capstract:2611.00068v1 [cs.LG] 20 Nov 2026

Thermal-State Thresholds for Early Stopping in Iterative Regression

Elira Tavrennic
Department of Computer Science
Brackenmere Institute for Statistical Learning

Abstract

We study whether the temperature of a cooling cup of tea can select a useful stopping time for gradient descent. A simulated cup follows Newtonian cooling, and training stops when its temperature crosses a calibrated threshold. For quadratic objectives, the normalized thermal trace exactly matches the optimization residual of a single Hessian eigendirection at a corresponding curvature. We also prove that, with fixed thermal conditions and update rate, the stopping rule is identical to a fixed computational budget. In synthetic linear regression, a threshold calibrated on 30 tasks attains mean excess prediction risk 0.416 on 100 new tasks from the same distribution, compared with 0.420 for per-task validation checkpoint selection and 0.964 at the final checkpoint. A matched clock reproduces the cup exactly. When the label-noise standard deviation doubles, cup risk increases to 1.181 versus 0.836 for validation. Altering the simulated cooling rate or update throughput changes the selected checkpoint. The results characterize thermal stopping as a transferable, externally timed regularizer and identify the distributional and timing assumptions on which it depends.

1 Introduction

Early stopping turns the duration of an optimization run into a regularization parameter. For least squares, the early iterates suppress directions that are learned slowly, often reducing the variance incurred by fitting noise (Ali et al., 2019; Raskutti et al., 2014). The difficulty is choosing when to stop. A validation set can estimate that point, but it consumes labeled examples and can be noisy. Rules based on training gradients avoid a validation set but still require access to the optimizer's internal state (Mahsereci et al., 2017).

We consider a stopping signal that requires neither access during the target run: the temperature of a cup of tea. At the first checkpoint for which the tea has cooled below a preset threshold, training stops. We study this rule in a controlled regression setting with a simulated cup, rather than claiming measurements from a physical vessel. The cup and the optimizer evolve on a shared clock; nothing is added to the training data or gradient.

The choice is motivated by a structural similarity. Temperature under Newtonian cooling decays exponentially toward ambient temperature. Each eigendirection of gradient descent on a quadratic loss also approaches its limiting value at a geometric rate. Under a suitable correspondence between cooling rate and curvature, a normalized temperature trace exactly tracks the remaining error in one eigendirection. The correspondence is exact for that direction and does not require a claim that tea responds to the data.

We calibrate one temperature threshold on 30 independent regression tasks and evaluate it on 100 new tasks at each of three noise levels. The calibrated cup attains excess prediction risk 0.416 on the matched distribution, against 0.420 for per-task validation checkpoint selection and 0.964 at the final checkpoint. Its risk is identical to a clock set to the same checkpoint. When the noise standard deviation doubles, the cup's risk rises to 1.181 while per-task validation yields 0.836. Changing

cooling rate or computational throughput also changes the selected checkpoint. The controlled setting therefore permits a precise account of both what the cup accomplishes and what information it lacks.

2 Related work

Early stopping is a form of iterative regularization: restricting optimization time can improve statistical performance even when the training objective continues to decrease. Ali et al. (2019) analyze gradient flow for least squares and its relation to ridge regression. Raskutti et al. (2014) study data-dependent stopping for nonparametric regression; Celisse and Wahl (2021) analyze discrepancy-based rules for spectral estimators. These results rely on properties of the learning problem, whereas the thermal rule below is independent of the target task once calibrated.

Validation-based procedures select or stop on estimates of generalization error; their operating characteristics depend on the criterion and evaluation frequency (Prechelt, 1998). Mahsereci et al. (2017) instead use statistics of stochastic gradients. Reusing a holdout for many adaptive choices can itself introduce selection effects (Feldman et al., 2019). Our cup uses validation data during calibration across tasks, but reads no target-task labels or gradients at deployment. The matched-clock control tests whether the thermal state provides any additional task-specific information.

3 Thermal stopping

3.1 Cooling model

Let T_0 be the initial cup temperature, T_a the ambient temperature, and $\tau > 0$ a cooling time constant in minutes. We model the cup by

$$T(t) = T_a + (T_0 - T_a)e^{-t/\tau}. \tag{1}$$

We set $T_0 = 90\,^\circ\mathrm{C}$, $T_a = 22\,^\circ\mathrm{C}$ and $\tau = 12$ minutes. These are assumed parameters of a simulation, not measurements of a particular tea, cup, room, or thermometer. A run proceeds at a simulated rate r updates per minute; the primary condition uses $r = 4$. At checkpoint k, the controller reads $T(k/r)$. For a threshold T_* between T_a and T_0, it selects the first checkpoint with $T(k/r) \le T_*$. If none crosses, it selects the final checkpoint.

The threshold is chosen once using other tasks, then held fixed across every evaluation condition. We call the procedure thermal stopping. It is validation-free only for the target run: the calibration procedure itself uses validation labels.

3.2 A spectral correspondence

Consider full-batch gradient descent on $L(w) = \|Xw - y\|^2/(2n)$, initialized at zero. Write $H = X^\top X/n$ and assume H is positive definite. Let $\hat{w} = H^{-1}X^\top y/n$ and let v_j be an eigenvector of H with eigenvalue λ_j. For a stable step size, the residual along this direction satisfies

$$v_j^\top(\hat{w} - w_k) = (1 - \eta\lambda_j)^k v_j^\top \hat{w}. \tag{2}$$

For $0 < \eta\lambda_j < 1$, the normalized residual equals the normalized temperature $(T(k/r) - T_a)/(T_0 - T_a)$ at every update when

$$\lambda_j = \frac{1 - e^{-1/(r\tau)}}{\eta}. \tag{3}$$

This follows by substitution in Eq. (2) and Eq. (1). With $\eta = 0.15$, $r = 4$ and $\tau = 12$, the matched curvature is approximately 0.137. Other eigenvalues have different decay rates; an entire multivariate loss curve cannot in general be represented by one cooling trace. The equality also concerns optimization error relative to the empirical solution, not prediction error relative to the unknown data-generating parameter.

3.3 Clock equivalence

The following observation holds for any fixed monotone cooling curve, irrespective of the model being trained.

Proposition 1 (Equivalence to a fixed budget)**.** *For a fixed temperature trace, threshold, rate and checkpoint schedule, thermal stopping selects the same checkpoint on every training problem. A clock that stops at this checkpoint returns exactly the same model as thermal stopping, provided the training trajectory is otherwise unchanged.*

Proof. The crossing index is a deterministic function of the trace, threshold and checkpoint schedule. It has no argument involving the data or model. Both procedures return the iterate at that index. The result also holds when a finite horizon is used as the fallback for a missing crossing. ∎

The proposition does not say a fixed budget is always poor. A budget selected on a representative distribution may perform well on new draws from that distribution. It says that the cup cannot adapt to a particular draw unless the thermal trace or threshold is itself made data-dependent. The equality provides an exact negative control for the experiments.

Before rounding to the checkpoint grid, the threshold corresponds to $k_* = r\tau \log[(T_0 - T_a)/(T_* - T_a)]$ updates. Its local sensitivity is $\partial k_*/\partial T_* = -r\tau/(T_* - T_a)$. For the primary parameters this is about -0.87 updates per degree Celsius at the selected threshold. The implemented controller instead changes in increments of five updates, so small threshold changes can have no effect until a checkpoint boundary is crossed. This distinction matters when comparing cups with different cooling constants.

4 Experimental setup

4.1 Data and optimization

Each task draws a coefficient vector $\beta \in \mathbb{R}^{32}$ with independent entries distributed as $N(0, 1/32)$. Training and validation sets each contain 64 independent examples. Features are independent standard Gaussian coordinates, and responses satisfy $y = x^\top \beta + \epsilon$, with $\epsilon \sim N(0, \sigma^2)$. All examples and noises are independently drawn within a task. We evaluate $\sigma \in \{0.5, 1, 2\}$; tasks with different noise levels reuse their random seeds for comparable underlying draws.

We run deterministic full-batch gradient descent at step size $\eta = 0.15$ from $w_0 = 0$ for 160 updates, recording checkpoints every five updates, including the initial model. The gradient is computed from the empirical Gram matrix and target vector. There is no explicit weight decay. For each checkpoint we evaluate the exact population excess squared prediction risk

$$R(w) = \mathbb{E}_x[(x^\top w - x^\top \beta)^2] = \|w - \beta\|_2^2, \tag{4}$$

where the expectation uses the known standard Gaussian feature distribution. This removes test-sample noise and excludes the irreducible response noise σ^2. The unknown β is used only for evaluation and the oracle comparator, never for cup calibration or validation selection.

4.2 Calibration and comparisons

We generate 30 calibration tasks with $\sigma = 1$ using seeds 0 through 29. For each checkpoint, we average the 30 validation mean squared errors and choose the checkpoint with the smallest average. We convert that checkpoint to its temperature under Eq. (1) and use it as T_*. Ties, if any, go to the earlier checkpoint. The resulting threshold is 77.212 °C, corresponding to update 10 or 2.5 simulated minutes. No evaluation task contributes to that selection.

For each noise level we then generate 100 evaluation tasks using seeds 1000 through 1099. We compare the cup against (i) a clock fixed at update 10, (ii) per-task selection of the lowest validation MSE over all 33 checkpoints, (iii) the first nonzero checkpoint, (iv) the final checkpoint, and (v) an oracle that chooses the minimum population excess risk on the same grid. The validation baseline is an offline checkpoint selector: it examines the complete trajectory and is not an online stopping rule. The oracle also examines the full trajectory and is an unattainable lower bound over this grid. The two fixed budgets are reference points, not tuned competitors.

We report task means and standard errors across 100 independent tasks per condition. For differences between the cup and validation, a paired percentile bootstrap resamples task pairs 2,000 times; intervals describe evaluation-task sampling variability conditional on the single calibration set. They do not include uncertainty from recalibrating the cup. Risk curves use the same tasks and are therefore correlated across checkpoints.

4.3 Thermal and throughput interventions

To isolate the clock mechanism, we hold the calibrated threshold fixed and vary the cooling constant τ or update rate r, halving or doubling each in turn. All other quantities, including the regression trajectories, stay fixed. This changes the number of updates completed before crossing but does not change the per-update learning dynamics. The simulation has no thermal measurement noise, transport delay, evaporation, reheating, or room-temperature drift.

We also recalibrate the threshold separately for $\sigma = 0.5$ and $\sigma = 2$, each time using 30 calibration tasks drawn from that condition and the same checkpoint-selection procedure. These diagnostic calibrations use prior knowledge of the new noise condition; they do not inspect the evaluation tasks. Their purpose is to distinguish a failure of the fixed threshold to transfer from a failure of the temperature parameterization to represent a better budget.

5 Results

5.1 Matched task distribution

Figure 1 shows the temperature trajectory alongside mean risk on the calibration distribution. The crossing occurs at update 10, near the bottom of the mean risk curve. Table 1 gives the evaluation results. On $\sigma = 1$ tasks, the cup achieves 0.416 ± 0.010 excess risk, compared with 0.420 ± 0.011 for validation and 0.392 ± 0.009 for the oracle. The paired difference between cup and validation is -0.0048, with a 95% bootstrap interval of $[-0.0172, 0.0075]$. This interval includes zero and does not establish that the methods are equivalent.

The final checkpoint has risk 0.964. The cup reduces risk by 56.9% relative to this checkpoint, an absolute reduction of 0.549. The matched clock agrees with the cup for every task, as required by Proposition 1. Per-task validation chooses a median of 10 updates but can vary across tasks. Its additional adaptivity does not yield a clear average advantage in this condition, where selecting among noisy validation estimates has a cost.

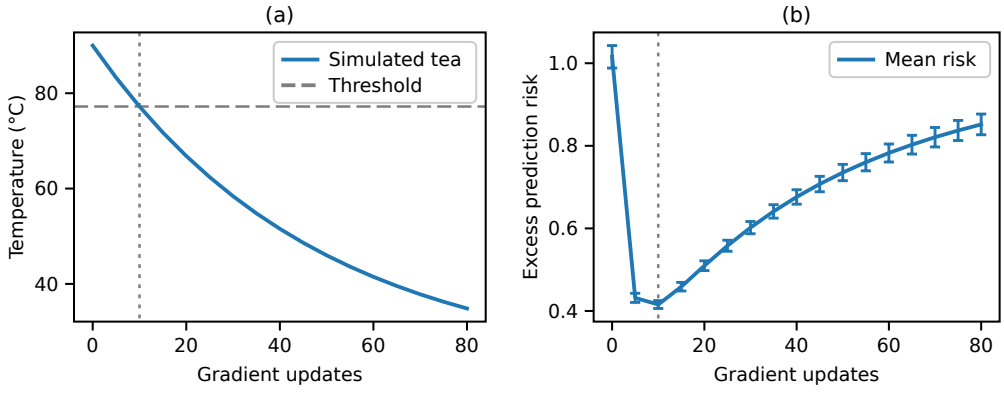


Figure 1: Simulated cooling and excess prediction risk for the $\sigma = 1$ tasks. The left panel shows the temperature at each checkpoint and the calibrated threshold. The right panel shows mean population excess risk over 100 evaluation tasks; error bars are one standard error. The vertical line marks the selected checkpoint.

Table 1: Population excess risk, mean $\pm$ standard error over 100 tasks. Cup threshold and matched clock are calibrated only at $\sigma = 1$. Validation and oracle select over the full checkpoint path. Lower is better.

Selector	$\sigma = 0.5$	$\sigma = 1$	$\sigma = 2$
Cup	0.227 ± 0.007	0.416 ± 0.010	1.181 ± 0.026
Matched clock	0.227 ± 0.007	0.416 ± 0.010	1.181 ± 0.026
Validation	0.172 ± 0.005	0.420 ± 0.011	0.836 ± 0.023
Fixed, update 5	0.343 ± 0.011	0.432 ± 0.011	0.795 ± 0.017
Fixed, update 160	0.241 ± 0.008	0.964 ± 0.032	3.860 ± 0.127
Oracle	0.163 ± 0.004	0.392 ± 0.009	0.760 ± 0.016

5.2 Transfer across noise levels

The same threshold is less suitable outside its calibration distribution. At $\sigma = 0.5$, risk continues to fall after update 10, and per-task validation selects a median of 25 updates. The cup exceeds validation by 0.055 excess-risk units, with a paired 95% interval of $[0.0428, 0.0680]$. At $\sigma = 2$, the median validation selection is update 5. The cup exceeds validation by 0.345, with interval $[0.2915, 0.3973]$. Figure 2 plots the mean risk paths. Under high noise, even update 5 is better than the adaptive validation selector on average; it was not chosen using these high-noise tasks and is included to show the dependence on the reference budget.

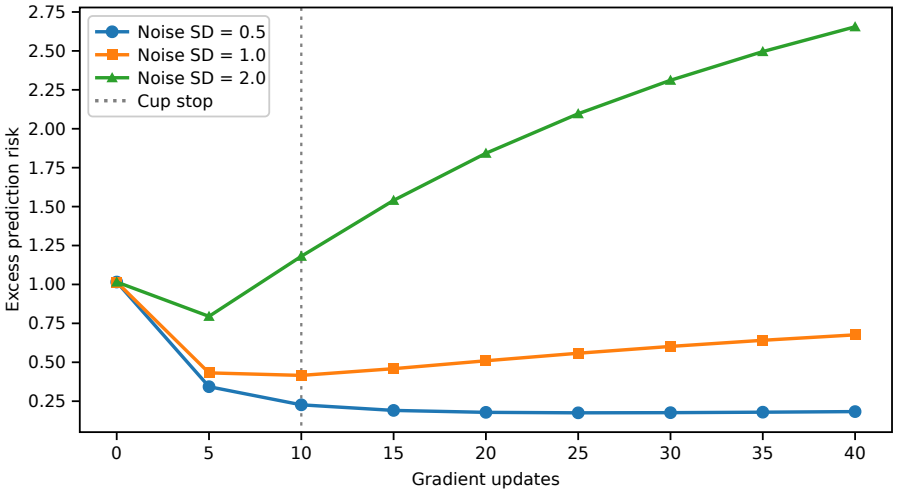


Figure 2: Mean population excess risk under three label-noise levels. The same calibrated threshold selects update 10 in every condition. Each curve averages 100 tasks and is shown through update 40 for readability.

When given calibration tasks from each new condition, the controller chooses a different temperature for each noise level (Table 2). At low noise the selected temperature falls to 58.398 °C, and at high noise it rises to 83.273 °C. The corresponding risks are 0.176 and 0.795. Thus a warmer stopping temperature accompanies more label noise in this setting. This relationship is induced by the calibration procedure: neither the temperature trace nor its crossing observes the labels of the evaluation task.

Table 2: Distribution-specific calibration using 30 separate tasks per noise level. The threshold is rounded for display. Risk is mean ± standard error over 100 evaluation tasks.

Noise SD	Threshold (°C)	Selected update	Excess risk
0.5	58.398	30	0.176 ± 0.004
1.0	77.212	10	0.416 ± 0.010
2.0	83.273	5	0.795 ± 0.017

5.3 Vessel and throughput sensitivity

Table 3 shows two ways to reach each of three update counts. Halving τ or r selects update 5, while doubling either selects update 20. The corresponding risk changes from 0.416 to 0.432 or 0.510. Because only the product $r\tau$ appears in the crossing condition, the paired changes produce exactly the same checkpoint in this simulation. The differences are properties of the time-to-update conversion, not evidence that the simulated beverage has observed the loss.

Table 3: Interventions at $\sigma = 1$, using the original threshold. Risk is mean ± standard error over the same 100 tasks. τ is in minutes and r in updates per minute.

τ	r	Selected update	Excess risk
6	4	5	0.432 ± 0.011
12	2	5	0.432 ± 0.011
12	4	10	0.416 ± 0.010
12	8	20	0.510 ± 0.012
24	4	20	0.510 ± 0.012

5.4 Calibration cost and compute

The main threshold requires 30 calibration trajectories, each with its own training and validation set. The two diagnostic recalibrations add 60 trajectories; the main evaluation comprises 300 additional trajectories. All 390 trajectories are calculated to update 160 so that each selector can be evaluated on the same path; this is 62,400 gradient updates. A real online cup-controlled run would stop at its crossing, but the validation comparator as implemented needs the complete path. The simulated four updates per minute defines the coupling to the cooling model and is not the measured execution speed. The complete experiment ran on one CPU in approximately seven seconds using Python's standard library; no accelerator was used.

6 Discussion

The central result is a control result. Under fixed thermal conditions, the cup implements a fixed budget, so its apparent validation-free performance comes from transferring a stopping choice across tasks. The threshold can be useful when the distribution and time-to-update mapping remain stable. The comparison to the clock prevents attributing that utility to a special property of the beverage. The single-mode correspondence explains why the two trajectories can have the same mathematical form; it does not make the cup a proxy for all modes or for generalization risk.

The primary calibration is substantial relative to a single target task. It uses 1,920 validation examples across 30 tasks and a grid of 33 candidate checkpoints; each diagnostic recalibration uses the same number of examples from its own condition. Per-task validation uses 64 examples on each evaluation task. The two procedures therefore differ in where their validation data are spent; the thermal controller should not be described as eliminating validation data in general. Its threshold of 77.212 °C is specific to the assumed parameters, checkpoint spacing, update rate and training distribution. Rounding the displayed value is for reporting; the simulation uses the unrounded threshold.

These experiments involve synthetic linear regression and an idealized cooling law. We have not measured a real cup, assessed different beverages or tested neural networks, variable step times, hardware contention, or noisy temperature sensing. A physical implementation would require measuring the temperature trace and synchronization error, and a model that learns from a changing distribution would need task-specific information to adapt. The oracle comparison is conditional on the same finite checkpoint grid and is not a bound on all possible estimators.

Reproducibility statement

The experiment uses the standard-library pseudorandom generator with the seeds listed in Section 4. A separate generator seeded 2701 produces the bootstrap resamples. All training and validation samples are generated afresh for each seed and noise condition. The full procedure, including curve aggregation and threshold selection, is specified by Eq. (1), Eq. (4), the update rule in Section 3, and the experimental settings above. No human participants or external datasets were used.

References

Alnur Ali, J. Zico Kolter, and Ryan J. Tibshirani. A Continuous-Time View of Early Stopping for Least Squares Regression. In *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89, pages 1370–1378, 2019.

Alain Celisse and Martin Wahl. Analyzing the Discrepancy Principle for Kernelized Spectral Filter Learning Algorithms. *Journal of Machine Learning Research*, 22(76):1–59, 2021.

Vitaly Feldman, Roy Frostig, and Moritz Hardt. The Advantages of Multiple Classes for Reducing Overfitting from Test Set Reuse. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pages 1892–1900, 2019.

Maren Mahsereci, Lukas Balles, Christoph Lassner, and Philipp Hennig. Early Stopping without a Validation Set. *arXiv preprint arXiv:1703.09580*, 2017.

Lutz Prechelt. Automatic Early Stopping Using Cross Validation: Quantifying the Criteria. *Neural Networks*, 11(4):761–767, 1998. doi: 10.1016/S0893-6080(98)00010-0.

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*, 15:335–366, 2014.