THE CURVATURE FLOOR: A GEOMETRIC CRITERION FOR EARLY STOPPING

Anonymous authors
Paper under double-blind review

ABSTRACT

Early stopping needs a clean validation set, which self-supervised pretraining and training with noisy labels often lack. Rules without one watch the loss, the gradient or the prediction changes, which keep changing after the model stops improving. We instead monitor the effective learning rate in units of the stability threshold of gradient descent, $s_t = \eta_t^{\text{eff}} \lambda_{\max}/2$, where $\lambda_{\max}$ is the top eigenvalue of the Fisher information, preconditioned for adaptive methods and estimated with labels sampled from the model. In seven training regimes, from ResNet-50 on ImageNet to MAE pretraining, BERT fine-tuning and 40% label noise, s_t falls by at least a factor of five and settles at a floor. It crosses 0.05 at 5% to 15% of the budget before the best checkpoint, a lead that varies almost 700-fold in updates, so a rule should wait for a time that scales with the budget. We prove that when the statistic depends only on the budget fraction, a threshold rule on a linearly smoothed statistic stops at the same fraction of every budget, for every profile of the statistic, if and only if its kernel is rescaled with the budget at every lag within the run. We also bound the delay of the exponential moving average in closed form. The resulting rule, Curvature Floor Stopping (CFS), uses no labels while it runs and adds about 5% memory on models up to 340M parameters. With four constants set per regime on development labels of the same runs, it matches or exceeds validation-based early stopping in five of seven regimes, stays within 0.02 points in the other two and cuts the total compute by 20% to 59%, monitoring included. With one budget-scaled time constant on held-out runs, it is on par with a stronger validation rule that returns its best checkpoint, while costing less than it in six of the seven regimes.

1 INTRODUCTION

Networks are trained for a fixed budget of updates, and the last part of the budget can add little or even hurt accuracy, for example under label noise (Arpit et al., 2017). Early stopping (Prechelt, 1998) removes this part by watching a validation set and stopping once the validation metric stops improving. A clean validation set is not always at hand. Self-supervised pretraining has no labels for the downstream task (He et al., 2022; Chen et al., 2020), a validation set drawn from noisy labels is itself noisy (Yuan et al., 2024), and federated training cannot pool one (McMahan et al., 2017). Rules without a validation set watch the run itself, through the loss, the gradient norm (Mahsereci et al., 2017), the gradient disparity between mini-batches (Forouzesh & Thiran, 2021) or the prediction changes (Yuan et al., 2024). These quantities follow the fit to the training data. They keep changing while the network fits hard or mislabeled examples (Arpit et al., 2017), so rules built on them tend to stop late. We look for a quantity of the run that warns when further updates will soon stop helping.

An update can help only if it changes the predictive distribution of the model. How much it does depends on the step size and on how sharply the predictions respond to the parameters, a curvature that for the predictive distribution is the Fisher information, the local metric of the KL divergence (Amari, 1998; Martens, 2020). The Fisher needs no true labels. We sample the labels from the model, and in expectation the result equals the Gauss-Newton matrix (Martens, 2020). Unlike the Hessian, it is positive semidefinite and its products with a vector need no second derivatives. On a quadratic, gradient descent with step η contracts the error along a direction of curvature λ only if $\eta\lambda < 2$ (Wu et al., 2018; Cohen et al., 2021), so we measure the effective learning rate η_t^{eff} in units of this threshold, $s_t = \eta_t^{\text{eff}} \lambda_{\max}/2$, where $\lambda_{\max}$ is the top eigenvalue of the Fisher (Section 3.1). A value of

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.

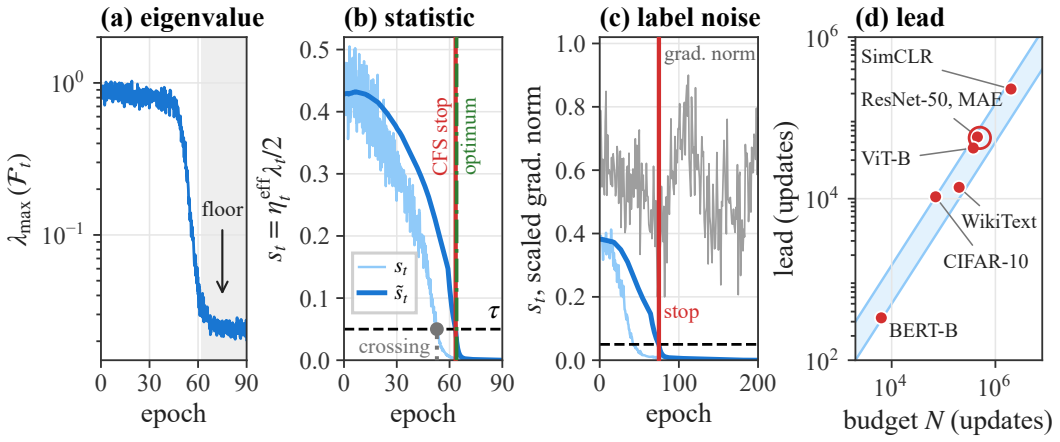

Figure 1: **The curvature floor.** (a) Top eigenvalue of the sampled-label Fisher for ResNet-50 on ImageNet (SGD with a cosine schedule, seed 42), which falls from about 0.7 at epoch 45 to 0.03 at epoch 62 and then stays flat (gray band). (b) The statistic s_t of (2) and its EMA $\tilde{s}_t$, same seed. The raw value crosses $\tau = 0.05$ at epoch 52.8 and CFS stops at epoch 63.5, just before the optimum, the minimum of the development loss at epoch 64. (c) ResNet-18 on CIFAR-10 with 40% symmetric label noise (seed 42). The gradient norm, rescaled to the axis, stays high while s_t reaches its floor, and CFS stops at epoch 74.8 of 200. (d) Lead of the crossing over the best checkpoint against the budget N for all seven regimes (ring: MAE). The lead grows from 335 to 228,000 updates but stays between 5% and 15% of the budget (band), so the wait after the crossing should scale with the budget.

one marks the stability limit of plain gradient descent, and $s_t \ll 1$ means the step is far below this limit along the sharpest direction of the predictions. We can compute s_t on any run and follow it.

We measured s_t in seven training regimes, from ResNet-50 on ImageNet to MAE pretraining, BERT fine-tuning and 40% label noise. In every regime s_t falls steeply, by at least a factor of five, and stays flat at a low level, which we call the curvature floor (Figure 1). The fall is more than the decay of the learning rate, since on ResNet-50 the top eigenvalue itself drops more than 20-fold, and under label noise s_t reaches 0.05 while the learning rate is still at 88% of its peak (Figure 1c). The statistic first falls below 0.05, an event we call the crossing, at 5% to 15% of the budget before the best checkpoint. A rule that stops at the crossing stops too soon, and the measured lead shows how long it should wait. In updates it ranges from 335 on BERT-B to 228,000 on SimCLR, a factor of almost 700, while as a share of the budget it stays between 5% and 15% (Figure 1d). No single wait fixed in updates fits all seven regimes, so the wait after the crossing should instead scale with the budget of the whole run.

Designing the rule then comes down to setting this wait. A practical rule smooths s_t and stops when the smoothed value falls below a threshold. The smoothing averages out the noise of single estimates, and its lag delays the stop past the crossing, so the lag can serve as the wait. We prove that when the statistic depends only on the budget fraction, such a rule stops at the same fraction of every budget, for every profile of the statistic, if and only if its smoothing kernel is rescaled with the budget at every lag within the run (Theorem 1). For the exponential moving average (EMA) this means a time constant proportional to the budget, and we bound its delay after the crossing in closed form (Proposition 1). A rank argument shows that a small Fisher batch biases the statistic upward and tends to delay the stop (Proposition 2). These results lead to Curvature Floor Stopping (CFS), which estimates $\lambda_{\max}$ with Fisher-vector products, smooths s_t with a time constant proportional to the budget, and stops once the smoothed value stays below $\tau = 0.05$ for a few consecutive checks.

CFS uses no labels while it runs. We set four of its constants per regime with development labels of the same runs, and held-out runs share one such time constant. CFS matches or exceeds validation-based early stopping in five of seven regimes, stays within 0.02 points in the other two, and saves 20% to 59% of the compute. It carries over to nine ImageNet architectures, seven GLUE tasks and language models up to 141B parameters, and on 3000-update runs, where a time constant fixed in updates often fails to stop, it stops in every run. We make three contributions. (i) We identify a label-free, step-normalized curvature statistic whose crossing of a fixed threshold precedes the best checkpoint

by 5% to 15% of the budget. (ii) We show that budget invariance requires a rescaled kernel, bound the delay of the EMA in closed form, show that its stop is stable under small deviations and under estimation noise, and explain why a small Fisher batch tends to delay the stop. (iii) We build a practical rule on these results and test each part against its removal or its most direct replacement.

2 Related Work

Early stopping without validation data. Early stopping with a validation set is a standard regularizer (Prechelt, 1998), well understood for linear and kernel models (Yao et al., 2007; Raskutti et al., 2014). Rules without validation data replace the held-out metric by a statistic of the run. Mahsereci et al. (2017) test whether the gradient is statistically distinguishable from zero, Forouzesh & Thiran (2021) use the gradient disparity between mini-batches, Bonet et al. (2021) use channel-wise NNK polytope interpolation and Jamshidi et al. (2025) sample an approximate posterior. For noisy labels, Label Wave (Yuan et al., 2024) follows the prediction changes on the training set, and Instance-dependent Early Stopping (Yuan et al., 2025) removes examples whose training loss has converged. These are our closest baselines, and we compare with both. For self-supervised pretraining, the effective rank or the spectral decay of the embeddings can select models without labels (Garrido et al., 2023; Agrawal et al., 2022). Liao et al. (2020) track the spectrum of an approximate Fisher to describe training, while we normalize its top eigenvalue by the step size and derive from it when to stop training.

Curvature along the training trajectory. Under full-batch gradient descent the top Hessian eigenvalue rises to the stability threshold $2/\eta$ and hovers there, a regime known as the edge of stability (Cohen et al., 2021; Damian et al., 2023). The preconditioned top eigenvalue of adaptive methods such as Adam behaves the same way (Cohen et al., 2022). Stochastic training selects minima whose sharpness fits the step size and the batch size (Wu et al., 2018; Jastrzębski et al., 2020), warmup moves the early trajectory to flatter regions (Gilmer et al., 2022; Kalra & Barkeshli, 2024), and flatter minima often generalize better (Hochreiter & Schmidhuber, 1997; Keskar et al., 2017; Foret et al., 2021). We use the end of this evolution, where the normalized curvature settles at a floor, as the signal to stop, and we analyze when it happens in the setting of budgeted training (Li et al., 2020b).

Estimating curvature at scale. Hessian-vector products differentiate through the gradient computation (Pearlmutter, 1994), which Gauss-Newton and Fisher-vector products avoid (Schraudolph, 2002; Martens, 2010). With labels sampled from the model the Fisher equals the Gauss-Newton matrix in expectation, unlike the empirical Fisher (Martens, 2020; Kunstner et al., 2019), and Lanczos or power iteration gives its top eigenvalues for large networks (Ghorbani et al., 2019; Papyan, 2019; Yao et al., 2020). We compute the product with forward-mode automatic differentiation (Baydin et al., 2018) and bound the bias of a small Fisher batch with matrix concentration bounds (Tropp, 2015).

3 The Curvature Floor

3.1 A Step-Normalized Curvature Statistic

Let $\theta_t \in \mathbb{R}^d$ be the parameters after t updates, N the budget in updates and $p_\theta(y \mid x)$ the predictive distribution of the model. At a check, an update where we measure the statistic, we draw a Fisher batch of B_p training inputs $x_1, \dots, x_{B_p}$, sample one label per input from the model, $\hat{y}_i \sim p_{\theta_t}(\cdot \mid x_i)$, and form the scores $g_i = \nabla_\theta \log p_{\theta_t}(\hat{y}_i \mid x_i)$. The sampled-label Fisher is

$$\mathcal{F}_t = \frac{1}{B_p} \sum_{i=1}^{B_p} g_i \, g_i^\top. \tag{1}$$

Unlike the empirical Fisher (Kunstner et al., 2019), it uses no true labels. For SGD with momentum μ the effective learning rate is $\eta_t^{\text{eff}} = \eta_t/(1-\mu)$ (Sutskever et al., 2013), and we monitor $\mathcal{C}_t = \mathcal{F}_t$. AdamW scales the step by $P_t = \operatorname{diag}\left((\sqrt{\hat{v}_t} + \epsilon)^{-1}\right)$, where $\hat{v}_t$ is its second-moment estimate, so we monitor $\mathcal{C}_t = P_t^{1/2} \mathcal{F}_t P_t^{1/2}$ with $\eta_t^{\text{eff}} = \eta_t$ (Cohen et al., 2022). Let $\lambda_t = \lambda_{\max}(\mathcal{C}_t)$. Gradient descent on a quadratic with this curvature is stable only if $\eta_t^{\text{eff}} \lambda_t < 2$, so we define

$$\boxed{s_t = \tfrac{1}{2} \, \eta_t^{\text{eff}} \, \lambda_t} \tag{2}$$

with $s_t = 1$ at the stability threshold of plain gradient descent. Momentum moves this threshold to $s_t = (1+\mu)/(1-\mu)$ for SGD, and to $(1+\beta_1)/(1-\beta_1)$ for Adam with first-moment decay β_1 (Cohen et al., 2022). This is 19 at $\mu = \beta_1 = 0.9$ (Section B.1), so s_t means the same for both optimizers. The curvature term of a step is also at most s_t times its linear term (Section B.1), so a small value of s_t means that curvature no longer limits the size of the update along any direction.

3.2 The Floor and Its Timing

Figure 1a shows λ_t for a ResNet-50 run. It stays between about 0.6 and 1.0 for 45 epochs, falls to 0.03 by epoch 62 and then stays flat. Since the learning rate decays at the same time, s_t falls even faster (Figure 1b). It crosses $\tau = 0.05$ at epoch 52.4 ± 1.9 over five seeds. We call the best checkpoint of a run its optimum. For ResNet-50 it is the minimum of the development loss, at epoch 64 in every seed, and elsewhere the checkpoint with the best held-out metric (Table 11). On ResNet-50 the crossing thus precedes the optimum by 12.9% of the budget. The same fall appears in every regime, from above 0.15 to at most 0.028 at the stop (Table 11), and we plot the floor for five of them (Figures 1 to 3). For ViT-B, BERT-B, MAE, SimCLR, WikiText-103 and CIFAR-10 with 40% label noise, the crossing leads the optimum by 11.3%, 5.3%, 11.3%, 11.4%, 6.9% and 15.0% of the budget. The crossing is an early warning, with a lead of 5.3% to 15.0% of the budget in all seven regimes.

3.3 Interpreting the Statistic

The timing could depend on the curvature matrix. On ResNet-18 with CIFAR-10, the top eigenvalues of the Hessian, the Gauss-Newton matrix and the Fisher fall together from about 2.5 to 0.1, with the Fisher within 1.2% of the Gauss-Newton matrix and 2% to 20% above the Hessian, and with AdamW the preconditioned statistic falls in the same way (Figure 3). The lemma below, whose part (i) is known (Martens, 2020), explains this agreement and what drives the fall (proof in Section B.2).

Lemma 1 (Sampled-label Fisher). *Let the model output the natural parameters $f_\theta(x) \in \mathbb{R}^m$ of an exponential family with sufficient statistic $T(y)$, let the labels $\hat{y}_i$ be drawn independently, and let $J_i = \partial f_\theta(x_i)/\partial\theta$ and $\Lambda_i = \mathrm{Cov}_{y\sim p_\theta(\cdot|x_i)}[T(y)]$. Then (i) $\mathbb{E}[\mathcal{F}_t \mid \theta_t, x_{1:B_p}] = G_t := \frac{1}{B_p}\sum_i J_i^\top \Lambda_i J_i$, the Gauss-Newton matrix of the Fisher batch, and (ii) for softmax classification with predicted probabilities p_i,*

$$\lambda_{\max}(G_t) \;\le\; \mathbb{E}\big[\lambda_{\max}(\mathcal{F}_t) \,\big|\, \theta_t, x_{1:B_p}\big] \;\le\; \frac{1}{B_p}\sum_{i=1}^{B_p}\big(1 - \|p_i\|_2^2\big)\|J_i\|_2^2. \tag{3}$$

Part (i) is why the statistic needs no labels. Part (ii) bounds the expected curvature by the Gini impurity $1 - \|p_i\|_2^2$ of the model's own predictions. Once the network puts most of its probability on one class for each training input, λ_t is small in expectation unless the Jacobians grow, whether or not the labels are correct. Under 40% label noise (Figure 1c) the gradient norm stays high while the network fits the wrong labels, but s_t crosses τ near epoch 45, and CFS stops at epoch 75.1 on average, close to the mean optimum at epoch 74.9. The lemma also shows a limit. In classification the statistic measures confidence on training inputs, so later gains in accuracy go unseen (Section J).

4 Curvature Floor Stopping

4.1 Estimating the Top Eigenvalue

The rule needs λ_t at hundreds of checks per run, so the estimate must be cheap in memory and in time. A Hessian-vector product differentiates through the backward pass and raises the peak memory of ResNet-50 by 232%, while a Fisher-vector product does not. For a direction v, $\mathcal{F}_t v = \frac{1}{B_p}\sum_i a_i g_i$ with $a_i = g_i^\top v$, the directional derivative of $\log p_\theta(\hat{y}_i \mid x_i)$. One forward-mode pass returns all a_i, and one backward pass of $\frac{1}{B_p}\sum_i \mathrm{sg}(a_i)\log p_\theta(\hat{y}_i \mid x_i)$, where sg is the stop-gradient operator, returns $\mathcal{F}_t v$ (Section B.3). No per-example gradient is stored, and the peak memory rises by 4.7% to 5.0% on seven models from 12M to 340M parameters (Section I). Power iteration starts from the eigenvector of the previous check, which cuts the products per check from 12.8 to 2.5 on average (Section B.11). Each estimate comes from one Fisher batch, so we also measure its estimation noise.

The two halves of the batch give Rayleigh quotients $\hat{\lambda}^{(1)}$ and $\hat{\lambda}^{(2)}$ of the final direction, and

$$\hat{\lambda} = \tfrac{1}{2}\big(\hat{\lambda}^{(1)} + \hat{\lambda}^{(2)}\big), \qquad \mathrm{SNR} = \frac{\hat{\lambda}^2}{\tfrac{1}{2}\big(\hat{\lambda}^{(1)} - \hat{\lambda}^{(2)}\big)^2 + \epsilon_0} \tag{4}$$

is a split-half signal-to-noise ratio with offset $\epsilon_0 = 10^{-8}$. Any check with $\mathrm{SNR} < \kappa = 10$ or $\hat{\lambda} \le 0$ is rejected, and in the seven main regimes this test keeps 94.4% to 99.8% of all checks (Section B.4).

4.2 The Stopping Rule

Let $t_1 < t_2 < \cdots$ be the accepted checks and s_j the statistic (2) at check j. The rule keeps an EMA with decay factor $\beta \in [0, 1)$, started at the first accepted value,

$$\tilde{s}_1 = s_1, \qquad \tilde{s}_j = \beta\, \tilde{s}_{j-1} + (1 - \beta)\, s_j, \tag{5}$$

and stops at the first check j with

$$\boxed{\tilde{s}_{j-k} < \tau \ \text{ for } k = 0, \ldots, K-1 \quad \text{and} \quad \max_{i \le j} \tilde{s}_i \ge 3\tau} \tag{6}$$

The first condition asks for K consecutive values below τ, and we call K the patience. The second, a start condition, prevents premature stops when the first estimates are low. Checks are Δ_1 updates apart while $\tilde{s} \ge 3\tau$, Δ_2 while $\tau \le \tilde{s} < 3\tau$ and Δ_3 once $\tilde{s} < \tau$, or 600, 150 and 44 on ResNet-50. With a check interval Δ, β acts as a time constant $\ell = \Delta/(-\ln \beta)$ updates. We set $\beta = \exp(-\Delta_3/(cN))$, so the time constant of the shortest interval is a fraction c of the budget and interval Δ_z has the time constant $\ell_z = (\Delta_z/\Delta_3)\, cN$. Close to the threshold the EMA runs at the middle interval, with $\ell_2 \approx 0.04N$ on ResNet-50. We use $\tau = 0.05$ and $\kappa = 10$ for every task and model (Algorithm 1).

4.3 Theoretical Analysis

Nothing in (6) refers to the optimum. The stop lands near it because the crossing leads the optimum and the smoothing adds a delay whose length c sets. We derive how the delay must depend on the budget (proofs in Section B). Following budgeted training (Li et al., 2020b), we define the schedule on the budget fraction, and we assume that the noise-free statistic depends on this fraction alone.

> **Assumption 1** (Budgeted training)**.** Runs that share a recipe, including the Fisher batch size, and a schedule $\eta_t = \eta_0 h(t/N)$ with a fixed shape h have a noise-free statistic $\mathbb{E}[s_t \mid \theta_t, P_t] = S(t/N)$ for a bounded measurable function $S \ge 0$ on $[0, 1]$ that does not depend on the budget N.

Figure 1d is consistent with this assumption, although it compares different recipes, not one recipe at several budgets. So are the calibrated values of c, which stay between 0.012 and 0.017 while budgets range from 6K to 2M updates. Consider any rule that smooths the statistic with a causal kernel $k_N \ge 0$ of unit mass, $\tilde{s}(t) = \int_0^\infty k_N(r)\, s(\max\{t - r, t_0\})\, \mathrm{d}r$ with $s(t) = S(t/N)$, and stops at $t_{\mathrm{stop}} = \inf\{t \in (t_0, N] : \tilde{s}(t) < \tau\}$, or $t_{\mathrm{stop}} = \infty$ if this set is empty. The first check is at $t_0 = u_0 N$ with the same u_0 for every budget. The EMA is the kernel $k_N(r) = e^{-r/\ell}/\ell$ in continuous time. Let $\bar{k}_N(x) = \int_x^\infty k_N(r)\, \mathrm{d}r$ be the tail mass of the kernel k_N and $u_{\mathrm{stop}} = t_{\mathrm{stop}}/N$ the stop fraction.

> **Theorem 1** (Budget invariance)**.** *Let Assumption 1 hold. (i) If the kernel is rescaled with the budget, $k_N(r) = k(r/N)/N$ for a fixed k, then*
>
> $$\tilde{s}(uN) = \int_0^\infty k(w)\, S\big(\max\{u - w, u_0\}\big)\, \mathrm{d}w, \tag{7}$$
>
> *so u_{stop} is the same for every budget N. (ii) Conversely, if u_{stop} is the same for every N whenever $S(u) = a$ for $u < u_1$ and $S(u) = 0$ otherwise, with $a > \tau$ and $u_0 < u_1 < 1$, then $\bar{k}_N(x) = \bar{k}_{N'}(xN'/N)$ for all budgets N, N' and all lags $x \le (1 - u_0)N$. In particular, an EMA stops at the same fraction for every N and every such S if and only if ℓ/N is constant.*

The same holds for the rule (6), with its three intervals, start condition and patience, when every interval is a fixed fraction of the budget (Corollary 1). CFS instead fixes the intervals and sets β from the budget, so its time constants are proportional to N and only the discretization of the checks departs from invariance (Section B.5). The theorem fixes how the kernel must scale. Where the stop falls within the budget depends on the delay after the crossing, which we bound next for the EMA.

Proposition 1 (Delay of the EMA). *Let the smoother be an EMA with time constant ℓ, write $\tilde{S}(u) = \tilde{s}(uN)$, and let $u_c = \inf\{u > u_0 : S(u) < \tau\} > u_0$, with $S \geq \tau$ on $[u_0, u_c)$ and $S < \tau$ on $(u_c, 1]$. If $\underline{S} \leq S$ on $(u_c, 1]$ and $S \leq \bar{S}$ on $(u_f, 1]$ for constants $0 \leq \underline{S} \leq \bar{S} < \tau$ and some $u_f \geq u_c$, then*

$$\frac{\ell}{N} \ln \frac{\tilde{S}(u_c) - \underline{S}}{\tau - \underline{S}} \;\leq\; u_{\text{stop}} - u_c \;\leq\; (u_f - u_c) + \frac{\ell}{N} \ln \frac{\tilde{S}(u_c) - \bar{S}}{\tau - \bar{S}}, \tag{8}$$

where the upper bound applies whenever it is below $1 - u_c$. Among right-continuous kernels, the exponential kernel is the only one for which, for every lag $x > 0$ and every bounded history, the smoothed value at t and s on $[t, t + x]$ determine the smoothed value at $t + x$ (Lemma 2).

The last statement is the memoryless property of the exponential distribution (Feller, 1971), and it is why we use the EMA: for other linear smoothers the delay depends in general on the whole history before the crossing, not only on $\tilde{S}(u_c)$. If S is non-increasing, u_{stop} does not decrease as ℓ/N grows, and if S is continuous, $u_{\text{stop}} \to u_c$ as $\ell/N \to 0$ (Corollary 2). A time constant ℓ fixed in updates thus fails in two ways. As N grows the stop moves to the crossing, early by the whole lead, and once $N(1 - u_c) < \ell \ln(\tilde{S}(u_c)/\tau)$ the noise-free rule never stops. With three check intervals the EMA decays with ℓ_1 while $\tilde{s} \geq 3\tau$ and with ℓ_2 below. If the statistic is zero after the crossing $t_c = u_c N$ and $\tilde{s}(t_c) \geq 3\tau$, the EMA in continuous time first falls below τ at $t_c + D$ with

$$D = cN \Big[\frac{\Delta_1}{\Delta_3} \ln \frac{\tilde{s}(t_c)}{3\tau} + \frac{\Delta_2}{\Delta_3} \ln 3 \Big], \tag{9}$$

and the discrete rule, accepting every check, stops at most $\Delta_1 + (K - 1)\Delta_3$ later. A positive statistic after the crossing only lengthens D (Corollary 3), linking c to the lead. The rule is an exponentially weighted moving average chart with a lower control limit (Roberts, 1959). Our analysis adds how its time constant must scale with the budget and how far a deviation from S can move the stop: given a margin, a deviation $\leq \varepsilon$ moves the stop fraction of an EMA with one time constant by at most $(\ell/N)\,\varepsilon/(\tau - \bar{S} - \varepsilon)$ (Proposition 3), 0.5% of the budget for $\ell = 0.04N$ and $\varepsilon = \bar{S} = 0.005$.

Proposition 2 (Finite-sample bias). *Condition on the parameters θ_t and P_t ($P_t = I$ for SGD), let $\bar{\mathcal{C}}_t = \mathbb{E}\,\mathcal{C}_t$, and let $\rho_t = \operatorname{tr} \bar{\mathcal{C}}_t / \lambda_{\max}(\bar{\mathcal{C}}_t)$ be its effective rank. (i) Each Fisher batch gives $\lambda_{\max}(\mathcal{C}_t) \geq \operatorname{tr}(\mathcal{C}_t)/B_p$, and $\mathbb{E}\,\lambda_{\max}(\mathcal{C}_t) \geq \lambda_{\max}(\bar{\mathcal{C}}_t) \max\{1, \rho_t/B_p\}$. (ii) If the pairs $(x_i, \hat{y}_i)$ are exchangeable, for instance i.i.d., $\mathbb{E}\,\lambda_{\max}(\mathcal{C}_t)$ does not grow with the Fisher batch size B_p.*

Multiplying (i) by $\eta_t^{\text{eff}}/2$ gives $\mathbb{E}\,s_t \geq (\rho_t/B_p)\, s_t^\star$ with $s_t^\star = \eta_t^{\text{eff}} \lambda_{\max}(\bar{\mathcal{C}}_t)/2$, a lower bound that no smoothing removes, so the noise-free statistic can fall below τ only if $B_p > \rho_t s_t^\star/\tau$. By (ii) a smaller batch does not lower the noise-free statistic, so in continuous time its EMA stays at least as high, and the stop does not come earlier if both EMAs meet the start condition before falling below τ. For bounded independent scores a Bernstein bound shows that the bias vanishes as B_p grows (Section B.9), and Proposition 4 in Section B.10 shows that the EMA keeps a share of the variance of the estimation noise that falls to $(1 - \beta)/(1 + \beta)$, which is 0.4% at $\beta = 0.9921$ on ResNet-50 after about 640 checks. Corollary 4 turns this into a bound on how far the noise can move the stop.

5 Experiments

5.1 Experimental Setup

Datasets and Models. We use seven training regimes with five seeds each: ResNet-50 (He et al., 2016) and ViT-B/16 (Dosovitskiy et al., 2021) on ImageNet-1K (Deng et al., 2009) with SGD and AdamW (Loshchilov & Hutter, 2019); MAE (He et al., 2022) and SimCLR (Chen et al., 2020) pretraining on ImageNet, evaluated by linear probing of the frozen encoder; BERT-B (Devlin et al., 2019) fine-tuned on SST-2 (Wang et al., 2019); ResNet-18 on CIFAR-10 (Krizhevsky, 2009) with 40% symmetric label noise; and a Transformer language model on WikiText-103 (Merity et al., 2017). Budgets, optimizers and schedules follow standard recipes (Section C). ImageNet selection uses a 10K development split of the validation set, and the main results are reported on the other 40K images. SST-2 has no public test labels, so its development set serves both for selection and for reporting.

Table 1: **Main results** (mean and standard deviation, five seeds): top-1 accuracy for ResNet-50 and ViT-B, linear-probe accuracy for MAE and SimCLR, accuracy for BERT-B and ResNet-18, and perplexity for WikiText-103 (lower is better). Small gray: stop in epochs (WikiText-103: thousands of updates). Labels: the rule reads held-out labels. Best rule in bold, second underlined. §No labels while running; constants set per regime. †Trains for the whole budget and saves compute by dropping examples (none on ViT-B, BERT-B). *Linear probe every 50 epochs, patience five. ‡Lower bound: mean over seeds of the better of epoch 64 and the CFS stop. Bottom: CFS with 40 new draws of the Fisher batches (metric, cost relative to the full run), the best common stop, the best epoch for all five runs chosen with labels at no more cost than CFS (metric, epoch), and the total-cost speedup.

Method	Labels	ResNet-50 ImageNet	ViT-B/16 ImageNet	MAE ImageNet	SimCLR ImageNet	BERT-B SST-2	ResNet-18 CIFAR-10, 40%	Transformer WikiText-103
Full run		76.84±0.19 90	81.81±0.19 300	68.04±0.21 1600	69.34±0.30 800	92.45±0.36 3	52.39±0.79 200	25.07±0.11 200
Loss plateau	✗	76.55±0.18 86±3	81.91±0.18 286±8	67.88±0.23 1520	69.41±0.27 760±18	92.88±0.24 2.2±0.1	63.20±1.10 164±12	24.92±0.10 188±6
Gradient norm	✗	76.20±0.16 82	81.86±0.18 284±10	68.02±0.20 1480±45	69.36±0.29 785±12	<u>92.90</u>±0.27 2.3±0.2	52.39±0.79 200	24.94±0.11 191±5
Label Wave	✗	75.80±0.27 68	80.85±0.31 220	n/a	n/a	92.75±0.30 2.0±0.1	81.56±0.49 78	n/a
IES†	✗	76.9±0.2 90	81.81±0.19 300	n/a	n/a	92.45±0.36 3	52.5±0.8 200	n/a
Validation ES	✓	<u>77.01</u>±0.10 65.0±0.0	<u>82.03</u>±0.07 245	<u>68.16</u>±0.18 1380±27*	**69.55**±0.20 675±18	92.66±0.28 2.4	**81.99**±0.45 82	<u>24.79</u>±0.06 172
CFS (ours)	✗§	**77.18**±0.11 63.2±1.2	**82.20**±0.11 209±4	**68.32**±0.19 1096±21	69.53±0.22 619±10	**93.00**±0.18 2.25±0.04	81.98±0.21 75.1±1.9	**24.69**±0.10 159±2
Oracle	✓	≥77.24‡	82.23±0.09 209±3	68.33±0.18 1101±30	69.54±0.22 621±16	93.00±0.18 2.27±0.04	82.31±0.30 74.9±2.2	24.68±0.10 158±2
CFS, 40 batch draws	✗	77.16 0.72	82.21 0.74	68.32 0.69	69.53 0.78	93.00 0.78	82.00 0.41	24.685 0.80
Best common stop	✓	77.14 64.1	82.11 207.8	68.32 1105	69.52 622	92.98 2.26	82.11 74.5	24.688 159.6
Speedup of CFS		1.39×	1.36×	1.45×	1.29×	1.28×	2.44×	1.24×

Baselines. We compare with the full run, loss plateau and gradient-norm rules, Label Wave (Yuan et al., 2024), Instance-dependent Early Stopping (IES, Yuan et al., 2025), validation-based early stopping (validation ES) and the oracle, which picks the best saved checkpoint of each run with held-out labels. Validation ES evaluates periodically, stops once the metric no longer improves and returns that model (Section C.2). Section G adds a stronger variant returning its best checkpoint.

Implementation Details. CFS does not change the run, so we log Fisher-batch estimates of s_t along each run and apply the rule and all its variants to these logs, each evaluated at its own stop. Where the rule also ran online, the stops agree within 0.2 epochs (Table 16). For each regime we choose c, the check intervals, K and the Fisher batch size once, on its five runs, by minimizing the total cost plus the gap to the development-set oracle (Section C.3). Table 1 thus reports CFS on the runs used for this choice, the only step that uses development labels. Table 9 lists the chosen constants. Held-out runs (Section 5.3) use $c = 0.012$ and none of their labels. All reported costs include monitoring.

5.2 Main Results

Table 1 compares all rules. In all seven regimes CFS, whose constants are set per regime, is the best rule that reads no labels while running. With the thresholds of Section C.2, the loss plateau and gradient-norm rules usually stop later than CFS or not at all, and under label noise they keep training while the network memorizes wrong labels (63.20 and 52.39 against 81.98). Label Wave, designed for label noise, does well there, but on ResNet-50 it stops at epoch 68, in the accuracy dip of Table 12, 1.38 points below CFS. Although CFS uses no labels while it runs, it matches or exceeds validation ES in five regimes and is within 0.02 points in the other two. Its gap to the oracle is at most 0.04 points on ViT-B, MAE, SimCLR and BERT-B, 0.01 in perplexity on WikiText-103, 0.33 points under label noise and at least 0.06 on ResNet-50, and it saves 20% to 59% of the compute. The stop is stable. With 40 new draws of the Fisher batches the metric moves by at most 0.02 points, and the best common stop, chosen with labels at no more cost than CFS, beats these draws only under label noise.

Self-supervised pretraining shows why a rule on the pretext loss stops late (Figure 2). For MAE the reconstruction loss keeps falling until the end, from 0.366 at the CFS stop to 0.352. At the stop the linear-probe accuracy is already 68.32, against 68.33 for the oracle and 68.04 after the full run, and

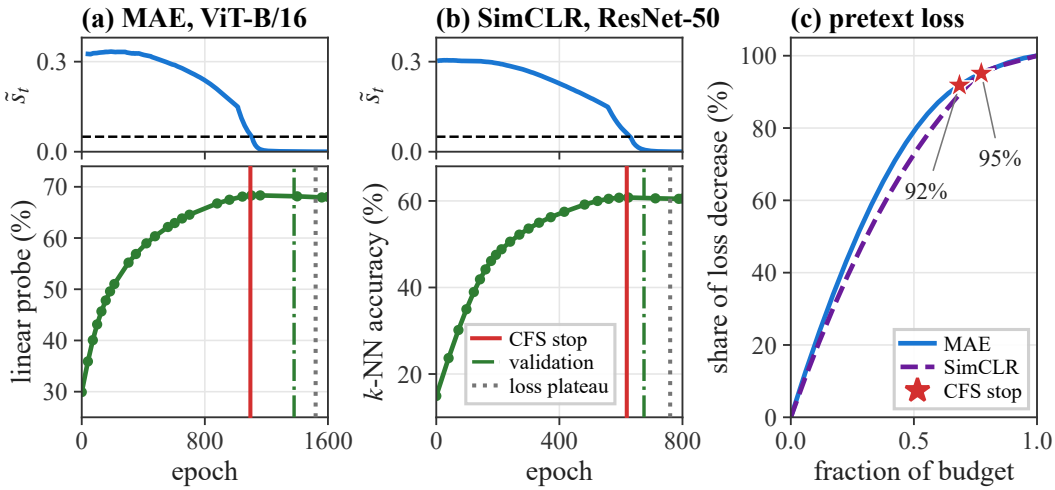

Figure 2: **Self-supervised pretraining.** (a) MAE with ViT-B/16. Top: the EMA $\tilde{s}_t$ of seed 42 and the threshold τ. Bottom: linear-probe accuracy and the stops of CFS, validation ES and the loss plateau, all averaged over five seeds. (b) The same for SimCLR with ResNet-50 and the k-NN accuracy. (c) Share of the total decrease of the pretext loss L_t after t updates (MSE for MAE, NT-Xent for SimCLR), $(L_0 - L_t)/(L_0 - L_N)$, against the budget fraction. At the CFS stop (red stars) 92% and 95% of the decrease has happened, while the downstream accuracy is within 0.01 points of its peak.

the fine-tuning accuracy matches that of the full run (83.59 against 83.58). A loss plateau rule waits until epoch 1520. SimCLR shows the same pattern for the contrastive loss and the k-NN accuracy. At the CFS stop 92% (MAE) and 95% (SimCLR) of the decrease of the pretext loss has happened, and the rest takes 32% and 23% of the budget and adds at most 0.01 points of downstream accuracy.

5.3 Generality

Architectures, tasks and language models. With the same threshold, CFS gives a speedup in updates of 1.15× to 1.61× on nine ImageNet architectures from 5.3M to 304M parameters, seven of them without calibration, and within a family the speedup grows with size. On seven GLUE tasks it averages 81.50, against 81.55 for validation ES, 81.93 for the oracle and 81.14 for the full run. In supervised fine-tuning of five language models from 7.6B to 141B parameters, it stops after 1.5 to 2.1 of three epochs, within -0.15 to $+0.18$ points of validation ES, with a measured speedup of 1.43× to 1.92×. Section F adds detection, fairness, label noise, transfer and plug-ins for noisy-label methods.

Short budgets. Proposition 1 predicts that a time constant set for long runs fails on a short budget. We test this on 3000-update runs of five open language models: Qwen2.5-7B and 14B (Yang et al., 2024), Mistral-7B (Jiang et al., 2023), Mixtral-8x7B (Jiang et al., 2024) and Mamba2-2.7B (Dao & Gu, 2024). Each model has twelve runs under each of three conditions, with no, moderate or strong decay of a knowledge score late in training (Section H). CFS is compared with a matched rule that checks at the same updates, every ten, and differs only in β. CFS uses $\beta = 0.76$, which gives $\ell = cN = 36$ updates, while the matched rule keeps $\beta = 0.99$, typical of long runs, which gives $\ell = 995$ updates, a third of the budget. The matched rule stops in 98 of 180 runs, and CFS in all 180 runs, on average at 58% to 68% of the budget, with scores near validation-based stopping (Figure 10).

Held-out runs. On twelve further runs per regime not used to set any constant, CFS with the shared value $c = 0.012$ is on par with a validation rule that returns its best checkpoint. Paired differences range from -0.01 to $+0.10$ points, and every 95% confidence interval contains zero and lies within 0.3 points of it. CFS costs less than this rule except on BERT-B, where it costs 9.4% more and the loss plateau and gradient-norm rules are slightly ahead at lower cost. It also beats the full run in all 84 paired comparisons and lies on the Pareto front of cost and performance in six regimes (Section G).

Table 2: **Ablations** on ResNet-50 with five seeds unless noted: each component of CFS against its removal or most direct replacement (B: the training batch size; stops in epochs). The interval row uses the log of seed 42, with ℓ fixed by adjusting β in the CFS column and β fixed in the changed one. The short-budget row gives the composite score (0.5 task $+ 0.3$ knowledge $+ 0.2$ factual) minus the oracle under no, moderate and strong decay, averaged over the five models (Section H). The start-condition row uses a stress test of 200 runs, 24 of which have early estimates of s_t biased low.

Component	Change	Measure	CFS	Changed
EMA	removed ($\beta = 0$)	stop / top-1	63.2 / 77.18	52.5 / 75.98
	$\beta = 0.995$	stop / top-1	63.2 / 77.18	73.5 / 75.88
Time constant ℓ	β fixed, intervals $\times\frac{1}{2}/\times 2$	stop	63.4 / 63.5	56.2 / 81.3
	$\beta = 0.99$, 3000 updates	score gap	$-0.43/-0.07/-0.11$	$-0.12/-0.60/-1.71$
Start condition	removed	premature stops	0 / 200	24 / 200
Fisher batch	$B_p = 4B \to B/4$	stop / top-1	63.2 / 77.18	78.1 / 76.03
Threshold	$\tau = 0.04$ / 0.06	stop	63.2	66.6 / 60.4
Patience, SNR test	$K = 1$ / $\kappa = 0$	stop	63.2	63.2 / 63.1
Fisher products	Hessian products	peak memory	$+5\%$	$+232\%$
Warm start	cold start	power steps	2.5	12.8

5.4 Ablation Studies

Table 2 removes or replaces each part of CFS in turn, and Section E gives the complete sweeps.

Smoothing is necessary, and its time constant sets the stop. Without smoothing the rule stops after the crossing, at epoch 52.5, and loses 1.2 points. Factors up to $\beta = 0.95$ barely delay the stop, larger ones move it later, to epoch 59.9 at 0.99 and 73.5 at 0.995, and $\beta = 0.999$ never stops, as Proposition 1 predicts. The stop depends on β and the intervals only through ℓ. Halving or doubling all intervals at fixed β moves the stop to epoch 56.2 or 81.3, while at fixed ℓ the same change moves it by at most 0.1 epochs. The closed form (9) stays within 1.1 epochs of the discrete rule, and its slope $\ell_1 = 15.1$ epochs in $\ln \tau$ matches the sweep: for $0.02 \le \tau \le 0.10$ the stop follows $63.2 - 15.1 \ln(\tau/0.05)$ epochs within 0.5 epochs, and accuracy peaks at $\tau = 0.05$, where c was set.

A time constant set for long runs is too slow on short ones. On 3000-update runs, $\beta = 0.99$ trails the oracle by 0.60 and 1.71 points under moderate and strong decay, CFS by 0.07 and 0.11. Without decay the optimum lies at 93% to 94% of the budget, where $\beta = 0.99$ is 0.31 points above CFS.

The start condition removes premature stops. In a stress test of 200 runs, 24 of which start with estimates biased low, 12% of the runs stop prematurely without the start condition, with or without the SNR test. With it none of the 200 runs does, and the mean accuracy rises from 68.40 to 77.03.

The Fisher batch must be large enough. Shrinking the Fisher batch size from four training batches to a quarter of one moves the stop later, from epoch 63.2 to epoch 78.1, in line with Proposition 2.

The stop is insensitive to K and κ. They move the stop by less than one epoch over $K \in [1, 5]$ and $\kappa \in [0, 20]$, since near the stop the EMA keeps only 0.4% of the noise variance (Proposition 4). We still keep the SNR test as a safeguard for small Fisher batches, where it rejects 4% to 6% of the checks (Table 6). **Fisher products and warm starts keep the cost low.** Fisher products add 5% memory where Hessian products add 232%, and warm starts cut the power steps per check fivefold.

6 Conclusion

The effective learning rate in units of the curvature stability threshold, with the Fisher estimated from sampled labels, falls through a fixed threshold 5% to 15% of the budget before the optimum and then settles at a floor in all seven main regimes. A rule built on it should wait for a time proportional to the budget, and an EMA with a budget-scaled time constant makes this wait the same fraction of every budget and bounds it in closed form. The resulting rule uses no labels while it runs, matches validation-based stopping and saves 20% to 59% of the compute across these regimes. **Limitations.** The statistic follows the confidence of the model on its training inputs, so cyclic learning rates and gains that come after the floor can make it stop too early (Section J). Its constants are set per regime with development labels, although one shared value of c also works well on our held-out runs.

Reproducibility Statement

The statistic and the rule are given by Equations (1), (2) and (4) to (6) and Algorithm 1, the estimator by Section B.3, the constants of every regime by Sections C and C.3, the per-seed stops by Section D, and the proofs by Section B. The hardware and the software versions are listed in Section C.1.

AI Use Statement

Generative AI tools were used only to edit the wording of text written by the authors, to search for related work and check bibliographic details, and to write code for plots, tables and typesetting. They were not used for any task that the ICLR 2027 AI policy requires authors to disclose: developing theory, formulating or proving mathematical claims, proposing or refining hypotheses, designing, implementing or running the method or experiments, generating, cleaning or analyzing data, translating text, or interpreting results. The models in our experiments are objects of study, not writing aids. The authors reviewed all AI-assisted work and take full responsibility for the content of this paper.

References

Kumar Krishna Agrawal, Arnab Kumar Mondal, Arna Ghosh, and Blake A. Richards. α-ReQ: Assessing representation quality in self-supervised learning by measuring eigenspectrum decay. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, 2022.

Shun-ichi Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10(2):251–276, 1998.

Devansh Arpit, Stanisław Jastrzębski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S. Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, and Simon Lacoste-Julien. A closer look at memorization in deep networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, volume 70 of *Proceedings of Machine Learning Research*, pp. 233–242, 2017.

Atılım Güneş Baydin, Barak A. Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind. Automatic differentiation in machine learning: a survey. *Journal of Machine Learning Research*, 18(153):1–43, 2018.

David Bonet, Antonio Ortega, Javier Ruiz-Hidalgo, and Sarath Shekkizhar. Channel-wise early stopping without a validation set via NNK polytope interpolation. In *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pp. 351–358, 2021.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1597–1607, 2020.

Jeremy M. Cohen, Simran Kaur, Yuanzhi Li, J. Zico Kolter, and Ameet Talwalkar. Gradient descent on neural networks typically occurs at the edge of stability. In *International Conference on Learning Representations (ICLR)*, 2021.

Jeremy M. Cohen, Behrooz Ghorbani, Shankar Krishnan, Naman Agarwal, Sourabh Medapati, Michal Badura, Daniel Suo, David Cardoze, Zachary Nado, George E. Dahl, and Justin Gilmer. Adaptive gradient methods at the edge of stability. *arXiv preprint arXiv:2207.14484*, 2022.

Alex Damian, Eshaan Nichani, and Jason D. Lee. Self-stabilization: The implicit bias of gradient descent at the edge of stability. In *International Conference on Learning Representations (ICLR)*, 2023.

Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 10041–10071, 2024.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 248–255, 2009.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*, pp. 4171–4186, 2019.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021.

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference (ITCS)*, pp. 214–226, 2012.

Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The PASCAL visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010.

William Feller. *An Introduction to Probability Theory and Its Applications*, volume II. John Wiley & Sons, second edition, 1971.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations (ICLR)*, 2021.

Mahsa Forouzesh and Patrick Thiran. Disparity between batches as a signal for early stopping. In *Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*, volume 12976 of *Lecture Notes in Computer Science*, pp. 217–232. Springer, 2021.

Quentin Garrido, Randall Balestriero, Laurent Najman, and Yann LeCun. RankMe: Assessing the downstream performance of pretrained self-supervised representations by their rank. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, volume 202 of *Proceedings of Machine Learning Research*, pp. 10929–10974, 2023.

Behrooz Ghorbani, Shankar Krishnan, and Ying Xiao. An investigation into neural net optimization via Hessian eigenvalue density. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2232–2241, 2019.

Justin Gilmer, Behrooz Ghorbani, Ankush Garg, Sneha Kudugunta, Behnam Neyshabur, David Cardoze, George E. Dahl, Zachary Nado, and Orhan Firat. A loss curvature perspective on training instabilities of deep learning models. In *International Conference on Learning Representations (ICLR)*, 2022.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15979–15988, 2022.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations (ICLR)*, 2021.

Sepp Hochreiter and Jürgen Schmidhuber. Flat minima. *Neural Computation*, 9(1):1–42, 1997.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

Arash Jamshidi, Lauri Seppäläinen, Katsiaryna Haitsiukevich, Hoang Phuc Hau Luu, Anton Björklund, and Kai Puolamäki. GRADSTOP: Early stopping of gradient descent via posterior sampling. In *Proceedings of the 28th European Conference on Artificial Intelligence (ECAI)*, volume 413 of *Frontiers in Artificial Intelligence and Applications*, pp. 2057–2064. IOS Press, 2025.

Stanisław Jastrzębski, Maciej Szymczak, Stanislav Fort, Devansh Arpit, Jacek Tabor, Kyunghyun Cho, and Krzysztof Geras. The break-even point on optimization trajectories of deep neural networks. In *International Conference on Learning Representations (ICLR)*, 2020.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7B. *arXiv preprint arXiv:2310.06825*, 2023.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

Dayal Singh Kalra and Maissam Barkeshli. Why warmup the learning rate? Underlying mechanisms and improvements. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 111760–111801, 2024.

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. In *International Conference on Learning Representations (ICLR)*, 2017.

Vladimir Koltchinskii and Karim Lounici. Concentration inequalities and moment bounds for sample covariance operators. *Bernoulli*, 23(1):110–133, 2017.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Frederik Kunstner, Philipp Hennig, and Lukas Balles. Limitations of the empirical Fisher approximation for natural gradient descent. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, pp. 4158–4169, 2019.

Junnan Li, Richard Socher, and Steven C. H. Hoi. DivideMix: Learning with noisy labels as semi-supervised learning. In *International Conference on Learning Representations (ICLR)*, 2020a.

Mengtian Li, Ersin Yumer, and Deva Ramanan. Budgeted training: Rethinking deep neural network training under resource constraints. In *International Conference on Learning Representations (ICLR)*, 2020b.

Zhiyuan Li, Tianhao Wang, and Sanjeev Arora. What happens after SGD reaches zero loss? –A mathematical framework. In *International Conference on Learning Representations (ICLR)*, 2022.

Zhibin Liao, Tom Drummond, Ian Reid, and Gustavo Carneiro. Approximate Fisher information matrix to characterize the training of deep neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(1):15–26, 2020.

Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 3214–3252, 2022.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*, volume 8693 of *Lecture Notes in Computer Science*, pp. 740–755. Springer, 2014.

Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 20331–20342, 2020.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *IEEE International Conference on Computer Vision (ICCV)*, pp. 3730–3738, 2015.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019.

Maren Mahsereci, Lukas Balles, Christoph Lassner, and Philipp Hennig. Early stopping without a validation set. *arXiv preprint arXiv:1703.09580*, 2017.

Stephan Mandt, Matthew D. Hoffman, and David M. Blei. Stochastic gradient descent as approximate Bayesian inference. *Journal of Machine Learning Research*, 18(134):1–35, 2017.

James Martens. Deep learning via Hessian-free optimization. In *Proceedings of the 27th International Conference on Machine Learning (ICML)*, pp. 735–742, 2010.

James Martens. New insights and perspectives on the natural gradient method. *Journal of Machine Learning Research*, 21(146):1–76, 2020.

H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 54 of *Proceedings of Machine Learning Research*, pp. 1273–1282, 2017.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations (ICLR)*, 2017.

Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. Deep double descent: Where bigger models and more data hurt. In *International Conference on Learning Representations (ICLR)*, 2020.

Vardan Papyan. Measurements of three-level hierarchical structure in the outliers in the spectrum of deepnet Hessians. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, volume 97 of *Proceedings of Machine Learning Research*, pp. 5012–5021, 2019.

Barak A. Pearlmutter. Fast exact multiplication by the Hessian. *Neural Computation*, 6(1):147–160, 1994.

Boris T. Polyak. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17, 1964.

Lutz Prechelt. Early stopping—but when? In *Neural Networks: Tricks of the Trade*, volume 1524 of *Lecture Notes in Computer Science*, pp. 55–69. Springer, 1998.

Garvesh Raskutti, Martin J. Wainwright, and Bin Yu. Early stopping and non-parametric regression: An optimal data-dependent stopping rule. *Journal of Machine Learning Research*, 15(11):335–366, 2014.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 28, pp. 91–99, 2015.

S. W. Roberts. Control chart tests based on geometric moving averages. *Technometrics*, 1(3):239–250, 1959.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations (ICLR)*, 2020.

Nicol N. Schraudolph. Fast curvature matrix-vector products for second-order gradient descent. *Neural Computation*, 14(7):1723–1738, 2002.

Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum in deep learning. In *Proceedings of the 30th International Conference on Machine Learning (ICML)*, volume 28 of *Proceedings of Machine Learning Research*, pp. 1139–1147, 2013.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following LLaMA model. https://github.com/tatsu-lab/stanford_alpaca, 2023.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Joel A. Tropp. An introduction to matrix concentration inequalities. *Foundations and Trends in Machine Learning*, 8(1–2):1–230, 2015.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations (ICLR)*, 2019.

Lei Wu, Chao Ma, and Weinan E. How SGD selects the global minima in over-parameterized learning: A dynamical stability perspective. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, pp. 8289–8298, 2018.

Tong Xiao, Tian Xia, Yi Yang, Chang Huang, and Xiaogang Wang. Learning from massive noisy labeled data for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2691–2699, 2015.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Yuan Yao, Lorenzo Rosasco, and Andrea Caponnetto. On early stopping in gradient descent learning. *Constructive Approximation*, 26(2):289–315, 2007.

Zhewei Yao, Amir Gholami, Kurt Keutzer, and Michael W. Mahoney. PyHessian: Neural networks through the lens of the Hessian. In *IEEE International Conference on Big Data*, pp. 581–590, 2020.

Yi Yu, Tengyao Wang, and Richard J. Samworth. A useful variant of the Davis–Kahan theorem for statisticians. *Biometrika*, 102(2):315–323, 2015.

Suqin Yuan, Lei Feng, and Tongliang Liu. Early stopping against label noise without validation data. In *International Conference on Learning Representations (ICLR)*, 2024.

Suqin Yuan, Runqi Lin, Lei Feng, Bo Han, and Tongliang Liu. Instance-dependent early stopping. In *International Conference on Learning Representations (ICLR)*, 2025.

Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations (ICLR)*, 2018.

APPENDIX

A Notation and Algorithm **16**

B Proofs and Derivations **17**

B.1 Effective Learning Rate under Momentum and AdamW 17

B.2 Proof of Lemma 1 18

B.3 Fisher-Vector Products without Double Backpropagation 19

B.4 The Split-Half SNR Test 19

B.5 Proof of Theorem 1 20

B.6 Proof of Proposition 1 21

B.7 Delay with Three Check Intervals and Discrete Checks 22

B.8 Approximate Budget Invariance 23

B.9 Proof of Proposition 2 24

B.10 Smoothing of Estimation Noise 25

B.11 Power Iteration with Warm Starts 26

B.12 Negative Curvature 26

B.13 A Local Quadratic Model of the Statistic 27

C Experimental Details **27**

C.1 Datasets, Models and Hardware 27

C.2 Baselines 27

C.3 Calibration 28

D Results on the Main Regimes **29**

D.1 Per-Seed Results 29

D.2 Curvature Matrices and AdamW 29

D.3 Accuracy around the Stop 30

D.4 Consistency across Seeds 30

D.5 Monitored Quantities of One Run 30

D.6 Self-Supervised Pretraining 30

D.7 Stability under New Fisher Batches 30

E Ablation Studies **33**

E.1 Smoothing and the Time Constant 33

E.2 The Time Constant and the Budget 34

E.3 The Start Condition and the SNR Test 34

E.4 Random Hyperparameter Settings 35

E.5 Estimator Choices 35

F Generality **36**

F.1 Architectures and GLUE . . . 36
F.2 Fine-Tuning Large Language Models . . . 36
F.3 Detection, Fairness, Real Label Noise and Transfer . . . 36
F.4 CFS as a Plug-In . . . 37
F.5 Learning-Rate Schedules . . . 37
G Held-Out Runs . . . 38
H Short-Budget Runs of Language Models . . . 41
I Computational Cost . . . 44
J Failure Cases and Limitations . . . 44

A Notation and Algorithm

Table 3 collects the symbols of the paper, and Algorithm 1 gives the full procedure of CFS. In Algorithm 1, the EMA starts at the first accepted value, so it is defined from its first check on. Values below τ count toward the patience only after the start condition holds. A rejected check leaves the EMA and the counter unchanged, and the next check interval is then set by the last accepted value.

Table 3: **Notation.** Symbols of the main text and the appendix, grouped by the data and the model, the curvature matrices, the optimizer, the statistic and the rule, and the analysis, in this order.

Symbol	Meaning
$\theta_t \in \mathbb{R}^d$, t, N	parameters after t updates, update index, budget in updates
B, B_p	training batch size, Fisher batch size
$p_\theta(y \mid x)$, $\hat{y}_i$, g_i	predictive distribution, label sampled from it, score $\nabla_\theta \log p_\theta(\hat{y}_i \mid x_i)$
f_θ, m, $T(y)$	natural parameters of the output, their dimension, sufficient statistic
$\mathcal{F}_t$, G_t, H_t	sampled-label Fisher (1), Gauss-Newton matrix of the Fisher batch, Hessian
J_i, Λ_i, p_i	Jacobian of the outputs, covariance of $T(y)$, predicted probabilities of input i
η_t, h, μ	learning rate, schedule shape with $\eta_t = \eta_0 h(t/N)$, SGD momentum
β_1, $\hat{v}_t$, ϵ, P_t	decay rate of the first moment of Adam, second-moment estimate, its offset, AdamW preconditioner
η_t^{eff}, $\mathcal{C}_t$, $\bar{\mathcal{C}}_t$	effective learning rate, monitored matrix ($\mathcal{F}_t$ for SGD, $P_t^{1/2}\mathcal{F}_t P_t^{1/2}$ for AdamW), its mean
λ_t, ρ_t	top eigenvalue of $\mathcal{C}_t$, effective rank $\operatorname{tr}\bar{\mathcal{C}}_t/\lambda_{\max}(\bar{\mathcal{C}}_t)$
$\hat{\lambda}$, $\hat{\lambda}^{(1)}$, $\hat{\lambda}^{(2)}$, ϵ_0	estimate of λ_t, Rayleigh quotients on the two halves of the Fisher batch, offset of the SNR
s_t, $\tilde{s}_t$	normalized statistic (2) and its EMA (5)
τ, K, κ	threshold, patience (consecutive checks with $\tilde{s} < \tau$), SNR threshold
β	decay factor of the EMA per accepted check (unrelated to β_1)
$\Delta_1, \Delta_2, \Delta_3$; ℓ, ℓ_z	check intervals; time constant $\ell = \Delta/(-\ln\beta)$ and ℓ_z of interval Δ_z
c, γ	time constant of the shortest interval as a fraction of the budget, $\ell_3 = cN$; $\gamma = \ell/N$ for one time constant
$u = t/N$, u_0, S, $\tilde{S}$	budget fraction, first check, noise-free statistic and its EMA as functions of u
u_c, t_c; u_{stop}, t_{stop}; D	crossing; stop (∞ if the rule never stops); time from t_c to the first $\tilde{s} < \tau$
$\underline{S}$, $\bar{S}$, u_f, ε	lower and upper bounds on S after the crossing, fraction from which $S \le \bar{S}$, deviation from S
k_N, $\bar{k}_N$	smoothing kernel at budget N and its tail mass $\bar{k}_N(x) = \int_x^\infty k_N$
ν, q	bound on $\|P_t^{1/2} g_i\|_2^2$, acceptance rate of the SNR test
s_j°, ξ_j, σ^2, σ_ψ^2	noise-free statistic at check j, its estimation noise, bound on the variance and sub-Gaussian proxy of the noise

Algorithm 1 Curvature Floor Stopping (CFS)

Require: budget N, threshold τ, patience K, SNR threshold κ, check intervals $(\Delta_1, \Delta_2, \Delta_3)$, fraction c, Fisher batch size B_p

$\beta \leftarrow \exp(-\Delta_3/(cN))$; $\tilde{s} \leftarrow \varnothing$; $n \leftarrow 0$; active $\leftarrow$ false; $v \leftarrow$ random unit vector; $t' \leftarrow \Delta_1$
for $t = 1, 2, \ldots, N$ **do**
  one optimizer update
  if $t = t'$ **then**
    draw a Fisher batch of B_p training inputs, sample $\hat{y}_i \sim p_\theta(\cdot \mid x_i)$, split the batch into two halves
    power iteration on $\mathcal{C}_t$ from v with Fisher-vector products until the relative change of the Rayleigh quotient is below 10^{-3}
    $\hat{\lambda}^{(1)}, \hat{\lambda}^{(2)} \leftarrow$ Rayleigh quotients of v on the two halves; $\hat{\lambda} \leftarrow \frac{1}{2}(\hat{\lambda}^{(1)} + \hat{\lambda}^{(2)})$
    if $\hat{\lambda} > 0$ **and** $\mathrm{SNR} \geq \kappa$ **then**
      $s \leftarrow \frac{1}{2}\eta_t^{\mathrm{eff}}\hat{\lambda}$; $\tilde{s} \leftarrow s$ if $\tilde{s} = \varnothing$, else $\tilde{s} \leftarrow \beta\tilde{s} + (1-\beta)s$
      active $\leftarrow$ active **or** $\tilde{s} \geq 3\tau$; $n \leftarrow n+1$ if (active and $\tilde{s} < \tau$), else $n \leftarrow 0$
      if $n \geq K$ **then**
        return θ_t
      end if
    end if
    $t' \leftarrow t + \Delta_1$ if $\tilde{s} = \varnothing$ or $\tilde{s} \geq 3\tau$; $t + \Delta_2$ if $\tau \leq \tilde{s} < 3\tau$; $t + \Delta_3$ otherwise
  end if
end for
return θ_N

B Proofs and Derivations

B.1 Effective Learning Rate under Momentum and AdamW

Heavy-ball momentum (Polyak, 1964) keeps a velocity $m_{t+1} = \mu m_t + \nabla\mathcal{L}(\theta_t)$ and updates $\theta_{t+1} = \theta_t - \eta\, m_{t+1}$. For a gradient that changes slowly over the memory of the velocity, $\nabla\mathcal{L}(\theta_t) \approx g$, unrolling the velocity gives

$$\begin{aligned} m_{t+1} &= \mu m_t + g = \mu\big(\mu m_{t-1} + g\big) + g = \mu^2 m_{t-1} + (1+\mu)\, g \\ &= \mu^3 m_{t-2} + \big(1 + \mu + \mu^2\big)\, g = \cdots = \mu^{t+1} m_0 + \Big(\sum_{j=0}^{t} \mu^j\Big) g \\ &= \mu^{t+1} m_0 + \frac{1-\mu^{t+1}}{1-\mu}\, g \;\xrightarrow[t\to\infty]{}\; \frac{g}{1-\mu}, \end{aligned} \tag{10}$$

so each update moves the parameters by $-\eta g/(1-\mu)$ in the steady state, and $\eta^{\mathrm{eff}} = \eta/(1-\mu)$. For stability we look at one quadratic mode x with curvature λ. Eliminating the velocity from $m_{t+1} = \mu m_t + \lambda x_t$ and $x_{t+1} = x_t - \eta m_{t+1}$ gives

$$\begin{aligned} x_{t+1} &= x_t - \eta\big(\mu m_t + \lambda x_t\big) = x_t - \mu\big(x_{t-1} - x_t\big) - \eta\lambda\, x_t \\ &= \big(1 + \mu - \eta\lambda\big)\, x_t - \mu\, x_{t-1}, \end{aligned} \tag{11}$$

with characteristic polynomial $r^2 + \alpha_1 r + \alpha_0$, where $\alpha_1 = \eta\lambda - 1 - \mu$ and $\alpha_0 = \mu$. Both roots lie in the unit disk if and only if $|\alpha_0| < 1$ and $|\alpha_1| < 1 + \alpha_0$, that is $0 < \eta\lambda < 2(1+\mu)$. In units of $s = \frac{1}{2}\eta^{\mathrm{eff}}\lambda$ this reads $s < (1+\mu)/(1-\mu)$, which is 19 for $\mu = 0.9$ and reduces to $s < 1$ for $\mu = 0$.

For Adam the first moment is a normalized average. On one preconditioned mode with curvature λ, $m_{t+1} = \beta_1 m_t + (1-\beta_1)\lambda x_t$ and $x_{t+1} = x_t - \eta\, m_{t+1}$, and eliminating m as above gives

$$x_{t+1} = \big(1 + \beta_1 - (1-\beta_1)\eta\lambda\big)\, x_t - \beta_1 x_{t-1}. \tag{12}$$

The same conditions give $0 < \eta\lambda < 2(1+\beta_1)/(1-\beta_1)$, which is 38 for $\beta_1 = 0.9$, the limit that Cohen et al. (2022) observe for the preconditioned sharpness. A constant gradient gives $m \to g$ for Adam and $m \to g/(1-\mu)$ for heavy ball, hence $\eta^{\mathrm{eff}} = \eta$ and $\eta^{\mathrm{eff}} = \eta/(1-\mu)$. In units of $s = \frac{1}{2}\eta^{\mathrm{eff}}\lambda$ both limits read $s < (1+\mu)/(1-\mu)$, with $\mu = \beta_1$ for Adam. The limit exceeds

one because momentum amplifies a constant gradient by $1/(1-\mu)$ but a gradient that alternates in sign only by $1/(1+\mu)$. We treat the preconditioner as frozen and neglect bias correction and weight decay. We keep the plain threshold $\eta^{\text{eff}}\lambda = 2$ of gradient descent as the unit of s_t, so τ is the same fraction of the stability limit for both families, namely $\tau/19 \approx 1/380$ of it for $\mu = \beta_1 = 0.9$.

For AdamW the update is $\theta_{t+1} = \theta_t - \eta_t P_t \hat{m}_t$ with $P_t = \operatorname{diag}((\sqrt{\hat{v}_t}+\epsilon)^{-1})$. On a quadratic with curvature matrix $\mathcal{F}$, with momentum omitted and P frozen, the linear update is $e_{t+1} = (I - \eta P\mathcal{F})e_t$ for the error e_t, and $P\mathcal{F}$ is not symmetric. Since $P \succ 0$ we can write $P\mathcal{F} = P^{1/2}\big(P^{1/2}\mathcal{F}P^{1/2}\big)P^{-1/2}$, so $P\mathcal{F}$ is similar to the symmetric positive semidefinite matrix $P^{1/2}\mathcal{F}P^{1/2}$ and has the same real, nonnegative eigenvalues. Equivalently, in the coordinates $\theta' = P^{-1/2}\theta$ the update is plain gradient descent with step η and curvature $P^{1/2}\mathcal{F}P^{1/2}$. Power iteration on the symmetric form costs one Fisher-vector product and two scalings per step, $P^{1/2}\mathcal{F}P^{1/2}v = P^{1/2}\big(\mathcal{F}(P^{1/2}v)\big)$. Table 4 checks this numerically on a 3×3 example with the identity and two other diagonal preconditioners.

Table 4: **The AdamW matrix on an example** (3×3 positive definite $\mathcal{F}$, $\eta = 10^{-3}$). The symmetric form $P^{1/2}\mathcal{F}P^{1/2}$ and product $P\mathcal{F}$ have the same top eigenvalue for each P (Difference: their gap).

P	$\operatorname{diag}(P)$	$\lambda_{\max}(\mathcal{F})$	$\lambda_{\max}(P^{1/2}\mathcal{F}P^{1/2})$	$\lambda_{\max}(P\mathcal{F})$	$\eta\lambda_{\max}/2$	Difference
identity	(1, 1, 1)	4.6134	4.6134	4.6134	2.31×10^{-3}	2.7×10^{-15}
anisotropic	(0.1, 2, 5)	4.6134	5.3221	5.3221	2.66×10^{-3}	3.6×10^{-15}
small scale	(0.02, 0.03, 0.07)	4.6134	0.1086	0.1086	5.43×10^{-5}	1.4×10^{-17}

What s_t bounds. For a step $\Delta\theta = -\eta_t^{\text{eff}} g$ along a gradient g, the curvature term of the change is bounded by the linear term,

$$\tfrac{1}{2}\,\Delta\theta^\top \mathcal{C}_t\,\Delta\theta = \tfrac{1}{2}(\eta_t^{\text{eff}})^2\, g^\top \mathcal{C}_t\, g \;\le\; \tfrac{1}{2}(\eta_t^{\text{eff}})^2 \lambda_t \|g\|_2^2 = s_t \cdot \eta_t^{\text{eff}} \|g\|_2^2, \tag{13}$$

where $\eta_t^{\text{eff}}\|g\|_2^2$ is the first-order decrease of the loss. When $\mathcal{C}_t$ is the Fisher, the left side is the KL divergence between the predictive distributions before and after the step, to second order. A small value of s_t thus means that the curvature no longer limits the size of the update along any direction.

B.2 Proof of Lemma 1

Let the negative log-likelihood be $L(f, y) = -\log p(y \mid f)$ with $p(y \mid f) = b(y)\exp\big(f^\top T(y) - A(f)\big)$ for a base measure b and the log-partition function A. Then $\nabla_f \log p(y \mid f) = T(y) - \nabla A(f) = T(y) - \mathbb{E}_{p(\cdot|f)}[T]$ and $\nabla_f^2 L = \nabla^2 A(f) = \operatorname{Cov}_{p(\cdot|f)}[T(y)] = \Lambda$. With $f = f_\theta(x_i)$ the score is $g_i = J_i^\top\big(T(\hat{y}_i) - \mathbb{E}[T]\big)$, and taking the expectation over $\hat{y}_i \sim p_\theta(\cdot \mid x_i)$ gives

$$\begin{aligned}\mathbb{E}\big[g_i g_i^\top\big] &= \mathbb{E}\Big[J_i^\top\big(T(\hat{y}_i) - \mathbb{E}[T]\big)\big(T(\hat{y}_i) - \mathbb{E}[T]\big)^\top J_i\Big] \\ &= J_i^\top\,\mathbb{E}\Big[\big(T(\hat{y}_i) - \mathbb{E}[T]\big)\big(T(\hat{y}_i) - \mathbb{E}[T]\big)^\top\Big]\,J_i = J_i^\top \Lambda_i J_i.\end{aligned} \tag{14}$$

Averaging over the B_p inputs of the Fisher batch gives $\mathbb{E}[\mathcal{F}_t \mid \theta_t, x_{1:B_p}] = G_t$. This is the Gauss-Newton matrix, since the Hessian of one example splits as $\nabla_\theta^2 L_i = J_i^\top \Lambda_i J_i + \sum_j \partial_{f_j} L\, \nabla_\theta^2 f_j$ and the first term is its positive semidefinite part (Martens, 2020). This proves part (i) of Lemma 1.

For part (ii), softmax classification has $T(y) = e_y$ and $\Lambda_i = \operatorname{diag}(p_i) - p_i p_i^\top$, which is positive semidefinite with trace $\sum_y p_{iy} - \sum_y p_{iy}^2 = 1 - \|p_i\|_2^2$. The first inequality of (3) is Jensen's inequality, since $\lambda_{\max}$ is convex (Section B.9). The second holds because $\mathcal{F}_t \succeq 0$ gives $\lambda_{\max}(\mathcal{F}_t) \le \operatorname{tr}\mathcal{F}_t$ and $\mathbb{E}[\operatorname{tr}\mathcal{F}_t \mid \theta_t, x_{1:B_p}] = \operatorname{tr} G_t$. For positive semidefinite A and B, $\operatorname{tr}(AB) \le \lambda_{\max}(B)\operatorname{tr} A$, so

$$\operatorname{tr} G_t = \frac{1}{B_p}\sum_i \operatorname{tr}\left(\Lambda_i J_i J_i^\top\right) \;\le\; \frac{1}{B_p}\sum_i \lambda_{\max}\left(J_i J_i^\top\right)\operatorname{tr}\Lambda_i = \frac{1}{B_p}\sum_i \left(1 - \|p_i\|_2^2\right)\|J_i\|_2^2, \tag{15}$$

where $\|J_i\|_2$ is the spectral norm. This proves part (ii). The Gini impurity is within a factor of two of the error of the top class, since $\|p_i\|_2^2 \le \max_y p_{iy}$ and $1 - \|p_i\|_2^2 \le 1 - (\max_y p_{iy})^2 \le 2(1 - \max_y p_{iy})$. For AdamW, replace J_i by $J_i P_t^{1/2}$, since the monitored matrix is $P_t^{1/2}\mathcal{F}_t P_t^{1/2}$.

For Gaussian regression with fixed variance σ_y^2 and the mean f as the natural parameter, $T(y) = y/\sigma_y^2$ and $\Lambda_i = I/\sigma_y^2$, so $G_t = \frac{1}{\sigma_y^2 B_p} \sum_i J_i^\top J_i$ does not depend on the residuals. The empirical Fisher with the true labels is $\frac{1}{\sigma_y^4 B_p} \sum_i J_i^\top (y_i - f_i)(y_i - f_i)^\top J_i$ and grows with the squared residuals. Table 5 shows both effects on a linear and a quadratic model. The sampled-label Fisher equals the Gauss-Newton matrix at every residual scale. The top eigenvalue of the empirical Fisher grows from 0 to 88.5 as the residual grows to 4, and the Hessian of the quadratic model turns negative definite.

Table 5: **Why labels are sampled from the model** ($n = 384$ samples, $d = 16$ parameters, Gaussian noise of variance 0.5; residual scales 0, 1 and 4; top eigenvalues unless noted). The mean sampled Fisher equals the Gauss-Newton matrix at every scale, while the empirical Fisher grows with it.

Model	Residual scale	$\lambda_{\max}(H)$	$\lambda_{\max}(G)$	$\lambda_{\max}(\mathbb{E}\mathcal{F})$	$\lambda_{\max}$ (empirical Fisher)	$\lambda_{\min}(H)$
linear	0 / 1 / 4	2.765	2.765	2.765	0 / 5.53 / 88.5	1.366
quadratic	0 / 1 / 4	2.807 / 1.989 / −0.170	2.807	2.807	0 / 5.61 / 89.8	1.358 / 0.944 / −0.592

Likelihoods for self-supervised pretraining. The two self-supervised regimes need a likelihood from which to sample. MAE minimizes the mean squared error over the M masked targets. We divide the reconstruction and the target by $\sqrt{M}$ and sample Gaussian pseudo-targets with variance $1/2$ in these units, so that $G = 2J^\top J/M$ is exactly the Gauss-Newton matrix of the loss averaged over the masked targets. For SimCLR each of the $2B_p$ augmented views i defines the softmax of the contrastive loss over the other views, $p_\theta(j \mid i)$, from which we sample a view $\hat{j}_i$ and form $g_i = \nabla_\theta \log p_\theta(\hat{j}_i \mid i)$ and $\mathcal{F} = \frac{1}{2B_p} \sum_i g_i g_i^\top$. The rank bound of Proposition 2(i) then holds with $2B_p$ in place of B_p, while the upper bound (31) does not apply as stated, since the views of one batch share candidates. For the MAE likelihood the Gauss-Newton matrix does not depend on the predictions, so there the fall of the statistic comes from the Jacobians and the learning-rate decay.

B.3 Fisher-Vector Products without Double Backpropagation

Write $\varphi_i(\theta) = \log p_\theta(\hat{y}_i \mid x_i)$ with the label $\hat{y}_i$ fixed after sampling, so that $g_i = \nabla_\theta \varphi_i$ and no derivative passes through the sampling. For a direction v and the scalar $a_i = g_i^\top v$, which is the derivative of $\alpha \mapsto \varphi_i(\theta + \alpha v)$ at $\alpha = 0$ and comes from one forward-mode pass (Baydin et al., 2018), define $\Phi(\theta; v) = \frac{1}{B_p} \sum_i \mathrm{sg}(a_i)\, \varphi_i(\theta)$. Then

$$\begin{aligned} \nabla_\theta \Phi(\theta; v) &= \frac{1}{B_p} \sum_{i=1}^{B_p} \Big(\underbrace{\nabla_\theta\, \mathrm{sg}(a_i)}_{=0}\, \varphi_i(\theta) + \mathrm{sg}(a_i)\, \nabla_\theta \varphi_i(\theta) \Big) \\ &= \frac{1}{B_p} \sum_{i=1}^{B_p} a_i\, g_i = \frac{1}{B_p} \sum_{i=1}^{B_p} g_i \big(g_i^\top v \big) = \mathcal{F}_t v, \end{aligned} \tag{16}$$

so one ordinary backward pass of Φ returns the product. Without the stop-gradient the extra term $\frac{1}{B_p} \sum_i \varphi_i \nabla_\theta a_i$ would contain second derivatives and would need double backpropagation. No per-example gradient is stored, since the forward-mode pass carries one tangent and the backward pass one weighted sum. The same pass also gives the split-half test. For a unit vector v the Rayleigh quotient on half h of the batch is $\hat{\lambda}^{(h)} = \frac{2}{B_p} \sum_{i \in h} a_i^2$. For AdamW the product is applied to $P_t^{1/2} v$ and scaled by $P_t^{1/2}$ after. The peak-memory overhead is 4.7% to 5.0% on seven models from 12M to 340M parameters, against as much as 232% for a Hessian-vector product on ResNet-50 (Section I).

B.4 The Split-Half SNR Test

Let $\hat{\lambda}^{(1)}$ and $\hat{\lambda}^{(2)}$ be the Rayleigh quotients of the same unit vector v on two disjoint halves of the Fisher batch. For independent examples and a fixed v, the two are independent and unbiased for $\lambda(v) = v^\top \bar{\mathcal{C}}_t v$, each with variance σ_λ^2. Their mean $\hat{\lambda}$ has variance $\sigma_\lambda^2/2$, and the denominator of the

SNR is unbiased for σ_λ^2,

$$\mathbb{E}\Big[\tfrac{1}{2}\big(\hat\lambda^{(1)}-\hat\lambda^{(2)}\big)^2\Big] = \tfrac{1}{2}\Big(\mathbb{E}\big[(\hat\lambda^{(1)})^2\big]+\mathbb{E}\big[(\hat\lambda^{(2)})^2\big]-2\,\mathbb{E}\big[\hat\lambda^{(1)}\big]\,\mathbb{E}\big[\hat\lambda^{(2)}\big]\Big)$$
$$= \tfrac{1}{2}\Big(\big(\sigma_\lambda^2+\lambda^2\big)+\big(\sigma_\lambda^2+\lambda^2\big)-2\lambda^2\Big) = \sigma_\lambda^2. \tag{17}$$

The SNR thus estimates $\lambda^2/\sigma_\lambda^2$, with one degree of freedom in the denominator, so it is a coarse filter. If the two halves are Gaussian, their mean and their difference are independent, and $\mathrm{SNR} = \frac{1}{2}F'_{1,1}(2\lambda^2/\sigma_\lambda^2)$ follows a scaled noncentral F distribution with one and one degrees of freedom. A check is then accepted at $\kappa = 10$ with probability 0.25, 0.46, 0.67, 0.88 and 0.998 when $\lambda^2/\sigma_\lambda^2$ is 1, 4, 10, 25 and 100. The test also selects on the noise, since small estimates are rejected more often. Among accepted checks the mean is biased upward by 43%, 11%, 3.6% and 0.9% at the first four of these ratios, which is consistent with the small delay of the stop as κ grows (Table 17). The acceptance rates of 94.4% to 99.8% in Table 6 thus indicate that most checks have a high ratio. Since v comes from the whole batch, the two quotients are also slightly biased upward and weakly dependent, and for SimCLR the views of the two halves share candidates. The test controls the spread of one estimate, and Proposition 2 bounds its bias. The three regimes with a Fisher batch smaller than four training batches reject 4% to 6% of their checks, and the other four reject at most 0.6%.

Table 6: **SNR test acceptance** at $\kappa = 10$ (five seeds, up to the stop) and Fisher batch B_p/B.

	ResNet-50	ViT-B	MAE	BERT-B	CIFAR-10, 40%	SimCLR	WikiText-103
Accepted / checks	3192 / 3202	6227 / 6264	1169 / 1175	623 / 651	1006 / 1008	5597 / 5852	1520 / 1610
Acceptance rate	99.7%	99.4%	99.5%	95.7%	99.8%	95.6%	94.4%
B_p/B	4	4	4	0.5	4	2	2

B.5 Proof of Theorem 1

Part (i). Under Assumption 1 the statistic is $s(t') = S(t'/N)$ and the smoother starts at $t_0 = u_0 N$. With the rescaled kernel $k_N(r) = k(r/N)/N$ and the substitution $r = wN$,

$$\tilde s(uN) = \int_0^\infty k_N(r)\, s\big(\max\{uN-r,\, u_0N\}\big)\,\mathrm{d}r = \int_0^\infty \frac{1}{N}\,k\Big(\frac{r}{N}\Big)\,S\Big(\frac{\max\{uN-r,\,u_0N\}}{N}\Big)\,\mathrm{d}r$$
$$= \int_0^\infty k(w)\,S\Big(\frac{\max\{uN-wN,\,u_0N\}}{N}\Big)\,\mathrm{d}w = \int_0^\infty k(w)\,S\big(\max\{u-w,\,u_0\}\big)\,\mathrm{d}w, \tag{18}$$

which is (7). The right-hand side does not involve N, so the set $\{u > u_0 : \tilde s(uN) < \tau\}$ and its infimum u_{stop} are the same for every budget. For the EMA kernel, $k(w) = e^{-w/\gamma}/\gamma$ with $\gamma = \ell/N$, and a second substitution $w = \gamma w'$ turns (18) into $\tilde s(uN) = \int_0^\infty e^{-w'} S(\max\{u-\gamma w', u_0\})\,\mathrm{d}w'$.

Part (ii). Take the step $S(u) = a$ for $u < u_1$ and $S(u) = 0$ for $u \ge u_1$, with $a > \tau$ and $u_0 < u_1 < 1$. For $t < u_1 N$ every argument of s in the smoother lies below $u_1 N$, so $\tilde s(t) = a > \tau$. For $t \ge u_1 N$ the argument $\max\{t-r, t_0\}$ lies below $u_1 N$ exactly when $r > t - u_1 N$, because $t_0 < u_1 N$, and therefore

$$\tilde s(t) = a\int_{t-u_1N}^\infty k_N(r)\,\mathrm{d}r = a\,\bar k_N(t-u_1N),$$
$$u_{\mathrm{stop}} = u_1 + \frac{Q_N(y)}{N}, \qquad Q_N(y) := \inf\{x \ge 0 : \bar k_N(x) < y\}, \qquad y = \frac{\tau}{a} \in (0,1), \tag{19}$$

if $u_1 + Q_N(y)/N < 1$, and $u_{\mathrm{stop}} = \infty$ otherwise, since then $\tilde s(t) \ge \tau$ for all $t \le N$. Fix two budgets N and N' and some $y \in (0,1)$. If $Q_{N'}(y)/N' < 1 - u_0$, choose $u_1 \in \big(u_0, 1 - Q_{N'}(y)/N'\big)$, so that the stop of budget N' lies inside the budget, and invariance gives $Q_N(y)/N = Q_{N'}(y)/N'$. If $Q_{N'}(y)/N' \ge 1 - u_0$, invariance for every $u_1 > u_0$ gives $Q_N(y)/N \ge 1 - u_0$. Hence

$$\min\{Q_N(y)/N,\, 1-u_0\} = \min\{Q_{N'}(y)/N',\, 1-u_0\} \qquad \text{for all } y \in (0,1). \tag{20}$$

Let $\Psi(x) = \bar k_{N'}(xN'/N)$, which is continuous and non-increasing with $\Psi(0) = 1$ and has the quantile function $(N/N')\,Q_{N'}$. Suppose that $\bar k_N(x^*) < \Psi(x^*)$ for some $x^* \in \big(0, (1-u_0)N\big)$,

and take y strictly between the two values. Continuity of $\bar{k}_N$ gives $Q_N(y) < x^*$, so the left side of (20) is below x^*/N. Since $\Psi > y$ on $[0, x^*]$, $(N/N')\,Q_{N'}(y) \geq x^*$, so the right side is at least $\min\{x^*/N, 1-u_0\} = x^*/N$, a contradiction. The case $\bar{k}_N(x^*) > \Psi(x^*)$ is symmetric, and continuity extends the equality to the closed interval, so $\bar{k}_N(x) = \bar{k}_{N'}(xN'/N)$ for $x \leq (1-u_0)N$. Since $\bar{k}_N$ is absolutely continuous with derivative $-k_N$, this is equivalent to $k_N(r) = (N'/N)\,k_{N'}(rN'/N)$ for almost every $r < (1-u_0)N$. For the EMA, $e^{-x/\ell_N} = e^{-xN'/(N\ell_{N'})}$ on an interval of positive length if and only if $\ell_N/N = \ell_{N'}/N'$, and part (i) gives the converse. □

Lags beyond $(1-u_0)N$ enter the smoother on $[u_0, 1]$ only through their total mass, so no condition on them can be derived. For $u_0 = 0$, the kernel $k_1(r) = e^{-r}$ and a kernel k_2 that equals $e^{-r/2}/2$ on $[0, 2]$ and spreads its remaining mass e^{-1} evenly over $(2, 3)$ give the same stop fractions for $N = 1$ and $N = 2$, although $\bar{k}_2(x) \neq \bar{k}_1(x/2)$ on $(2, 3)$. The rule of CFS is not a linear smoother, since it switches between check intervals at fixed levels of $\tilde{s}$. The next result extends part (i) to this rule.

Corollary 1 (Budget invariance with proportional intervals)**.** *Let Assumption 1 hold, remove the estimation noise, and let the check intervals be fixed fractions of the budget, $\Delta_z = \omega_z N$ for $z = 1, 2, 3$, with ω_z and β independent of N. Then the fractions t_j/N of the checks, the smoothed values $\tilde{s}_j$, the interval chosen after each check, the start condition, the counter of the patience and the stop fraction of Algorithm 1 are the same for every budget N.*

Proof. By induction over the checks. The first check is at $t_1/N = \omega_1$. If $t_1/N, \ldots, t_j/N$ do not depend on N, neither do the values $s_i = S(t_i/N)$ for $i \leq j$, the smoothed values given by (5), the start condition $\max_{i \leq j} \tilde{s}_i \geq 3\tau$, the counter, and the interval $\omega_{z_j} N$ chosen after check j. Hence $t_{j+1}/N = t_j/N + \omega_{z_j}$ does not depend on N either. The stop is the first check at which the counter reaches K, so its fraction is the same for every N. □

CFS instead fixes the intervals in updates and sets $\beta = \exp(-\Delta_3/(cN))$, so the time constants $\ell_z = (\Delta_z/\Delta_3)\,cN$ are proportional to N. In continuous time the EMA then solves $(\ell_z/N)\,\mathrm{d}\tilde{S}/\mathrm{d}u = S - \tilde{S}$ in phase z, which does not involve N. The discrete checks add a further error, at most $\Delta_1 + (K-1)\Delta_3$ after the crossing check as $s \downarrow 0$ (Corollary 3). We do not bound it for general S.

B.6 Proof of Proposition 1

Recursive form of the EMA. In continuous time the EMA solves $\ell\,\dot{\tilde{s}}(t) = s(t) - \tilde{s}(t)$ with $\tilde{s}(t_0) = s(t_0)$. Multiplying by the integrating factor $e^{t/\ell}$ and integrating from any $t' \geq t_0$ to t gives, in budget fractions with $\gamma = \ell/N$ and $\tilde{S}(u) = \tilde{s}(uN)$,

$$\frac{\mathrm{d}}{\mathrm{d}t}\Big(e^{t/\ell}\,\tilde{s}(t)\Big) = \frac{1}{\ell}\,e^{t/\ell}\,s(t) \quad\Longrightarrow\quad \tilde{s}(t) = e^{-(t-t')/\ell}\,\tilde{s}(t') + \frac{1}{\ell}\int_{t'}^{t} e^{-(t-r)/\ell}\,s(r)\,\mathrm{d}r,$$

$$\tilde{S}(u) = e^{-(u-u')/\gamma}\,\tilde{S}(u') + \frac{1}{\gamma}\int_{u'}^{u} e^{-(u-w)/\gamma}\,S(w)\,\mathrm{d}w \qquad (u \geq u' \geq u_0). \tag{21}$$

The smoothed value at u' and the statistic on $[u', u]$ thus fully determine the smoothed value at u.

The delay. On $[u_0, u_c)$, $\tilde{S}$ is an average of values of S that are at least τ, so $\tilde{S} \geq \tau$ there and, by continuity, at u_c. Hence $u_{\mathrm{stop}} \geq u_c$. For $u \geq u_c$, (21) with $u' = u_c$ and $S \geq \underline{S}$ gives

$$\tilde{S}(u) \;\geq\; \tilde{S}(u_c)\,e^{-(u-u_c)/\gamma} + \underline{S}\big(1 - e^{-(u-u_c)/\gamma}\big) \;=\; \underline{S} + \big(\tilde{S}(u_c) - \underline{S}\big)\,e^{-(u-u_c)/\gamma}, \tag{22}$$

which is at least τ as long as $u - u_c \leq \gamma \ln \frac{\tilde{S}(u_c) - \underline{S}}{\tau - \underline{S}}$. This is the lower bound. For the upper bound there is nothing to prove if $u_{\mathrm{stop}} \leq u_f$. Otherwise $\tilde{S} \geq \tau > S$ on $(u_c, u_f]$, so $\tilde{S}$ decreases there and $\tilde{S}(u_f) \leq \tilde{S}(u_c)$. For $u \geq u_f$, (21) with $u' = u_f$ and $S \leq \bar{S}$ gives

$$\tilde{S}(u) \;\leq\; \bar{S} + \big(\tilde{S}(u_f) - \bar{S}\big)\,e^{-(u-u_f)/\gamma} \;\leq\; \bar{S} + \big(\tilde{S}(u_c) - \bar{S}\big)\,e^{-(u-u_f)/\gamma}, \tag{23}$$

which is below τ once $u - u_f > \gamma \ln \frac{\tilde{S}(u_c) - \bar{S}}{\tau - \bar{S}}$. This gives (8). With $\underline{S} = \bar{S} = 0$ and a jump of S to zero at u_c, both bounds equal $\gamma \ln(\tilde{S}(u_c)/\tau)$. Keeping both exponentials gives the sharper bound $\tilde{S}(u) \leq \bar{S} + (\tau - \bar{S})e^{-(u-u_f)/\gamma} + (\tilde{S}(u_c) - \tau)e^{-(u-u_c)/\gamma}$ for $u \geq u_f$. The last statement of Proposition 1 is Lemma 2 below. □

Corollary 2 (Monotonicity and limits). *Under the assumptions of Proposition 1: (a) if S is non-increasing on $[u_0, 1]$, u_{stop} does not decrease as $\gamma = \ell/N$ grows; (b) if S is continuous on $[u_c, 1]$, then $u_{\text{stop}} \to u_c$ as $\gamma \to 0$; (c) if $\gamma \ln(\tilde{S}(u_c)/\tau) > 1 - u_c$, the noise-free rule does not stop before the end of the budget.*

Proof. (a) By part (i) of Theorem 1, $\tilde{S}_\gamma(u) = \int_0^\infty e^{-w} S(\max\{u - \gamma w, u_0\})\,\mathrm{d}w$, and each integrand does not decrease as γ grows. Hence $\tilde{S}_{\gamma_1} \le \tilde{S}_{\gamma_2}$ for $\gamma_1 < \gamma_2$, and every fraction with $\tilde{S}_{\gamma_2} < \tau$ has $\tilde{S}_{\gamma_1} < \tau$. (b) Fix $\vartheta \in (0, 1 - u_c)$, and let $\bar{S} = \max_{[u_c+\vartheta,1]} S$, which is below τ, and $M = \sup S$. Proposition 1 with $u_f = u_c + \vartheta$ gives $u_c \le u_{\text{stop}} \le u_c + \vartheta + \gamma \ln \frac{M - \bar{S}}{\tau - \bar{S}}$ for small γ, and the right side tends to $u_c + \vartheta$. (c) This is the lower bound of Proposition 1 with $\underline{S} = 0$. □

Lemma 2 (Recursive linear smoothers). *Let k be a probability density on $[0, \infty)$ and $\tilde{s}(t) = \int_0^\infty k(r)\, s(t - r)\,\mathrm{d}r$ for bounded measurable $s \ge 0$ on $\mathbb{R}$. Suppose that for every lag $x > 0$ the value $\tilde{s}(t + x)$ is a function of $\tilde{s}(t)$ and of s on $(t, t + x]$, for all such s and all t. Then $k(r) = e^{-r/\ell}/\ell$ for almost every r and some $\ell > 0$, and for every r if k is right-continuous.*

Proof. Fix $x > 0$ and a bounded measurable χ on $[0, \infty)$ with $\int_0^\infty k(r)\chi(r)\,\mathrm{d}r = 0$. Let $M \ge \sup|\chi|$, and take two histories that agree on $(t, t + x]$ and satisfy $s_1(t - r) = M + \chi(r)$ and $s_2(t - r) = M$ for $r \ge 0$. They give the same $\tilde{s}(t) = M$ and the same input on $(t, t + x]$, hence the same $\tilde{s}(t + x)$. Splitting the smoother at lag x,

$$\begin{aligned}\tilde{s}(t + x) &= \int_0^x k(r)\, s(t + x - r)\,\mathrm{d}r + \int_x^\infty k(r)\, s(t + x - r)\,\mathrm{d}r \\ &= \underbrace{\int_0^x k(r)\, s(t + x - r)\,\mathrm{d}r}_{\text{depends on } s \text{ on } (t, t+x]} + \int_0^\infty k(x + r)\, s(t - r)\,\mathrm{d}r, \end{aligned} \tag{24}$$

shows that $\int_0^\infty k(x + r)\chi(r)\,\mathrm{d}r = 0$. The linear functional $L_x(\chi) = \int_0^\infty k(x + r)\chi(r)\,\mathrm{d}r$ thus vanishes on the kernel of $L_0(\chi) = \int_0^\infty k(r)\chi(r)\,\mathrm{d}r$, which is not identically zero, so $L_x = \phi(x)L_0$ for a constant $\phi(x)$. Applying this to $\chi = \mathbf{1}_{[a,b]}$ for all $a < b$ gives $k(x + r) = \phi(x)k(r)$ for almost every $r \ge 0$. Integrating over $r \in [y, \infty)$ gives $\bar{k}(x + y) = \phi(x)\bar{k}(y)$ for the tail $\bar{k}(x) = \int_x^\infty k$, and $y = 0$ gives $\phi(x) = \bar{k}(x)$. Hence

$$\bar{k}(x + y) = \bar{k}(x)\,\bar{k}(y) \qquad \text{for all } x, y \ge 0. \tag{25}$$

The tail $\bar{k}$ is continuous with $\bar{k}(0) = 1$. If $\bar{k}(x_0) = 0$ for some $x_0 > 0$, then $\bar{k}(x_0) = \bar{k}(x_0/2^n)^{2^n}$ gives $\bar{k}(x_0/2^n) = 0$ for all n, which contradicts continuity at 0. So $\ln \bar{k}$ is finite, continuous and additive, hence linear, and $\bar{k}(x) = e^{-x/\ell}$ with $\ell > 0$ because $\bar{k}(x) \to 0$ as $x \to \infty$. Then $k = -\bar{k}' = e^{-r/\ell}/\ell$ almost everywhere, and everywhere if k is right-continuous. □

The identity (25) is the memoryless property that characterizes the exponential distribution (Feller, 1971). The lemma concerns linear smoothers. A recursive rule $\tilde{s}_j = \Phi(\tilde{s}_{j-1}, s_j)$, such as the smoother of CFS with three check intervals, also carries its past in one number, and Corollary 3 uses exactly this at the switch between intervals. The conclusion fails if the property is required for one lag only. Every $k(r) = p(r)e^{-r/\ell}/Z$ with $p > 0$ of period Δ satisfies $k(\Delta + r) = e^{-\Delta/\ell}k(r)$, hence $\tilde{s}(t + \Delta) = \int_0^\Delta k(r)s(t + \Delta - r)\,\mathrm{d}r + e^{-\Delta/\ell}\tilde{s}(t)$. With checks at a fixed check interval Δ, the geometric weights of (5) are the discrete counterpart of the exponential kernel, with $\beta = e^{-\Delta/\ell}$.

B.7 Delay with Three Check Intervals and Discrete Checks

The rule of Section 4.2 uses one factor β for all checks, so interval Δ_z has the time constant $\ell_z = \Delta_z/(-\ln\beta)$, and $\beta = \exp(-\Delta_3/(cN))$ gives $\ell_z = (\Delta_z/\Delta_3)\,cN$. After the crossing $t_c = u_c N$, the EMA of s remains in the first phase while $\tilde{s} \ge 3\tau$ and in the second phase while $\tau \le \tilde{s} < 3\tau$.

Corollary 3 (Delay with three check intervals). *In continuous time let the smoothed value solve $\ell(\tilde{s})\,\dot{\tilde{s}} = s - \tilde{s}$, with $\ell(y) = \ell_1$ for $y \ge 3\tau$ and $\ell(y) = \ell_2$ for $\tau \le y < 3\tau$. Let $\tilde{s}(t_c) \ge 3\tau$ at the*

crossing t_c, and $\underline{S} \le s \le \bar{S} < \tau$ after it. The time D from t_c to the first time with $\tilde{s} < \tau$ satisfies

$$\ell_1 \ln \frac{\tilde{s}(t_c) - \underline{S}}{3\tau - \underline{S}} + \ell_2 \ln \frac{3\tau - \underline{S}}{\tau - \underline{S}} \;\le\; D \;\le\; \ell_1 \ln \frac{\tilde{s}(t_c) - \bar{S}}{3\tau - \bar{S}} + \ell_2 \ln \frac{3\tau - \bar{S}}{\tau - \bar{S}}. \tag{26}$$

With checks at the intervals of Algorithm 1, the factor β per check, and $s = 0$ after the crossing check with every such check accepted (the limit $s \downarrow 0$), the delay to the first check with $\tilde{s} < \tau$ exceeds the value for $\underline{S} = \bar{S} = 0$ by an amount in $(0, \Delta_1]$, and the patience adds $(K-1)\Delta_3$.

Proof. While $\tilde{s} \ge \tau > \bar{S} \ge s$ we have $\dot{\tilde{s}} < 0$, so $\tilde{s}$ decreases strictly until it reaches τ, and $(\tilde{s} - \bar{S})/\ell(\tilde{s}) \le -\dot{\tilde{s}} \le (\tilde{s} - \underline{S})/\ell(\tilde{s})$. Changing the variable from t to $y = \tilde{s}(t)$ gives $D = \int_\tau^{\tilde{s}(t_c)} \mathrm{d}y/(-\dot{\tilde{s}})$, which lies between $\int_\tau^{\tilde{s}(t_c)} \ell(y)\,\mathrm{d}y/(y - \underline{S})$ and $\int_\tau^{\tilde{s}(t_c)} \ell(y)\,\mathrm{d}y/(y - \bar{S})$. Splitting both integrals at 3τ gives (26). For the checks, $s = 0$ gives $\tilde{s}_j = \beta^j \tilde{s}_0$ at the j-th check after the crossing check, where $\tilde{s}_0$ is the EMA at that check. Let $r_1 = \ln(\tilde{s}_0/3\tau)/(-\ln\beta)$ and $r = \ln(\tilde{s}_0/\tau)/(-\ln\beta)$. Then $M_1 = \lfloor r_1 \rfloor + 1$ checks have $\tilde{s}_j \ge 3\tau$ and $M = \lfloor r \rfloor + 1$ checks have $\tilde{s}_j \ge \tau$, and the interval after check j is Δ_1 for $j < M_1$ and Δ_2 for $M_1 \le j < M$. The discrete delay is $M_1\Delta_1 + (M - M_1)\Delta_2$, the continuous one is $\ell_1 \ln \frac{\tilde{s}_0}{3\tau} + \ell_2 \ln 3 = r_1\Delta_1 + (r - r_1)\Delta_2$, and their difference

$$(M_1 - r_1)(\Delta_1 - \Delta_2) + (M - r)\,\Delta_2 \tag{27}$$

lies in $(0, \Delta_1]$ because $M_1 - r_1$ and $M - r$ lie in $(0, 1]$. □

With $\underline{S} = \bar{S} = 0$ both sides of (26) equal (9). If the statistic reaches the level $\bar{S}$ only at $t_f > t_c$, the upper bound holds with $t_f - t_c$ added, since $\tilde{s}(t_f) \le \tilde{s}(t_c)$. The corollary rests on the recursive form of the EMA: the value 3τ at the switch of intervals carries the whole past, so the second phase always takes between $\ell_2 \ln \frac{3\tau - \underline{S}}{\tau - \underline{S}}$ and $\ell_2 \ln \frac{3\tau - \bar{S}}{\tau - \bar{S}}$. A rejected check advances time without decaying the EMA, so with an acceptance rate q the time constants become ℓ_z/q, at most 6% above ℓ_z here (Table 6). Table 16 shows that (9) lies 0.5 to 1.1 epochs below the discrete rule with smoothing.

As a check on the calibration, take seed 42 of ResNet-50, with $\tilde{s}(t_c) = 0.216$ and the interval ratios $\Delta_1/\Delta_3 = 13.6$ and $\Delta_2/\Delta_3 = 3.4$. The value of c at which (9) equals the lead of 11.2 epochs, from the crossing at epoch 52.8 to epoch 64, is 0.014. The calibrated $c = 0.0123$ is slightly smaller for two reasons. The closed form (9) neglects the statistic after the crossing and so underestimates the delay of the discrete rule, and the objective (39) also charges the cost, which favors earlier stops.

B.8 Approximate Budget Invariance

Assumption 1 is an idealization, since runs with different budgets never follow exactly one profile. The next result shows that the stop fraction moves only a little when a run deviates from the profile S.

Proposition 3 (Approximate budget invariance)**.** *Let $\tilde{S}$ be the EMA of S in budget fractions with $\gamma = \ell/N$, started at $\tilde{S}(u_0) = S(u_0)$, and let $u^* \in (u_0, 1)$ be its stop. Assume that $S \le \bar{S} < \tau$ on $[u_f, 1]$ for some $u_f \le u^* - \gamma\varepsilon/(\tau - \bar{S})$, that $\tilde{S} \ge \tau + \varepsilon$ on $[u_0, u_f]$, and that $u^* + \gamma\varepsilon/(\tau - \bar{S} - \varepsilon) < 1$. If a run of budget N has a noise-free statistic S_N with $\sup_{[u_0,1]} |S_N - S| \le \varepsilon < \tau - \bar{S}$, the stop u_N^* of the EMA of S_N with the same γ satisfies*

$$-\frac{\gamma\varepsilon}{\tau - \bar{S}} \;\le\; u_N^* - u^* \;\le\; \frac{\gamma\varepsilon}{\tau - \bar{S} - \varepsilon}. \tag{28}$$

Proof. The EMA has a nonnegative kernel of unit mass and starts at the first value, so $\sup |\tilde{S}_N - \tilde{S}| \le \varepsilon$. On $[u_f, u^*]$ we have $\tilde{S} \ge \tau$ and $S \le \bar{S}$, hence $\gamma\,\mathrm{d}\tilde{S}/\mathrm{d}u = S - \tilde{S} \le -(\tau - \bar{S})$, and since $\tilde{S}(u^*) = \tau$ by continuity, $\tilde{S}(u^* - r) \ge \tau + r(\tau - \bar{S})/\gamma$ for $0 \le r \le u^* - u_f$. With $r_0 = \gamma\varepsilon/(\tau - \bar{S})$, this gives $\tilde{S} \ge \tau + \varepsilon$ on $[u_f, u^* - r_0]$, and the assumption gives it on $[u_0, u_f]$. So $\tilde{S}_N \ge \tau$ on $(u_0, u^* - r_0]$ and $u_N^* \ge u^* - r_0$. After u^*, as long as $\tilde{S} \ge \tau - \varepsilon$, we have $\gamma\,\mathrm{d}\tilde{S}/\mathrm{d}u \le -(\tau - \varepsilon - \bar{S})$, so $\tilde{S}(u^* + r) < \tau - \varepsilon$ for $r > \gamma\varepsilon/(\tau - \bar{S} - \varepsilon)$, and it stays there because $S \le \bar{S} < \tau - \varepsilon$. There $\tilde{S}_N < \tau$, which gives the upper bound. □

With a time constant of $0.04N$, the value of ℓ_2 on ResNet-50, $\bar{S} = 0.005$ and $\varepsilon = 0.005$, the stop moves by at most 0.5% of the budget, about half an epoch. The bound holds for one time constant.

The smoother of CFS switches its time constant at 3τ, so two paths that start δ apart can separate by more than δ near the switch, and the proof does not transfer directly to the rule of Algorithm 1.

B.9 Proof of Proposition 2

Part (i). Write $\tilde{g}_i = P_t^{1/2} g_i$ for the preconditioned scores, with $\tilde{g}_i = g_i$ for SGD, so that $\mathcal{C}_t = \frac{1}{B_p}\sum_i \tilde{g}_i \tilde{g}_i^\top$. This matrix has rank at most B_p, so it has at most $\min\{B_p, d\}$ nonzero eigenvalues $\lambda_1(\mathcal{C}_t) \geq \lambda_2(\mathcal{C}_t) \geq \cdots \geq 0$, and they sum to $\operatorname{tr}\mathcal{C}_t$. The largest is at least their mean,

$$\lambda_{\max}(\mathcal{C}_t) = \lambda_1(\mathcal{C}_t) \;\geq\; \frac{1}{\min\{B_p, d\}} \sum_k \lambda_k(\mathcal{C}_t) \;=\; \frac{\operatorname{tr}\mathcal{C}_t}{\min\{B_p, d\}} \;\geq\; \frac{\operatorname{tr}\mathcal{C}_t}{B_p} \tag{29}$$

for every Fisher batch. Taking expectations gives $\mathbb{E}\,\lambda_{\max}(\mathcal{C}_t) \geq \operatorname{tr}(\bar{\mathcal{C}}_t)/B_p$. The top eigenvalue is convex on symmetric matrices, as the supremum of the linear functions $A \mapsto w^\top A w$ over unit vectors w, so Jensen's inequality gives $\mathbb{E}\,\lambda_{\max}(\mathcal{C}_t) \geq \lambda_{\max}(\bar{\mathcal{C}}_t)$, the second bound of part (i).

Part (ii). For a batch of b pairs write $\mathcal{C}^{(b)} = \frac{1}{b}\sum_{i=1}^{b} \tilde{g}_i \tilde{g}_i^\top$ and, for $j \leq b+1$, let $\mathcal{C}^{(b)}_{-j}$ be the average over the b indices of $\{1, \ldots, b+1\} \setminus \{j\}$. Each index appears in b of these $b+1$ averages, so

$$\mathcal{C}^{(b+1)} = \frac{1}{b+1} \sum_{j=1}^{b+1} \mathcal{C}^{(b)}_{-j}. \tag{30}$$

By convexity of $\lambda_{\max}$, $\lambda_{\max}(\mathcal{C}^{(b+1)}) \leq \frac{1}{b+1}\sum_j \lambda_{\max}(\mathcal{C}^{(b)}_{-j})$, and every $\mathcal{C}^{(b)}_{-j}$ has the law of $\mathcal{C}^{(b)}$ by exchangeability. Taking expectations gives $\mathbb{E}\,\lambda_{\max}(\mathcal{C}^{(b+1)}) \leq \mathbb{E}\,\lambda_{\max}(\mathcal{C}^{(b)})$, which is part (ii).

An upper bound. If in addition the pairs are independent and $\|\tilde{g}_i\|_2^2 \leq \nu$ for every i, then

$$\mathbb{E}\,\lambda_{\max}(\mathcal{C}_t) \;\leq\; \lambda_{\max}(\bar{\mathcal{C}}_t) + \sqrt{\frac{2\nu\,\lambda_{\max}(\bar{\mathcal{C}}_t)\ln d}{B_p}} + \frac{\nu \ln d}{3B_p}. \tag{31}$$

To prove it, write $\mathcal{C}_t - \bar{\mathcal{C}}_t = \sum_{i=1}^{B_p} X_i$ with $X_i = (\tilde{g}_i\tilde{g}_i^\top - \bar{\mathcal{C}}_t)/B_p$, which are independent, symmetric and of mean zero. Since $-\bar{\mathcal{C}}_t \preceq 0$, Weyl's inequality gives $\lambda_{\max}(X_i) \leq \|\tilde{g}_i\|_2^2/B_p \leq \nu/B_p$. For the variance,

$$\mathbb{E}X_i^2 = \frac{1}{B_p^2}\Big(\mathbb{E}\big[\|\tilde{g}_i\|_2^2\,\tilde{g}_i\tilde{g}_i^\top\big] - \bar{\mathcal{C}}_t^2\Big) \;\preceq\; \frac{1}{B_p^2}\,\mathbb{E}\big[\|\tilde{g}_i\|_2^2\,\tilde{g}_i\tilde{g}_i^\top\big] \;\preceq\; \frac{\nu}{B_p^2}\,\bar{\mathcal{C}}_t,$$

$$\Omega := \Big\|\sum_{i=1}^{B_p} \mathbb{E}X_i^2\Big\| \;\leq\; \frac{\nu\,\lambda_{\max}(\bar{\mathcal{C}}_t)}{B_p}. \tag{32}$$

The matrix Bernstein inequality for a sum of independent, zero-mean, symmetric $d \times d$ matrices with top eigenvalues bounded by R (Tropp, 2015, Ch. 6) states $\mathbb{E}\,\lambda_{\max}(\sum_i X_i) \leq \sqrt{2\Omega \ln d} + \frac{1}{3}R\ln d$. With $R = \nu/B_p$ and Weyl's inequality once more,

$$\mathbb{E}\,\lambda_{\max}(\mathcal{C}_t) \;\leq\; \lambda_{\max}(\bar{\mathcal{C}}_t) + \mathbb{E}\,\lambda_{\max}(\mathcal{C}_t - \bar{\mathcal{C}}_t) \;\leq\; \lambda_{\max}(\bar{\mathcal{C}}_t) + \sqrt{\frac{2\nu\,\lambda_{\max}(\bar{\mathcal{C}}_t)\ln d}{B_p}} + \frac{\nu\ln d}{3B_p}. \tag{33}$$

The argument uses only the form $\mathcal{C}_t = \frac{1}{B_p}\sum_i \tilde{g}_i\tilde{g}_i^\top$, so it holds for SGD and for AdamW alike. □

Remarks. Since the labels are sampled from the model, $\mathbb{E}\,g_i = 0$, and $\mathcal{C}_t$ is the sample covariance of the scores. The lower bound of part (i) and the upper bound (31) differ by at least a factor $\ln d$ in the batch size they call for, because $\nu \geq \mathbb{E}\|\tilde{g}_i\|_2^2 = \operatorname{tr}\bar{\mathcal{C}}_t$ gives $\nu\ln d/\lambda_{\max}(\bar{\mathcal{C}}_t) \geq \rho_t \ln d$. The intrinsic-dimension form of the matrix Bernstein inequality (Tropp, 2015, Ch. 7) replaces $\ln d$, up to constants, by the logarithm of the intrinsic dimension of the variance proxy $(\nu/B_p)\bar{\mathcal{C}}_t$, which equals ρ_t. Both sides are then governed by the effective rank, as for sample covariance matrices (Koltchinskii & Lounici, 2017). The upper bound needs independent scores, which fails for SimCLR, whose views share candidates, and for language models, whose sampled tokens come from shared sequences. Part (i) still holds with $2B_p$ or the number of sampled tokens in place of B_p. A floor $\underline{S}$ of the statistic after the crossing enters the lower bound of (8), which diverges as $\underline{S} \to \tau$, so a batch near the minimum size delays the stop by a large share of the budget, as Table 15 also shows.

B.10 Smoothing of Estimation Noise

Proposition 4 (Smoothing of estimation noise). *Let the accepted values be $s_j = s_j^\circ + \xi_j$ with noise-free values s_j°, and let $\mathcal{F}_{j-1}$ be the training run together with the estimates before check j. Assume that $\mathbb{E}[\xi_j \mid \mathcal{F}_{j-1}] = 0$, $\mathbb{E}[\xi_j^2 \mid \mathcal{F}_{j-1}] \le \sigma^2$ and $\mathbb{E}[e^{\zeta \xi_j} \mid \mathcal{F}_{j-1}] \le e^{\zeta^2 \sigma_\psi^2/2}$ for all real ζ. Let $\tilde{s}_j^\circ$ be the EMA (5) of the s_j° and $n_j = \tilde{s}_j - \tilde{s}_j^\circ$. Then*

$$\mathrm{Var} n_j \le \sigma^2 v_j, \qquad \mathbb{E}\, e^{\zeta n_j} \le e^{\zeta^2 \sigma_\psi^2 v_j/2}, \qquad v_j = \frac{1-\beta+2\beta^{2j-1}}{1+\beta} \;\downarrow\; \frac{1-\beta}{1+\beta} = \tanh\frac{\Delta}{2\ell}, \tag{34}$$

with $\Delta/\ell = -\ln\beta$ in every interval. At a check with $\tilde{s}_j^\circ \ge \tau + \delta$ for a margin $\delta > 0$, the probability that $\tilde{s}_j < \tau$ is at most $\exp\big(-\delta^2/(2\sigma_\psi^2 v_j)\big)$, and over a set $\mathcal{J}$ of checks with margins $\delta_j > 0$ it is at most $\sum_{j\in\mathcal{J}} \exp\big(-\delta_j^2/(2\sigma_\psi^2 v_j)\big)$.

Proof. Since $\tilde{s}_1 = s_1$ and $n_j = \beta n_{j-1} + (1-\beta)\xi_j$,

$$n_j = \beta^{j-1}\xi_1 + (1-\beta)\sum_{i=2}^{j} \beta^{j-i}\xi_i,$$

$$\beta^{2(j-1)} + (1-\beta)^2 \sum_{i=2}^{j} \beta^{2(j-i)} = \beta^{2(j-1)} + \frac{(1-\beta)\big(1-\beta^{2(j-1)}\big)}{1+\beta} = \frac{1-\beta+2\beta^{2j-1}}{1+\beta} = v_j,$$

$$\frac{1-\beta}{1+\beta} = \frac{1-e^{-\Delta/\ell}}{1+e^{-\Delta/\ell}} = \frac{e^{\Delta/2\ell}-e^{-\Delta/2\ell}}{e^{\Delta/2\ell}+e^{-\Delta/2\ell}} = \tanh\frac{\Delta}{2\ell}. \tag{35}$$

The weights w_i of the first line are fixed and their squares sum to v_j. The ξ_i are martingale differences, so the cross terms of $\mathbb{E}\, n_j^2$ vanish and $\mathrm{Var} n_j \le \sigma^2 v_j$, and conditioning on $\mathcal{F}_{j-1}, \mathcal{F}_{j-2}, \dots$ in turn gives $\mathbb{E}\, e^{\zeta n_j} \le e^{\zeta^2 \sigma_\psi^2 \sum_i w_i^2/2}$. The value falls below τ at a check with margin δ only if $n_j < -\delta$, so the Chernoff bound gives the first probability and the union bound the second. □

At $\beta = 0.9921$ on ResNet-50 the stationary factor is 0.004, so the EMA keeps 0.4% of the variance and 6.3% of the standard deviation of the estimation noise. The factor v_j starts at one and comes within 1% of its stationary value only after about 640 checks, near the stop on ResNet-50, so early in the run the smoothed statistic is much noisier, which the start condition covers. The factor per check is the same in every interval, because β acts per accepted check. Three assumptions are idealized. Under $\|\tilde{g}_i\|_2^2 \le \nu$ the Rayleigh quotient of a fixed unit vector is an average of B_p independent terms in $[0, \nu]$, hence sub-Gaussian with proxy $\nu^2/(4B_p)$ by Hoeffding's lemma, which gives $\sigma_\psi^2 = (\eta_t^{\mathrm{eff}}/2)^2 \nu^2/(4B_p)$ for the statistic, but the estimated direction depends on the batch. Warm starts also couple consecutive estimates through the starting vector. The SNR test selects on the noise (Section B.4), which biases the accepted values slightly upward. The next result combines Proposition 4 with the delay of the noise-free rule to bound how far the noise can move the stop.

Corollary 4 (Stop under estimation noise). *Index the accepted checks by j, let j° be the first check with $\tilde{s}_j^\circ < \tau$, and let $0 < \delta < \tau - \bar{S}$. For an index j_1, assume that $\tilde{s}_j^\circ \ge 3\tau + \delta$ at some check $j < j_1$, that $\tilde{s}_j^\circ \ge \tau + \delta$ for all $j < j_1$ and that $s_j^\circ \le \bar{S} < \tau$ for $j \ge j_1$. On the event $\{\max_{j \le j_{\max}} |n_j| \le \delta\}$, whose probability is at least $1 - 2\sum_{j \le j_{\max}} \exp\big(-\delta^2/(2\sigma_\psi^2 v_j)\big)$, the stop j_{stop} of the rule with patience K satisfies*

$$j^\circ - \frac{\delta}{(1-\beta)(\tau-\bar{S})} - 1 \;\le\; j_{\mathrm{stop}} \;\le\; j^\circ + \frac{\delta}{(1-\beta)(\tau-\bar{S}-\delta)} + K, \tag{36}$$

whenever $j_{\max}$ exceeds the right-hand side.

Proof. On the event, the start condition holds before check j_1. For $j_1 \le j < j^\circ$ we have $\tilde{s}_j^\circ \ge \tau$ and $s_{j+1}^\circ \le \bar{S}$, so $\tilde{s}_j^\circ - \tilde{s}_{j+1}^\circ = (1-\beta)(\tilde{s}_j^\circ - s_{j+1}^\circ) \ge (1-\beta)(\tau - \bar{S})$. Counting back from $j^\circ - 1$, $\tilde{s}_j^\circ \ge \tau + \delta$ for $j \le j^\circ - 1 - \delta/((1-\beta)(\tau-\bar{S}))$, and by assumption for $j < j_1$. At these checks $\tilde{s}_j \ge \tilde{s}_j^\circ - \delta \ge \tau$, so none of them lies below τ, which gives the lower bound. For $j \ge j^\circ$, $\tilde{s}_{j+1}^\circ - \bar{S} \le \beta(\tilde{s}_j^\circ - \bar{S})$, so $\tilde{s}_j^\circ - \bar{S} < \beta^{j-j^\circ}(\tau - \bar{S})$. This is at most $\tau - \bar{S} - \delta$ once $j - j^\circ \ge \ln\frac{\tau-\bar{S}}{\tau-\bar{S}-\delta}\big/(-\ln\beta)$, which

holds for $j - j^\circ \geq \delta/((1-\beta)(\tau - \bar{S} - \delta))$ because $\ln \frac{a}{a-\delta} \leq \frac{\delta}{a-\delta}$ and $-\ln \beta \geq 1 - \beta$. From that check on, $\tilde{s}_j \leq \tilde{s}_j^\circ + \delta < \tau$ at every check, and the K-th consecutive value below τ follows within K checks. □

In updates, one check near the threshold spans Δ_2 before the first value below τ and Δ_3 after it, and $\Delta/(1-\beta)$ is close to ℓ, so the spread of the stop is about $\ell_2\delta/(\tau - \bar{S})$ updates. This is why new draws of the Fisher batches move the stop, and the metric, so little in every regime (Table 14).

B.11 Power Iteration with Warm Starts

Let $\mathcal{C}$ have eigenpairs (λ_k, w_k) with $\lambda_1 > \lambda_2 \geq \cdots \geq 0$, and start from $v^{(0)} = \sum_k b_k w_k$ with $b_1 \neq 0$. Then

$$\mathcal{C}^n v^{(0)} = \sum_k b_k \lambda_k^n w_k = \lambda_1^n \Big(b_1 w_1 + \sum_{k \geq 2} b_k \Big(\frac{\lambda_k}{\lambda_1} \Big)^n w_k \Big),$$
$$\tan \psi_n \leq \Big(\frac{\lambda_2}{\lambda_1} \Big)^n \tan \psi_0, \qquad 0 \leq \lambda_1 - v^\top \mathcal{C} v \leq \lambda_1 \sin^2 \psi, \tag{37}$$

where ψ_n is the angle between the normalized iterate and w_1, and the last inequality holds for any unit vector v at angle ψ from w_1. A relative accuracy ϵ_{rel} is thus guaranteed after $n \geq \ln\big(\tan \psi_0 / \sqrt{\epsilon_{\mathrm{rel}}}\big) / \ln(\lambda_1/\lambda_2)$ steps. From a random start $\tan \psi_0 \approx \sqrt{d}$, since the coordinate of a random unit vector on w_1 is of order $1/\sqrt{d}$, which gives about eleven steps for ResNet-50 at $\epsilon_{\mathrm{rel}} = 10^{-3}$ and $\lambda_2/\lambda_1 = 0.32$. From the eigenvector of the previous check at update t', $\sin \psi_0 \leq 2\|\mathcal{C}_t - \mathcal{C}_{t'}\|_2/(\lambda_1 - \lambda_2)$ (Yu et al., 2015), which is small when the change of the matrix between the two checks, the drift of the parameters together with the sampling noise of the two Fisher batches, is small compared with the eigengap. The ratio λ_2/λ_1 is about 0.3 on our models (Table 7). Averaged over the phases of a run, a cold start takes 10.5 to 15.2 steps and a warm start 1.8 to 3.2, and the mean per check falls from 12.8 to 2.5, about a fivefold saving of Fisher-vector products.

Table 7: **Power iteration.** Left: ratio of the top two eigenvalues of the monitored matrix at one checkpoint per model, and the steps to a relative accuracy of 10^{-3} there from a cold and a warm start. Right: mean steps per check in each phase of a run and their ratio, the saving of the warm start.

Model	λ_2/λ_1	Cold / warm	Phase of training	Cold	Warm	Ratio
ResNet-50, ImageNet	0.32	8 / 2	early (0% to 30%)	15.2	3.2	4.8×
ViT-B, ImageNet	0.28	7 / 2	middle (30% to 70%)	12.8	2.5	5.1×
BERT-B, GLUE	0.33	8 / 2	late (70% to 100%)	10.5	1.8	5.8×
			average	12.8	2.5	5.1×

B.12 Negative Curvature

The Fisher and the Gauss-Newton matrix are positive semidefinite. If the Hessian is used instead, the Rayleigh quotient can be negative, and power iteration converges to the eigenvalue of largest magnitude, which may be the most negative one. Algorithm 1 rejects every check whose Rayleigh quotient is not positive, which keeps s_t nonnegative and leaves the EMA unchanged. Table 8 gives small examples. For an indefinite matrix with a dominant negative eigenvalue, power iteration returns -5, and without the check the statistic would be -0.25. It also rejects numerical noise near zero.

Table 8: **The positivity check** on one positive semidefinite matrix and three that are not ($\eta^{\mathrm{eff}} = 0.1$). Every check whose Rayleigh quotient is not positive is rejected (✗), so the statistic stays nonnegative.

Matrix	$\lambda_{\min}$	Largest eigenvalue	Rayleigh quotient	Spectral radius	Accepted	s without check	s with check
positive semidefinite	0.12	1.7	1.7	1.7	✓	0.085	0.085
indefinite	-5	2	-5	5	✗	-0.25	✗
negative definite	-3	-0.2	-3	3	✗	-0.15	✗
near zero	-10^{-12}	10^{-12}	-4×10^{-13}	10^{-12}	✗	-2×10^{-14}	✗

B.13 A Local Quadratic Model of the Statistic

In the local quadratic model behind the stability threshold, s_t has a direct reading. Let $e_t = \theta_t - \theta^\star$ near a reference point, $\mathcal{L}(\theta^\star + e) \approx \mathcal{L}(\theta^\star) + \frac{1}{2} e^\top H e$ with $H \succeq 0$, and consider the stochastic update $e_{t+1} = (I - \eta^{\text{eff}} H) e_t - \eta^{\text{eff}} \xi_t$, where the noise ξ_t has mean zero and covariance Σ and is independent of e_t and across updates. In the eigenbasis $H = U \operatorname{diag}(\lambda_k) U^\top$ each mode obeys $z_{t+1}^{(k)} = (1 - 2s^{(k)}) z_t^{(k)} - \eta^{\text{eff}} \tilde{\xi}_t^{(k)}$ with $s^{(k)} = \eta^{\text{eff}} \lambda_k / 2$. For the covariance $V_t = \mathbb{E}[e_t e_t^\top]$ and $A = I - \eta^{\text{eff}} H$,

$$V_{t+1} = A V_t A^\top + (\eta^{\text{eff}})^2 \Sigma, \qquad V_{t+n} = A^n V_t (A^\top)^n + (\eta^{\text{eff}})^2 \sum_{j=0}^{n-1} A^j \Sigma (A^\top)^j,$$

$$V_{t+1}^{(k)} = \big(1 - 2s^{(k)}\big)^2 V_t^{(k)} + (\eta^{\text{eff}})^2 \Sigma^{(k)}, \qquad V_\infty^{(k)} = \frac{(\eta^{\text{eff}})^2 \Sigma^{(k)}}{1 - (1 - 2s^{(k)})^2} = \frac{(\eta^{\text{eff}})^2 \Sigma^{(k)}}{4 s^{(k)} (1 - s^{(k)})}, \quad (38)$$

so the expected quadratic energy $\mathcal{E}_t = \frac{1}{2} \sum_k \lambda_k V_t^{(k)}$ approaches its stationary value $\mathcal{E}_\infty = \sum_{k:\lambda_k > 0} \eta^{\text{eff}}\, \Sigma^{(k)} / (4(1 - s^{(k)}))$ at the rate $(1 - 2s^{(k)})^2$ per mode, the stationary covariance of SGD on a quadratic (Mandt et al., 2017), since modes with $\lambda_k = 0$ carry no energy. At $s = \tau = 0.05$ the top mode of gradient descent with step η^{eff} still contracts by $(1 - 0.1)^2 = 0.81$ per update. With heavy-ball momentum $\mu = 0.9$ this mode is underdamped, and its energy contracts by $\mu = 0.9$ per update. Both roots of $r^2 - (1 + \mu - \eta\lambda) r + \mu$ then have modulus $\sqrt{\mu}$, since $(1 - \sqrt{\mu})^2 < \eta\lambda = 2s(1 - \mu) = 0.01 < (1 + \sqrt{\mu})^2$. In both cases optimization still makes progress at the threshold. There the step is twenty times below the stability limit of plain gradient descent, and under our schedules the statistic then stays at its floor for the rest of the run. The stop then follows from the delay of Proposition 1 and the measured lead of Section 3.2, not from the quadratic model.

C Experimental Details

C.1 Datasets, Models and Hardware

Table 9 lists the seven regimes and the calibrated protocol. Batch sizes follow the standard recipes: 256 for ResNet-50, 1024 for ViT-B, 4096 for MAE, 32 for BERT-B, 128 for CIFAR-10 and 512 for SimCLR. The learning rate follows a cosine schedule to zero in all regimes except BERT-B, which decays linearly. ImageNet-1K has 1.28M training images. For the seven main regimes, its 50K validation images are split once into a 10K development set, used for all selection, and a 40K test split, used for all reported numbers. CIFAR-10 uses 45K training images, 5K development images and the 10K test set. SST-2 has 67K training sentences and the 872-sentence development set. WikiText-103 has 103M training tokens, 217K validation tokens and 245K test tokens, and its Fisher batch holds 1K sampled tokens. ImageNet, MAE, SimCLR and WikiText-103 run on eight A100 GPUs with 80GB, and CIFAR-10 and SST-2 on one A100 with 40GB, with PyTorch 2.1 and CUDA 12.1.

Table 9: **Training regimes and the calibrated protocol.** Budget in epochs and updates per epoch; initial learning rate η_0 (effective rate η_0^{eff} for SGD); warmup in epochs; check intervals in updates; β per check; time constant ℓ_2 of the middle interval (updates, % of budget); Fisher batch size; cost of one check in update equivalents. $K = 2$ on SimCLR, 3 elsewhere; $\tau = 0.05$ and $\kappa = 10$ everywhere.

Regime	Model	Optimizer	Epochs	Updates/epoch	η_0 (η_0^{eff})	Warmup	$(\Delta_1, \Delta_2, \Delta_3)$	β	ℓ_2 (updates, %)	B_p/B	Cost/check
ImageNet	ResNet-50	SGD, μ=0.9	90	5000	0.1 (1.0)	0	(600, 150, 44)	0.9921	18 900 (4.2)	4	12
ImageNet	ViT-B/16	AdamW	300	1250	10^{-3}	5	(250, 62, 19)	0.9961	15 900 (4.2)	4	12
ImageNet, MAE	ViT-B/16	AdamW	1600	313	1.5×10^{-4}	40	(1804, 448, 136)	0.9811	23 500 (4.7)	4	12
SST-2	BERT-B	AdamW	3	2105	2×10^{-5}	0	(54, 16, 6)	0.933	231 (3.7)	0.5	1.5
CIFAR-10, 40%	ResNet-18	SGD, μ=0.9	200	352	0.1 (1.0)	0	(188, 48, 16)	0.9842	3 010 (4.3)	4	12
ImageNet, SimCLR	ResNet-50	SGD, μ=0.9	800	2500	0.3 (3.0)	10	(1733, 433, 130)	0.9961	111 000 (5.5)	2	6
WikiText-103	Transformer	AdamW	200	1000	3×10^{-4}	2	(718, 178, 54)	0.981	9 300 (4.6)	2	6

C.2 Baselines

Validation ES starts at the first scheduled evaluation after warmup, stops at the first evaluation that does not improve on the best value so far and returns that checkpoint, which is patience one. It

evaluates once per epoch for ResNet-50, ViT-B and CIFAR-10, every 0.1 epoch for BERT-B and every 1000 updates for WikiText-103. For MAE it evaluates a linear probe on 5K development images every 50 epochs and waits for five evaluations without improvement. The last improvement comes at epoch 1100 in two seeds and 1150 in three, so it stops at epoch 1350 or 1400. Table 10 shows the development loss of ResNet-50 around its minimum. The loss improves up to epoch 64 in every seed, so validation ES stops at epoch 65 and returns its checkpoint, 0.13 points below the best.

Table 10: **Validation ES on ResNet-50.** Development loss on the 10K set near its minimum (in bold), and top-1 on the 40K test split at the best checkpoint (epoch 64) and at the returned one (epoch 65).

	Development loss at epoch				Top-1 at epoch	
Seed	62	63	64	65	64 (best)	65 (returned)
42	1.3835	1.3802	**1.3758**	1.4020	77.19	77.14
43	1.3435	1.3400	**1.3332**	1.3570	77.02	76.88
44	1.4200	1.4158	**1.4095**	1.4470	77.26	77.07
45	1.3660	1.3615	**1.3567**	1.3810	77.09	76.95
46	1.3991	1.3932	**1.3883**	1.4190	77.14	77.01
Mean					77.14	77.01

The loss plateau rule stops when the training loss changes by less than 10^{-4} over ten consecutive epochs, and the gradient-norm rule stops when $\|\nabla \mathcal{L}\|_2 < 10^{-3}$. We implement Label Wave (Yuan et al., 2024) with a threshold on the prediction changes, and it stops when the rate of prediction changes on the training set over a window of five epochs falls below 0.01. On BERT-B, whose budget is three epochs, these windows count evaluations made every 0.1 epoch. IES (Yuan et al., 2025) removes an example when its second-order loss difference falls below 10^{-3} on ImageNet and 10^{-2} on CIFAR-10, and trains on the rest for the whole budget. Its saving is the ratio of total to active examples, $1.31\times$ on ResNet-50. On ViT-B and BERT-B it removes no examples and equals the full run. The oracle picks the best saved checkpoint of each run. For ResNet-50 we give a lower bound on the oracle, 77.24, the mean over seeds of the better of the checkpoints at epoch 64 and the CFS stop.

C.3 Calibration

For each regime we compare 35 to 48 candidates. They combine four or five interval settings between one half and twice a base interval, fractions c between 0.006 and 0.019, a patience $K \in \{1, \dots, 5\}$ and a Fisher batch of 1, 2, 4 or 8 times a reference batch, which is the training batch and a quarter of it on SST-2. Each candidate is run on the statistic logged in the five training runs, with 20 draws of the Fisher batches per run, and scored by

$$J = \overline{\text{cost}} + 0.10\,\frac{\overline{\text{gap}}}{\max(\sigma_{\text{ref}}, 0.1)} + 0.05\,\frac{\overline{\text{gap}}_{\text{worst }20\%}}{\max(\sigma_{\text{ref}}, 0.1)}, \tag{39}$$

where the cost is the total cost ratio with monitoring, the gap is the distance to the development-set oracle of the same run, and σ_{ref} is a scale of the gap for each regime (0.1, 0.1, 0.2, 0.3, 0.4, 0.2 and 0.1 points in the order of Table 9). With these weights, a gap of one reference scale weighs as much as 10% of the budget. The minimizer of J is the protocol of Table 9. Its fraction c follows from $c = \Delta_3/(-N \ln \beta)$ and is 0.0123, 0.0130, 0.0142, 0.0137, 0.0143, 0.0166 and 0.0141 in the same order. The held-out runs use $c = 0.012$, close to the smallest of these values. Calibration uses only the 10K ImageNet development set, the CIFAR-10 development images, the WikiText-103 validation tokens and the SST-2 development set, on which the SST-2 results are also reported in Table 1.

The experiments beyond the seven calibrated regimes, namely the other architectures, the other GLUE tasks, the noise rates, Clothing1M, detection, CelebA, transfer and the plug-ins, use a reference protocol without calibration. It keeps $\tau = 0.05$, $\kappa = 10$ and $K = 3$. Runs of at least 5×10^4 updates use $\beta = 0.99$, a Fisher batch equal to the training batch and intervals $\Delta_z = 2\lfloor \min(b_z, N/r_z) \rfloor$ with $b = (2000, 500, 100)$ and $r = (1500, 6000, 20000)$, which gives $(600, 150, 44)$ on 90-epoch ImageNet runs. Shorter runs use $\beta = 0.95$, a quarter of the training batch and $r = (600, 2400, 10000)$, with every interval at least one update. The caps N/r_z bind for budgets below 2×10^6 updates, which covers every run of this protocol, so the intervals are fixed fractions of the budget up to rounding, and the time constant $\Delta/(-\ln \beta)$ is proportional to the budget, as part (i) of Theorem 1 requires.

D RESULTS ON THE MAIN REGIMES

D.1 PER-SEED RESULTS

Table 11 gives the crossing, the stop and the statistic at the stop for every seed of the seven regimes. The stop is at the granularity of updates, and we evaluate the model at that update. The smoothed value at the stop lies between 0.044 and 0.050 in all 35 runs, just below τ, while the raw value lies between 0.003 and 0.028, well below τ, so the raw statistic has already reached its floor by the stop.

Table 11: **Per-seed results** of the calibrated protocol. Crossing: the first accepted check with $s_t < \tau$. Stop: first check with K consecutive $\tilde{s} < \tau$. Then come the raw and smoothed statistic and the SNR at the stop, the checks up to the stop, the metric at the stop and the optimum, which is the minimum of the development loss for ResNet-50 and the oracle checkpoint elsewhere. Crossings, stops and optima are in epochs (for SST-2 out of three epochs, for WikiText-103 in thousands of updates).

Regime	Seed	Crossing	Stop	s at stop	$\tilde{s}$ at stop	SNR	Checks	Metric	Optimum
ResNet-50	42	52.80	63.47	0.0032	0.0492	246	641	77.09	64
	43	54.84	64.85	0.0033	0.0491	154	654	77.14	64
	44	53.28	63.55	0.0052	0.0492	89	647	77.08	64
	45	50.16	61.81	0.0039	0.0491	9×10^4	628	77.27	64
	46	51.00	62.38	0.0035	0.0491	50	632	77.32	64
ViT-B	42	172.4	207.6	0.0043	0.0495	87	1246	82.28	206.8
	43	175.4	209.5	0.0032	0.0495	1423	1250	82.28	208.1
	44	171.8	206.3	0.0030	0.0496	17	1234	82.09	207.3
	45	182.6	215.4	0.0034	0.0495	9502	1282	82.07	214.5
	46	171.6	207.3	0.0054	0.0495	1.8×10^4	1252	82.27	206.6
MAE	42	951	1107	0.0049	0.0477	168	237	68.16	1122
	43	934	1114	0.0053	0.0478	4.8×10^4	239	68.46	1125
	44	922	1096	0.0044	0.0476	5.1×10^4	235	68.25	1104
	45	922	1101	0.0057	0.0481	34	236	68.57	1102
	46	870	1060	0.0053	0.0483	21	228	68.16	1051
BERT-B	42	2.12	2.29	0.0275	0.0466	14	133	92.78	2.30
	43	2.06	2.18	0.0180	0.0446	135	126	93.23	2.21
	44	2.14	2.26	0.0250	0.0470	272	126	93.12	2.27
	45	2.12	2.28	0.0239	0.0453	40	131	92.89	2.28
	46	2.15	2.27	0.0237	0.0464	1366	135	93.00	2.31
CIFAR-10, 40%	42	42.2	74.8	0.0096	0.0485	75	200	82.31	77.9
	43	47.5	74.7	0.0104	0.0487	129	202	81.94	72.8
	44	46.5	76.3	0.0088	0.0485	9661	203	81.80	74.8
	45	44.9	77.4	0.0081	0.0482	4301	205	81.82	75.9
	46	43.3	72.5	0.0103	0.0483	1662	198	82.01	72.9
SimCLR	42	547	631	0.0135	0.0497	87	1217	69.77	644
	43	533	624	0.0106	0.0498	306	1197	69.55	623
	44	523	619	0.0077	0.0498	928	1157	69.18	622
	45	516	604	0.0078	0.0498	101	1132	69.52	600
	46	528	618	0.0054	0.0497	117	1149	69.64	614
WikiText-103	42	144.2	158.3	0.0193	0.0485	112	310	24.577	156.4
	43	144.5	159.9	0.0280	0.0488	323	331	24.728	159.1
	44	141.9	156.4	0.0221	0.0485	24	317	24.827	156.4
	45	144.5	158.9	0.0223	0.0488	43	322	24.655	158.1
	46	145.6	160.8	0.0266	0.0489	31	330	24.643	159.8

D.2 CURVATURE MATRICES AND ADAMW

Figure 3a compares the top eigenvalues of the three curvature matrices on ResNet-18 trained on CIFAR-10, computed by Lanczos at four epochs with 32 label draws per input for the Fisher. The Fisher stays within 1.2% of the Gauss-Newton matrix and exceeds the Hessian by 2% to 20%, as Lemma 1 leads us to expect. Figure 3b shows the preconditioned statistic of BERT-B with AdamW, which falls and levels off like the statistic of SGD in Figure 1b and stops seed 42 at epoch 2.29.

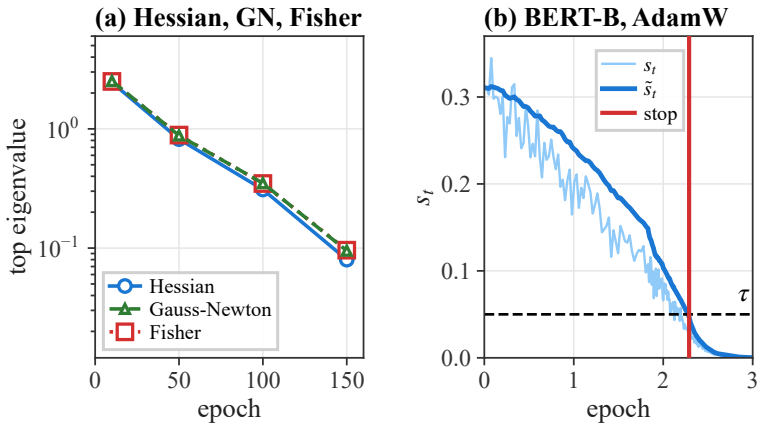

Figure 3: **Curvature matrices and AdamW.** (a) Top eigenvalues of the Hessian, the Gauss-Newton matrix and the sampled-label Fisher at four epochs (ResNet-18 on CIFAR-10, Lanczos, 32 label draws per input for the Fisher). (b) BERT-B on SST-2 with AdamW: s_t computed from $P_t^{1/2}\mathcal{F}_t P_t^{1/2}$, its EMA $\tilde{s}_t$, the threshold τ and the stop of seed 42 at epoch 2.29, close to its oracle checkpoint.

D.3 Accuracy around the Stop

Table 12 evaluates the five ResNet-50 runs at fixed epochs. The accuracy rises to 77.14 at epoch 64, dips to 75.80 at epoch 68 and recovers only to 76.84 at the end of the schedule. CFS stops at epoch 63.2 on average with 77.18, at the top of this curve, and its per-seed stops between epochs 61.8 and 64.9 do slightly better than any single fixed epoch. Label Wave stops at epoch 68, right in the dip.

Table 12: **ResNet-50 at fixed epochs** (mean top-1 of the five runs), with CFS and the oracle bound from Table 1. The epochs include 68 and 82, the stops of Label Wave and of the gradient-norm rule.

Epoch	30	45	50	55	60	61	62	63	64	65	68	82	88	90	CFS (63.2)	Oracle
Top-1	72.63	75.17	75.74	76.22	76.71	76.83	77.00	77.10	77.14	77.01	75.80	76.20	76.74	76.84	77.18	$\geq$77.24

D.4 Consistency across Seeds

Figure 4 shows the five ResNet-50 runs. The raw statistic of the five seeds follows the same path, with pairwise correlations between 0.985 and 0.992. The crossing moves up to 4.7 epochs between seeds and the stop moves with it, so the delay after the crossing stays between 10.0 and 11.7 epochs.

D.5 Monitored Quantities of One Run

Figure 5 collects the quantities monitored on one ImageNet run, seed 42 of ResNet-50. The development loss reaches its minimum at epoch 64 and validation ES stops at epoch 65. When CFS stops at epoch 63.5, the learning rate is at 20% of its peak and the top eigenvalue has fallen by a factor of more than 20. Of the 822 checks plotted in (f), two have SNR below $\kappa = 10$ and are rejected.

D.6 Self-Supervised Pretraining

Table 13 adds the downstream metrics that are not in Table 1. For MAE, the fine-tuning accuracy at the CFS stop matches that of the full run (83.59 against 83.58) and is 0.08 points below that at the stop of validation ES. For SimCLR, the k-NN accuracy at the stop is 60.79, close to 60.78 for the oracle and above 60.55 for the full run, while the contrastive loss falls from 4.22 at the stop to 4.12.

D.7 Stability under New Fisher Batches

Calibration uses 20 draws of the Fisher batches per run, so we run the chosen protocol with 40 new draws on the same five training runs (Table 14). The mean metric differs from the reported runs by at most 0.02 points, as Corollary 4 suggests, and the cost stays between 0.41 and 0.80 of a full run.

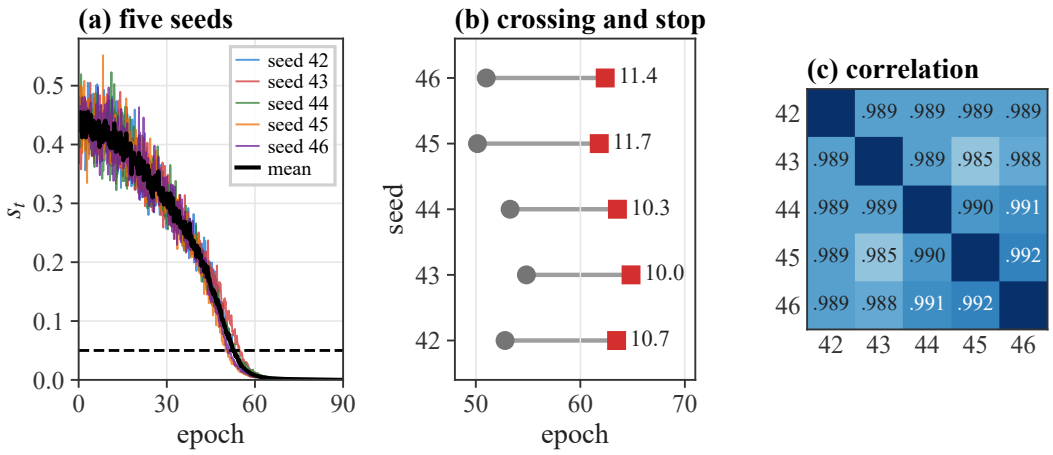

Figure 4: **Consistency across seeds on ResNet-50.** (a) Raw statistic s_t of the five seeds and their mean, with the threshold τ (dashed). (b) Crossing (gray circle) and stop (red square) of each seed, with the delay between them in epochs. (c) Pairwise Pearson correlation of the five raw trajectories.

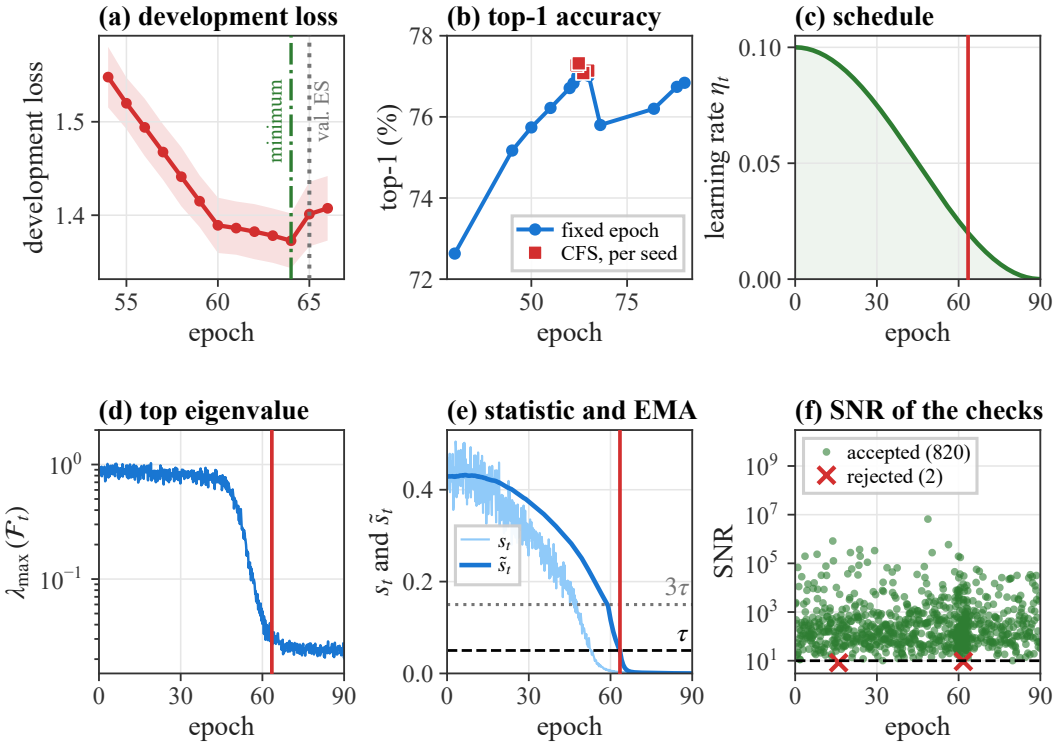

Figure 5: **One ImageNet run in detail** (ResNet-50, seed 42; five-seed means where marked). (a) Development loss over epochs 54 to 66 (mean and standard deviation); the minimum is at epoch 64 and validation ES stops at 65. (b) Top-1 accuracy at fixed epochs (mean) and at the CFS stop of each seed. (c) Cosine learning-rate schedule. (d) Top eigenvalue on a log scale. (e) s_t and $\tilde{s}_t$ with τ and the start level 3τ. (f) SNR of the 822 checks with the threshold κ (dashed), two of them rejected. Red lines in (c) to (e) mark the CFS stop of seed 42 at epoch 63.5, just before the loss minimum.

Table 13: **Self-supervised pretraining.** Left: MAE fine-tuning accuracy and reconstruction loss (MSE) at each stop. Right: SimCLR contrastive loss (NT-Xent) and k-NN accuracy at the stop of each rule. Stops in epochs; values are means over five seeds, with standard deviations for accuracies.

MAE	Stop	Fine-tune	MSE	SimCLR	Stop	NT-Xent	k-NN
full run	1600	83.58±0.04	0.352	full run	800	4.12	60.55±0.19
loss plateau	1520	83.43±0.05	0.353	oracle	621	4.22	60.78±0.18
validation ES	1380	83.67±0.07	0.356				
CFS	1096	83.59±0.05	0.366	CFS	619	4.22	60.79±0.17

Table 14: **CFS with 40 new draws of the Fisher batches** on the five runs: mean and standard deviation of the metric over the draws, total cost relative to the full run, and the reported metric.

	40 new draws		
Regime	Metric	Cost	Reported runs
ResNet-50	77.16±0.10	0.719	77.18
ViT-B	82.21±0.10	0.736	82.20
MAE	68.32±0.17	0.691	68.32
BERT-B	93.00±0.17	0.783	93.00
CIFAR-10, 40%	82.00±0.32	0.410	81.98
SimCLR	69.53±0.20	0.778	69.53
WikiText-103 (↓)	24.685±0.088	0.803	24.686

E Ablation Studies

This section gives the full version of Section 5.4, with the same order of conclusions as there.

E.1 Smoothing and the Time Constant

Table 15 varies one quantity at a time on ResNet-50 and Figure 6 plots the same values. Without smoothing the stop falls at the crossing. Factors up to $\beta = 0.95$ still stop almost at the crossing, since their time constants are short compared with the lead. From $\beta = 0.99$ on the stop moves later monotonically, and $\beta = 0.999$ never stops within the budget, which by Corollary 2(c) happens once $\ell \ln(\tilde{s}(t_c)/\tau)$ exceeds the remaining budget. The top-1 accuracy is highest at the calibrated $\beta = 0.9921$ and falls on both sides of it, by 0.5 points at $\beta = 0.99$ and by 1.3 points at $\beta = 0.995$.

Table 15: **One quantity at a time** on ResNet-50 (five seeds; default in bold): stop in epochs and top-1 accuracy; for the check intervals with β fixed, the number of checks and the monitoring time as a share of a run, not the stop. β and the intervals enter only through the time constants $\ell_z = \Delta_z/(-\ln\beta)$. For $\tau = 0.15$ three of the five seeds never reach the start level 3τ and run to the end, epoch 90.

Threshold τ			Factor β			SNR threshold κ			Patience K			Fisher batch B_p/B			Intervals, β fixed			
Value	Stop	Top-1	Value	Stop	Top-1	Value	Stop	Top-1	Value	Stop	Top-1	Value	Stop	Top-1	$(\Delta_1, \Delta_2, \Delta_3)$	Checks	Time	Top-1
0.02	77.4	76.01	0	52.5	75.98	0	63.1	77.17	1	63.2	77.18	0.25	78.1	76.03	(300, 75, 22)	1102	2.9%	76.29
0.03	71.1	75.83	0.9	52.8	76.00	5	63.1	77.17	2	63.2	77.18	0.5	71.4	75.82	(450, 112, 33)	785	2.1%	76.61
0.04	66.6	76.33	0.95	53.1	76.03	**10**	**63.2**	**77.18**	**3**	**63.2**	**77.18**	1	66.4	76.42	**(600, 150, 44)**	**640**	**1.7%**	**77.18**
0.05	**63.2**	**77.18**	0.99	59.9	76.68	15	63.4	77.13	4	63.2	77.18	2	63.7	77.11	(900, 225, 66)	508	1.4%	75.84
0.06	60.4	76.74	**0.9921**	**63.2**	**77.18**	20	63.8	77.10	5	63.2	77.18	**4**	**63.2**	**77.18**	(1200, 300, 88)	445	1.2%	76.17
0.08	56.0	76.30	0.995	73.5	75.88													
0.10	53.2	76.05	0.999	90	76.84													
0.15	73.4	76.36																

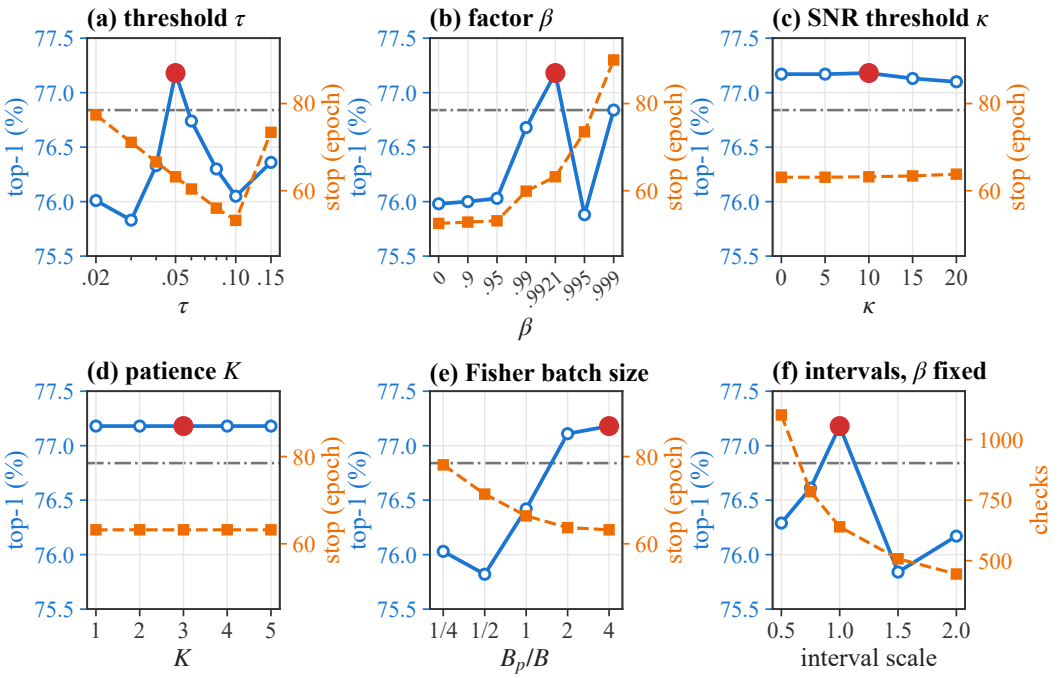

Figure 6: **One quantity at a time** on ResNet-50 (values of Table 15). Top-1 accuracy (blue, left axis) and stop epoch (orange, right axis; the number of checks in (f)) against the threshold τ, the factor β, the SNR threshold κ, the patience K, the Fisher batch size B_p/B and the interval scale at fixed β. The red marker is the default setting and the gray dash-dotted line marks the full run at 76.84.

The intervals matter only through the time constant. Table 16 runs the rule on the logged statistic of seed 42. With β fixed, halving or doubling every interval halves or doubles ℓ and moves the stop

to epoch 56.2 or 81.3. With ℓ fixed, obtained by adjusting β, the same change moves the stop by at most 0.1 epochs, as the continuous-time description predicts: the stop depends on the intervals only through ℓ_1 and ℓ_2, up to one interval (Corollary 3). Run on the logged values, the rule reproduces the stops of the training runs within 0.2 epochs, and the closed form (9) is a lower bound within 1.1 epochs of it. Halving the intervals at fixed ℓ costs more checks for no change of the stop, and the default of 640 checks takes 1.7% of the time of a run, against 2.9% at half intervals and fixed β.

Table 16: **The stop depends on the intervals only through the time constant** (the rule on the logged statistic of ResNet-50, seed 42). Time constants ℓ_1, ℓ_2 in epochs; $\tilde{s}(t_c)$ is the EMA at the crossing t_c (epoch 52.8, or 52.9 for 1.5× intervals); the closed form is $t_c + D$ from (9), or $t_c + \ell_2 \ln(\tilde{s}(t_c)/\tau)$ when $\tilde{s}(t_c) < 3\tau$. Discrete rule: run on the log; training run: the online stop, where recorded.

Setting	β	$(\Delta_1, \Delta_2, \Delta_3)$	ℓ_1	ℓ_2	$\tilde{s}(t_c)$	Closed form	Discrete rule	Training run
no smoothing	0	(600, 150, 44)	0	0	0.049	52.8	52.8	52.8
fast	0.95	(600, 150, 44)	2.3	0.58	0.060	52.9	53.4	53.3
	0.99	(600, 150, 44)	11.9	3.0	0.193	59.1	60.2	60.2
default	0.9921	(600, 150, 44)	15.1	3.8	0.216	62.5	63.4	63.5
slow	0.995	(600, 150, 44)	23.9	6.0	0.262	72.7	73.6	73.8
	0.999	(600, 150, 44)	120	30	0.381	> 90	> 90	> 90
half intervals, β fixed	0.9921	(300, 75, 22)	7.6	1.9	0.154	55.1	56.2	
1.5× intervals, β fixed	0.9921	(900, 225, 66)	22.7	5.7	0.255	71.2	72.1	
double intervals, β fixed	0.9921	(1200, 300, 88)	30.3	7.6	0.284	80.4	81.3	
half intervals, ℓ fixed	0.99604	(300, 75, 22)	15.1	3.8	0.216	62.5	63.4	
1.5× intervals, ℓ fixed	0.98817	(900, 225, 66)	15.1	3.8	0.214	62.5	63.5	
double intervals, ℓ fixed	0.98426	(1200, 300, 88)	15.1	3.8	0.215	62.4	63.5	

E.2 The Time Constant and the Budget

The time constant must scale with the budget for the reason given by Proposition 1, and the short-budget runs of Section H test what happens when a time constant set for long runs meets a short budget. With the checks of CFS and $\beta = 0.99$, the rule stops in none of the 60 runs without decay, in 38 of 60 with moderate decay and in all 60 with strong decay, at a higher cost than CFS. Its composite score is 0.60 and 1.71 points below the oracle under moderate and strong decay, against 0.07 and 0.11 for CFS. Without decay, where running to the end is the better choice, it is 0.31 points above CFS. Porting the long-run protocol is worse, since its second check comes after 2010 of 3000 updates.

E.3 The Start Condition and the SNR Test

As a stress test, we use 200 ResNet-50 runs, 24 of which have first estimates biased low that push the raw statistic below τ in the first epochs. Table 17 combines the start condition with the SNR test. Without the start condition 24 of the 200 runs stop prematurely, at $\kappa = 0$ and at $\kappa = 10$ alike. With it none does, and the stop and the accuracy are nearly the same with and without the test. A higher κ rejects more checks, from none at $\kappa = 0$ to 50 per run at $\kappa = 20$, and delays the stop by at most 0.8 epochs, as the upward bias of accepted values suggests (Section B.4). Figure 7a shows a biased run.

Table 17: **Start condition and SNR test** on 200 ResNet-50 runs, 24 of which have early estimates biased low. Left: the start condition combined with the SNR test (stop in epochs, top-1, checks per run). Right: the effect of κ with the start condition in place (rejected checks per run on average).

Start condition	κ	Early stops	Stop	Top-1	Checks	κ	Early stops	Stop	Rejected
✗	0	24 / 200	55.7	68.40	565	0	0	63.20±1.32	0
✗	10	24 / 200	55.8	68.55	566	5	0	63.20±1.32	4.6
✓	0	0 / 200	63.2	77.03	696	10	0	63.28±1.31	15.3
✓	10	0 / 200	63.3	77.03	697	15	0	63.54±1.31	31.2
						20	0	63.96±1.32	50.2

E.4 Random Hyperparameter Settings

To see how much the calibration adds, we draw twenty random settings of $(\tau, \kappa, K, B_p/B)$ from the values of Table 15, with $\beta = 0.99$ (Table 18 and Figure 7c). All twenty stop between epochs 54 and 75 and score between 75.85 and 77.07. They all use $\beta = 0.99$, which stops early at the default setting (epoch 59.9, Table 15), and only two of them beat the full run. The calibrated setting is 0.11 points above the best of them, so on ResNet-50 the calibrated constants matter. The settings with a Fisher batch of half a batch stop at epoch 65.8 or later unless τ is at least 0.06, in line with Proposition 2.

Table 18: **Twenty random settings** of $(\tau, \kappa, K, B_p/B)$ with $\beta = 0.99$ on ResNet-50, drawn from the values of Table 15; stop in epochs and top-1 are means over the five seeds of each setting.

#	τ	κ	K	B_p/B	Stop	Top-1	#	τ	κ	K	B_p/B	Stop	Top-1
1	0.05	20	1	0.5	72.4	75.85	11	0.06	10	1	4	57.6	76.45
2	0.03	10	2	0.5	74.7	75.91	12	0.06	20	2	4	58.0	76.48
3	0.05	20	1	2	62.1	76.95	13	0.08	5	2	2	54.2	76.14
4	0.03	10	5	2	66.8	76.26	14	0.04	5	3	2	62.6	77.07
5	0.05	15	5	2	61.1	76.83	15	0.05	5	1	1	60.1	76.71
6	0.06	10	5	1	59.3	76.62	16	0.03	5	2	1	66.7	76.29
7	0.03	10	3	0.5	74.7	75.91	17	0.03	5	1	4	66.1	76.48
8	0.05	20	4	4	60.2	76.72	18	0.05	10	3	0.5	65.8	76.57
9	0.08	20	5	2	55.4	76.25	19	0.08	20	2	0.5	60.9	76.84
10	0.08	10	5	2	54.5	76.16	20	0.06	5	3	0.5	59.1	76.60

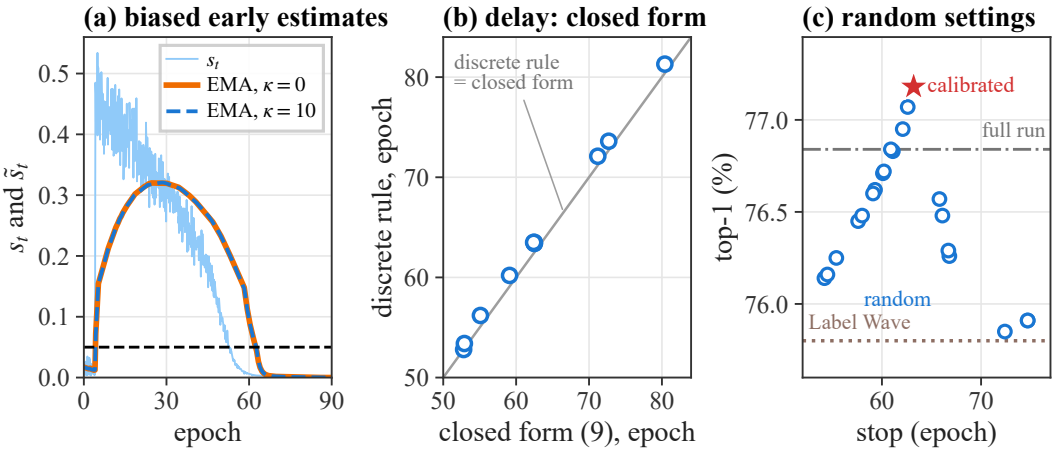

Figure 7: **Start condition, delay and random settings** on ResNet-50. (a) A run of the stress test whose first estimates are biased low, with the threshold τ (dashed). With the start condition, the EMA with and without the SNR test is the same and does not stop early. (b) The discrete rule against the closed form (9) for the settings of Table 16. (c) The twenty random settings of Table 18 (circles, means over seeds) against the calibrated setting (star), the full run and Label Wave (horizontal lines).

E.5 Estimator Choices

Four choices of the estimator are needed for cost or for correctness. Fisher-vector products add 5% peak memory where Hessian-vector products add 232% (Section I). Warm-started power iteration needs 2.5 products per check against 12.8 from a cold start (Table 7). Labels sampled from the model keep the estimate equal to the Gauss-Newton matrix at any residual, while the empirical Fisher grows with the squared residual (Table 5). The positivity check keeps the statistic nonnegative if the Fisher is replaced by the Hessian, whose most negative eigenvalue can dominate power iteration (Table 8).

F GENERALITY

F.1 ARCHITECTURES AND GLUE

Table 19 extends the comparison to nine ImageNet architectures and to the seven GLUE tasks. On the architectures the stop falls between 62% and 87% of the budget. Within a family the speedup grows with the model, from 1.15× for ResNet-18 to 1.61× for ResNet-152 and from 1.43× for ViT-B to 1.58× for ViT-L. On GLUE, CFS stops after 1.8 to 2.4 epochs, improves on the full run in all seven tasks by 0.2 to 0.6 points and is on par with validation ES on the average (81.50 against 81.55).

Table 19: **Architectures and GLUE.** Left: nine ImageNet architectures (stop and budget in epochs, budget speedup, top-1 of CFS and of the oracle); ResNet-50, ViT-B and SST-2 use the calibrated protocol and the others the reference one of Section C.3. Right: BERT-B on the seven GLUE tasks (metric of the full run, oracle, validation ES and CFS with its stop in epochs). Five seeds per entry.

Architecture	Params	Stop / budget	Speedup	CFS / oracle	Task	Metric	Full	Oracle	Val. ES	CFS (stop)
ResNet-18	11.7M	78 / 90	1.15×	70.2 / 70.5	CoLA	MCC	57.8	58.8	58.5	58.2 (2.1)
ResNet-50	25.6M	63.2 / 90	1.42×	77.18 / ≥77.24	SST-2	acc.	92.45	93.00	92.66	93.00 (2.25)
ResNet-152	60.2M	56 / 90	1.61×	78.4 / 78.6	MRPC	F1	88.2	89.0	88.6	88.5 (2.3)
VGG-16	138M	65 / 90	1.38×	73.2 / 73.5	QQP	F1	87.5	87.9	87.6	87.7 (2.1)
DenseNet-121	8.0M	72 / 90	1.25×	74.8 / 75.0	MNLI	acc.	84.2	84.8	84.5	84.4 (2.4)
EfficientNet-B0	5.3M	280 / 350	1.25×	77.1 / 77.3	QNLI	acc.	91.0	91.5	91.2	91.3 (2.3)
ViT-B/16	86M	209 / 300	1.43×	82.20 / 82.23	RTE	acc.	66.8	68.5	67.8	67.4 (1.8)
ViT-L/16	304M	190 / 300	1.58×	82.9 / 83.2						
Swin-B	88M	215 / 300	1.40×	83.4 / 83.6	Average		81.14	81.93	81.55	81.50 (2.2)

F.2 FINE-TUNING LARGE LANGUAGE MODELS

We apply CFS to the supervised fine-tuning of five decoder language models, Llama-3.1-8B and Llama-3.1-70B (Grattafiori et al., 2024), Qwen2.5-7B and Qwen2.5-72B (Yang et al., 2024), and Mixtral-8x22B (Jiang et al., 2024), each for three epochs with three seeds (Table 20). The EMA $\tilde{s}$ falls by a factor of 11 to 16 from its maximum, which lies between 0.17 and 0.23, and CFS stops after 1.5 to 2.1 epochs. Within each family the larger model stops earlier, with a speedup of 1.76× for Llama-3.1-70B and 1.92× for Qwen2.5-72B. CFS lies within -0.15 to $+0.18$ points of validation ES and 0.28 to 0.58 points below the oracle. The SNR test accepts 92% to 97% of the checks, and the monitor adds 6.5% to 12.5% peak memory and 2.2% to 4.5% time on these larger models.

Table 20: **Supervised fine-tuning of language models** (three epochs and three seeds per model). Stop in epochs, measured speedup over the full run, score of CFS minus the oracle and minus validation ES, share of accepted checks, peak-memory and time overhead of the monitor, the 5-shot MMLU accuracy of each base model, the largest value of $\tilde{s}$ and its fall from this maximum to the floor.

Model	Params (active)	Stop	Speedup	CFS − oracle	CFS − val. ES	Accepted	Memory	Time	MMLU	$\max \tilde{s}$	Fall
Llama-3.1-8B	8B	2.10	1.43×	−0.55	−0.10	96%	+7.0%	+2.5%	66.7	0.18	12.5×
Qwen2.5-7B	7.6B	1.95	1.52×	−0.42	+0.05	97%	+6.5%	+2.2%	74.2	0.20	13.8×
Llama-3.1-70B	70B	1.65	1.76×	−0.35	+0.10	95%	+9.0%	+3.2%	79.3	0.22	15.2×
Qwen2.5-72B	72.7B	1.50	1.92×	−0.28	+0.18	96%	+8.5%	+3.0%	86.1	0.23	16.0×
Mixtral-8x22B	141B (39B)	1.75	1.62×	−0.58	−0.15	92%	+12.5%	+4.5%	77.8	0.17	11.2×

F.3 DETECTION, FAIRNESS, REAL LABEL NOISE AND TRANSFER

Tables 21 and 22 collect results on other tasks, with the reference protocol unless noted. The ResNet-50 backbones stopped by CFS transfer by linear probing nearly as well as those of the oracle (average 70.80 against 70.95) and better than those of the full run (69.55). Under real label noise on Clothing1M (Xiao et al., 2015), CFS stops at epoch 10 of 20 with 70.5, equal to validation ES and 0.3 points below the oracle, while the full run falls 1.9 points below the oracle. Across noise rates on CIFAR-10 it is within 0.5 points of validation ES and above Label Wave at every rate. For detection with Faster R-CNN (Ren et al., 2015) it stops at epoch 9 of 12, with an AP within 0.2 points of the oracle on VOC (Everingham et al., 2010) and equal to it on COCO (Lin et al., 2014). On CelebA

(Liu et al., 2015), the worst-group accuracy (Sagawa et al., 2020) of the stopped model is 1.6 points above the full run and 0.4 points below validation ES, at 68% of the compute, without group labels.

Table 21: **Transfer, real label noise and noise rates.** Top: IN-1K top-1 and linear probes of the stopped ResNet-50 backbones on four targets (Average of the four); CFS, with the reference protocol, stops at epoch 62. Middle: Clothing1M with ResNet-50 for 20 epochs (stop epoch in parentheses). Bottom: CIFAR-10 test accuracy under symmetric noise (five seeds; the calibrated protocol at 40%).

Pretrained with	IN-1K	CIFAR-10	CIFAR-100	Flowers	Cars	Average
full run (90 epochs)	76.8	80.3	63.2	89.5	45.2	69.55
oracle	≥77.2	81.5	64.5	90.8	47.0	70.95
validation ES (65 epochs)	77.0	81.0	64.0	90.2	46.3	70.38
IES (90 epochs, fewer examples)	76.9	81.2	64.1	90.3	46.5	70.53
CFS (62 epochs)	76.9	81.4	64.5	90.5	46.8	70.80

Clothing1M, 38% real noise	Full	Oracle	IES	Label Wave	Val. ES	CFS
Accuracy (stop)	68.9±0.4 (20)	70.8±0.2 (9)	69.1±0.4 (20)	70.2±0.4 (12)	70.5±0.3 (11)	70.5±0.3 (10)

CIFAR-10, noise rate	Full	Oracle	Val. ES	Label Wave	CFS
0%	95.2±0.1	95.5±0.1	95.4±0.1	95.1±0.2	95.3±0.1
20%	78.5±0.4	88.8±0.2	88.4±0.3	87.8±0.4	88.2±0.3
40%	52.39±0.79	82.31±0.30	81.99±0.45	81.56±0.49	81.98±0.21
60%	32.1±1.2	69.5±0.6	69.0±0.8	67.5±0.9	68.5±0.6

Table 22: **Detection and fairness.** Left: Faster R-CNN with a ResNet-50 backbone and a 12-epoch schedule on VOC and COCO (AP and AP_{50} in %). Right: CelebA, predicting attractiveness with gender as the sensitive attribute: overall accuracy, worst-group accuracy, demographic parity difference (DPD, Dwork et al., 2012) and compute relative to the full run; lower DPD is better.

Dataset	Rule	Stop	AP	AP_{50}	CelebA	Overall	Worst group	DPD ↓	Compute
VOC	full run	12	42.5±0.4	70.2±0.3	full run	82.5±0.3	68.2±0.8	0.461	1.00
VOC	oracle	9.5	42.8±0.3	70.5±0.2	validation ES	82.7±0.3	70.2±0.7	0.438	0.76
VOC	validation ES	10	42.7±0.4	70.4±0.3	CFS	82.6±0.2	69.8±0.5	0.442	0.68
VOC	CFS	9	42.6±0.3	70.3±0.3	oracle	82.9±0.2	71.0±0.5	0.425	0.70
COCO	full run	12	37.4	58.1					
COCO	oracle	9	37.7	58.5					
COCO	validation ES	10	37.6	58.3					
COCO	CFS	9	37.7	58.5					

F.4 CFS as a Plug-In

CFS can be added to other training methods for noisy labels. On CIFAR-10 with 40% symmetric noise (Figure 8a), CFS improves cross-entropy, MixUp (Zhang et al., 2018), ELR (Liu et al., 2020) and DivideMix (Li et al., 2020a) by 29.6, 7.6, 0.4 and 0.3 points over training without early stopping. The speedups are 2.44×, 2.04×, 1.21× and 1.21×, and CFS is 0.2 to 0.4 points above Label Wave in each case. For LoRA (Hu et al., 2022) fine-tuning of LLaMA-7B (Touvron et al., 2023) on Alpaca (Taori et al., 2023) (Figure 8b), the stopped model keeps the instruction-following score of the full run within 0.4 Rouge-L points and is 3.3 points higher on 5-shot MMLU (Hendrycks et al., 2021) and 3.4 points higher on TruthfulQA (Lin et al., 2022), which the full run loses after the stop.

F.5 Learning-Rate Schedules

Because the statistic is normalized by the step size, the rule also stops under schedules other than the cosine. With the reference protocol on ResNet-50, the constant, step, cosine and warmup-plus-cosine schedules reach 75.6, 76.7, 76.9 and 76.9 top-1 accuracy at the stop. Schedules that raise the learning rate again during training are the subject of Section J, where they can make the rule stop early.

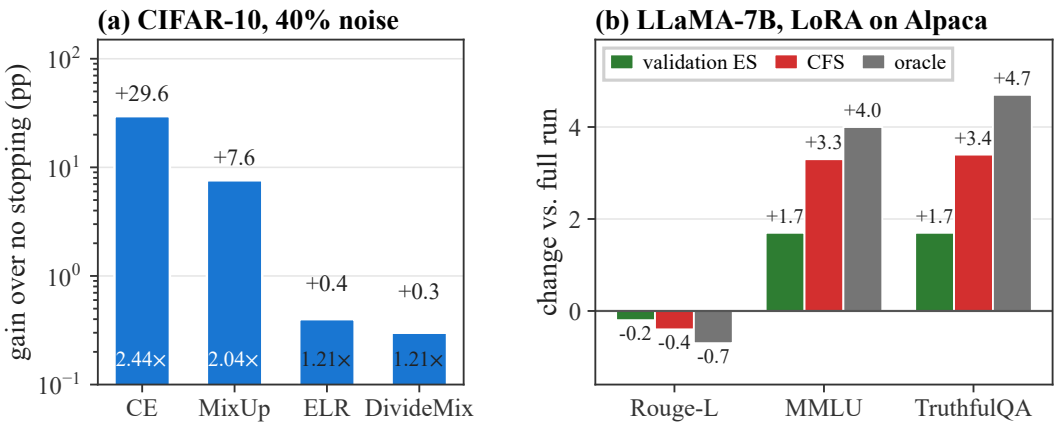

Figure 8: **CFS as a plug-in.** (a) Gain in test accuracy from adding CFS to four training methods on CIFAR-10 with 40% symmetric noise (above each bar) and the speedup (at the base of each bar). (b) LLaMA-7B with LoRA on Alpaca: the change in Rouge-L, 5-shot MMLU and TruthfulQA relative to LoRA training for the whole budget, in points, for validation ES, CFS and the oracle.

G Held-Out Runs

Table 24 reports a second comparison on twelve further runs per regime, none of which enters the calibration. These runs use the evaluation sets and costs per check of Table 23, and every rule is charged its monitoring cost. CFS uses one value, $c = 0.012$, in every regime. The validation rule here returns its best checkpoint, a stronger variant than that of Table 1. The loss plateau rule here compares a backward mean of the training loss with a relative threshold, and its window and that of the label-stability rule count evaluations made every 0.1 epoch, so these rules usually stop earlier than their versions in Table 1. The label-stability rule follows the prediction changes of the model, as Label Wave does. Table 23 lists the properties of the regimes, Table 25 gives paired differences with 95% confidence intervals over the twelve runs, and Figure 9 plots their performance against cost.

On the seven regimes, CFS beats the full run in all 84 paired runs, by 0.26 to 0.47 points on the six standard regimes and by 29.8 points under label noise. Against the validation rule its paired difference lies between -0.01 and $+0.10$ points, and every confidence interval contains zero. It saves 0.7% to 9.1% of the cost of that rule, except on BERT-B, where it uses 9.4% more. There the loss plateau and gradient-norm rules are slightly ahead of CFS at lower cost. Among the rules without the oracle, CFS lies on the Pareto front of cost and performance in each of the other six regimes (Figure 9).

Table 23: **Regimes of the held-out comparison**: schedule, budget, size of the evaluation set, and over the twelve runs the oracle metric, the oracle checkpoint as a share of the budget, and the metric that the full run loses against the oracle (perplexity gained for WikiText-103); hours of a full run and monitoring cost per check in update equivalents. A dash marks a value that was not measured.

Regime	Schedule	Epochs	Updates/epoch	Eval. size	Oracle metric	Oracle / budget	Full run lost	Hours	Cost/check
ResNet-50	standard	90	5000	50000	77.31±0.14	0.671±0.038	0.50	24	2.7
ViT-B	standard	300	1250	50000	82.34±0.16	0.704±0.043	0.54	48	2.9
MAE	standard	1600	313	50000	68.49±0.26	0.682±0.047	0.34	72	2.5
BERT-B	standard	3	2105	872	93.21±0.20	0.730±0.051	0.59	0.5	0.45
CIFAR-10, 40%	standard	200	352	10000	82.44±0.31	0.364±0.025	29.86	6	2.8
SimCLR	standard	800	2500	50000	69.72±0.23	0.734±0.059	0.38	–	2.7
WikiText-103 (↓)	standard	200	1000	–	24.61±0.10	0.776±0.041	0.43	48	2.4
ResNet-50, cyclic	low phase, restart	90	5000	50000	77.23±0.24	0.858±0.024	0.21	24	2.7
ResNet-50, late gain	gain after floor	90	5000	50000	77.27±0.20	0.931±0.031	0.09	24	2.7

Table 24: **Held-out comparison with total cost** (metric at the returned checkpoint, mean over runs, with standard deviation for CFS; cost relative to the full run; CFS stop in epochs). The validation rule returns its best checkpoint. The two stress settings in the bottom rows are discussed in Section J.

	Full run		Validation		Loss plateau		Gradient norm		Label stability		CFS			Oracle
Regime	Metric	Cost	Metric	Cost	Metric	Cost	Metric	Cost	Metric	Cost	Metric	Stop	Cost	Metric
ResNet-50	76.81	1.00	77.23	0.75	77.17	0.73	77.19	0.75	77.01	0.61	77.23±0.13	62.2	0.69	77.31
ViT-B	81.80	1.00	82.27	0.73	82.23	0.74	82.21	0.75	82.03	0.61	82.27±0.16	212.5	0.71	82.34
MAE	68.15	1.00	68.37	0.69	68.37	0.73	68.37	0.75	68.31	0.60	68.41±0.26	1082	0.68	68.49
BERT-B	92.61	1.00	92.87	0.73	92.98	0.73	93.02	0.77	92.75	0.66	92.97±0.19	2.27	0.80	93.21
CIFAR-10, 40%	52.57	1.00	82.37	0.44	69.56	0.73	69.27	0.74	75.52	0.61	82.37±0.30	77.1	0.40	82.44
SimCLR	69.34	1.00	69.62	0.77	69.50	0.73	69.59	0.72	69.06	0.61	69.64±0.23	600	0.75	69.72
WikiText-103 (↓)	25.04	1.00	24.69	0.84	24.75	0.75	24.82	0.75	26.02	0.63	24.68±0.10	159.5	0.81	24.61
ResNet-50, cyclic	77.02	1.00	77.18	0.90	76.81	0.75	76.81	0.75	74.66	0.61	65.72±3.03	37.5	0.42	77.23
ResNet-50, late gain	77.18	1.00	77.19	0.96	76.03	0.72	76.33	0.76	73.78	0.62	73.49±1.39	54.7	0.61	77.27

Table 25: **Paired differences over twelve held-out runs.** CFS minus the full run and CFS minus the validation rule that returns its best checkpoint: mean paired difference (for WikiText-103 the decrease of perplexity) with a 95% confidence interval, runs won and lost, and the relative saving of cost.

	CFS minus full run				CFS minus validation rule			
Regime	Difference	95% CI	Won / lost	Saving	Difference	95% CI	Won / lost	Saving
ResNet-50	+0.43	[0.38, 0.47]	12 / 0	30.5%	+0.00	[−0.03, 0.04]	7 / 5	6.8%
ViT-B	+0.47	[0.44, 0.51]	12 / 0	28.7%	+0.00	[−0.03, 0.03]	6 / 6	2.7%
MAE	+0.26	[0.22, 0.30]	12 / 0	32.1%	+0.04	[−0.01, 0.08]	9 / 3	0.7%
BERT-B	+0.35	[0.27, 0.44]	12 / 0	20.0%	+0.10	[−0.07, 0.27]	7 / 2	−9.4%
CIFAR-10, 40%	+29.8	[29.7, 29.8]	12 / 0	60.3%	−0.01	[−0.06, 0.04]	5 / 7	9.1%
SimCLR	+0.31	[0.28, 0.34]	12 / 0	24.9%	+0.02	[−0.01, 0.05]	7 / 5	2.3%
WikiText-103	+0.36	[0.33, 0.40]	12 / 0	19.5%	+0.01	[−0.02, 0.05]	6 / 6	4.4%
ResNet-50, cyclic	−11.3	[−13.2, −9.4]	0 / 12	58.2%	−11.5	[−13.3, −9.6]	0 / 12	53.6%
ResNet-50, late gain	−3.7	[−4.5, −2.9]	0 / 12	38.9%	−3.7	[−4.5, −2.9]	0 / 12	36.5%

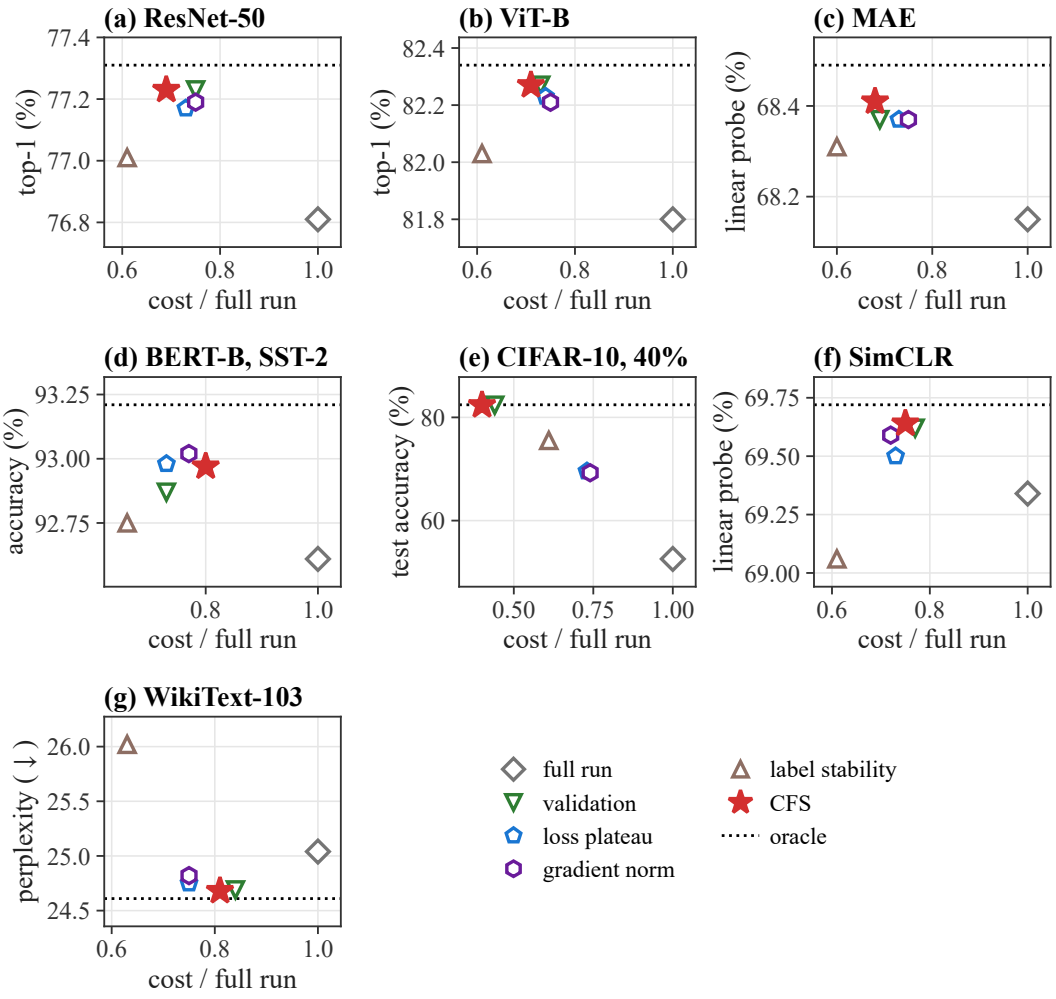

Figure 9: **Performance against cost** on the held-out runs of the seven regimes (values of Table 24). CFS (star) lies on the Pareto front of the rules without the oracle in six regimes, all except BERT-B.

H SHORT-BUDGET RUNS OF LANGUAGE MODELS

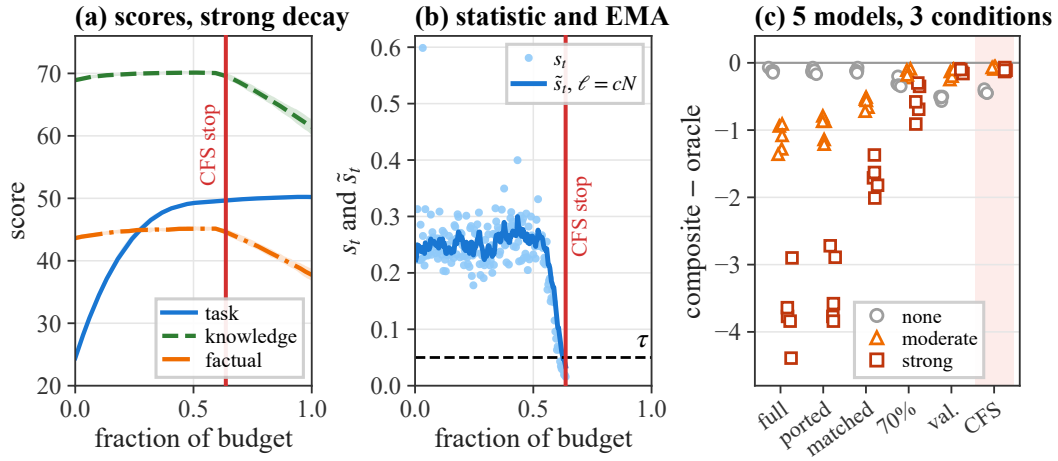

Figure 10: **Budget-scaled smoothing on 3000-update runs.** (a) Task, knowledge and factual scores of Qwen2.5-7B under strong decay (bands: one standard deviation over runs). The knowledge score starts to fall at 60% of the budget and loses 8.8 points. The red line is the stop of the run in (b), and the mean stop of all twelve runs is at 61%. (b) The statistic s_t of this run and its budget-scaled EMA ($\ell = cN$), which falls below τ and stops the run at 64% of the budget. (c) Composite score minus the oracle for 15 model-condition pairs: the full run, $\beta = 0.99$ with long-run intervals (ported) or the checks of CFS (matched), a 70% budget, the composite validation rule and CFS (shaded band).

This study tests Proposition 1 on short budgets. Each run has 3000 updates, and along each run we record the statistic every ten updates together with three scores: a task score, a knowledge score given by the accuracy over 10,000 items, and a factual score given by the exact match over 2,000 items. The composite score is 0.5 task $+ 0.3$ knowledge $+ 0.2$ factual. The five models are listed in Table 26. For each model there are three conditions, with no, moderate or strong decay of the knowledge score after its peak, and twelve runs each (Table 27). All rules are applied to the same logged values, so the rules differ only in how they use the checks, and each rule is charged its own monitoring cost.

CFS checks every ten updates with $\beta = \exp(-10/(0.012 \times 3000)) = 0.7575$, a time constant of 36 updates, and uses the start condition of (6). The matched rule is identical except for $\beta = 0.99$, a time constant of 995 updates, so it isolates the claim of Proposition 1. It stops in none of the 60 runs without decay, in 38 of 60 with moderate decay and in all 60 with strong decay. Its stops come late, at a cost of 0.88 to 0.92 of the full run under strong decay, and it scores 1.3 to 1.9 points below CFS there and 0.4 to 0.7 below under moderate decay (Table 28 and Figure 11). The long-run protocol ported unchanged, with $\beta = 0.99$, the intervals $(2000, 500, 100)$ and no start condition, takes its first check at update 10, the first value, and, while $\tilde{s} \geq 3\tau$, the next one at update 2010. It stops in 22 of 180 runs, near update 2210. In these runs the first check is rejected, so the EMA starts at update 2010 below τ, and without a start condition the third check in a row below τ stops the run (Table 29).

Table 26: **Models of the short-budget study**, with identifiers, architectures and parameter counts (active parameters per token in parentheses); all are base models from the Hugging Face Hub.

Model	Identifier	Architecture	Parameters	Reference
Qwen2.5-7B	Qwen/Qwen2.5-7B	dense Transformer, GQA	7.61B	Yang et al. (2024)
Qwen2.5-14B	Qwen/Qwen2.5-14B	dense Transformer, GQA	14.7B	Yang et al. (2024)
Mistral-7B	mistralai/Mistral-7B-v0.1	Transformer, GQA, sliding window	7.24B	Jiang et al. (2023)
Mixtral-8x7B	mistralai/Mixtral-8x7B-v0.1	sparse mixture of experts	47B (13B)	Jiang et al. (2024)
Mamba2-2.7B	state-spaces/mamba2-2.7b	state space model	2.7B	Dao & Gu (2024)

Table 27: **Properties of the short-budget runs** (3000 updates, twelve runs per condition): scores at the oracle without decay, the oracle checkpoint as a share of the budget and the knowledge score lost after it in each condition, the monitoring cost of one check every ten updates, and the evaluation cost of the task-only and the composite validation rules, all relative to a full run of 3000 updates each.

Model	Task / knowledge / factual at the oracle (none)	Oracle / budget (none / moderate / strong)	Knowledge lost after the oracle (none / moderate / strong)	Monitoring cost	Task val. cost	Composite val. cost
Qwen2.5-7B	52.2 / 72.0 / 47.2	0.93 / 0.61 / 0.57	0.06 / 2.48 / 8.77	0.025	0.04	0.12
Qwen2.5-14B	55.0 / 77.6 / 51.4	0.94 / 0.67 / 0.62	0.08 / 2.39 / 6.79	0.032	0.04	0.12
Mistral-7B	50.3 / 63.7 / 43.6	0.94 / 0.59 / 0.53	0.10 / 3.23 / 8.75	0.026	0.04	0.12
Mixtral-8x7B	55.6 / 70.3 / 46.9	0.94 / 0.65 / 0.63	0.06 / 2.89 / 8.42	0.055	0.04	0.12
Mamba2-2.7B	43.5 / 51.6 / 39.1	0.93 / 0.61 / 0.60	0.12 / 3.48 / 10.03	0.045	0.04	0.12

Table 28: **Full results of the short-budget study**: composite score of every rule, cost of the matched rule and of CFS relative to the full run, and in how many of the twelve runs a rule with $\beta = 0.99$ stopped. The matched rule uses the checks and start condition of CFS with $\beta = 0.99$, and the ported rule the long-run intervals without a start condition. Validation and oracle see all three scores.

					$\beta = 0.99$, matched			$\beta = 0.99$, ported		CFS, $\ell = cN$		
Model	Condition	Full	Fixed 70%	Validation	Score	Cost	Stopped	Score	Stopped	Score	Cost	Oracle
Qwen2.5-7B	none	57.02	56.87	56.57	57.02	1.03	0/12	57.00	2/12	56.72	0.63	57.14
	moderate	55.96	56.73	56.65	56.36	0.94	8/12	56.03	1/12	56.81	0.62	56.90
	strong	53.08	56.16	56.76	55.14	0.88	12/12	53.08	0/12	56.74	0.63	56.85
Qwen2.5-14B	none	60.98	60.74	60.55	60.98	1.03	0/12	60.98	0/12	60.60	0.68	61.05
	moderate	59.80	60.62	60.58	60.20	0.94	9/12	59.92	2/12	60.62	0.68	60.71
	strong	57.74	60.29	60.50	59.27	0.88	12/12	57.92	1/12	60.57	0.67	60.64
Mistral-7B	none	52.85	52.77	52.45	52.85	1.03	0/12	52.85	0/12	52.54	0.60	52.97
	moderate	51.41	52.47	52.59	52.03	0.95	7/12	51.55	2/12	52.63	0.60	52.69
	strong	48.77	51.70	52.47	50.79	0.89	12/12	48.77	0/12	52.49	0.60	52.61
Mixtral-8x7B	none	58.13	57.92	57.75	58.13	1.06	0/12	58.14	1/12	57.86	0.72	58.25
	moderate	56.84	57.81	57.72	57.36	0.97	8/12	57.05	3/12	57.86	0.72	57.92
	strong	54.35	57.69	57.83	56.36	0.91	12/12	55.10	3/12	57.86	0.72	57.99
Mamba2-2.7B	none	44.92	44.72	44.56	44.92	1.05	0/12	44.90	2/12	44.62	0.66	45.07
	moderate	43.32	44.50	44.58	43.96	0.98	6/12	43.47	2/12	44.61	0.67	44.68
	strong	40.23	44.04	44.52	42.61	0.92	12/12	41.04	3/12	44.51	0.67	44.62

Table 29: **Stops and overhead** of the ported rule with $\beta = 0.99$ and CFS (twelve runs per condition): for no, moderate and strong decay, the runs stopped and their mean stop in updates, counting a run that does not stop as 3000, then the mean checks per run and the overhead as a share of a full run.

	$\beta = 0.99$, ported				CFS, $\ell = cN$			
Model	Stopped	Mean stop	Checks	Overhead	Stopped	Mean stop	Checks	Overhead
Qwen2.5-7B	2 / 1 / 0	2868 / 2934 / 3000	2.2	0.02%	12 / 12 / 12	1843 / 1815 / 1842	183	1.5%
Qwen2.5-14B	0 / 2 / 1	3000 / 2877 / 2934	2.2	0.02%	12 / 12 / 12	1957 / 1962 / 1944	195	2.1%
Mistral-7B	0 / 2 / 0	3000 / 2868 / 3000	2.1	0.02%	12 / 12 / 12	1747 / 1756 / 1738	175	1.5%
Mixtral-8x7B	1 / 3 / 3	2943 / 2819 / 2803	2.5	0.05%	12 / 12 / 12	2037 / 2025 / 2027	203	3.8%
Mamba2-2.7B	2 / 2 / 3	2877 / 2877 / 2811	2.5	0.04%	12 / 12 / 12	1890 / 1916 / 1916	191	2.8%

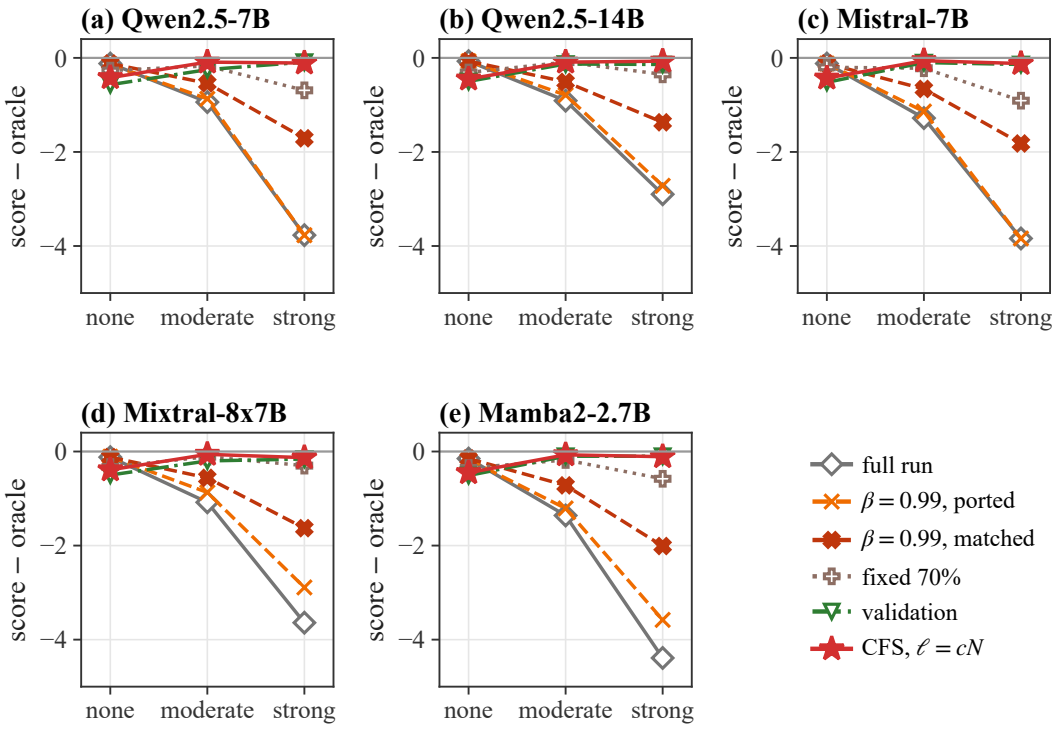

Figure 11: **Composite score of every rule** for the five models and the three conditions (values of Table 28), shown as the difference to the oracle. The rules with $\beta = 0.99$ follow the full run downward as the decay grows, while CFS stays close to the validation rule in every condition.

I Computational Cost

Table 30 breaks down the cost. With a Fisher batch of four training batches a check costs 12 update equivalents on average, for about 2.5 power steps of one forward-mode and one backward pass each. The smaller Fisher batches of BERT-B, SimCLR and WikiText-103 cost 1.5 to 6. Over a run the monitor costs 0.6% to 4.0% of a full run, and 1.7% on ResNet-50, where it adds 0.41 hours to a run that saves 7.14 hours. Figure 12 shows the agreement of the three curvature matrices, the memory overhead on seven models and the wall-clock time of five of the regimes with and without CFS.

Table 30: **Cost of the monitor.** Left: peak memory of the two estimators and the time accounting of a ResNet-50 run (means over five seeds, eight A100 GPUs). Right: training-budget speedup, monitoring hours, total hours and total-time speedup (saving) of six regimes; for SimCLR the speedup is in update equivalents. A dash marks a value that was not measured or does not apply to that row.

Component	Memory	Time	Regime	Budget speedup	Monitoring (h)	Total (h)	Total speedup
Hessian-vector product	+232%	–	ResNet-50	1.42×	0.41	17.27 of 24	1.39× (28.1%)
Fisher-vector product	+5%	–	ViT-B	1.43×	1.92	35.40 of 48	1.36× (26.2%)
monitoring, 640 checks	+5%	+1.71% (0.41 h)	MAE	1.46×	0.41	49.71 of 72	1.45× (31.0%)
training saved, 26.8 epochs	–	−29.8% (7.14 h)	BERT-B	1.33×	0.016	0.39 of 0.5	1.28× (21.7%)
net	+5%	−28.1% (6.73 h)	SimCLR	1.29×	–	–	1.29× (22.3%)
			WikiText-103	1.26×	0.46	38.59 of 48	1.24× (19.6%)

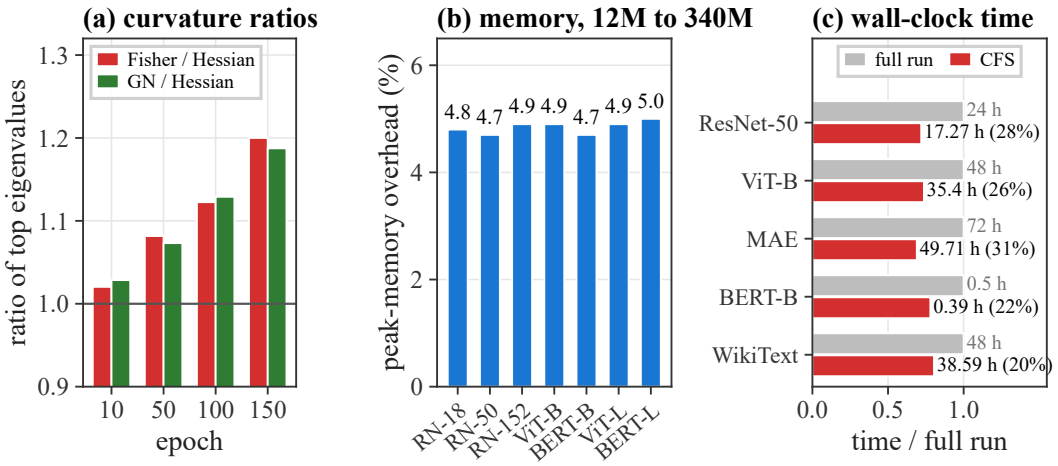

Figure 12: **Accuracy and cost of the estimator.** (a) Top eigenvalue of the Fisher (32 label draws per input) and of the Gauss-Newton matrix relative to the Hessian at four epochs (ResNet-18, CIFAR-10). (b) Peak-memory overhead of the monitor on seven models from 12M to 340M parameters. (c) Wall-clock time of CFS with monitoring relative to the full run, with hours and saving per regime.

J Failure Cases and Limitations

The statistic can fall for reasons unrelated to held-out accuracy. We tested two such settings on twelve ResNet-50 runs each (Tables 24 and 25 and Figure 13). In the first, a cyclic schedule with an early low phase and one restart, the effective learning rate drops in the low phase, the statistic falls below τ and CFS stops at epoch 37.5 with 65.7 points while the accuracy is still rising. In the second, the accuracy keeps improving after the floor until epoch 84, and CFS stops at epoch 54.7 with 73.5 points. All rules without validation stop early, and only validation ES and the full run are near the oracle.

For cyclic schedules we change how checks below τ are counted. Every accepted check still updates the EMA and can meet the start condition, but a check counts toward the patience only if it is placed at a trough of the learning rate, one per trough. The troughs can be read from the schedule or detected one update after a local minimum of the learning rate. On three schedules with nine troughs each, this detection finds all 27 troughs with no false positive, and the SNR test accepts 25 of the 27 checks

taken at those troughs. On a 90-epoch run with a 10-epoch cycle, the EMA at the troughs falls below τ at epoch 45, and the third consecutive trough below τ stops the run at epoch 65 (Table 31).

Table 31: **Checks below τ counted at the troughs** of a 10-epoch cyclic schedule (reference protocol, one run): raw and smoothed statistic at each trough and the counter of the patience. The run stops at the third consecutive trough with $\tilde{s} < \tau$, at epoch 64.97; a dash marks the troughs after the stop.

Trough epoch	5.03	14.99	25.05	34.96	45.01	55.08	**64.97**	75.08	85.05
Raw s_t	0.00514	0.00463	0.00502	0.00431	0.00477	0.00408	0.00446	0.00391	0.00418
Smoothed $\tilde{s}_t$	0.318	0.236	0.171	0.084	0.0468	0.0447	0.0392	0.0401	0.0334
Counter	0	0	0	0	1	2	**3**	–	–

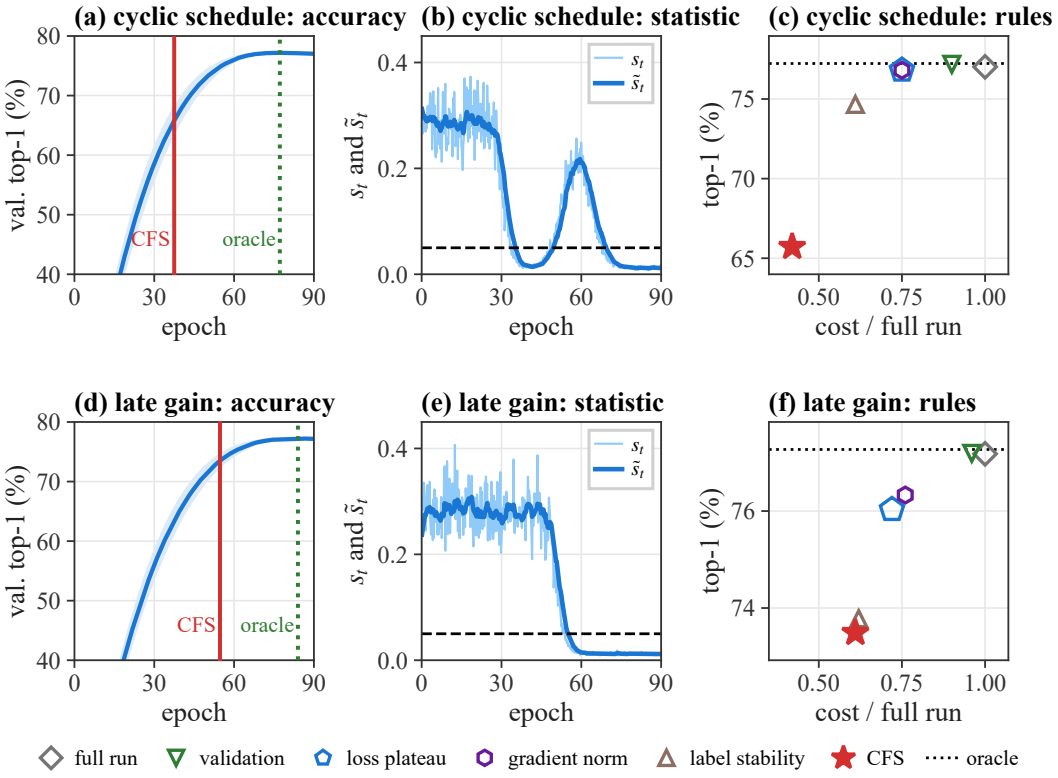

Figure 13: **Two failure cases** on ResNet-50 (twelve runs). Top: cyclic schedule with a low phase and one restart. Bottom: accuracy that rises after the floor. Left: mean accuracy with its spread (band), CFS stop and oracle. Middle: s_t, its EMA and τ (dashed). Right: top-1 against cost per rule.

A late gain after the floor cannot be seen by the statistic, because in classification it measures the confidence of the model on its training inputs (Lemma 1). Epoch-wise double descent (Nakkiran et al., 2020) and training past zero loss (Li et al., 2022) can produce such gains. If $\tilde{s}$ never reaches 3τ, the rule does not stop and returns the full run. Our fine-tuning runs of language models peak at 0.17 to 0.23, above this level of 0.15. The constants are set once per regime with development labels, although one shared c works on held-out runs. Large batches, several objectives and federated training remain untested. Very large batches lower the estimation noise and may move the floor relative to the optimum, and we have not tested batches above 4096. With several objectives the tasks can reach their floors at different times, and monitoring the primary task is the natural choice. In federated training the checks need the inputs and the outputs of the local model, so they have to run on the clients, which can then average their estimates of λ_t before the server applies the rule.