WHY SOME MODELS RESIST UNLEARNING: A LINEAR STABILITY PERSPECTIVE

Anonymous authors
Paper under double-blind review

ABSTRACT

Machine unlearning—the ability to erase the effect of specific training samples without retraining from scratch—is critical for privacy, regulation, and efficiency. However, most progress in unlearning has been empirical, with little theoretical understanding of when and why unlearning works. We tackle this gap by framing unlearning through the lens of asymptotic linear stability to capture the interaction between optimization dynamics and data geometry. The key quantity in our analysis is data coherence - the cross-sample alignment of loss-surface directions near the optimum. We decompose coherence along three axes: within the retain set, within the forget set, and between them, and prove tight stability thresholds that separate convergence from divergence. To further link data properties to forgettability, we study a two-layer ReLU CNN under a signal-plus-noise model and show that stronger memorization makes forgetting easier: when the signal-to-noise ratio (SNR) is lower, cross-sample alignment is weaker, reducing coherence and making unlearning easier; conversely, high-SNR, highly aligned models resist unlearning. For empirical verification, we show that Hessian tests and CNN heatmaps align closely with the predicted boundary, mapping the stability frontier of gradient-based unlearning as a function of batching, mixing, and data/model alignment. Our analysis is grounded in random matrix theory tools and provides the first principled account of the trade-offs between memorization, coherence, and unlearning.

1 INTRODUCTION

Machine unlearning – the ability to erase specific training samples' influence from a model – is critical for compliance, privacy, and model maintenance. Practically, retraining an N-sample model from scratch after removing even one sample incurs prohibitive cost, motivating a flurry of approximate unlearning methods. (Shen et al., 2024b;a; Hatua et al., 2024; Bourtoule et al., 2020; Cao & Yang, 2015; Golatkar et al., 2020; Ginart et al., 2019; Golatkar et al., 2021; Graves et al., 2020; Sekhari et al., 2021). However, despite this rapid progress, most unlearning work remains empirical and ad-hoc. We lack a unifying theoretical framework to predict when and why a given model can be efficiently unlearned. Notably, even initial theoretical treatments (e.g. from a differential-privacy viewpoint providing deletion guarantees (Sekhari et al., 2021; Chien et al., 2025)) do not explain the dynamics of forgetting or the interaction between the forget and retain sets. This gap motivates our work: we seek a principled understanding of the optimization dynamics of unlearning, grounded in the geometry of the model's loss landscape.

Our approach. We frame the unlearning process through the lens of asymptotic linear stability analysis in optimization. In this work, we analyze the asymptotic behavior of the stability of an unlearning solution, which fundamentally differs from the fine-tuning perspective (Ding et al., 2025) typically adopted in unlearning, and show that this stability is what ultimately determines whether a solution is unlearnable or not. Intuitively, unlike standard training which begins at random initialization, unlearning starts from a pre-trained model near a local minimum of the loss. We analyze small perturbation dynamics around that optimum to determine whether a "forgetting update" (e.g. gradient steps that increase loss on the forget set) will remain confined to a neighborhood of the original minimum (stable minima, no unlearning) or cause the model to drift off and diverge (unstable minima, unlearning possible). This linear stability perspective, inspired by prior analyses of SGD near minima (Ma & Ying, 2021; Dexter et al., 2024), provides a tractable characterization of the

transition between convergence vs. catastrophic forgetting. The key quantities in our analysis are a set of data coherence measures that quantify the alignment of loss gradients across samples near the optimum. We formally decompose coherence along three axes – (i) within the retain set, (ii) within the forget set, and (iii) between retain and forget sets – and derive stability thresholds in terms of these coherence values. Our theory thus for the first time links interactions between retain and forget data to unlearning success: for example, if the gradient directions of forget-set samples are highly aligned with those of retain-set samples (high retain-forget coherence), the model will resist unlearning because any parameter change that increases forget-set loss will also significantly hurt retain-set loss. Conversely, if the forget-set gradients live in a subspace largely independent from the retain-set (low inter-coherence), we prove the existence of a stable update direction that forgets the target data while leaving the rest of the model performance intact. These results yield a stability frontier in terms of data coherence: a boundary in data-geometry space separating regimes where gradient-based unlearning can succeed from where it fails.

Our framework also yields an intriguing insight into the relationship between training memorization and subsequent forgettability. Interestingly and perhaps surprisingly, we find that stronger memorization can make forgetting easier. In our framework, memorization corresponds to a regime of complex data where the model fits idiosyncratic details. We formalize this by adapting a two-layer ReLU CNN signal-plus-noise model from prior work on benign overfitting (Kou et al., 2023). Using random matrix theory tools, we prove that when the signal-to-noise ratio (SNR) in the data is lower (i.e. the model has to memorize more spurious noise), the cross-sample alignment of gradients is weaker – reducing the coherence terms, allowing effective forgetting of those samples. In contrast, a model trained on high-SNR data (with strongly aligned, dominant features) has very coherent gradients that push it to the edge of the stability frontier, making it resist unlearning – any attempt to forget one sample's influence will strongly perturb many others. We analytically identify this coherence-controlled stability boundary, and our experimental results confirm the trend: e.g. Hessian eigenvalue tests and CNN heatmaps of forget vs. retain influence align with the predicted boundary, mapping out how changes in batch size, mixing of forget/retain data, or network alignment affect the outcome of unlearning.

Contributions To summarize, our contributions are as follows: (1) We develop the first theoretical framework for machine unlearning based on linear stability analysis to address *what local optimization dynamics govern unlearning?*. We derive precise conditions (in terms of Hessian spectra and data coherence) under which standard gradient-based unlearning converges/diverges. (2) We address *how do the retain and forget sets interact, quantitatively, in determining stability?* Towards this goal, we introduce novel coherence metrics to quantify the retain–forget interaction, and prove how each coherence component (retain-retain, forget-forget, retain-forget) influences the stability of the unlearning process. These results formally characterize the joint role of data geometry and data distribution in forgetting dynamics. (3) To address *how does a model's propensity to memorize interact with its ability to forget?*, we establish a surprising link between memorization and forgettability: using a two-layer CNN with controllable noise, we prove that increased memorization (lower SNR) expands the range of stable unlearning (making forgetting easier), whereas high SNR (less overfitting) shrinks it (forgetting becomes harder). This is, to our knowledge, the first result to rigorously connect a model's generalization/memorization properties to its unlearning behavior. (4) Our empirical tests measure stability indicators (e.g. sharpness via Hessian eigenvalues) and unlearning performance under various conditions, and show strong agreement with the theoretical stability frontier. Taken together, our results provide the first principled account of the trade-offs between memorization, data coherence, and unlearning in modern ML models.

Related Work Prior works (Wu et al., 2022; 2018; Wu & Su, 2023) utilize the linear stability framework to understand the relation between converging-diverging boundary and alignment of noise and loss landscape. Furthermore, Wu et al. (2022); Wu & Su (2023) connect the alignment properties to the simplicity bias that occurs in generic SGD. Additionally, Ma & Ying (2021) extend the framework by incorporating higher-order moments of the noise, revealing subtle implicit regularization effects on parameter evolution. Dexter et al. (2024) introduced the notion of data coherence, which directly quantifies the alignment of sample-specific gradients in the loss landscape, offering a fine-grained tool to analyze sample interactions. Compared to alternative theoretical approaches, such as gradient flow or dynamical system approximations, linear stability has the distinct advantage of making explicit connections between model architecture, data distribution, and optimization algorithm. Unlike the standard learning scenario, unlearning requires analyzing

the interleaving interaction between retain and forget sets, which introduces new dynamics not present in classical stability analysis. To address these challenges, we introduce new analytical tools and definitions that generalize coherence to mixed retain–forget settings. In doing so, we not only provide stability criteria for unlearning but also establish the first formal connection between memorization and forgetting, thereby broadening the scope of linear stability analysis beyond its traditional application to standard training.

2 Theory

2.1 Background

Linear stability around minima. Linear stability provides a principled lens for analyzing the local dynamics of iterative optimization near a critical point (e.g., local minima or saddles) by linearizing the update map and studying the resulting linear time-varying system to characterize convergence/divergence behavior of stochastic iterative algorithms for that critical point. This perspective underlies modern convergence analyses of SGD and its variants and, more recently, has proved effective for characterizing generalization-relevant phenomena such as rapid escape from sharp minima (Wu et al., 2018; Dexter et al., 2024). In our context, we consider a loss $L(w) = \frac{1}{n}\sum_{i=1}^{n} \ell_i(w)$ for model parameters $w \in \mathbb{R}^d$. Let w^* be a local minimum. For a small perturbation δ around w^*, a first-order Taylor expansion gives

$$\nabla L(w^* + \delta) \approx \nabla^2 L(w^*)\, \delta,$$

since $\nabla L(w^*) = 0$. This linearization suggests that near w^*, the gradient is approximately given by the Hessian $H = \nabla^2 L(w^*)$ times the perturbation. Since we are only interested in the dyanamics of the optimizer (rather than its absolute position), without loss of generality we take $w^* = 0$ as in prior works.

Stochastic gradient updates. We are interested in the dynamics of stochastic gradient descent (SGD) near w^*. A generic SGD update can be written as

$$w_{t+1} = w_t - \eta\, g_t,$$

where $\eta > 0$ is the learning rate and g_t is the stochastic gradient at step t. In the neighborhood of w^*, using the linear approximation, we can write $g_t \approx H_t\, w_t$, where H_t is a random Hessian matrix (a mini-batch estimate of H). Thus the update becomes

$$w_{t+1} = w_t - \eta\, H_t\, w_t = (I - \eta H_t)\, w_t\,. \tag{1}$$

Here $H_t = \frac{1}{B}\sum_{i\in D_t} H_i$ is the average Hessian over the mini-batch D_t of size B, and $H_i = \nabla^2 \ell_i(w^*)$. By construction $H = \frac{1}{n}\sum_{i=1}^{n} H_i$. Following Dexter et al. (2024), we model mini-batch sampling via Bernoulli selection (each data point is included in the batch independently with probability B/n).

Unlearning update rule. To analyze machine unlearning (where a subset of the training data, called the *forget set*, is to be "forgotten" while the remaining data in the *retain set* is preserved), we adopt the update rule of Kurmanji et al. (2023). In this scheme, each step performs simultaneous gradient descent on the retain set and ascent on the forget set. Intuitively, this means we take a step that decreases loss on retained data while increasing loss on data that should be unlearned. Many gradient-based unlearning algorithms can be viewed as variants of this approach with different weighting of these components. We use n_f, f_r for number of forget and retain samples respectively. In our linearized framework, the update with forget importance hyper-parameter $\alpha \in [0, 1]$ is:

$$w_{k+1} = w_k - \eta\Big[(1-\alpha)\frac{1}{B}\sum_{i\in D_{r,k}} H_i\, w_k \;-\; \alpha\,\frac{1}{B}\sum_{i\in D_{f,k}} H_i\, w_k\Big], \tag{2}$$

where $D_{r,k}$ and $D_{f,k}$ denote the mini-batch of retain-set and forget-set examples at step k, respectively. This can be rewritten in operator form as

$$w_{k+1} = J_k\, w_k, \qquad J_k \;=\; I \;-\; \eta(1-\alpha)\frac{1}{B}\sum_{i\in D_{r,k}} H_i \;+\; \eta\alpha\,\frac{1}{B}\sum_{i\in D_{f,k}} H_i\,. \tag{3}$$

The random linear operator J_k captures the combined effect of the retain and forget gradients at step k. Note that J_k is itself random due to sampling of a mini-batch from each set.

A central question in linear stability analysis is whether the iterates remain near the original optimum or diverge away. To quantify this, we examine the expected squared norm of the parameters after k steps, $\mathbb{E}\|w_k\|^2 = \mathbb{E}[w_k^T w_k]$. Starting from an isotropic small perturbation w_0 (we assume $w_0 \sim \mathcal{N}(0, I)$ without loss of generality), one can expand $w_k = J_{k-1} \cdots J_0 \, w_0$. This yields

$$\mathbb{E}\|w_k\|^2 \;=\; \mathbb{E}[w_0^T (J_0^T \cdots J_{k-1}^T J_{k-1} \cdots J_0) w_0] \;=\; \mathbb{E}\,\mathrm{Tr}\big(J_{k-1} \cdots J_0 J_0^T \cdots J_{k-1}^T\big)\,, \tag{4}$$

where we used $\mathbb{E}[w_0 w_0^T] = I$ in the final equality. Eq (4) is the key quantity we will analyze to determine stability: if $\mathbb{E}\|w_k\|^2$ remains bounded (or decays) as $k \to \infty$, the unlearning process is *stable* (convergent) around w^*, whereas if $\mathbb{E}\|w_k\|^2 \to \infty$, the process is *unstable* (diverges, escaping w^*).

2.2 Coherence Measures for Unlearning

Coherence in single-dataset SGD. Before introducing our new coherence measures tailored to unlearning, we briefly review the original notion of *Hessian coherence* from Dexter et al. (2024) for standard (single dataset) learning. The coherence quantifies the alignment between per-sample Hessians in the training set. Intuitively, if all samples induce very aligned curvature directions, SGD will experience less "randomness" and more stable updates, whereas if each sample's loss landscape is oriented differently, the optimization dynamics are more erratic.

Definition 1 (Coherence, single set (Dexter et al., 2024))**.** *Given a collection of positive semidefinite (PSD) Hessian matrices $\{H_i : i \in [n]\}$ for n training samples, the* coherence matrix $S \in \mathbb{R}^{n \times n}$ *is defined by*

$$S_{ij}^{\textit{single}} \;=\; \| \, H_i^{1/2} \, H_j^{1/2} \, \|_F \,,$$

the Frobenius norm of the product of the square-root Hessians of sample i and j. The associated coherence measure *is*

$$\sigma^{\textit{single}} \;=\; \frac{\lambda_{\max}(S^{\textit{single}})}{\max_{i \in [n]} \lambda_{\max}(H_i)} \,,$$

i.e. the largest eigenvalue of $S^{\textit{single}}$ normalized by the largest individual sample Hessian eigenvalue.

Intuitively, σ^{single} close to 1 indicates that the top curvature directions of all samples are closely aligned (high coherence), whereas a small σ^{single} indicates disparate or orthogonal curvatures across samples. Prior work showed that higher coherence σ^{single} correlates with greater stability of SGD. In other words, when the loss landscapes of different samples "point" in similar directions, gradient steps reinforce each other and it is harder for SGD to diverge away from the optimum.

Coherence with retain and forget sets. In an unlearning scenario, we have two disjoint sets of samples: the retain set D_r (of size n_r) and the forget set D_f (of size n_f). Coherence within each set (retain vs. forget) is not sufficient to describe the behavior of the combined ascent-descent dynamics. We need to also quantify the interaction *between* the two sets. We therefore introduce a series of definitions that extend coherence to the multi-set setting.

First, we define a weighted combination of Hessians from a retain-forget pair, which will serve as an effective "mixing Hessian:"

Definition 2 (Mix-Hessian)**.**

$$D \coloneqq \frac{1}{n_r n_f} \sum_{r \in D_r, f \in D_f} \frac{C_r^{\frac{1}{2}}}{C_r^{\frac{1}{2}} + C_f^{\frac{1}{2}}} H_r + \frac{C_f^{\frac{1}{2}}}{C_r^{\frac{1}{2}} + C_f^{\frac{1}{2}}} H_f = \frac{1}{n_r n_f} \sum_{rf} D_{rf}, \tag{5}$$

Here $C_r = \eta^2 (1-\alpha)^2 \frac{1}{n_r}(\frac{1}{B} - \frac{1}{n_r})$, $C_f = \eta^2 \alpha^2 \frac{1}{n_f}(\frac{1}{B} - \frac{1}{n_f})$, $D_{rf} = \frac{C_r^{\frac{1}{2}}}{C_r^{\frac{1}{2}} + C_f^{\frac{1}{2}}} H_r + \frac{C_f^{\frac{1}{2}}}{C_r^{\frac{1}{2}} + C_f^{\frac{1}{2}}} H_f$. The constants C_r and C_f reflect the relative contribution of retain vs forget Hessians to the second-moment dynamics of w_k (these arise from the SGD noise analysis in Lemma 2.1 later). The mix-Hessian D aggregates the pairwise influence of retain and forget sets; it effectively summarizes how the two sets jointly affect curvature when considered together in the update. Next, analogous to the single-set case, we define a coherence matrix that captures alignment across *pairs* of retain/forget examples:

Definition 3 (Mix-coherence matrix)**.** *Construct an index set for all retain–forget pairs:* $\mathcal{P} = \{(r, f) : r \in D_r,\ f \in D_f\}$ *of size* $|\mathcal{P}| = n_r n_f$*. The* mix-coherence matrix $S \in \mathbb{R}^{|\mathcal{P}|\times|\mathcal{P}|}$ *is defined entrywise by*

$$S_{(r,f),(r',f')} \;=\; \big\| D_{rf}^{1/2}\, D_{r'f'}^{1/2} \big\|_F \,,$$

for any $(r, f), (r', f') \in \mathcal{P}$*.*

In words, S measures the alignment between every pair of mixed Hessians D_{rf} and $D_{r'f'}$. Finally, we define an overall coherence measure for unlearning, generalizing the single-set σ:

Definition 4 (Unlearning Coherence Measure)**.** *The unlearning coherence is*

$$\sigma \;=\; \frac{\lambda_{\max}(S)}{\max_{(r,f)\in\mathcal{P}}\ \lambda_{\max}(D_{rf})} \,,$$

the leading eigenvalue of the mix-coherence matrix S *normalized by the largest eigenvalue among all individual* D_{rf} *matrices.*

This definition reduces to the original coherence measure in the limit where one of the sets is absent (e.g. $n_f = 0$ or $\alpha = 0$ yields only a retain set). It simultaneously captures the within-set coherences and the cross-set coupling. Intuitively, if the retain and forget sets are highly *aligned* in terms of curvature directions, the mix-coherence σ will be large. In that case, performing ascent on forget and descent on retain will tend to cancel out: the update directions from the two sets are similar but with opposite sign, leading to minimal movement away from w^*. This predicts stability for the current optimum (i.e. resistance to unlearning). Conversely, if the two sets are incoherent (small σ), their Hessians push in different directions; the ascent on forget set will not be canceled by descent on retain, making it easier for the iterates w_k to escape the original minimum. In summary, our multi-set coherence measure σ quantifies how conducive the data geometry is to either divergence or convergence during unlearning. To our knowledge, this is the first work to explicitly incorporate multiple data subsets into a stability analysis of optimization.

2.3 Linear Stability Analysis of Unlearning

In the theory section, we study a more fundamental problem regarding whether a set is unlearnable in asymptotic manner through optimization instead of in a fine-tuning regime. Ding et al. (2025). We now leverage the above framework to derive conditions under which the unlearning dynamics (2) will converge or diverge. In our work, A key technical challenge is that, unlike in the single-set case, the influence of SGD noise cannot be captured by a simple closed-form recursion. This is because the gradient noise now comes from two interleaved sources (retain and forget sets) with potentially different magnitudes.

We begin with a lemma that describes the evolution of the second moment $\mathbb{E}\|w_k\|^2$ in terms of a recursive sequence of matrices N_k. This lemma generalizes the stability condition from Dexter et al. (2024) to account for the alternating ascent/descent updates.

Lemma 2.1 (Stability recurrence for unlearning)**.** *Consider the unlearning update operator* J_k *defined in* (3)*. Define a sequence of PSD matrices* $\{N_k\}_{k\geq 0}$ *by* $N_0 = I$ *and for* $k \geq 1$*:*

$$N_k \;=\; C_f \sum_{i\in D_f} H_i\, N_{k-1}\, H_i \;+\; C_r \sum_{i\in D_r} H_i\, N_{k-1}\, H_i \,, \tag{6}$$

with C_r, C_f *as given in Definition 2. Also let* $M_k = J^{2k} + N_k$*. (*$J = I - \eta(1-\alpha)H_R + \eta\alpha H_F$ *where* H_R *and* H_F *ar full Hessian of retain and forget set. See definition 6) Then:*

1. ***(Lower bound)*** $\mathbb{E}\,\mathrm{Tr}\big(J_0^T \cdots J_{k-1}^T J_{k-1} \cdots J_0\big) \;\geq\; \mathrm{Tr}(M_k)$*. Moreover, if* $\mathrm{Tr}(N_k) \to \infty$ *as* $k \to \infty$*, then* $\mathbb{E}\|w_k\|^2 \to \infty$ *as well.*

2. ***(Upper bound)*** *If at each step* J_k *is spectrally bounded as* $(1-\epsilon)I \succeq J \succeq -(1-\epsilon)I$ *for some* $\epsilon \in (0, 1)$ *(i.e. all eigenvalues of* J *lie in* $[-(1-\epsilon),\, 1-\epsilon]$*), then*

$$\mathbb{E}\,\mathrm{Tr}\big(J_0^T \cdots J_{k-1}^T J_{k-1} \cdots J_0\big) \;\leq\; \sum_{r=0}^{k-1} \binom{k}{r} (1-\epsilon)^{2(k-r)}\ \mathrm{Tr}(N_r)\,.$$

   If in addition $\mathrm{Tr}(N_r) \leq \epsilon$ *for all* r*, then* $\mathbb{E}\|w_k\|^2 \to 0$ *as* $k \to \infty$ *(the unlearning update converges in mean square).*

Discussion. Part (1) of Lemma 2.1 provides a sufficient condition for divergence: if the "noise accumulation" matrices N_k (which capture how SGD variance builds up over iterations) have unbounded trace, then the model will eventually blow up (escape the optimum). Part (2) gives a sufficient condition for convergence: if each J is a contraction (spectral norm < 1 by a margin ϵ) and the accumulated noise remains small, then the model's parameter norm will vanish (meaning w_k returns to the optimum). These statements generalize classical stability results to the unlearning case. Importantly, the recursion (6) for N_k does not admit a simple closed form because N_{k-1} appears inside sums over both sets D_r and D_f. This coupling between retain and forget sets is what makes analyzing unlearning challenging. By introducing the coherence measures (Definition 4 and related definition), we overcome this hurdle: the coherence will allow us to relate $\mathrm{Tr}(N_k)$ to data-dependent quantities like $\lambda_{\max}(D)$ and thereby derive interpretable stability criteria.

Using the coherence framework, we can now state our main stability thresholds. The first result is a condition under which the unlearning dynamics *diverge* (fail to stay at the original minimum):

Theorem 2.2 (Divergence criterion for unlearning)**.** *Under the setup of Lemma 2.1, the unlearning process will diverge if the mix-Hessian eigenvalue exceeds a threshold determined by the coherence. In particular, if*

$$\lambda_{\max}(D) \;\geq\; \frac{\sqrt{2}\,\sigma}{\eta\Big((1-\alpha)\,n_f\,\sqrt{\frac{n_r}{B}-1}\;+\;\alpha\,n_r\,\sqrt{\frac{n_f}{B}-1}\Big)}\,, \tag{7}$$

then $\lim_{k\to\infty}\mathbb{E}\|w_k\|^2 = \infty$. *Equivalently, condition* (7) *guarantees the unlearning algorithm will* escape *the original minima (diverge) due to the stochastic dynamics.*

In plain terms, Theorem 2.2 says that if the influence of the forget–retain interaction (measured by $\lambda_{\max}(D)$) is sufficiently large relative to the stabilizing effect of coherence σ (and other factors like batch size B and relative set sizes), then the gradient ascent on the forget set will overpower the descent on the retain set, leading to instability. The inequality (7) can be viewed as a quantitative stability limit or "edge of chaos" for unlearning: beyond this point, the original solution w^* cannot hold.

Our next theorem establishes a matching lower bound, showing that the above divergence condition is essentially tight. It guarantees that when $\lambda_{\max}(D)$ is below a certain threshold (of the same order as in (7)), one can find a scenario where the unlearning process converges, thereby demonstrating that the threshold cannot be significantly improved in general:

Theorem 2.3 (Convergence condition (matching lower bound))**.** *Suppose* $\lambda_{\max}(D)$ *and* σ *satisfy*

$$\lambda_{\max}(D) \;\leq\; \frac{2\,\sigma}{\eta\,C_r'\,\big(\sigma + n_f\big(\frac{n_r}{B}-1\big)\big)}\,, \tag{8}$$

where $C_r' = \sqrt{C_r}/(\sqrt{C_r}+\sqrt{C_f})$ *(with* C_r, C_f *from Definition 2). Then there exists a choice of PSD Hessians* $\{H_i\}$ *for the retain and forget sets such that the unlearning update converges (i.e.* $\lim_{k\to\infty}\mathbb{E}\|w_k\|^2 = 0$*) under those Hessians.*

The convergence condition (8) mirrors the divergence condition in its dependence on σ, n_r, n_f, and B. The existence of a construction that achieves convergence when (8) holds indicates that our divergence criterion in Theorem 2.2 is tight up to constant factors. In summary, Theorems 2.2 and 2.3 together pin down a theoretical threshold curve in the space of data coherence and algorithm parameters that separates stable (convergent) unlearning from unstable (divergent) unlearning.

Implications for unlearning methods. The above results shed light on why certain unlearning methods succeed or fail, by interpreting them through the lens of coherence and the weighting imbalance between retain and forget sets.

We can now interpret some common unlearning strategies:

Naive negative gradient. A straightforward unlearning baseline is to set $\alpha = 1$ and run gradient ascent on the forget set alone. Our framework explains why this often fails. If the forget set has high internal coherence, its gradients align with the curvature at w^*, so ascent follows a single stable direction and does not escape the minimum due to lack of stability. If the forget and retain sets are

also highly coherent, the overall coherence stays large even without the retain set. In both cases divergence is inhibited, matching empirical reports that naive negative-gradient unlearning typically stagnates or oscillates, hurting retained data while barely reducing forget-set performance (Ding et al., 2025; Fan et al., 2025; Ding et al., 2025).

Random label perturbation. Another strategy is to add randomness to the forgetting process, for instance by using mislabeled data or injecting noise into the forget set's gradients (see, e.g., random label unlearning). In our terms, this deliberately *breaks the coherence* of the forget set: if labels are randomized, the gradients from forget-set samples become effectively uncorrelated, dramatically lowering the forget-set's internal σ. This, in turn, allows the model to escape the original minimum much faster, because the forget-set ascent directions will fluctuate rather than consistently opposing the retain-set descent. Moreover, randomizing forget labels also reduces the coupling between forget and retain sets (since the forget-set gradient is now essentially random noise orthogonal to the retain-set Hessians). Thus, random label methods improve unlearning by driving the coherence measure σ downward, so the divergence criterion is more easily satisfied. (Graves et al., 2020)

Min-Max (targeted forget) methods. More sophisticated approaches pick a subset of model weights or directions that are most "responsible" for the forget set's performance, and then apply ascent/descent on those components. This can be seen as applying projection matrices P_F and P_R to the Hessians H_f, H_r respectively, focusing updates on certain eigen-directions. Such projections effectively reduce the overlap between forget-set and retain-set update directions (since $P_F H_f$ and $P_R H_r$ act in different subspaces), thereby reducing the cross-coherence between the two sets. In our framework, this corresponds to a smaller overall σ as well. By isolating the forgetting dynamics, Min-Max methods thus decrease the ability of the retain set to interfere with forgetting (and vice versa), making the unlearning process more effective. (Tang & Khanna, 2025; Fan et al., 2024)

2.4 Memorization and Forgetting

So far, our analysis has focused on the role of stochastic gradient noise (from mini-batch sampling). We now turn to another key factor: the inherent *signal vs. noise* structure of the data itself. We ask: if a model has *memorized* certain training examples (as opposed to learning a shared signal), does that make it easier or harder to forget those examples? We will show a theoretical connection between a model's tendency to memorize (which occurs when data has low signal-to-noise ratio) and the ease of unlearning. To make this concrete, we consider a specific data model and network, inspired by the theoretical construction by Kou et al. (2023). The data distribution is designed so that each example contains a mixture of a common signal and independent noise. This is formalized as follows:

Definition 5 (Data Setup)**.** *Let $\mu \in \mathbb{R}^d$ be a fixed unit-norm signal vector. Each training example consists of a feature pair $x = [x^{(1)}; x^{(2)}] \in \mathbb{R}^{2d}$ (concatenation of two d-dimensional parts) and a label $y \in \{-1, +1\}$. The example is generated by:*

1. *Sample y as a Rademacher random variable ($\Pr(y = +1) = \Pr(y = -1) = \frac{1}{2}$).*
2. *Sample a noise vector $\xi \sim \mathcal{N}(0, \sigma^2 I_d)$ in $\mathbb{R}^d$, where σ^2 is the noise variance.*
3. *With equal probability, set either $x^{(1)} = y\,\mu$ and $x^{(2)} = \xi$, or $x^{(1)} = \xi$ and $x^{(2)} = y\,\mu$. In other words, one of the two halves of x carries the signal $y\,\mu$ and the other carries independent noise.*

We then consider a two-layer convolutional neural network (CNN) with ReLU activations operating on this data.. The network has two sets of convolutional filters (for the positive and negative class) and outputs a score $f(W, x)$ whose sign determines the predicted label. Specifically, let $W^{(+1)}$ and $W^{(-1)}$ be the weight matrices for the two classes, each of shape $m \times d$ (with m filters). The network's output is

$$\begin{aligned} f(W, x) \;=\; & \frac{1}{m} \sum_{r=1}^{m} \Big(\mathrm{ReLU}(\langle w_r^{(+1)}, x^{(1)} \rangle) + \mathrm{ReLU}(\langle w_r^{(+1)}, x^{(2)} \rangle) \Big) \\ & - \frac{1}{m} \sum_{r=1}^{m} \Big(\mathrm{ReLU}(\langle w_r^{(-1)}, x^{(1)} \rangle) + \mathrm{ReLU}(\langle w_r^{(-1)}, x^{(2)} \rangle) \Big), \end{aligned} \tag{9}$$

and the model is trained with logistic loss $L_S(W) = \frac{1}{n}\sum_{i=1}^{n} \log\big(1 + \exp(-y_i\, f(W, x_i))\big)$. We focus on the case where the network can fit the training data perfectly (interpolating regime) and potentially overfits.

In this setting, we can analyze the coherence of the Hessians at the trained solution. The following result provides an upper bound on the coherence in terms of the *signal-to-noise ratio* (SNR) of the data, defined as $\mathrm{SNR} = \frac{\|\mu\|}{\sigma\sqrt{d}}$ (which measures the strength of the common signal relative to noise in each example):

Theorem 2.4 (Coherence bound in the CNN memorization model)**.** *Under the data model of Definition 5 and the two-layer ReLU CNN defined above, suppose the network is trained to near-zero training loss. Then with probability at least $1 - 8\delta$ (over the random draw of the dataset), the largest eigenvalue of the coherence matrix S for the retain/forget split satisfies*

$$\lambda_{\max}(S) \;\le\; \mathcal{O}\Big(n_r\, n_f\, d\sigma^2\Big[(\sqrt{C_r'} + \sqrt{C_f'})^2\,(\mathrm{SNR})^2 + (C_r' + C_f')\Big]\Big)\,, \tag{10}$$

$$\max_{rf} \lambda_{\max}(D_{rf}) \le \mathcal{O}((C_r' + C_f')(d\sigma^2(\mathrm{SNR})^2 + 1)), \tag{11}$$

where C_r' and C_f' are the normalized retain/forget weight fractions as defined in Theorem 2.3. Consider division of two quantities and we can find that for small SNR limit and large SNR limit:

$$\lim_{SNR\to 0} \frac{\lambda_{\max}(S)^{upper}}{\max_{rf} D_{rf}^{upper}} = \mathcal{O}(n_r n_f)\;\;, \lim_{SNR\to\infty} \frac{\lambda_{\max}(S)^{upper}}{\max_{rf} D_{rf}^{upper}} = \mathcal{O}(n_r n_f(1 + \frac{2\sqrt{C_r' C_f'}}{C_r' + C_f'})). \tag{12}$$

Discussion Theorem 2.4 shows the surprising role of SNR in stability of the optimizer through its control over the coherence. In particular, if the data has a very low SNR (meaning μ is small relative to the noise σ), then the network is likely to memorize the noise. In that regime, high-dimensional random noise vectors are nearly orthogonal to each other, so Hessians for different samples align poorly. Our bound indicates that coherence measure is larger in large SNR limit compared to small SNR limit, so a smaller SNR yields a smaller coherence. Consequently, when the model has memorized (low SNR), the unlearning process becomes easier: the model can move away from the original fit with less resistance, as formalized by our earlier stability criteria. Conversely, if the data has high SNR (dominant signal shared across examples), the model will latch onto that signal, resulting in large coherence, and the unlearning process will be much harder (the model resists leaving the optimum since all samples agree on the direction).

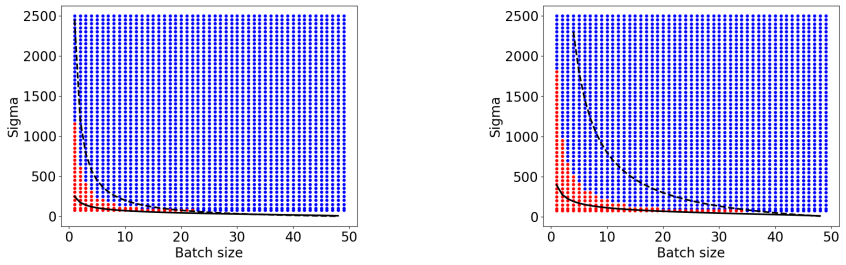
This gives a rigorous basis for a perhaps counter-intuitive aphorism: *the more you memorize, the easier you forget*. In other words, models that rely heavily on idiosyncratic features of individual data points (memorization) are in fact less stable at those points and can forget them with less effort, whereas models that have learned a strong global structure (signal) are more stable and resistant to having a single sample's influence removed. Our work is the first to formally establish this connection between memorization (in terms of data geometry) and unlearning. We believe this provides valuable insight into the trade-offs inherent in machine unlearning.

3 Experiments

3.1 Diverging and converging condition.

Experimental setup. In this section, we simulate experiments to test Theorems 2.2 and 2.3. We fix $n_f = n_r = 50$ and set $\alpha = 0.1$. Say Q is a hyper-parameter constant. We will set Q to different values to control various quantities in the experiments. For the retain set, Hessians are defined as $H_i = me_1e_1^T$ for $i \in [Q]$, and $H_i = me_{i-Q+1}e_{i-Q+1}^T$ otherwise, with $m = 2n_r/Q$; the forget set uses the same construction. This ensures $\lambda_1(H_R) = \lambda_1(H_F) = \lambda_1(D) = 2$, controlling sharpness. We choose $\eta \le 1$ to avoid divergence from the standard criterion $\eta \ge 2/\lambda_1$, so any escaping behavior stems solely from stochasticity, consistent with our theorem.

To vary coherence, we change Q and compute (B, σ) pairs by adjusting batch size. For each pair, we randomly initialize w, run 1000 updates, and record $\|w_{1000}\|$. Runs with $\|w_{1000}\|/\|w_0\| \ge 1000$ are marked as diverging. Each experiment is repeated 10 times, and the majority outcome determines convergence/divergence.



(a) Learning rate $\eta = 0.5$ (b) Learning rate $\eta = 0.8$

Figure 1: **Tight upper and lower bounds.** Blue = convergence, red = divergence. The dashed line is the lower bound (Theorem 2.3), the solid line the divergence criterion (Theorem 2.2). Both closely track the true boundary.

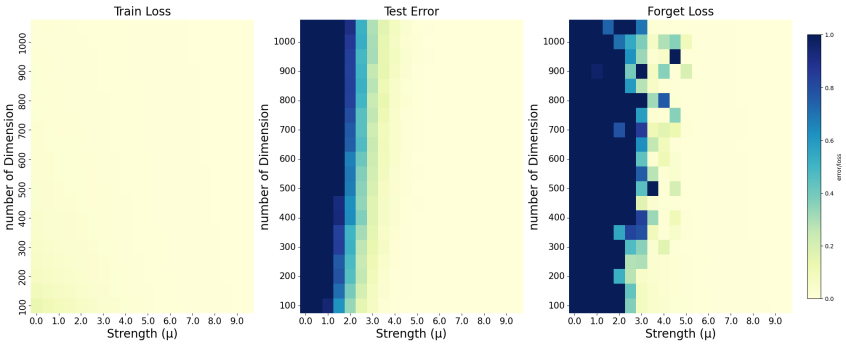


Figure 2: **(Left)** Training loss. **(Middle)** Test error. **(Right)** Forget loss. Memorization and forgetting regions strongly overlap: the left and middle panels show training loss and test error, while the right shows forget loss under different combinations of dimension and signal strength.

Bounds on divergence. Figure 1 shows that both our upper and lower bounds predict the divergence region accurately; the bounds are tighter for batch sizes ≥ 10. The divergence criterion in particular matches the true boundary, demonstrating that our coherence-based measure captures the essential optimization dynamics accurately. This highlights coherence as a meaningful lens on unlearning dynamics, with potential applications beyond our scope.

3.2 Relation between memorization and forgetting.

Experimental setup. We generate data as in Definition 5 along with the 2 layer CNN. The dataset has 50 training samples without label noise. We set $\boldsymbol{\mu} = \|\boldsymbol{\mu}\|_2[1, 0, \ldots, 0]$ and add Gaussian noise $\xi \sim \mathcal{N}(0, \sigma^2 I_d)$ with $\sigma = 1.0$. We vary the value of $d \in [100, 1100]$ to verify our results with varying levels of over-parametrization. The CNN has $m = 10$ filters and is trained by full-batch gradient descent for 100 epochs at learning rate 0.1, ensuring training loss ≤ 0.1.

We record training loss and test error (on 1000 unseen samples). For unlearning, out of total 50 samples, we use 25 samples to form the forget set and the other 25 to build the retain set. We apply mini-batch unlearning (batch size 5) using the negative-gradient method with learning rate 0.1 and $\alpha = 0.3$ for 90 steps, then record the average forget loss. Each experiment is repeated 20 times and averaged.

Memorization–forgetting overlap. Figure 2 shows heatmaps over signal strength and dimension. Memorization is identified where training loss is low but test error high (left, middle). Strikingly, these regions coincide with high loss on the forget (right), confirming our prediction: memorization corresponds to low coherence, making solutions unstable and easier to escape, thus making unlearning easier. This provides strong evidence for our framework and, to our knowledge, is the first work to connect memorization and forgetting through coherence.

REFERENCES

Idan Attias, Gintare Karolina Dziugaite, Mahdi Haghifam, Roi Livni, and Daniel M. Roy. Information complexity of stochastic convex optimization: Applications to generalization and memorization, 2024. URL https://arxiv.org/abs/2402.09327.

Rajendra Bhatia. *Matrix analysis*, volume 169. Springer Science & Business Media, 2013.

Stella Biderman, USVSN Sai Prashanth, Lintang Sutawika, Hailey Schoelkopf, Quentin Anthony, Shivanshu Purohit, and Edward Raff. Emergent and predictable memorization in large language models, 2023. URL https://arxiv.org/abs/2304.11158.

Lucas Bourtoule, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning, 2020. URL https://arxiv.org/abs/1912.03817.

Yinzhi Cao and Junfeng Yang. Towards making systems forget with machine unlearning. *2015 IEEE Symposium on Security and Privacy*, pp. 463–480, 2015. URL https://api.semanticscholar.org/CorpusID:5945696.

Nicholas Carlini, Úlfar Erlingsson, and Nicolas Papernot. Distribution density, tails, and outliers in machine learning: Metrics and applications, 2019. URL https://arxiv.org/abs/1910.13427.

Eli Chien, Haoyu Wang, Ziang Chen, and Pan Li. Langevin unlearning: A new perspective of noisy gradient descent for machine unlearning, 2025. URL https://arxiv.org/abs/2401.10371.

Gregory Dexter, Borja Ocejo, Sathiya Keerthi, Aman Gupta, Ayan Acharya, and Rajiv Khanna. A precise characterization of sgd stability using loss surface geometry, 2024. URL https://arxiv.org/abs/2401.12332.

Meng Ding, Rohan Sharma, Changyou Chen, Jinhui Xu, and Kaiyi Ji. Understanding fine-tuning in approximate unlearning: A theoretical perspective, 2025. URL https://arxiv.org/abs/2410.03833.

Chongyu Fan, Jiancheng Liu, Yihua Zhang, Eric Wong, Dennis Wei, and Sijia Liu. Salun: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation, 2024. URL https://arxiv.org/abs/2310.12508.

Xindi Fan, Jing Wu, Mingyi Zhou, Pengwei Liang, and Dinh Phung. Imu: Influence-guided machine unlearning, 2025. URL https://arxiv.org/abs/2508.01620.

Vitaly Feldman. Does learning require memorization? a short tale about a long tail, 2021. URL https://arxiv.org/abs/1906.05271.

Antonio Ginart, Melody Y. Guan, Gregory Valiant, and James Zou. Making ai forget you: Data deletion in machine learning, 2019. URL https://arxiv.org/abs/1907.05012.

Aditya Golatkar, Alessandro Achille, and Stefano Soatto. Eternal sunshine of the spotless net: Selective forgetting in deep networks, 2020. URL https://arxiv.org/abs/1911.04933.

Aditya Golatkar, Alessandro Achille, Avinash Ravichandran, Marzia Polito, and Stefano Soatto. Mixed-privacy forgetting in deep networks, 2021. URL https://arxiv.org/abs/2012.13431.

Laura Graves, Vineel Nagisetty, and Vijay Ganesh. Amnesiac machine learning, 2020. URL https://arxiv.org/abs/2010.10981.

Amartya Hatua, Trung T. Nguyen, Filip Cano, and Andrew H. Sung. Machine unlearning using forgetting neural networks, 2024. URL https://arxiv.org/abs/2410.22374.

Zachary Izzo, Mary Anne Smart, Kamalika Chaudhuri, and James Zou. Approximate data deletion from machine learning models, 2021. URL https://arxiv.org/abs/2002.10077.

Yiwen Kou, Zixiang Chen, Yuanzhou Chen, and Quanquan Gu. Benign overfitting for two-layer relu convolutional neural networks, 2023. URL https://arxiv.org/abs/2303.04145.

Meghdad Kurmanji, Peter Triantafillou, Jamie Hayes, and Eleni Triantafillou. Towards unbounded machine unlearning, 2023. URL https://arxiv.org/abs/2302.09880.

Hao Li, Di Huang, Ziyu Wang, and Amir M. Rahmani. Skewed memorization in large language models: Quantification and decomposition, 2025. URL https://arxiv.org/abs/2502.01187.

Chao Ma and Lexing Ying. On linear stability of sgd and input-smoothness of neural networks, 2021. URL https://arxiv.org/abs/2105.13462.

USVSN Sai Prashanth, Alvin Deng, Kyle O'Brien, Jyothir S V, Mohammad Aflah Khan, Jaydeep Borkar, Christopher A. Choquette-Choo, Jacob Ray Fuehne, Stella Biderman, Tracy Ke, Katherine Lee, and Naomi Saphra. Recite, reconstruct, recollect: Memorization in lms as a multifaceted phenomenon, 2025. URL https://arxiv.org/abs/2406.17746.

Ayush Sekhari, Jayadev Acharya, Gautam Kamath, and Ananda Theertha Suresh. Remember what you want to forget: Algorithms for machine unlearning, 2021. URL https://arxiv.org/abs/2103.03279.

Shaofei Shen, Chenhao Zhang, Alina Bialkowski, Weitong Chen, and Miao Xu. Camu: Disentangling causal effects in deep model unlearning, 2024a. URL https://arxiv.org/abs/2401.17504.

Shaofei Shen, Chenhao Zhang, Yawen Zhao, Alina Bialkowski, Weitong Tony Chen, and Miao Xu. Label-agnostic forgetting: A supervision-free unlearning in deep models, 2024b. URL https://arxiv.org/abs/2404.00506.

Haoran Tang and Rajiv Khanna. Sharpness-aware machine unlearning, 2025. URL https://arxiv.org/abs/2506.13715.

Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2018.

Alexander Warnecke, Lukas Pirch, Christian Wressnegger, and Konrad Rieck. Machine unlearning of features and labels, 2023. URL https://arxiv.org/abs/2108.11577.

Lei Wu and Weijie J. Su. The implicit regularization of dynamical stability in stochastic gradient descent, 2023. URL https://arxiv.org/abs/2305.17490.

Lei Wu, Chao Ma, et al. How sgd selects the global minima in over-parameterized learning: A dynamical stability perspective. *Advances in Neural Information Processing Systems*, 31, 2018.

Lei Wu, Mingze Wang, and Weijie Su. The alignment property of sgd noise and how it helps select flat minima: A stability analysis, 2022. URL https://arxiv.org/abs/2207.02628.