2/L$ *and any* $\sigma, \tau > 0$*, it contracts globally with factor at most* $\max\{|1-\eta L/2|, |1-\eta L|\}$*. In particular, at* $\eta = 4/(3L)$*, the contraction ratio of GAR is at most* $1/3$*.*

At the same step size $\eta = 4/(3L)$, uniform GD contracts by exactly $1/3$. By Theorem 3, when $\sigma^2/(L\tau) > 1/2$, exponential loss weighting instead converges to the nonzero attracting two-cycle from every initialization with $z_{2,0} \neq 0$. This gives a global comparison of the three dynamics on the same well-conditioned average objective. Appendix D proves Propositions 3–4 and Corollary 1.

6 Extensions to Momentum and AdamW

GAR can be adapted to optimizers with state by applying the same weight-selection principle to their one-step updates. We state the resulting choices in Algorithm 1 below; Appendix E gives the derivations.

Polyak momentum. Let $\Delta_t := \boldsymbol{x}_t - \boldsymbol{x}_{t-1}$ and $\boldsymbol{y}_t := \boldsymbol{x}_t + \beta\Delta_t$. With weights $\boldsymbol{w}$, the momentum update is

$$\boldsymbol{x}_{t+1} = \boldsymbol{y}_t - \eta \boldsymbol{P}^{-1}\boldsymbol{G}_t\boldsymbol{w}.$$

Expanding $\|\boldsymbol{x}_{t+1} - \boldsymbol{x}^\star\|_{\boldsymbol{P}}^2$ and using convexity gives Algorithm 1 with

$$s_{t,j}^{\mathrm{mom}} = d_{t,j} + \beta\langle \nabla f_{i_{t,j}}(\boldsymbol{x}_t), \Delta_t\rangle, \qquad \boldsymbol{R} = \boldsymbol{P}, \quad \alpha = \eta.$$

Note that $s_{t,j}^{\mathrm{mom}}$ can be viewed as a first-order approximation of $f_{i_{t,j}}(\boldsymbol{y}_t) - f_{i_{t,j}}(\boldsymbol{x}^\star)$. In practice, we use the observable look-ahead loss $f_{i_{t,j}}(\boldsymbol{y}_t)$ as a surrogate for the unavailable oracle gap.

AdamW. For weight selection, we freeze the Adam preconditioner at

$$\boldsymbol{P}_{t-1} := \mathrm{Diag}(\sqrt{\boldsymbol{v}_{t-1}} + \epsilon_{\mathrm{Adam}})$$

and omit decoupled weight decay, which is applied by the subsequent AdamW step. Define

$$\boldsymbol{y}_t := \boldsymbol{x}_t - \eta\beta_1 \boldsymbol{P}_{t-1}^{-1}\boldsymbol{m}_{t-1}, \qquad \alpha_t := \eta(1-\beta_1).$$

With weights $\boldsymbol{w}$, the corresponding approximation to the Adam update is $\widetilde{\boldsymbol{x}}_{t+1} = \boldsymbol{y}_t - \alpha_t \boldsymbol{P}_{t-1}^{-1}\boldsymbol{G}_t\boldsymbol{w}$. The same squared-distance calculation gives Algorithm 1 with

$$s_{t,j} = d_{t,j} + \langle \nabla f_{i_{t,j}}(\boldsymbol{x}_t), \boldsymbol{y}_t - \boldsymbol{x}_t\rangle, \qquad \boldsymbol{R} = \boldsymbol{P}_{t-1}, \quad \alpha = \alpha_t.$$

This score is a first-order approximation of $f_{i_{t,j}}(\boldsymbol{y}_t) - f_{i_{t,j}}(\boldsymbol{x}^\star)$. In practice, we therefore use the observable look-ahead losses $\ell_{t,j}^{\mathrm{LA}} := f_{i_{t,j}}(\boldsymbol{y}_t)$ as surrogates, solving

$$\boldsymbol{w}_t^{\mathrm{GAR}} = \mathrm{softmax}\left(\frac{\boldsymbol{\ell}_t^{\mathrm{LA}} - \alpha_t \boldsymbol{G}_t^\top \boldsymbol{P}_{t-1}^{-1}\boldsymbol{G}_t\boldsymbol{w}_t^{\mathrm{GAR}}}{\tau}\right). \tag{19}$$

After selecting $\boldsymbol{w}_t^{\mathrm{GAR}}$, we stop gradients through the weights and apply standard AdamW to $\sum_j w_{t,j}^{\mathrm{GAR}} f_{i_{t,j}}(\boldsymbol{x}_t)$. Thus $\boldsymbol{y}_t$ and $\boldsymbol{P}_{t-1}$ are used only for weight selection; the actual AdamW step updates its moments and applies decoupled weight decay normally.

7 EXPERIMENTS

We train GPT-style models with 120M, 210M, and 300M parameters from scratch for 20,000 steps on the seven-domain SlimPajama 6B subset, following the training and evaluation pipeline of Sow et al. (2025). Baselines are uniform AdamW and loss-only LinUpper (Sow et al., 2025). GAR evaluates per-example losses at the Adam look-ahead point and uses the previous-step preconditioner. To avoid storing full per-example gradients, we approximate $\boldsymbol{G}_t^\top \boldsymbol{P}_{t-1}^{-1} \boldsymbol{G}_t \boldsymbol{w}$ using only gradients from the last linear readout layer. At each reweighted step, we initialize the fixed-point iteration at uniform weights and perform exactly three iterations of (19) using VJP–preconditioner–JVP products. The resulting weights are detached (stop-gradient) before the weighted training loss is backpropagated.

All methods use the same underlying data mixture and uniform weights for the first 1,000 training steps. After this burn-in period, Uniform AdamW remains uniform, while LinUpper and GAR apply their respective within-minibatch weights. GAR uses a constant temperature τ thereafter. In distributed training, the weights are normalized over the global effective minibatch. All reported numbers are final-step evaluation log-perplexities. Appendix F gives implementation details.

Table 1: Domain-level evaluation log-perplexity at step 20,000 (lower is better).

Method	ArXiv	Books	C4	CC	GitHub	StackEx.	Wiki.	Avg.
120M parameters								
Uniform AdamW	2.482	4.207	4.450	4.420	1.815	2.649	3.114	3.305
LinUpper	2.481	4.159	4.401	4.367	1.840	2.651	3.089	3.284
GAR ($\tau = 1$)	2.654	4.147	4.377	4.340	2.054	2.838	3.230	3.377
GAR ($\tau = 3$)	2.485	4.129	4.355	4.317	1.879	2.660	3.078	**3.272**
GAR ($\tau = 10$)	2.505	4.177	4.418	4.380	1.842	2.654	3.114	3.298
210M parameters								
Uniform AdamW	2.379	4.031	4.268	4.228	1.679	2.500	2.878	3.138
LinUpper	2.354	3.952	4.183	4.137	1.780	2.525	2.873	3.115
GAR ($\tau = 1$)	2.469	3.969	4.181	4.142	1.904	2.672	2.995	3.190
GAR ($\tau = 3$)	2.338	3.958	4.167	4.120	1.704	2.499	2.847	**3.090**
GAR ($\tau = 10$)	2.332	3.996	4.221	4.179	1.677	2.493	2.855	3.108
300M parameters								
Uniform AdamW	2.328	4.032	4.264	4.225	1.666	2.485	2.880	3.126
LinUpper	2.354	3.952	4.193	4.154	1.749	2.522	2.874	3.114
GAR ($\tau = 1$)	2.470	3.972	4.188	4.141	1.895	2.664	3.002	3.190
GAR ($\tau = 3$)	2.330	3.942	4.161	4.120	1.694	2.488	2.830	**3.081**
GAR ($\tau = 10$)	2.315	3.983	4.218	4.176	1.661	2.476	2.848	3.097

Table 1 summarizes the domain-level evaluation results. GAR with $\tau = 3$ achieves the best average log-perplexity at all three model sizes: 3.272, 3.090, and 3.081, improving over LinUpper by 0.012, 0.025, and 0.033, respectively. The results also show a clear dependence on the temperature: $\tau = 1$ degrades performance on GitHub and StackExchange, whereas $\tau = 3$ gives the best average performance at all three scales.

8 CONCLUSION

While reweighted SGD admits a sharpened convergence guarantee under interpolation, without interpolation it incurs an additional gradient-deviation term. In deterministic full-batch optimization, the covariance of component gradients at the ERM stationary point can add local curvature that slows or destabilizes loss-only reweighting. Exponential reweighting can further transition to an attracting two-cycle. GAR incorporates gradient geometry into weight selection, achieves the standard uniform-SGD upper bound minus a nonnegative margin, and remains stable on our hard instance where exponential weighting cycles. Our language-model experiments suggest that the same correction can help after adapting GAR to AdamW.

Our theory does not cover the state-dependent preconditioning and momentum used in the AdamW variant, and computing the GAR weights introduces additional overhead relative to loss-only reweighting. Developing efficient approximations and extending the analysis and design principle to broader optimizer geometries, including normalization-based methods, are left for future work.

AI USE STATEMENT

In this work, we used generative AI tools (OpenAI GPT models and Google Gemini) to assist with several parts of the research and writing process. AI tools provided substantial assistance in developing mathematical derivations and proofs for the theoretical results beyond Proposition 1, including formulating and refining mathematical claims and hypotheses. Gemini was also used to assist with implementing the GAR method used in our experiments. In addition, generative AI tools were used for literature retrieval and review, early-stage manuscript drafting, and suggestions regarding the title of the paper. The core research problem and the GAR formulation were developed by the human authors. The experimental setup follows Sow et al. (2025), and generative AI was not used to design or interpret the experiments. All AI-assisted mathematical arguments were independently checked by the authors. AI-assisted code was inspected and tested by the authors, and literature references and associated claims were checked against the original sources. The authors substantially revised AI-assisted text and take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

All assumptions and complete proofs of the theoretical results are provided in the appendices. Section 7 and Appendix F describe the experimental setup and GAR-specific implementation details; unless otherwise noted, model configurations and training hyperparameters follow the publicly available implementation of Sow et al. (2025). The project-specific code modifications cannot be publicly released due to code-release restrictions.

REFERENCES

Guillaume Alain, Alex Lamb, Chinnadhurai Sankar, Aaron Courville, and Yoshua Bengio. Variance reduction in SGD by distributed importance sampling. *arXiv preprint arXiv:1511.06481*, 2015.

Jing An, Lexing Ying, and Yuhua Zhu. Why resampling outperforms reweighting for correcting sampling bias with stochastic gradients. In *International Conference on Learning Representations*, 2021.

Lukas Balles, Cedric Archambeau, and Giovanni Zappella. A negative result on gradient matching for selective backprop. *arXiv preprint arXiv:2312.05021*, 2023.

Jean-Francois Bonbhel. K-ABENA: K-adaptive backpropagation with error-based N-exclusion algorithm:(compensated loss-based sample exclusion with unbiased gradient estimation). *arXiv preprint arXiv:2607.05903*, 2026.

Zalán Borsos, Andreas Krause, and Kfir Y Levy. Online variance reduction for stochastic optimization. In *Conference On Learning Theory*, pp. 324–357. PMLR, 2018.

Ayoub El Hanchi, David Stephens, and Chris Maddison. Stochastic reweighted gradient descent. In *International Conference on Machine Learning*, pp. 8359–8374. PMLR, 2022.

Siddharth Gopal. Adaptive sampling for SGD by exploiting side information. In *International Conference on Machine Learning*, pp. 364–372. PMLR, 2016.

Robert Gower, Othmane Sebbouh, and Nicolas Loizou. SGD for structured nonconvex functions: Learning rates, minibatching and interpolation. In *International Conference on Artificial Intelligence and Statistics*, pp. 1315–1323. PMLR, 2021.

Angela H Jiang, Daniel L-K Wong, Giulio Zhou, David G Andersen, Jeffrey Dean, Gregory R Ganger, Gauri Joshi, Michael Kaminksy, Michael Kozuch, Zachary C Lipton, et al. Accelerating deep learning by focusing on the biggest losers. *arXiv preprint arXiv:1910.00762*, 2019.

Angelos Katharopoulos and François Fleuret. Biased importance sampling for deep neural network training. *arXiv preprint arXiv:1706.00043*, 2017.

Angelos Katharopoulos and François Fleuret. Not all samples are created equal: Deep learning with importance sampling. In *International conference on machine learning*, pp. 2525–2534. PMLR, 2018.

Kenji Kawaguchi and Haihao Lu. Ordered SGD: A new stochastic optimization framework for empirical risk minimization. In *International Conference on Artificial Intelligence and Statistics*, pp. 669–679. PMLR, 2020.

Ramnath Kumar, Kushal Majmundar, Dheeraj Nagaraj, and Arun Sai Suggala. Stochastic reweighted gradient descent via distributionally robust optimization. *Transactions on Machine Learning*, 2024.

Tian Li, Ahmad Beirami, Maziar Sanjabi, and Virginia Smith. On tilted losses in machine learning: Theory and applications. *Journal of Machine Learning Research*, 24(142):1–79, 2023.

Ilya Loshchilov and Frank Hutter. Online batch selection for faster training of neural networks. *arXiv preprint arXiv:1511.06343*, 2015.

Siyuan Ma, Raef Bassily, and Mikhail Belkin. The power of interpolation: Understanding the effectiveness of SGD in modern over-parametrized learning. In *International Conference on Machine Learning*, pp. 3325–3334. PMLR, 2018.

Sören Mindermann, Jan M Brauner, Muhammed T Razzak, Mrinank Sharma, Andreas Kirsch, Winnie Xu, Benedikt Höltgen, Aidan N Gomez, Adrien Morisot, Sebastian Farquhar, et al. Prioritized training on points that are learnable, worth learning, and not yet learnt. In *International Conference on Machine Learning*, pp. 15630–15649. PMLR, 2022.

Deanna Needell, Nathan Srebro, and Rachel Ward. Stochastic gradient descent, weighted sampling, and the randomized Kaczmarz algorithm. *Advances in neural information processing systems*, 27, 2014.

Neal Parikh and Stephen Boyd. Proximal algorithms. *Foundations and Trends in optimization*, 1(3): 127–239, 2014.

Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. Learning to reweight examples for robust deep learning. In *International conference on machine learning*, pp. 4334–4343. PMLR, 2018.

Daouda Sow, Herbert Woisetschläger, Saikiran Bulusu, Shiqiang Wang, Hans Arno Jacobsen, and Yingbin Liang. Dynamic loss-based sample reweighting for improved large language model pretraining. In *International Conference on Learning Representations*, volume 2025, pp. 89490–89520, 2025.

Sharan Vaswani, Francis Bach, and Mark Schmidt. Fast and faster convergence of SGD for overparameterized models and an accelerated perceptron. In *The 22nd international conference on artificial intelligence and statistics*, pp. 1195–1204. PMLR, 2019.

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. *Advances in Neural Information Processing Systems*, 37:131197–131223, 2024.

Shaobo Wang, Xuan Ouyang, Tianyi Xu, Yuzheng Hu, Jialin Liu, Guo Chen, Tianyu Zhang, Junhao Zheng, Kexin Yang, Xingzhang Ren, et al. OPUS: towards efficient and principled data selection in large language model pre-training in every iteration. In *International Conference on Machine Learning*, 2026.

Peilin Zhao and Tong Zhang. Stochastic optimization with importance sampling for regularized loss minimization. In *international conference on machine learning*, pp. 1–9. PMLR, 2015.

A Additional Discussion of Related Work

Unbiased importance sampling versus biased reweighting. Classical nonuniform sampling often pairs data-dependent probabilities with inverse-probability correction so that the stochastic update remains unbiased for a prescribed objective (Needell et al., 2014; Zhao & Zhang, 2015; Alain et al., 2015; Gopal, 2016; Borsos et al., 2018; Katharopoulos & Fleuret, 2018; El Hanchi et al., 2022). Our setting is different: a minibatch is sampled first and the examples inside it receive simplex weights, so the resulting direction is generally biased. This distinction is why the gradient-deviation term in Theorem 1 need not enjoy the cancellation of an unbiased minibatch estimator.

Loss-biased modified objectives. Several methods intentionally optimize an objective different from the original empirical mean. Katharopoulos & Fleuret (2017) use loss as an importance signal and discuss the soft-maximum objective induced by biased weighting. Kawaguchi & Lu (2020) analyze Ordered SGD through an ordered empirical-risk objective, while KL-DRO/RGD induces exponential current-loss weighting (Kumar et al., 2024). TERM studies a family of exponentially tilted empirical risks whose gradients induce exponential loss weighting (Li et al., 2023).

Selection failures and fixed-point stability. Selective Backprop prioritizes high-loss examples to save backward computation (Jiang et al., 2019). Balles et al. (2023) report that neither their loss-based nor gradient-matching variants consistently outperform random selection under the same selection budget. A concurrent K-ABENA paper (Bonbhel, 2026) studies a different negative phenomenon: uncompensated selection bias can prevent an ERM minimizer from remaining stationary. In our hard instance, by contrast, equal losses and symmetry make the weighted update exactly zero at the ERM optimum. The failure occurs because reweighting changes the local Jacobian around that fixed point, which can make it poorly conditioned or repelling without moving the fixed point itself.

Gradient-aware data weighting and selection. Meta-reweighting uses a clean validation gradient (Ren et al., 2018); GREATS selects subsets using a Taylor approximation of validation-loss improvement (Wang et al., 2024); and OPUS scores candidates in optimizer-induced update space while accounting for redundancy (Wang et al., 2026). Thus using gradient or optimizer information for data selection is not itself new. GAR addresses a narrower problem: continuous weighting within the minibatch already selected for training, using no external validation or proxy set, with a global comparison to uniform minibatch SGD and a local stability comparison to loss-only weighting.

B Proofs for the Global Weighted-SGD Results

Lemma 1 (Metric self-bounding). *If g is convex and L-smooth in $\|\cdot\|_{\boldsymbol{P}}$, then*

$$\|\nabla g(\boldsymbol{x}) - \nabla g(\boldsymbol{y})\|^2_{\boldsymbol{P}^{-1}} \le 2L\big(g(\boldsymbol{x}) - g(\boldsymbol{y}) - \langle \nabla g(\boldsymbol{y}), \boldsymbol{x} - \boldsymbol{y}\rangle\big).$$

If $\boldsymbol{y}$ minimizes g, then $\|\nabla g(\boldsymbol{x})\|^2_{\boldsymbol{P}^{-1}} \le 2L(g(\boldsymbol{x}) - g(\boldsymbol{y}))$.

Proof. Let $\boldsymbol{z} = \boldsymbol{P}^{1/2}\boldsymbol{x}$, $\boldsymbol{u} = \boldsymbol{P}^{1/2}\boldsymbol{y}$, and define

$$\widetilde{g}(\boldsymbol{z}) = g(\boldsymbol{P}^{-1/2}\boldsymbol{z}).$$

Then

$$\nabla\widetilde{g}(\boldsymbol{z}) = \boldsymbol{P}^{-1/2}\nabla g(\boldsymbol{P}^{-1/2}\boldsymbol{z}),$$

and hence

$$\|\nabla\widetilde{g}(\boldsymbol{z}) - \nabla\widetilde{g}(\boldsymbol{u})\| = \|\nabla g(\boldsymbol{x}) - \nabla g(\boldsymbol{y})\|_{\boldsymbol{P}^{-1}} \le L\|\boldsymbol{x} - \boldsymbol{y}\|_{\boldsymbol{P}} = L\|\boldsymbol{z} - \boldsymbol{u}\|.$$

Thus $\widetilde{g}$ is convex and L-smooth in the Euclidean norm.

Now define

$$h(\boldsymbol{z}) := \widetilde{g}(\boldsymbol{z}) - \langle\nabla\widetilde{g}(\boldsymbol{u}), \boldsymbol{z}\rangle.$$

The function h is also convex and L-smooth, and $\nabla h(\boldsymbol{u}) = \boldsymbol{0}$, so $\boldsymbol{u}$ is a minimizer of h. By the standard descent inequality for an L-smooth function,

$$h\left(\boldsymbol{z} - \frac{1}{L}\nabla h(\boldsymbol{z})\right) \le h(\boldsymbol{z}) - \frac{1}{2L}\|\nabla h(\boldsymbol{z})\|^2.$$

Since $\boldsymbol{u}$ minimizes h,

$$h(\boldsymbol{u}) \le h\left(\boldsymbol{z} - \frac{1}{L}\nabla h(\boldsymbol{z})\right),$$

and therefore

$$\|\nabla\widetilde{g}(\boldsymbol{z}) - \nabla\widetilde{g}(\boldsymbol{u})\|^2 \le 2L\big(\widetilde{g}(\boldsymbol{z}) - \widetilde{g}(\boldsymbol{u}) - \langle\nabla\widetilde{g}(\boldsymbol{u}), \boldsymbol{z} - \boldsymbol{u}\rangle\big).$$

Substituting $\boldsymbol{z} = \boldsymbol{P}^{1/2}\boldsymbol{x}$ and $\boldsymbol{u} = \boldsymbol{P}^{1/2}\boldsymbol{y}$ gives

$$\|\nabla g(\boldsymbol{x}) - \nabla g(\boldsymbol{y})\|^2_{\boldsymbol{P}^{-1}} \le 2L\big(g(\boldsymbol{x}) - g(\boldsymbol{y}) - \langle\nabla g(\boldsymbol{y}), \boldsymbol{x} - \boldsymbol{y}\rangle\big).$$

Finally, if $\boldsymbol{y}$ minimizes the differentiable convex function g, then $\nabla g(\boldsymbol{y}) = \boldsymbol{0}$, which yields

$$\|\nabla g(\boldsymbol{x})\|^2_{\boldsymbol{P}^{-1}} \le 2L\big(g(\boldsymbol{x}) - g(\boldsymbol{y})\big).$$

□

B.1 Proof of Proposition 1

Proof. Write $\boldsymbol{e}_t = \boldsymbol{x}_t - \boldsymbol{x}^\star$ and $\boldsymbol{g}_t = \boldsymbol{G}_t\boldsymbol{w}_t$. Exact expansion gives

$$\|\boldsymbol{e}_{t+1}\|^2_{\boldsymbol{P}} = \|\boldsymbol{e}_t\|^2_{\boldsymbol{P}} - 2\eta\langle\boldsymbol{g}_t, \boldsymbol{e}_t\rangle + \eta^2\|\boldsymbol{g}_t\|^2_{\boldsymbol{P}^{-1}}. \tag{20}$$

Convexity gives

$$\langle\boldsymbol{g}_t, \boldsymbol{e}_t\rangle \ge \sum_{j=1}^{b} w_{t,j}d_{t,j}.$$

Jensen's inequality and Lemma 1, using interpolation, give

$$\|\boldsymbol{g}_t\|^2_{\boldsymbol{P}^{-1}} \le \sum_{j=1}^{b} w_{t,j}\|\nabla f_{i_{t,j}}(\boldsymbol{x}_t)\|^2_{\boldsymbol{P}^{-1}} \le 2L\sum_{j=1}^{b} w_{t,j}d_{t,j}.$$

Substituting into (20) yields

$$\|\boldsymbol{e}_{t+1}\|^2_{\boldsymbol{P}} \le \|\boldsymbol{e}_t\|^2_{\boldsymbol{P}} - 2\eta(1 - \eta L)\sum_{j=1}^{b} w_{t,j}d_{t,j}.$$

Since $\eta \le 1/(2L)$, we have $2\eta(1 - \eta L) \ge \eta$, and hence

$$\|\boldsymbol{e}_{t+1}\|^2_{\boldsymbol{P}} \le \|\boldsymbol{e}_t\|^2_{\boldsymbol{P}} - \eta\sum_{j=1}^{b} w_{t,j}d_{t,j}.$$

Decomposing the weighted loss gap around uniform weights gives

$$\sum_{j=1}^{b} w_{t,j}d_{t,j} = \frac{1}{b}\sum_{j=1}^{b} d_{t,j} + \sum_{j=1}^{b}\left(w_{t,j} - \frac{1}{b}\right)d_{t,j}.$$

Therefore,

$$\|\boldsymbol{e}_{t+1}\|^2_{\boldsymbol{P}} \le \|\boldsymbol{e}_t\|^2_{\boldsymbol{P}} - \frac{\eta}{b}\sum_{j=1}^{b} d_{t,j} + \eta\sum_{j=1}^{b}\left(\frac{1}{b} - w_{t,j}\right)d_{t,j}.$$

Taking conditional expectation with respect to the minibatch draw at iteration t, and using uniform sampling with replacement,

$$\mathbb{E}_t\left[\frac{1}{b}\sum_{j=1}^{b} d_{t,j}\right] = f(\boldsymbol{x}_t) - f(\boldsymbol{x}^\star).$$

Hence

$$\eta\big(f(\boldsymbol{x}_t) - f(\boldsymbol{x}^\star)\big) \le \|\boldsymbol{e}_t\|^2_{\boldsymbol{P}} - \mathbb{E}_t\|\boldsymbol{e}_{t+1}\|^2_{\boldsymbol{P}} + \eta\,\mathbb{E}_t\left[\sum_{j=1}^{b}\left(\frac{1}{b} - w_{t,j}\right)d_{t,j}\right].$$

Taking total expectations and using the tower property yields

$$\eta\,\mathbb{E}\big[f(\boldsymbol{x}_t)-f(\boldsymbol{x}^\star)\big]\le \mathbb{E}\|\boldsymbol{e}_t\|_{\boldsymbol{P}}^2-\mathbb{E}\|\boldsymbol{e}_{t+1}\|_{\boldsymbol{P}}^2+\eta\,\mathbb{E}\left[\sum_{j=1}^{b}\left(\frac{1}{b}-w_{t,j}\right)d_{t,j}\right].$$

Summing over $t=0,\dots,T-1$ gives

$$\eta\sum_{t=0}^{T-1}\mathbb{E}\big[f(\boldsymbol{x}_t)-f(\boldsymbol{x}^\star)\big]\le \|\boldsymbol{x}_0-\boldsymbol{x}^\star\|_{\boldsymbol{P}}^2-\mathbb{E}\|\boldsymbol{x}_T-\boldsymbol{x}^\star\|_{\boldsymbol{P}}^2+\eta\sum_{t=0}^{T-1}\mathbb{E}\left[\sum_{j=1}^{b}\left(\frac{1}{b}-w_{t,j}\right)d_{t,j}\right].$$

Dropping the nonnegative final-distance term and dividing by ηT,

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\big[f(\boldsymbol{x}_t)-f(\boldsymbol{x}^\star)\big]\le \frac{\|\boldsymbol{x}_0-\boldsymbol{x}^\star\|_{\boldsymbol{P}}^2}{\eta T}+\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\sum_{j=1}^{b}\left(\frac{1}{b}-w_{t,j}\right)d_{t,j}\right].$$

Finally, by convexity of f,

$$f(\bar{\boldsymbol{x}}_T)\le\frac{1}{T}\sum_{t=0}^{T-1}f(\boldsymbol{x}_t),$$

so

$$\mathbb{E}\big[f(\bar{\boldsymbol{x}}_T)-f(\boldsymbol{x}^\star)\big]\le \frac{\|\boldsymbol{x}_0-\boldsymbol{x}^\star\|_{\boldsymbol{P}}^2}{\eta T}+\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\sum_{j=1}^{b}\left(\frac{1}{b}-w_{t,j}\right)d_{t,j}\right],$$

which proves the claim. □

B.2 Proof of Theorem 1

Proof. Because weights sum to one,

$$\boldsymbol{G}_t\boldsymbol{w}_t=\nabla f(\boldsymbol{x}_t)+\sum_{j=1}^{b}w_{t,j}(\nabla f_{i_{t,j}}(\boldsymbol{x}_t)-\nabla f(\boldsymbol{x}_t)).$$

Thus

$$\begin{aligned}\|\boldsymbol{G}_t\boldsymbol{w}_t\|_{\boldsymbol{P}^{-1}}^2&\le 2\|\nabla f(\boldsymbol{x}_t)\|_{\boldsymbol{P}^{-1}}^2+2\mathcal{V}_t\\&\le 4L(f(\boldsymbol{x}_t)-f(\boldsymbol{x}^\star))+2\mathcal{V}_t.\end{aligned}$$

Convexity also gives $\langle\boldsymbol{G}_t\boldsymbol{w}_t,\boldsymbol{e}_t\rangle\ge\sum_{j=1}^{b}w_{t,j}d_{t,j}$. Substitution into the exact distance expansion yields

$$\|\boldsymbol{e}_{t+1}\|_{\boldsymbol{P}}^2\le\|\boldsymbol{e}_t\|_{\boldsymbol{P}}^2-2\eta\sum_{j=1}^{b}w_{t,j}d_{t,j}+4\eta^2L(f(\boldsymbol{x}_t)-f(\boldsymbol{x}^\star))+2\eta^2\mathcal{V}_t.$$

Add and subtract the uniform minibatch gap and take conditional expectation:

$$\begin{aligned}\mathbb{E}_t\|\boldsymbol{e}_{t+1}\|_{\boldsymbol{P}}^2\le\|\boldsymbol{e}_t\|_{\boldsymbol{P}}^2-2\eta(1-2\eta L)(f(\boldsymbol{x}_t)-f(\boldsymbol{x}^\star))+2\eta^2\mathbb{E}_t\mathcal{V}_t\\+2\eta\mathbb{E}_t\sum_{j=1}^{b}\left(\frac{1}{b}-w_{t,j}\right)d_{t,j}.\end{aligned}$$

For $\eta\le 1/(4L)$, the descent coefficient is at least η. Taking total expectations, rearranging, summing over t, and applying convexity to the averaged iterate proves (2). □

B.3 Gradient-deviation bounds and tightness

Let $\boldsymbol{\delta}_i = \nabla f_i(\boldsymbol{x}_t) - \nabla f(\boldsymbol{x}_t)$. Jensen's inequality yields

$$\mathcal{V}_t \le \sum_{j=1}^{b} w_{t,j} \|\boldsymbol{\delta}_{i_{t,j}}\|_{\boldsymbol{P}^{-1}}^2 \le \sum_{j=1}^{b} \|\boldsymbol{\delta}_{i_{t,j}}\|_{\boldsymbol{P}^{-1}}^2.$$

Taking conditional expectations proves $\mathbb{E}_t \mathcal{V}_t \le b\sigma^2$. For uniform weights, independence and $n^{-1}\sum_i \boldsymbol{\delta}_i = \mathbf{0}$ give

$$\mathbb{E}_t \left\| \frac{1}{b} \sum_{j=1}^{b} \boldsymbol{\delta}_{i_{t,j}} \right\|_{\boldsymbol{P}^{-1}}^2 = \frac{1}{bn} \sum_{i=1}^{n} \|\boldsymbol{\delta}_i\|_{\boldsymbol{P}^{-1}}^2.$$

For tightness, take collinear deviations $\boldsymbol{\delta}_1 = \sqrt{n-1}\sigma \boldsymbol{v}$ and $\boldsymbol{\delta}_i = -\sigma \boldsymbol{v}/\sqrt{n-1}$ for $2 \le i \le n$, where $\|\boldsymbol{v}\|_{\boldsymbol{P}^{-1}} = 1$. They have mean zero and average squared dual norm σ^2. Give all weight to an occurrence of index 1 whenever at least one appears, and otherwise choose any sampled component. Since index 1 appears with probability $p_b = 1 - (1 - 1/n)^b$, this choice gives

$$\mathbb{E}_t \mathcal{V}_t = p_b(n-1)\sigma^2 + (1 - p_b)\frac{\sigma^2}{n-1},$$

which approaches $b\sigma^2$ when n is large relative to b.

C Proofs for the Deterministic Stability Results

C.1 Proof of Proposition 2

Proof. Because $\boldsymbol{d}(\boldsymbol{x}^\star) = \mathbf{0}$, permutation equivariance gives $\Phi(\mathbf{0}) = \boldsymbol{u} = n^{-1}\mathbf{1}$. Let $\boldsymbol{A} := D\Phi(\mathbf{0})$. Differentiating $\Phi(\Pi \boldsymbol{s}) = \Pi \Phi(\boldsymbol{s})$ at $\boldsymbol{s} = \mathbf{0}$ gives $\boldsymbol{A}\Pi = \Pi \boldsymbol{A}$ for every permutation matrix Π. Hence all diagonal entries of $\boldsymbol{A}$ are equal and all off-diagonal entries are equal, so $\boldsymbol{A} = a\boldsymbol{I} + c_0 \mathbf{1}\mathbf{1}^\top$. Because $\Phi(\boldsymbol{s}) \in \Delta_n$, $\mathbf{1}^\top \Phi(\boldsymbol{s}) = 1$ and therefore $\mathbf{1}^\top \boldsymbol{A} = 0$. Thus $c_0 = -a/n$. Defining $\gamma_\Phi := na$ gives

$$D\Phi(\mathbf{0}) = \frac{\gamma_\Phi}{n}\left(\boldsymbol{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top\right) = \frac{\gamma_\Phi}{n}\Pi_\perp.$$

Let $\boldsymbol{G}_\star = [\boldsymbol{g}_1^\star, \dots, \boldsymbol{g}_n^\star]$. Since $\nabla f(\boldsymbol{x}^\star) = 0$, $\boldsymbol{G}_\star \mathbf{1} = 0$, while $D\boldsymbol{d}(\boldsymbol{x}^\star) = \boldsymbol{G}_\star^\top$. Differentiating the weighted direction $\boldsymbol{G}(\boldsymbol{x})\Phi(\boldsymbol{d}(\boldsymbol{x}))$ at $\boldsymbol{x}^\star$ gives

$$\frac{1}{n}\sum_i \nabla^2 f_i(\boldsymbol{x}^\star) + \boldsymbol{G}_\star D\Phi(\mathbf{0})\boldsymbol{G}_\star^\top = \boldsymbol{H}_\star + \frac{\gamma_\Phi}{n}\boldsymbol{G}_\star \boldsymbol{G}_\star^\top = \boldsymbol{H}_\star + \gamma_\Phi \boldsymbol{C}_\star.$$

The original-coordinate update Jacobian is therefore $\boldsymbol{I} - \eta \boldsymbol{P}^{-1}(\boldsymbol{H}_\star + \gamma_\Phi \boldsymbol{C}_\star)$. Conjugating by $\boldsymbol{P}^{1/2}$ proves (6). □

C.2 Proof of Theorem 2

Proof. Fix $\boldsymbol{P} \succ 0$ and let $\boldsymbol{H}_0 = \mathrm{diag}(L/2, L)$. The hard instance (7) satisfies

$$\nabla f_s(\boldsymbol{x}) = \boldsymbol{P}^{1/2}(\boldsymbol{H}_0 \boldsymbol{z} + s\sigma \boldsymbol{e}_2), \qquad s \in \{-1, +1\}.$$

Hence both components are L-smooth and $(L/2)$-strongly convex in the $\boldsymbol{P}$-metric, their average has Hessian $\boldsymbol{H}_0$, and the two component-gradient deviations at every point are $\pm\sigma \boldsymbol{P}^{1/2}\boldsymbol{e}_2$, each with squared dual norm σ^2. At $\boldsymbol{x}^\star$, $f_{-1}(\boldsymbol{x}^\star) = f_{+1}(\boldsymbol{x}^\star) = c = \sigma^2/(2L)$, $\bar{\boldsymbol{H}}_\star = \boldsymbol{H}_0$, and $\bar{\boldsymbol{C}}_\star = \sigma^2 \boldsymbol{e}_2 \boldsymbol{e}_2^\top$. Hence $\boldsymbol{\ell}(\boldsymbol{x}) = \boldsymbol{d}(\boldsymbol{x}) + c\mathbf{1}$ for every $\boldsymbol{x}$, so shift invariance gives $\Phi(\boldsymbol{\ell}(\boldsymbol{x})) = \Phi(\boldsymbol{d}(\boldsymbol{x}))$. Proposition 2 therefore applies directly to the current-loss update and gives the eigenvalues (8).

For $\gamma_\Phi > 0$, the second stability condition dominates the first, so local linear stability requires $\eta < 2/(L + \gamma_\Phi \sigma^2)$. The optimal scalar step for the endpoint curvatures $\alpha = L/2$ and $\beta = L + \gamma_\Phi \sigma^2$ is $\eta_\star = 2/(\alpha + \beta)$, with

$$\rho_\Phi^\star = \frac{\beta - \alpha}{\beta + \alpha} = \frac{L/2 + \gamma_\Phi \sigma^2}{3L/2 + \gamma_\Phi \sigma^2}.$$

Writing $\kappa_\Phi = \beta/\alpha = 2 + 2\gamma_\Phi\sigma^2/L$ gives $\rho_\Phi^\star = (\kappa_\Phi - 1)/(\kappa_\Phi + 1)$. The number of linearized iterations needed to reduce a worst-direction perturbation by a factor ϵ is $\Theta(\kappa_\Phi \log(1/\epsilon))$. Uniform GD sees only $\boldsymbol{H}_0$ and, at $\eta = 4/(3L)$, contracts by $1/3$. If $\gamma_\Phi < 0$ and $\sigma^2 > L/|\gamma_\Phi|$, then $L + \gamma_\Phi\sigma^2 < 0$, so the second Jacobian eigenvalue $1 - \eta(L + \gamma_\Phi\sigma^2)$ is strictly larger than one for every $\eta > 0$. This proves the theorem. □

C.3 A BROADER CYCLE RESULT FOR MONOTONE LOSS-ONLY WEIGHTING

For a permutation-equivariant and shift-invariant weighting map $\Phi : \mathbb{R}^2 \to \Delta_2$, define

$$\psi(s) := \Phi_1(s, -s) - \Phi_2(s, -s).$$

Permutation equivariance makes ψ odd.

Proposition 5 (Attracting cycles beyond exponential weighting)**.** *Suppose ψ is nontrivial, nondecreasing, and continuously differentiable on $(0, \infty)$. Then for every $0 < \eta < 2/L$ for which uniform GD is stable on (7), there exists $\sigma > 0$ such that the Φ-reweighted dynamics on the same hard instance have a nonzero locally asymptotically stable two-cycle. This statement can include schemes with $\gamma_\Phi = 0$ or schemes whose weighting maps are nonsmooth at exact equality, provided the stated regularity holds for positive loss differences.*

C.4 PROOF OF PROPOSITION 5

Proof. By shift invariance, on (7) the weights depend only on the centered pair $(\sigma z_2, -\sigma z_2)$. Thus the exact dynamics are

$$z_1^+ = (1 - \eta L/2)z_1, \qquad z_2^+ = F(z_2) := (1 - \eta L)z_2 - \eta\sigma\psi(\sigma z_2).$$

ψ is odd by permutation equivariance and bounded by one because Φ takes values in the simplex. Since it is nontrivial and nondecreasing on $(0, \infty)$, there exists $s_0 > 0$ with $\psi(s_0) > 0$. We claim that for some $s > 0$,

$$0 \le r_s := \frac{s\psi'(s)}{\psi(s)} < 1. \tag{21}$$

If instead $s\psi'(s) \ge \psi(s)$ whenever $\psi(s) > 0$, then $(\psi(s)/s)' \ge 0$ wherever $\psi(s) > 0$. Starting from any s_0 with $\psi(s_0) > 0$ would imply $\psi(s) \ge [\psi(s_0)/s_0]s$ for all sufficiently large s, contradicting boundedness. Hence (21) holds at some differentiability point (indeed everywhere is a differentiability point under the stated C^1 assumption).

Fix any $0 < \eta < 2/L$ and choose

$$\sigma^2 = \left(\frac{2}{\eta} - L\right)\frac{s}{\psi(s)} > 0, \qquad a := \frac{s}{\sigma}.$$

Then

$$(2 - \eta L)a = \eta\sigma\psi(s),$$

so $F(a) = -a$. Oddness gives $F(-a) = a$, hence $\{a, -a\}$ is a nonzero two-cycle. Its one-step derivative at either cycle point is

$$F'(a) = 1 - \eta L - \eta\sigma^2\psi'(s) = -1 + (2 - \eta L)(1 - r_s).$$

Because $0 < 2 - \eta L < 2$ and $0 \le r_s < 1$, we have $-1 < F'(a) < 1$. Thus the second-coordinate multiplier of the two-step map is $F'(a)^2 < 1$. The first-coordinate multiplier is $(1 - \eta L/2)^2 < 1$ for $0 < \eta < 2/L$. Therefore the full two-cycle is locally asymptotically stable and has an open basin. □

C.5 PROOF OF THEOREM 3

Proof. For $\Phi_\tau(\boldsymbol{\ell}) = \mathrm{softmax}(\boldsymbol{\ell}/\tau)$, we have $\psi(s) = \tanh(s/\tau)$, so (9) follows. A fixed point satisfies $z_1 = 0$ and

$$Lz_2 + \sigma\tanh(\sigma z_2/\tau) = 0,$$

whose unique solution is $z_2 = 0$ because the two terms have the same sign. Thus $\boldsymbol{x}^\star$ is the unique fixed point. Define $F_\eta(z) := (1-\eta L)z - \eta\sigma\tanh(\sigma z/\tau)$. Its derivative satisfies

$$1 - \eta(L + \sigma^2/\tau) \le F'_\eta(z) \le 1 - \eta L.$$

If $0 < \eta < \eta_c = 2/(L + \sigma^2/\tau)$, the global Lipschitz constant of the full map is exactly (10). Balancing the endpoint curvatures gives $\eta_\star = 2/(3L/2 + \sigma^2/\tau)$ and $\rho_\star = (L/2 + \sigma^2/\tau)/(3L/2 + \sigma^2/\tau)$, yielding the stated $\Theta((1 + \sigma^2/(L\tau))\log(1/\epsilon))$ global complexity.

Now suppose $\eta_c < \eta < 2/L$. A symmetric two-cycle is determined by a positive solution of $F_\eta(a) = -a$. With $u = \sigma a/\tau$, this is

$$\eta(\sigma^2/\tau)\tanh u = (2 - \eta L)u.$$

The left-minus-right side is strictly concave on $u > 0$, has value zero and positive derivative at zero, and tends to $-\infty$, so there is a unique positive root u_η. At the corresponding a_η,

$$F'_\eta(a_\eta) = -1 + (2 - \eta L)\left(1 - \frac{2u_\eta}{\sinh(2u_\eta)}\right) \in (-1, 1),$$

so the cycle is locally asymptotically stable. If also $\eta \ge 1/L$, then for $z > 0$, $F_\eta(z) < 0$ and the magnitude map

$$G_\eta(z) := -F_\eta(z) = (\eta L - 1)z + \eta\sigma\tanh(\sigma z/\tau)$$

is increasing and strictly concave, with derivative above one at zero and asymptotic slope below one. Its unique positive fixed point is a_η, and monotonicity/concavity imply $G^t_\eta(z) \to a_\eta$ for every $z > 0$. Odd symmetry handles $z < 0$, while $z_{1,t} \to 0$. Finally, at $\eta = 4/(3L)$ the condition $\eta > \eta_c$ is exactly $\sigma^2/(L\tau) > 1/2$. □

C.6 Relation to the normalized practical LinUpper scheme

The released implementation of Sow et al. (2025) uses current per-example losses, global-minibatch range normalization, the capped LinUpper transform, and exponential scaling. Its exact piecewise weighting map is therefore not identical to the smooth class in Proposition 2. The broader cycle result above allows some nonsmoothness at equality and weighting maps that are locally flat away from it, but the implementation also contains a numerical range floor and additional piecewise operations. We therefore treat LinUpper as a motivating loss-only baseline rather than claiming it as a formal corollary of the stability theorems.

D Proofs and Further Properties of Geometry-Aware Reweighting

D.1 Stationarity of the entropy-regularized problem

For $\tau > 0$, the minimizer lies in the simplex interior. Introducing a multiplier λ for $\mathbf{1}^\top \boldsymbol{w} = 1$, stationarity of (12) gives

$$-\boldsymbol{d}_t + \eta \boldsymbol{G}_t^\top \boldsymbol{P}^{-1}\boldsymbol{G}_t\boldsymbol{w} + \tau(\mathbf{1} + \log \boldsymbol{w}) + \lambda\mathbf{1} = \mathbf{0}.$$

Exponentiating and normalizing gives (13).

D.2 Proof of Proposition 4

Proof. Fix t and define

$$\boldsymbol{r}_t(\boldsymbol{y}) := \boldsymbol{d}_t + \boldsymbol{G}_t^\top(\boldsymbol{y} - \boldsymbol{x}_t), \qquad \boldsymbol{w}(\boldsymbol{y}) := \mathrm{softmax}(\boldsymbol{r}_t(\boldsymbol{y})/\tau).$$

Differentiating (16) gives $\nabla\mathcal{M}_t(\boldsymbol{y}) = \boldsymbol{G}_t\boldsymbol{w}(\boldsymbol{y})$. At $\boldsymbol{y} = \boldsymbol{x}_t$, $\boldsymbol{w}(\boldsymbol{x}_t) = \mathrm{softmax}(\boldsymbol{d}_t/\tau)$, which yields the explicit identity in (17).

Let $\widehat{\boldsymbol{w}} = \boldsymbol{w}_t^{\mathrm{GAR}}$ and $\widehat{\boldsymbol{y}} = \boldsymbol{x}_t - \eta\boldsymbol{P}^{-1}\boldsymbol{G}_t\widehat{\boldsymbol{w}}$. The first-order condition for (12) gives

$$\widehat{\boldsymbol{w}} = \mathrm{softmax}\left(\frac{\boldsymbol{d}_t - \eta\boldsymbol{G}_t^\top\boldsymbol{P}^{-1}\boldsymbol{G}_t\widehat{\boldsymbol{w}}}{\tau}\right) = \mathrm{softmax}\left(\frac{\boldsymbol{d}_t + \boldsymbol{G}_t^\top(\widehat{\boldsymbol{y}} - \boldsymbol{x}_t)}{\tau}\right) = \boldsymbol{w}(\widehat{\boldsymbol{y}}).$$

Hence

$$\boldsymbol{P}(\widehat{\boldsymbol{y}} - \boldsymbol{x}_t)/\eta + \nabla \mathcal{M}_t(\widehat{\boldsymbol{y}}) = \boldsymbol{0},$$

which is the implicit identity in (17) and the first-order condition for minimizing $\mathcal{M}_t(\boldsymbol{y}) + (2\eta)^{-1}\|\boldsymbol{y} - \boldsymbol{x}_t\|_{\boldsymbol{P}}^2$. The quadratic term makes this objective strongly convex, so the minimizer is unique. □

D.3 Proof of Proposition 3

Proof. In the full-batch setting define, for current point $\boldsymbol{x}$ and candidate next point $\boldsymbol{y}$,

$$m_i(\boldsymbol{y}; \boldsymbol{x}) := d_i(\boldsymbol{x}) + \langle \nabla f_i(\boldsymbol{x}), \boldsymbol{y} - \boldsymbol{x} \rangle, \qquad \mathcal{M}_{\boldsymbol{x}}(\boldsymbol{y}) := \tau \log \left(\frac{1}{n} \sum_i e^{m_i(\boldsymbol{y}; \boldsymbol{x})/\tau} \right),$$

and let $\boldsymbol{S}(\boldsymbol{x}, \boldsymbol{y}) := \nabla_{\boldsymbol{y}} \mathcal{M}_{\boldsymbol{x}}(\boldsymbol{y})$. Proposition 4 gives the implicit update map $\boldsymbol{T}(\boldsymbol{x})$ through

$$\boldsymbol{T}(\boldsymbol{x}) = \boldsymbol{x} - \eta \boldsymbol{P}^{-1} \boldsymbol{S}(\boldsymbol{x}, \boldsymbol{T}(\boldsymbol{x})). \tag{22}$$

Here, (22) represents the GAR update: the weights implicit in $\boldsymbol{S}(\boldsymbol{x}, \boldsymbol{T}(\boldsymbol{x}))$ are generally nonuniform and become uniform only at $(\boldsymbol{x}^\star, \boldsymbol{x}^\star)$. At $(\boldsymbol{x}^\star, \boldsymbol{x}^\star)$, $m_i(\boldsymbol{x}^\star; \boldsymbol{x}^\star) = 0$ for every i, the softmax weights are uniform, and $\boldsymbol{S}(\boldsymbol{x}^\star, \boldsymbol{x}^\star) = \nabla f(\boldsymbol{x}^\star) = \boldsymbol{0}$.

Holding $\boldsymbol{x}$ fixed, differentiating with respect to $\boldsymbol{y}$ gives

$$D_{\boldsymbol{y}} \boldsymbol{S}(\boldsymbol{x}^\star, \boldsymbol{x}^\star) = \frac{1}{\tau} \boldsymbol{C}_\star.$$

For the partial derivative with respect to $\boldsymbol{x}$, note that

$$D_{\boldsymbol{x}} m_i(\boldsymbol{y}; \boldsymbol{x}) = \nabla^2 f_i(\boldsymbol{x})(\boldsymbol{y} - \boldsymbol{x})$$

when $\boldsymbol{y}$ is held fixed: the derivative of $d_i(\boldsymbol{x})$ cancels the derivative of $-\langle \nabla f_i(\boldsymbol{x}), \boldsymbol{x} \rangle$. Thus $D_{\boldsymbol{x}} m_i(\boldsymbol{x}^\star; \boldsymbol{x}^\star) = 0$, so the weights contribute no first-order term through their $\boldsymbol{x}$-dependence, while differentiating the component gradients yields

$$D_{\boldsymbol{x}} \boldsymbol{S}(\boldsymbol{x}^\star, \boldsymbol{x}^\star) = \frac{1}{n} \sum_i \nabla^2 f_i(\boldsymbol{x}^\star) = \boldsymbol{H}_\star.$$

Differentiating (22) therefore gives

$$\left(\boldsymbol{I} + \eta \boldsymbol{P}^{-1} \boldsymbol{C}_\star / \tau \right) D\boldsymbol{T}(\boldsymbol{x}^\star) = \boldsymbol{I} - \eta \boldsymbol{P}^{-1} \boldsymbol{H}_\star.$$

Conjugating by $\boldsymbol{P}^{1/2}$ proves (15). Since $\bar{\boldsymbol{C}}_\star \succeq 0$, all eigenvalues of $(\boldsymbol{I} + \eta \bar{\boldsymbol{C}}_\star / \tau)^{-1}$ lie in $(0, 1]$, proving the norm inequality. □

D.4 Proof of Theorem 4

Proof. Define $R_t(\boldsymbol{w}) := -\boldsymbol{d}_t^\top \boldsymbol{w} + (\eta/2)\|\boldsymbol{G}_t \boldsymbol{w}\|_{\boldsymbol{P}^{-1}}^2$. Since $h(\boldsymbol{w}) - h(\boldsymbol{u}) = \mathrm{KL}(\boldsymbol{w} \| \boldsymbol{u})$ for $\boldsymbol{u} = \frac{1}{b}\boldsymbol{1}$,

$$R_t(\boldsymbol{u}) - R_t(\boldsymbol{w}_t^{\mathrm{GAR}}) = Q_t(\boldsymbol{u}) - Q_t(\boldsymbol{w}_t^{\mathrm{GAR}}) + \tau \, \mathrm{KL}(\boldsymbol{w}_t^{\mathrm{GAR}} \| \boldsymbol{u}) = \Gamma_t.$$

Applying (11) with $\boldsymbol{w}_t^{\mathrm{GAR}}$ and replacing its R_t value by the uniform comparator gives

$$\|\boldsymbol{x}_{t+1} - \boldsymbol{x}^\star\|_{\boldsymbol{P}}^2 \le \|\boldsymbol{x}_t - \boldsymbol{x}^\star\|_{\boldsymbol{P}}^2 - 2\eta \boldsymbol{d}_t^\top \boldsymbol{u} + \eta^2 \|\boldsymbol{G}_t \boldsymbol{u}\|_{\boldsymbol{P}^{-1}}^2 - 2\eta \Gamma_t.$$

Conditionally, $\mathbb{E}_t[\boldsymbol{d}_t^\top \boldsymbol{u}] = f(\boldsymbol{x}_t) - f(\boldsymbol{x}^\star)$ and

$$\mathbb{E}_t \|\boldsymbol{G}_t \boldsymbol{u}\|_{\boldsymbol{P}^{-1}}^2 \le \|\nabla f(\boldsymbol{x}_t)\|_{\boldsymbol{P}^{-1}}^2 + \frac{\sigma^2}{b} \le 2L(f(\boldsymbol{x}_t) - f(\boldsymbol{x}^\star)) + \frac{\sigma^2}{b}.$$

Therefore,

$$\mathbb{E}_t \|\boldsymbol{x}_{t+1} - \boldsymbol{x}^\star\|_{\boldsymbol{P}}^2 \le \|\boldsymbol{x}_t - \boldsymbol{x}^\star\|_{\boldsymbol{P}}^2 - 2\eta(1 - \eta L)(f(\boldsymbol{x}_t) - f(\boldsymbol{x}^\star)) + \frac{\eta^2 \sigma^2}{b} - 2\eta \mathbb{E}_t[\Gamma_t]. \tag{23}$$

If $\eta \le 1/(2L)$, then $2(1 - \eta L) \ge 1$. Taking total expectations, rearranging, summing over t, and using convexity of f at $\bar{\boldsymbol{x}}_T$ proves (14). □

Corollary 2 (Strongly-convex minibatch guarantee)**.** *If, in addition, f is μ-strongly convex in $\|\cdot\|_{\boldsymbol{P}}$ and $\rho := 1 - \eta\mu(1-\eta L)$, then*

$$\mathbb{E}\|\boldsymbol{x}_T - \boldsymbol{x}^\star\|_{\boldsymbol{P}}^2 \le \rho^T \|\boldsymbol{x}_0 - \boldsymbol{x}^\star\|_{\boldsymbol{P}}^2 + \frac{\eta^2\sigma^2}{b}\frac{1-\rho^T}{1-\rho} - 2\eta \sum_{t=0}^{T-1} \rho^{T-1-t}\mathbb{E}[\Gamma_t]. \tag{24}$$

The first two terms recover the standard uniform-weight SGD bound under the same assumptions, while the final term is a discounted nonnegative GAR improvement. The contraction factor itself is independent of component-gradient variation and τ.

D.5 Proof of Corollary 2

If f is μ-strongly convex, then $f(\boldsymbol{x}_t) - f(\boldsymbol{x}^\star) \ge (\mu/2)\|\boldsymbol{x}_t - \boldsymbol{x}^\star\|_{\boldsymbol{P}}^2$. Substituting into (23) gives

$$\mathbb{E}_t\|\boldsymbol{x}_{t+1} - \boldsymbol{x}^\star\|_{\boldsymbol{P}}^2 \le \rho\|\boldsymbol{x}_t - \boldsymbol{x}^\star\|_{\boldsymbol{P}}^2 + \frac{\eta^2\sigma^2}{b} - 2\eta\mathbb{E}_t\Gamma_t, \qquad \rho = 1 - \eta\mu(1-\eta L).$$

Taking total expectations and unrolling the recurrence proves (24).

D.6 Proof of Corollary 1

Proof. On (7), the two component gradients in whitened coordinates are $\boldsymbol{H}_0\boldsymbol{z} \pm \sigma\boldsymbol{e}_2$. They have identical first coordinates, so GAR gives $z_1^+ = (1 - \eta L/2)z_1$. For the second coordinate, Proposition 4 says that the softmax weights are evaluated at the first-order models at $\boldsymbol{z}^+$. The difference of those two affine models is

$$[d_+(\boldsymbol{z}) - d_-(\boldsymbol{z})] + 2\sigma(z_2^+ - z_2) = 2\sigma z_2^+.$$

Hence $w_+ - w_- = \tanh(\sigma z_2^+/\tau)$, and the update gives (18).

Let $H(y) = y + \eta\sigma\tanh(\sigma y/\tau)$. This map is odd and strictly increasing, with $H'(y) = 1 + \eta(\sigma^2/\tau)\operatorname{sech}^2(\sigma y/\tau) \ge 1$. Therefore H^{-1} is 1-Lipschitz and $H(0) = 0$. Since $H(z_2^+) = (1 - \eta L)z_2$,

$$|z_2^+| \le |1 - \eta L|\,|z_2|.$$

Combining with the first coordinate proves the contraction bound. At $\eta = 4/(3L)$ both factors equal $1/3$ in absolute value. □

E Derivations for Momentum and AdamW Reweighting

E.1 Polyak momentum

Write $\Delta_t := \boldsymbol{x}_t - \boldsymbol{x}_{t-1}$. For a candidate weight vector $\boldsymbol{w}$, the heavy-ball update is

$$\boldsymbol{x}_{t+1}(\boldsymbol{w}) = \boldsymbol{x}_t + \beta\Delta_t - \eta\boldsymbol{P}^{-1}\boldsymbol{G}_t\boldsymbol{w}.$$

Denote the momentum-only point by $\boldsymbol{y}_t := \boldsymbol{x}_t + \beta\Delta_t$, so that $\boldsymbol{x}_{t+1}(\boldsymbol{w}) = \boldsymbol{y}_t - \eta\boldsymbol{P}^{-1}\boldsymbol{G}_t\boldsymbol{w}$. For any $\boldsymbol{w}$,

$$\|\boldsymbol{y}_t - \eta\boldsymbol{P}^{-1}\boldsymbol{G}_t\boldsymbol{w} - \boldsymbol{x}^\star\|_{\boldsymbol{P}}^2 = \|\boldsymbol{y}_t - \boldsymbol{x}^\star\|_{\boldsymbol{P}}^2 - 2\eta\langle\boldsymbol{G}_t\boldsymbol{w}, \boldsymbol{x}_t - \boldsymbol{x}^\star + \beta\Delta_t\rangle + \eta^2\|\boldsymbol{G}_t\boldsymbol{w}\|_{\boldsymbol{P}^{-1}}^2.$$

Convexity gives

$$\langle\nabla f_{i_{t,j}}(\boldsymbol{x}_t), \boldsymbol{x}_t - \boldsymbol{x}^\star\rangle \ge f_{i_{t,j}}(\boldsymbol{x}_t) - f_{i_{t,j}}(\boldsymbol{x}^\star).$$

Thus the weight-dependent upper bound uses the scores

$$f_{i_{t,j}}(\boldsymbol{x}_t) - f_{i_{t,j}}(\boldsymbol{x}^\star) + \beta\langle\nabla f_{i_{t,j}}(\boldsymbol{x}_t), \Delta_t\rangle$$

and retains the same quadratic term $\eta\|\boldsymbol{G}_t\boldsymbol{w}\|_{\boldsymbol{P}^{-1}}^2/2$. A first-order Taylor expansion shows that these scores equal

$$f_{i_{t,j}}(\boldsymbol{x}_t + \beta\Delta_t) - f_{i_{t,j}}(\boldsymbol{x}^\star) + O(\beta^2\|\Delta_t\|_{\boldsymbol{P}}^2),$$

under metric smoothness. This explains the look-ahead implementation in Section 6.

E.2 AdamW weight selection with the previous-step preconditioner

To derive the weight-selection surrogate, suppress bias correction and decoupled weight decay and consider the Adam-style moment/preconditioner proposal for a candidate weight vector $\boldsymbol{w}$:

$$\boldsymbol{g}_t(\boldsymbol{w}) = \boldsymbol{G}_t\boldsymbol{w}, \qquad \boldsymbol{m}_t(\boldsymbol{w}) = \beta_1\boldsymbol{m}_{t-1} + (1-\beta_1)\boldsymbol{g}_t(\boldsymbol{w}),$$
$$\boldsymbol{v}_t(\boldsymbol{w}) = \beta_2\boldsymbol{v}_{t-1} + (1-\beta_2)\boldsymbol{g}_t(\boldsymbol{w})^{\odot 2}, \qquad \boldsymbol{P}_t(\boldsymbol{w}) = \mathrm{Diag}(\sqrt{\boldsymbol{v}_t(\boldsymbol{w})} + \epsilon_{\mathrm{Adam}}),$$
$$\boldsymbol{x}_{t+1}(\boldsymbol{w}) = \boldsymbol{x}_t - \eta\boldsymbol{P}_t(\boldsymbol{w})^{-1}\boldsymbol{m}_t(\boldsymbol{w}).$$

Thus the exact current Adam preconditioner, and hence the optimization-variable step itself, depends on the candidate weights. For the reweighting problem we use the previous-step value

$$\boldsymbol{P}_{t-1} = \mathrm{Diag}(\sqrt{\boldsymbol{v}_{t-1}} + \epsilon_{\mathrm{Adam}})$$

as a proxy for the fixed preconditioner in the theory. Define

$$\boldsymbol{y}_t = \boldsymbol{x}_t - \eta\beta_1\boldsymbol{P}_{t-1}^{-1}\boldsymbol{m}_{t-1}, \qquad \alpha_t = \eta(1-\beta_1).$$

For the frozen-preconditioner candidate step

$$\widetilde{\boldsymbol{x}}_{t+1}(\boldsymbol{w}) = \boldsymbol{y}_t - \alpha_t\boldsymbol{P}_{t-1}^{-1}\boldsymbol{G}_t\boldsymbol{w},$$

we have

$$\|\widetilde{\boldsymbol{x}}_{t+1}(\boldsymbol{w}) - \boldsymbol{x}^\star\|_{\boldsymbol{P}_{t-1}}^2 = \|\boldsymbol{y}_t - \boldsymbol{x}^\star\|_{\boldsymbol{P}_{t-1}}^2 - 2\alpha_t\langle\boldsymbol{G}_t\boldsymbol{w}, \boldsymbol{y}_t - \boldsymbol{x}^\star\rangle + \alpha_t^2\|\boldsymbol{G}_t\boldsymbol{w}\|_{\boldsymbol{P}_{t-1}^{-1}}^2.$$

For $\boldsymbol{g}_{t,j} = \nabla f_{i_{t,j}}(\boldsymbol{x}_t)$, convexity gives

$$\langle\boldsymbol{g}_{t,j}, \boldsymbol{y}_t - \boldsymbol{x}^\star\rangle = \langle\boldsymbol{g}_{t,j}, \boldsymbol{x}_t - \boldsymbol{x}^\star\rangle + \langle\boldsymbol{g}_{t,j}, \boldsymbol{y}_t - \boldsymbol{x}_t\rangle \geq d_{t,j} + \langle\boldsymbol{g}_{t,j}, \boldsymbol{y}_t - \boldsymbol{x}_t\rangle.$$

This motivates the score

$$\widetilde{d}_{t,j} := d_{t,j} + \langle\boldsymbol{g}_{t,j}, \boldsymbol{y}_t - \boldsymbol{x}_t\rangle = d_{t,j} - \eta\beta_1\langle\boldsymbol{g}_{t,j}, \boldsymbol{P}_{t-1}^{-1}\boldsymbol{m}_{t-1}\rangle.$$

Dividing the weight-dependent terms in the distance expansion by $2\alpha_t$ gives Algorithm 1 with $\boldsymbol{s} = \widetilde{\boldsymbol{d}}_t$, $\boldsymbol{R} = \boldsymbol{P}_{t-1}$, and $\alpha = \alpha_t$. A first-order Taylor expansion gives

$$\widetilde{d}_{t,j} = f_{i_{t,j}}(\boldsymbol{y}_t) - f_{i_{t,j}}(\boldsymbol{x}^\star) + O(\|\boldsymbol{y}_t - \boldsymbol{x}_t\|_{\boldsymbol{P}_{t-1}}^2),$$

which motivates the look-ahead-loss implementation.

Once $\boldsymbol{w}_t$ is selected, the implementation forms $\sum_j w_{t,j} f_{i_{t,j}}(\boldsymbol{x}_t)$ and calls the standard AdamW optimizer. The selected weighted gradient updates $\boldsymbol{m}_t$ and $\boldsymbol{v}_t$, and the resulting $\boldsymbol{P}_t$ is used for the actual optimization-variable update. Hence the approximation $\boldsymbol{P}_t \approx \boldsymbol{P}_{t-1}$ is confined to the weight-selection calculation.

F Experimental Details

Unless otherwise noted, we follow the model configurations, training hyperparameters, and evaluation pipeline of the released implementation of Sow et al. (2025). GPT-style models with 120M, 210M, and 300M parameters are trained from scratch for 20,000 steps on the SlimPajama 6B multi-domain subset. The corpus contains ArXiv, Books, C4, CommonCrawl, GitHub, StackExchange, and Wikipedia. We do not apply an additional static domain-weighting method: uniform AdamW samples from the original mixture, and LinUpper and GAR alter weights only within each current minibatch. For the first 1,000 training steps, all runs use uniform weights. Thereafter, LinUpper and GAR switch to their respective dynamic weights, while Uniform AdamW remains uniform throughout. GAR uses a constant temperature after this burn-in. All reported values are evaluation log-perplexities at the final step.

The oracle gaps in the convex analysis are replaced by per-example language-model losses at the Adam look-ahead point $\boldsymbol{y}_t = \boldsymbol{x}_t - \eta\beta_1\boldsymbol{P}_{t-1}^{-1}\boldsymbol{m}_{t-1}$. The previous-step preconditioner $\boldsymbol{P}_{t-1}$ is available before processing the current minibatch and serves as the practical proxy for the fixed $\boldsymbol{P}$ in the reweighting problem. The optimization variable is temporarily moved to $\boldsymbol{y}_t$ for this forward pass

and then restored to $\boldsymbol{x}_t$ before the gradient computation. After the weights are selected, the weighted minibatch objective is passed to standard AdamW, which updates the current moments and applies decoupled weight decay in the usual way.

Let $\boldsymbol{\ell}_t^{\mathrm{LA}}$ collect the per-example losses at $\boldsymbol{y}_t$ and recall $\alpha_t = \eta(1-\beta_1)$. The practical fixed-point equation is

$$\boldsymbol{w} = \mathrm{softmax}\left(\frac{\boldsymbol{\ell}_t^{\mathrm{LA}} - \alpha_t \boldsymbol{G}_t^\top \boldsymbol{P}_{t-1}^{-1} \boldsymbol{G}_t \boldsymbol{w}}{\tau}\right). \tag{25}$$

To avoid storing full per-example gradients, we approximate $\boldsymbol{G}_t^\top \boldsymbol{P}_{t-1}^{-1} \boldsymbol{G}_t \boldsymbol{w}$ using only gradients from the last linear readout layer. Given $\boldsymbol{w}^{(k)}$, automatic differentiation computes $\boldsymbol{g}^{(k)} = \boldsymbol{G}_t \boldsymbol{w}^{(k)}$ with a VJP, applies $\boldsymbol{P}_{t-1}^{-1}$, and returns $\boldsymbol{G}_t^\top \boldsymbol{P}_{t-1}^{-1} \boldsymbol{g}^{(k)}$ with a JVP. Starting from $\boldsymbol{w}^{(0)} = \boldsymbol{u}$, we perform exactly three iterations of (25); each iteration evaluates one VJP–preconditioner–JVP product. We then detach the resulting weights (PyTorch `detach`), so the training backward pass does not differentiate through the fixed-point iterations. In distributed training, the weighted gradient and softmax maximum and normalizer are reduced across devices, so the weights are normalized over the global effective minibatch.