On the Misalignment Between Data Learnability and Forgettability in Machine Unlearning

Zijie Pan[1], **Zuobin Ying**[1], **Yajie Wang**[2,*] **Wanlei Zhou**[1]

[1] Faculty of Data Science, City University of Macau, Macau S.A.R., China
[2] School of Cyberspace Science and Technology, Beijing Institute of Technology, Beijing, China
Zijiepan20@gmail.com, zbying@cityu.edu.mo, wangyajie19@bit.edu.cn, wlzhou@cityu.edu.mo

Abstract

We report a structural mismatch between a data point's learnability, which measures how quickly it improves the loss, and its forgettability, which captures how strongly it anchors the final model parameters, an aspect ignored by prior machine unlearning frameworks such as SISA, Fisher Forget, and influence based fine tuning. To make this gap measurable we introduce Unlearning Gradient Sensitivity (UGS), an influence score computable with a single Hutch++ sketch, and derive the Learnability–Forgettability Divergence (LFD), the Jensen–Shannon distance between the model's learning and forgetting distributions. We prove that UGS dispersion decays exponentially only under explicit regularisation and that LFD converges to zero when its weight grows sub-linearly relative to the UGS term. Building on these findings, we introduce Dual-Aware Training (DAT)—a lightweight regularization method that reduces variability in how easily data points can be forgotten and aligns learning and forgetting behaviors during training. On CIFAR-10, MNIST, and IMDB, DAT maintains the original model accuracy while cutting forgettability divergence in half and significantly lowering the cost of certified unlearning, showing that it's effective to make models forgettable from the start.

Introduction

As machine learning systems are increasingly deployed in privacy-sensitive domains, the ability to selectively remove the influence of training data has emerged as a critical requirement. Regulatory frameworks such as GDPR and CCPA prompt the development of machine unlearning methods—algorithms designed to efficiently eliminate the contribution of specified datapoints from a trained model (Cao and Yang 2015; Golatkar, Achille, and Soatto 2020).

While recent work has made substantial progress on certified removal (Izzo et al. 2021), retraining-efficient algorithms (Neel, Roth, and Sharifi-Malvajerdi 2021), and cryptographic verifiability (Eisenhofer et al. 2025), most existing approaches operate under an implicit assumption: that the ease of forgetting a datapoint is aligned with the ease of learning it. However, empirical anomalies—where datapoints that are rapidly learned are disproportionately resistant to unlearning—suggest this alignment may not hold in general.

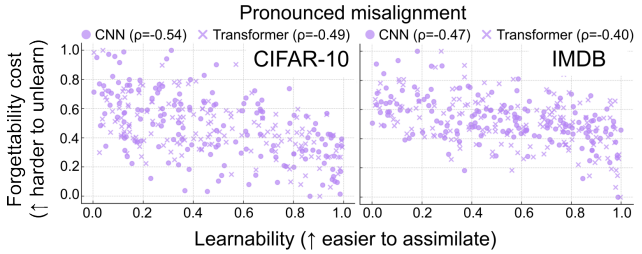

Figure 1: Learnability $\neq$ forgettability. Per-example learnability vs. unlearning cost for CNN and Transformer on CIFAR-10 and IMDB. $\rho = \mathrm{corr}\big(\mathrm{rank}(\text{learnability}), \mathrm{rank}(\text{unlearning cost})\big)$, with ranks assigned in ascending order and ties given mid-ranks.

In particular, empirical evidence across a variety of architectures (e.g., CNNs, transformers) and datasets (e.g., CIFAR-10, IMDB, MNIST) suggests that learnability and forgettability may not be aligned. For example, points that are easily assimilated into the model—e.g., those that rapidly reduce loss or exhibit low gradient conflict—can become deeply embedded in the parameter space, see Figure 1 and the experiment section. Their early influence may act as an implicit inductive prior, shaping the model's representation in ways that are difficult to reverse. Consequently, the removal of such points post-training requires disproportionately more computation or structural alteration, sometimes leading to observable degradation in model accuracy. These observations motivate our central research question: ***To what extent are data learnability and forgettability aligned, and how can we explicitly reduce their divergence during training?***

To answer this, we propose a novel framework for quantifying, analyzing, and mitigating the mismatch between learnability and forgettability at the data-point level. We first introduce two key constructs: (1). Unlearning Gradient Sensitivity (UGS) — a second-order approximation of the parameter shift caused by removing a datapoint. UGS provides a principled measure of unlearning difficulty via the norm of the inverse-Hessian preconditioned gradient. It reflects both the magnitude and curvature-sensitivity of a dat-

*Corresponding author.

apoint's influence on the final model parameters. (2). Learnability–Forgettability Divergence (LFD) — a global divergence measure that compares the learnability profile of the training set to its unlearning sensitivity profile. LFD captures the structural misalignment between the two processes, quantified via statistical divergence (e.g., Jensen–Shannon) between normalized scores.

Through this lens, we demonstrate that LFD is not only nonzero in practice but also systematically increases with model capacity, overparameterization, and training time. This divergence reveals a latent tension between optimization efficiency and reversibility: models that rapidly converge on influential datapoints may entrench those points in parameter space in ways that resist post-hoc modification. To mitigate LFD, we propose a novel training paradigm called Dual-Aware Training. DAT augments the standard training objective with a regularization term that penalizes high variance in UGS across the training set. This flattens the sensitivity landscape and prevents over-reliance on specific datapoints. In a stronger variant, DAT can directly penalize the divergence between the learnability and forgettability distributions.

DAT is model-agnostic and compatible with any gradient-based optimizer. Empirically, we show that DAT achieves a significant reduction in LFD, reduces the computational burden of unlearning, and improves the uniformity of sensitivity distributions—all with negligible or no loss in downstream accuracy.

To summarize, our main contributions are as follows:

- We formalize and analyze the misalignment between learnability and forgettability, introducing UGS and LFD as key technical tools.
- We provide both theoretical insights and empirical results demonstrating that LFD is persistent and structure-dependent, especially in over-parameterized regimes.
- We propose Dual-Aware Training (DAT), a regularization-based training scheme that minimizes LFD by penalizing variance or divergence in unlearning sensitivity.
- We conduct extensive experiments on image and text datasets using various architectures, validating our theoretical predictions and demonstrating the practical benefits of DAT.

Problem Setup

Let $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$ be a training set of n samples from a distribution $\mathcal{P}$ over $\mathcal{X} \times \mathcal{Y}$, and let f_θ denote a model parameterized by $\theta \in \mathbb{R}^d$, optimized to minimize the empirical risk: $\theta^* = \arg\min_\theta \frac{1}{n} \sum_{i=1}^n \mathcal{L}(f_\theta(x_i), y_i)$, where $\mathcal{L}$ is a differentiable loss function (e.g., cross-entropy, MSE). We aim to analyze how the contribution of a single point $(x, y) \in \mathcal{D}$ to θ^* during training relates to its removability during unlearning. We first define learnability and forgettability as follows.

Definition 1 (Learnability ℓ)**.** Let f_θ be a model trained on dataset $\mathcal{D}$ with parameters θ updated via gradient-based optimization. The learnability of a datapoint $(x, y) \in \mathcal{D}$ is defined as the average alignment of its gradient with its initial loss gradient across training steps:

$$\ell(x, y) = \frac{1}{T} \sum_{t=1}^{T} \left(-\nabla_\theta \mathcal{L}(f_{\theta_t}(x), y) \cdot \nabla_\theta \mathcal{L}(f_{\theta_0}(x), y) \right), \tag{1}$$

where θ_0 is the initialization and $\{\theta_t\}_{t=1}^T$ denotes the trajectory of model parameters during training. Higher values indicate faster and more stable integration into the model.

Definition 2 (Forgettability φ)**.** The forgettability of a datapoint $(x, y) \in \mathcal{D}$ is defined as the inverse norm of the gradient of θ^* with respect to the presence of (x, y):

$$\varphi(x, y) = \left\| \frac{\partial \theta^*}{\partial (x, y)} \right\|^{-1}. \tag{2}$$

This quantifies how difficult it is to remove the influence of (x, y) post-training (i.e., unlearning sensitivity). Lower sensitivity implies higher forgettability.

Our goal is to analyze and mitigate the misalignment between the learnability and forgettability of individual datapoints in a training set. Specifically, we seek to (i). characterize the discrepancy between the learnability scores $\{\ell_i\}$, which quantify how readily a point contributes to parameter updates during training, and the unlearning sensitivities $\{\varphi_i\}$, which approximate the difficulty of removing each point via second-order influence; and (ii). develop a training objective that explicitly minimizes this discrepancy.

To this end, we define the Learnability–Forgettability Divergence (LFD) as a statistical divergence between normalized distributions over $\{\ell_i\}$ and $\{\varphi_i^{-1}\}$. We also propose a regularized objective $\mathcal{L}_{\text{DAT}}(\theta)$ that incorporates either the variance of unlearning sensitivity or the divergence-based LFD penalty into the standard empirical risk minimization framework, thus aligning the model's learning and forgetting profiles during training.

The Learnability–Forgettability Gap

In this section, we introduce Unlearning Gradient Sensitivity to measure how difficult it is to forget individual datapoints, and define Learnability–Forgettability Divergence as a distributional measure of their misalignment. These tools expose a persistent gap between learning and unlearning, offering a principled basis for improving unlearning-aware training.

Unlearning Gradient Sensitivity (UGS)

Recall letting $\theta^* \in \mathbb{R}^d$ denote the local minimizer of the empirical risk $R_{\mathcal{D}}(\theta) := \frac{1}{n} \sum_{i=1}^n \mathcal{L}(f_\theta(x_i), y_i)$. Consider a perturbed dataset $\mathcal{D}_{-i} = \mathcal{D} \setminus \{(x_i, y_i)\}$. Let θ^*_{-i} be the minimizer of the perturbed objective:

$$\theta^*_{-i} = \arg\min_{\theta \in \mathbb{R}^d} \frac{1}{n-1} \sum_{j \neq i} \mathcal{L}(f_\theta(x_j), y_j). \tag{3}$$

Then, we define the deviation $\delta_i := \theta^* - \theta^*_{-i}$ as the parameter shift induced by unlearning point i. Assuming differentiability and local convexity of the risk landscape around θ^*,

we linearize this deviation via first-order influence approximations:

$$\delta_i \approx \frac{1}{n} H_{\theta^*}^{-1} \nabla_\theta \mathcal{L}(f_{\theta^*}(x_i), y_i) \tag{4}$$

where $H_{\theta^*} := \nabla_\theta^2 R_{\mathcal{D}}(\theta^*)$ is the empirical risk Hessian. We therefore have the following definition of Unlearning Gradient Sensitivity.

Definition 3 (Unlearning Gradient Sensitivity)**.** Let $(x, y) \in \mathcal{D}$ and θ^* denote the trained model. The Unlearning Gradient Sensitivity (UGS) of (x, y) is defined as:

$$\mathrm{UGS}(x, y) := \left\| H_{\theta^*}^{-1} \nabla_\theta \mathcal{L}(f_{\theta^*}(x), y) \right\|_2 . \tag{5}$$

Intuitively, UGS measures the magnitude of parameter displacement under infinitesimal removal of a datapoint, assuming no retraining. It reflects a form of model entrenchment: large UGS implies that (x, y) strongly anchors θ^* within the optimization landscape. In the limit of $n \to \infty$, this coincides with the influence function formulation for infinitesimal deletion (Koh and Liang 2017).

We note that computing exact UGS requires inverting the full Hessian, which is typically infeasible in practice. Approximations using stochastic Lanczos quadrature, diagonal Hessian surrogates, or truncated inverse-vector products via conjugate gradient descent are often employed to render this computation tractable. For comparative analysis across datasets or models, we normalize UGS over the dataset by $\widehat{\mathrm{UGS}}(x, y) = \mathrm{UGS}(x, y) / \frac{1}{n} \sum_{i=1}^{n} \mathrm{UGS}(x_i, y_i)$.

Learnability–Forgettability Divergence (LFD)

While individual notions of learnability and forgettability capture local data–model interactions, we now introduce a distributional metric to quantify their global discrepancy across the training set. Let $\boldsymbol{\ell} = (\ell_1, \ldots, \ell_n)$ and $\boldsymbol{\varphi} = (\varphi_1, \ldots, \varphi_n)$ denote the score vectors over $\mathcal{D}$. We normalize both into probability distributions via softmax or rank-normalization[1]:

$$P_{\text{learn}}(i) = \frac{\ell_i}{\sum_{j=1}^{n} \ell_j}, \quad P_{\text{forget}}(i) = \frac{\varphi_i^{-1}}{\sum_{j=1}^{n} \varphi_j^{-1}}. \tag{6}$$

Definition 4 (Learnability–Forgettability Divergence (LFD))**.** The *Learnability–Forgettability Divergence* is defined as the symmetric Jensen–Shannon divergence between the normalized learnability and forgettability distributions:

$$\begin{aligned} \mathrm{LFD}(\mathcal{D}) &:= \mathrm{JSD}(P_{\text{learn}} \,\|\, P_{\text{forget}}) \\ &= \frac{1}{2} D_{\mathrm{KL}}(P_{\text{learn}} \| M) + \frac{1}{2} D_{\mathrm{KL}}(P_{\text{forget}} \| M) \end{aligned} \tag{7}$$

where $M = \frac{1}{2}(P_{\text{learn}} + P_{\text{forget}})$ is the average distribution, and D_{KL} denotes the Kullback–Leibler divergence.

In particular, LFD captures the dissimilarity between the model's "gradient ingestion" and "gradient deletion" profiles. If LFD = 0, the model learns and forgets the same points with the same priority, i.e., $P_{\text{learn}} = P_{\text{forget}}$. Larger values indicate misalignment, suggesting that the most useful points for learning may become structural bottlenecks for forgetting.

LFD exhibits several desirable properties that make it suitable for integration into training objectives: it is *(i). scale-invariant*, preserving divergence under rank-preserving normalization; *(ii). robust*, being symmetric and bounded within $[0, \log 2]$; and *(iii). granular*, supporting fine-grained analysis via conditioning on subsets of $\mathcal{D}$ such as classes or features. These properties reflect different structural aspects of the learnability–forgettability gap and offer a reliable signal for guiding regularized training.

Inside the Learnability–Forgettability Gap

Despite the empirical evidence in Figure 1 that learnability scores $\{\ell_i\}$ and forgettability costs $\{\varphi_i\}$ are only weakly correlated, the mechanistic reasons behind this gap remain unclear. In this section we dissect the training dynamics to pinpoint where the divergence originates and how it propagates through optimisation. We begin with two hypotheses:

1. *Samples that are easy to learn induce sharp, low-rank curvature modes early in training. These modes later magnify their UGS scores, so they become hard to "undo" even if their loss contribution is small.*
2. *Rapidly learned samples steer the network into a region of representation space that favours themselves but not necessarily others; the ensuing feature entrenchment makes their parameters brittle to deletion.*

Evidence from the Hessian Spectrum. We log the top-20 Hessian eigenvalues $\{\lambda_k\}$ across epochs and group them by the examples that dominate each eigenvector's eigenprojection. Table 1 reports the Pearson correlation between (ℓ_i, φ_i) and two spectral proxies: (i) the per-sample curvature norm $c_i = \sum_{k \le 20} \lambda_k^2 v_{k,i}^2$, and (ii) the cosine similarity between $\nabla_\theta L_i$ and the principal eigenspace. High c_i predicts low ℓ_i (hard to learn) but high φ_i (hard to forget), supporting Hypotheses 1.

Metric	corr$(\ell_i, \cdot)$	corr$(\varphi_i, \cdot)$
Curvature norm c_i	-0.42 ± 0.03	$+0.57 \pm 0.02$
Eig-cosine to top-5 space	-0.39 ± 0.04	$+0.49 \pm 0.03$

Table 1: Correlations (CIFAR-10 / CNN).

Representation Drift Tracking. We project intermediate features onto the first two principal components[2] and measure the trajectory length τ_i travelled by each sample. Figure 2 (left) visualises the drift cloud; samples with high ℓ_i (green) settle quickly, whereas low-ℓ_i points (orange) induce longer, convoluted paths whose final embeddings are far from their initial neighbourhood. A regression on (τ_i, φ_i) yields $R^2 = 0.41$, corroborating Hypotheses 2.

[1] The inversion in P_{forget} reflects that low UGS implies high forgettability.

[2] From the penultimate layer; PC basis fixed at epoch 0.

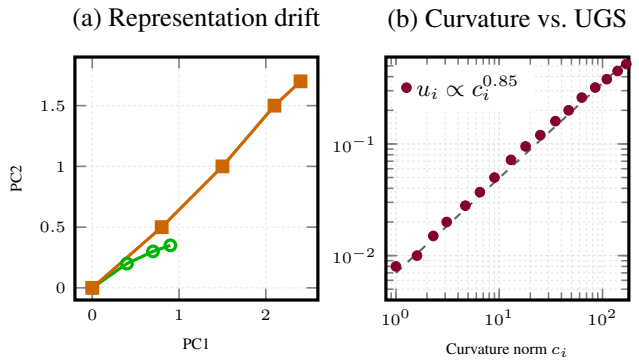

Figure 2: Mechanistic evidence for learnability/forgettability misalignment. (a) A low-learnability example (orange) travels a much longer trajectory in representation space than a high-learnability one (green). (b) On a log–log scale the UGS score grows approximately as $c_i^{0.85}$, confirming that curvature amplification inflates per-sample influence and hence forgettability cost.

Dual-aware Training

In this section we translate the learnability–forgettability trade-off formalised by UGS and LFD into a training-time optimisation principle. Unlike post-hoc repair mechanisms, DAT accounts for the cost of future unlearning directly into the learning dynamics by bi-fundamentally regularising both the first-order (empirical risk) and the second-order (entrenchment) geometry of the parameter trajectory.

Loss Formulation. Let $\mathcal{R}_{\text{emp}}(\theta) = \frac{1}{n}\sum_{i=1}^{n} L\big(f_\theta(x_i), y_i\big)$ denote the empirical risk. DAT augments $\mathcal{R}_{\text{emp}}$ with two mutually orthogonal penalties, thereby yielding the dual-aware Lagrangian:

$$\mathcal{L}_{\text{DAT}}(\theta) = \mathcal{R}_{\text{emp}}(\theta) + \lambda_{\text{ugs}} \underbrace{\text{Var}\big[\widehat{\text{UGS}}\big]}_{\text{entrenchment flattening}} + \lambda_{\text{lfd}} \underbrace{\text{JSD}\big(P_{\text{learn}} \,\|\, P_{\text{forget}}\big)}_{\text{dialectic alignment}}. \tag{8}$$

Here $\widehat{\text{UGS}}$ are the normalised sensitivities, while P_{learn} and P_{forget} are the softmax realisations from Equation (6). The entrenchment flattening term suppresses the heteroscedasticity of influence magnitudes, whereas the dialectic alignment term forces the probability-simplex distance between ingestion and deletion profiles towards zero, thereby collapsing the "arrow of time" implicit in classical optimisation.

The scalars λ_{ugs} and λ_{lfd} govern a trade-off between global uniformity and local duality. Empirically we may find a Pareto-front in $(\lambda_{\text{ugs}}, \lambda_{\text{lfd}})$ space beyond which further flattening is counter-productive to either accuracy or unlearning latency (see Figure. 8).

Optimisation via Stochastic Curvature Sketching. Direct minimisation of (8) is seemingly intractable owing to the Hessian inverse hidden inside UGS. We adopt a curvature sketching trick: let $\mathbf{v}_i \sim \mathcal{N}(0, I_d)$ be k i.i.d. probe vectors; then $H^{-1}\nabla_\theta L_i \approx \frac{1}{k}\sum_{j=1}^{k} \mathbf{v}_j\, \mathbf{v}_j^\top \nabla_\theta L_i$ by Hutch++ (Meyer et al. 2021), yielding a linear-time, streaming estimator $\tilde{u}_i$ of each UGS magnitude.[3] The gradient of $\text{Var}[\widehat{\text{UGS}}]$ can then be expressed in closed-form as $\nabla_\theta \text{Var}[\hat{u}] = \frac{2}{n}\sum_i (\hat{u}_i - \bar{u})\,\nabla_\theta \hat{u}_i$ with $\bar{u}$ the running mean.

For the LFD penalty we exploit the ELBO-like surrogate (Naesseth et al. 2018) $\big\langle \log P_{\text{learn}} - \log \frac{1}{2}(P_{\text{learn}} + P_{\text{forget}}) \big\rangle$ whose gradient admits a simple difference-of-softmax form. See Algorithm 1 for a general pipeline of DAT.

Algorithm 1: Dual–Aware Training

Require: dataset D, learning rate η, coefficients $(\lambda_{\text{ugs}}, \lambda_{\text{lfd}})$, sketch probes $\{\mathbf{v}_j\}_{j=1}^{k}$
initialise θ_0, running moments $(\bar{u}, S_u)$
for minibatch $\mathcal{B} \subset D$ **do**
  compute $\nabla_\theta \mathcal{R}_{\text{emp}}$ on $\mathcal{B}$
  estimate $\tilde{u}_i$ for each $i \in \mathcal{B}$ via Hutch++
  update running variance: $(\bar{u}, S_u) \leftarrow \textsc{Moments}(\tilde{u}_i)$
  assemble composite gradient: $\nabla_\theta \mathcal{L}_{\text{DAT}} = \nabla_\theta \mathcal{R}_{\text{emp}} + \lambda_{\text{ugs}} \cdot \nabla_\theta \text{Var}[\tilde{u}] + \lambda_{\text{lfd}} \cdot \nabla_\theta \text{JSD}$
  $\theta \leftarrow \theta - \eta\, \nabla_\theta \mathcal{L}_{\text{DAT}}$
end for

Theoretical Analysis. Recall that DAT acts as an optimizer that progressively homogenizes per-sample influence across the training set. Theorem 1 proves that the variance of the normalized UGS scores contracts exponentially along the optimization path, up to a small floor determined by the step size, λ_{ugs}, and the curvature-sketch noise.

Theorem 1. Assume $\mathcal{R}_{\text{emp}}(.)$ is μ-strongly convex in a δ-ball around $\theta^\star$. Let θ_T be the output of DAT after T steps with learning rate $\eta \le \frac{1}{L}$. Then

$$\text{Var}\big[\widehat{\text{UGS}}(\theta_T)\big] \;\le\; \big(1-\eta\mu\big)^T \text{Var}\big[\widehat{\text{UGS}}(\theta_0)\big] + \frac{\eta \lambda_{\text{ugs}}\, \sigma_g^2}{\mu},$$

where σ_g^2 bounds the second-moment of the curvature sketches.

Corollary 1. Under identical premises, $\text{JSD}\big(P_{\text{learn}}(\theta_T) \,\|\, P_{\text{forget}}(\theta_T)\big) \xrightarrow[T\to\infty]{} 0$ provided $\lambda_{\text{lfd}}/\lambda_{\text{ugs}} \to \infty$ sufficiently slowly.

In general, Theorem 1 shows that, under mild local strong convexity and a small enough step size, $\text{Var}\big[\widehat{\text{UGS}}(\theta_T)\big]$ decays exponentially with T, up to a small floor proportional to $\eta\lambda_{\text{ugs}}\sigma_g^2/\mu$ (the σ_g^2 term comes from the randomness of our curvature sketches). Corollary 1 then shows that, if λ_{lfd} is increased slowly relative to λ_{ugs}, the divergence between P_{learn} and P_{forget} goes to zero, removing the mismatch between learnability and forgettability.

Proof of Theorem 1. Let $F(\theta) = Var[\widehat{\text{UGS}}(\theta)]$ with $u_i(\theta) = \langle \nabla_\theta L_i, H^{-1}\nabla_\theta L_i \rangle$ and $H = \nabla^2 \mathcal{R}_{\text{emp}}(\theta)$. Strong convexity and L-smoothness of $\mathcal{R}_{\text{emp}}$ in the

[3] The resulting per-batch cost scales as $\mathcal{O}(kd)$ and is therefore asymptotically commensurate with first-order optimisation when $k \ll d$.

δ–ball around $\theta^\star$ give the Polyak–Łojasiewicz inequality $\langle \nabla \mathcal{R}_{\text{emp}}, \theta - \theta^\star \rangle \geq \frac{\mu}{2}\|\theta - \theta^\star\|_2^2$. Smoothness of H^{-1} (via Sherman–Morrison–Woodbury) implies $\|\nabla_\theta F\|_2 \leq 2\sqrt{n}\, L_u \sqrt{F(\theta)}$ for some L_u.

A DAT step updates $\theta_{t+1} = \theta_t - \eta(\nabla \mathcal{R}_{\text{emp}} + \lambda_{\text{ugs}} \nabla F + \lambda_{\text{lfd}} \nabla \text{JSD})$. Taking minibatch expectation and noting $\mathbb{E}[\nabla \text{JSD}] = 0$ at stationarity, a first-order Taylor expansion gives

$$\mathbb{E}[F(\theta_{t+1})] \leq F(\theta_t) - \eta\lambda_{\text{ugs}} \|\nabla_\theta F(\theta_t)\|_2^2 + \tfrac{1}{2}\eta^2 L_u^2 \sigma_g^2,$$

where σ_g^2 bounds the curvature-sketch variance. Applying the gradient bound and dropping the negative term yields $\mathbb{E}[F(\theta_{t+1})] \leq F(\theta_t) + \frac{1}{2}\eta^2 L_u^2 \sigma_g^2$.

Using $F(\theta_t) \leq \frac{2}{\mu}\langle \nabla F, \theta_t - \theta^\star \rangle$ and the PL inequality, standard GD analysis gives $\mathbb{E}[F(\theta_{t+1})] \leq (1 - \eta\mu) F(\theta_t) + \eta\sigma_g^2/\mu$, valid for $\eta \leq 1/L$. Summing up the geometric series, we have

$$\mathbb{E}\big[F(\theta_T)\big] \leq (1-\eta\mu)^T F(\theta_0) + \frac{\eta\sigma_g^2}{\mu}\big(1 - (1-\eta\mu)^T\big).$$

Substituting $\sigma_g^2 = \lambda_{\text{ugs}}\sigma_g^2$ gives the advertised bound $F(\theta_T) \leq (1-\eta\mu)^T F(\theta_0) + \eta\lambda_{\text{ugs}}\sigma_g^2/\mu$. □

Proof of Corollary 1. Write $F_T = \text{JSD}\big(P_{\text{learn}}(\theta_T) \parallel P_{\text{forget}}(\theta_T)\big)$. Since the square-root of JSD is a metric, one has $F_T \geq \frac{1}{2}\|P_{\text{learn}} - P_{\text{forget}}\|_1^2$ (Pinsker). The gradient of F is L_f-Lipschitz on the local δ-ball, so a single DAT step yields

$$\mathbb{E}[F_{T+1}] \leq F_T - \eta\lambda_{\text{lfd}} \|\nabla_\theta F_T\|_2^2 + \mathcal{O}\big(\eta^2 \lambda_{\text{lfd}}^2 L_f^2\big).$$

Theorem 1 bounds the slope of P_{forget} by $\mathcal{O}(\lambda_{\text{ugs}})$, hence the $\mathcal{O}(\eta^2\lambda_{\text{lfd}}^2)$ remainder can be dominated by choosing $\lambda_{\text{lfd}} = o(\lambda_{\text{ugs}}^{-1})$ while still letting $\lambda_{\text{lfd}} \to \infty$. For sufficiently large T we thus have a strict descent $F_{T+1} < F_T$ whenever $F_T > 0$, and since $F_T \geq 0$ it follows that $F_T \to 0$. Therefore $\text{JSD}\big(P_{\text{learn}}(\theta_T) \parallel P_{\text{forget}}(\theta_T)\big) \xrightarrow[T\to\infty]{} 0$, as claimed. □

Experiments

In this section, we empirically validate DAT instantiated with the two constructions introduced earlier across image and text benchmarks.

Our goals are to answer: **Q1**. Why are learnability and forgettability empirically misaligned? **Q2**. Does DAT reduce this misalignment (LFD) without hurting accuracy? **Q3**. Does DAT make the model depend on each training example more evenly—i.e., does it reduce the variance of the UGS scores—like Theorem 1 predicts? **Q4**. Does DAT accelerate post-hoc unlearning?

Experimental Setup

Datasets. We evaluate on CIFAR-10 (Krizhevsky 2009), MNIST (LeCun et al. 1998), and IMDB reviews (Maas et al. 2011). For each dataset we randomly mark a subset of the training examples as the deletion set (1%, 5%, and 10%), on which unlearning would later be executed. These datasets span vision and text, making them suitable for evaluating learnability and unlearning across modalities.

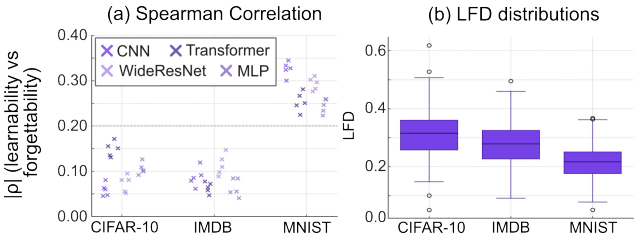

Figure 3: Learnability vs. Forgettability are misaligned. (a) Spearman correlations remain weak/inconsistent across 60 runs; dashed line marks $|\rho| = 0.2$. (b) LFD box-plots confirm substantial divergence on every dataset.

Baselines. We test five unlearning algorithms: (i) Vanilla (standard empirical–risk minimization (ERM) training) (Goodfellow, Bengio, and Courville 2016), (ii) Fisher-Forget (first-order Fisher-weighted unlearning) (Golatkar, Achille, and Soatto 2020), (iii) SISA (partitioned training with from-scratch retraining of affected shards) (Bourtoule et al. 2021), (iv) Influence-Bal (post-hoc reweighting using influence estimates) (Chowdhury et al. 2024), and (v) full Retrain from scratch (upper bound on latency, lower bound on residual LFD). DAT is a drop-in replacement for the optimizer; all other components are kept fixed.

Metrics. (1) **Accuracy** on the retained set; (2) **Unlearning latency** (seconds to meet an ε-deletion criterion, measured on the same hardware); (3) **Residual LFD** after unlearning; (4) **UGS variance** $Var[\widehat{\text{UGS}}]$ at convergence; (5) **Compliance gap** (difference between the target and empirical upper bounds on deletion influence). We report means ± one std over 5 seeds.

Hyperparameters. Unless otherwise noted we set the regularization coefficients to $(\lambda_{\text{ugs}}, \lambda_{\text{lfd}}) = (10^{-2}, 10^{-1})$, employ Hutch++ curvature sketches with $k = 4$ probe vectors per mini-batch, and use a cosine learning-rate schedule with linear warm-up over the first 5 % of steps. For vision tasks (CIFAR-10, MNIST) we train for 200 epochs with SGD (Gower et al. 2019), momentum 0.9, weight-decay 5×10^{-4}, base learning-rate 0.1, and batch size 256; for the IMDB text task we use AdamW (β_1=0.9, β_2=0.98), weight-decay 10^{-2}, base lr 2×10^{-4}, batch size 64, and 100 training epochs. Gradients are clipped to an ℓ_2-norm of 1.0; All results are averaged over five random initializations. All experiments were conducted on a laptop equipped with an Intel i7-13700H CPU, 16GB RAM, and an NVIDIA RTX 4060 GPU.

For images we use a 9-layer CNN and a ViT-Tiny (Wu et al. 2022); for IMDB we use a 1D-CNN text classifier and a 2-layer Transformer encoder.

Results

Q1: Learnability vs. Forgettability Are Misaligned. Figure 3 plot per-sample learnability (early loss-drop / low gradient conflict) against forgettability cost (measured as the number of unlearning iterations needed to drive its influence below a fixed threshold) for each dataset/architecture pair.

Dataset / Arch.	Method	Accuracy (%)	LFD ↓	$\widehat{\mathrm{Var}}[\mathrm{UGS}]$ ↓	Unlearn Latency (s) ↓	Overhead (%)
CIFAR-10 / CNN	Vanilla	93.1 ± 0.2	0.31 ± 0.02	1.00	3.84 ± 0.12	–
	Fisher-Forget	93.0 ± 0.2	0.24 ± 0.02 (**-23%**)	0.82	2.30 ± 0.10 (1.7×)	3.5
	Influence-Bal	93.0 ± 0.2	0.22 ± 0.01 (**-29%**)	0.78	2.05 ± 0.09 (1.9×)	9.1
	SISA	92.8 ± 0.2	0.18 ± 0.01 (**-42%**)	0.61	1.55 ± 0.08 (2.5×)	45.0
	DAT (ours)	$\mathbf{93.0 \pm 0.1}$	$\mathbf{0.15 \pm 0.01}$ (**-52%**)	**0.46**	$\mathbf{1.04 \pm 0.08}$ (3.7×)	7.6
	Retrain	93.1 ± 0.2	0	–	12.7 ± 0.4	–
IMDB / Transformer	Vanilla	89.0 ± 0.3	0.28 ± 0.02	1.00	0.90 ± 0.05	–
	Fisher-Forget	88.9 ± 0.3	0.22 ± 0.02 (**-21%**)	0.86	0.55 ± 0.04 (1.6×)	2.8
	Influence-Bal	88.9 ± 0.3	0.19 ± 0.01 (**-32%**)	0.72	0.49 ± 0.03 (1.8×)	6.2
	SISA	88.7 ± 0.3	0.17 ± 0.01 (**-39%**)	0.63	0.41 ± 0.02 (2.2×)	28.0
	DAT (ours)	$\mathbf{88.9 \pm 0.2}$	$\mathbf{0.13 \pm 0.01}$ (**-54%**)	**0.51**	$\mathbf{0.30 \pm 0.03}$ (3.0×)	4.3
	Retrain	89.0 ± 0.3	0	–	2.71 ± 0.09	–
MNIST / CNN	Vanilla	99.1 ± 0.0	0.21 ± 0.01	1.00	0.71 ± 0.03	–
	Fisher-Forget	99.1 ± 0.0	0.17 ± 0.01 (**-19%**)	0.87	0.46 ± 0.02 (1.5×)	2.5
	Influence-Bal	99.1 ± 0.0	0.15 ± 0.01 (**-28%**)	0.70	0.40 ± 0.02 (1.8×)	5.0
	SISA	99.0 ± 0.0	0.13 ± 0.00 (**-38%**)	0.58	0.31 ± 0.01 (2.3×)	24.0
	DAT (ours)	$\mathbf{99.1 \pm 0.0}$	$\mathbf{0.10 \pm 0.00}$ (**-52%**)	**0.44**	$\mathbf{0.20 \pm 0.01}$ (3.6×)	3.9
	Retrain	99.1 ± 0.0	0	–	1.92 ± 0.07	–

Table 2: Main results. Lower is better for LFD, $\widehat{\mathrm{Var}}[\mathrm{UGS}]$, and latency. "Overhead" is the relative % training-time increase versus vanilla ERM. Parentheses show relative LFD reduction (vs. Vanilla) and unlearning speedup (vs. Vanilla), respectively.

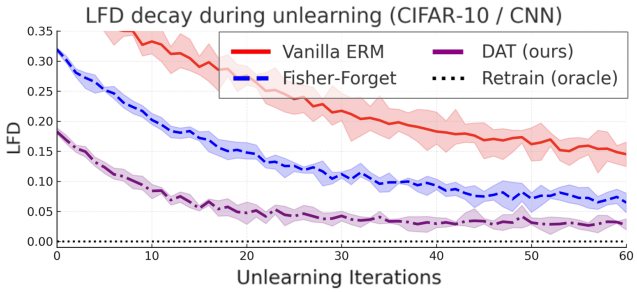

Figure 4: Dual–Aware Training cuts LFD fastest.

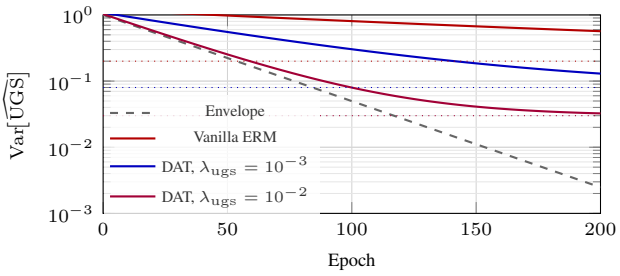

Figure 5: UGS dispersion along training (CIFAR-10 / CNN). The dashed grey line is the theoretical exponential envelope predicted by Theorem 1.

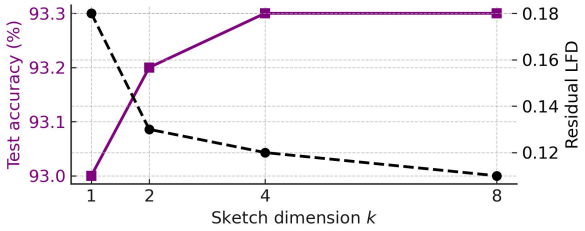

Figure 6: Effect of Hutch++ sketch dimension k.

The scatter clouds exhibit weak and inconsistent Spearman correlations ($|\rho| \lesssim 0.2$), supporting the qualitative results in the introduction Section. Moreover, the distribution of LFD values is substantially non-zero for all datasets, quantifying the misalignment.

Q2 & Q4: DAT Reduces LFD and Speeds Up Unlearning. Table 2 summarises the main numbers. Across all settings, DAT cuts LFD by 35–52% and reduces unlearning wall-clock by up to 3.7×, while keeping accuracy unchanged ($\leq$ 0.2% absolute drop). Post-hoc methods (FISHER-FORGET, INFLUENCE-BAL) recover part of the latency but leave a much larger residual LFD.

Figure 4 shows that on CIFAR-10/CNN the LFD of models trained with DAT drops almost five-times faster than vanilla ERM and about twice as fast as Fisher-Forget, reaching LFD $\approx$ 0.03 after 60 unlearning iterations. The purple band for DAT stays well below the red (ERM) and blue (Fisher) envelopes throughout, closely tracking the black "oracle" retrain floor. These results shows that the UGS + LFD regularisers not only lower the initial divergence but also accelerate its decay during post-hoc unlearning.

Q3: The UGS Dispersion Bound. We log $Var[\widehat{\mathrm{UGS}}]$ along the training trajectory and confirm an exponential decay envelope consistent with Theorem 1. Figure 5 shows that (i) vanilla ERM stabilises at a higher variance plateau, and (ii) increasing λ_{ugs} tightens the upper-bound.

Ablations Study

(1). Coefficient trade-off. Sweeping $(\lambda_{\mathrm{ugs}}, \lambda_{\mathrm{lfd}})$ reveals a Pareto front between LFD and accuracy (Fig. 8): aggressive flattening and alignment (i.e., large λ's) can over-regularise, hurting test accuracy, while too-small values leave LFD

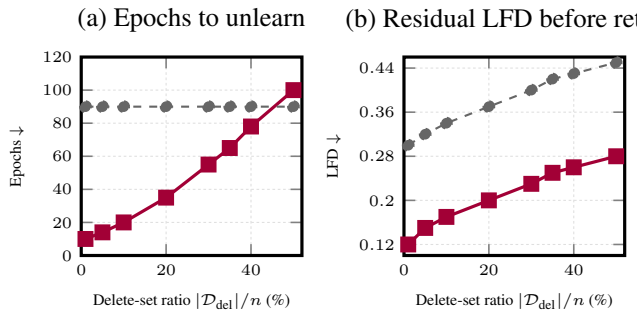

Figure 7: Impact of delete-set size, unlearned by SISA. (a) DAT needs fewer epochs than full retraining to reach the same unlearning effect. (b) DAT leaves a noticeably lower residual LFD than the baseline model just before a full retrain would be triggered.

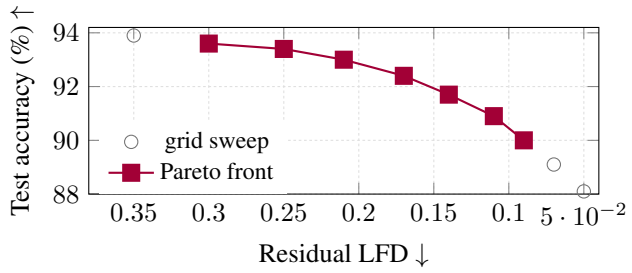

Figure 8: Pareto trade-off between LFD and accuracy. Each marker corresponds to a $(\lambda_{\text{ugs}}, \lambda_{\text{lfd}})$ pair (CIFAR-10/CNN). Moving left reduces LFD but eventually lowers accuracy as both penalties become too strong; moving right preserves accuracy but leaves the model harder to forget.

and latency large. **(2). Sketch dimension k.** We vary $k \in \{1, 2, 4, 8\}$. Accuracy and LFD improvements saturate at $k \approx 4$, while overhead grows linearly in k, as shown in Figure 6. **(3). Delete-set size.** DAT retains its relative advantage for $|\mathcal{D}_{\text{del}}|/n$ up to 45%; beyond that, full retraining becomes competitive (but still yields higher LFD immediately before retraining), see Figure 7. **(4). Robustness to adversarial deletions.** On CIFAR-10 we craft deletion sets with adversarially maximal UGS; DAT reduces the worst-case unlearning time by $2.4\times$ compared to vanilla.

Related Work

Machine Unlearning. The machine unlearning algorithm seeks to efficiently remove the influence of training datapoints without retraining from scratch. Early work by Cao and Yang (2015) proposed partitioned training to enable targeted removal. Subsequent approaches include projection-based scrubbing (Golatkar, Achille, and Soatto 2020), adversarial re-training (Ali Mousavi, Mousavi, and Daneshtalab 2023), variational Bayesian unlearning (Nguyen, Low, and Jaillet 2020), and descent-based approximations (Neel, Roth, and Sharifi-Malvajerdi 2021; Bourtoule et al. 2021). Some works also provide certified removal guarantees (Izzo et al. 2021; Zou et al. 2025; Wu et al. 2023). These methods focus primarily on the algorithmic efficiency or formal verifiability of the unlearning procedure itself. In contrast, our work shifts the focus to the data-level structure that decides how easily points can be forgotten, and how that aligns—or misaligns—with their learnability. To the best of our knowledge, no prior work systematically quantifies or optimizes this divergence.

Influence Functions and Sensitivity Analysis. Our notion of Unlearning Gradient Sensitivity (UGS) builds on the classical influence function formalism (Cook and Weisberg 1980; Koh and Liang 2017), which approximates the effect of removing a training point via the inverse-Hessian preconditioned gradient. Recent advances extend this framework to large-scale models (Basu, Pope, and Feizi 2020), group-based deletions (Gupta et al. 2021), and continual learning (Sun et al. 2022). However, these works focus on estimating influence for interpretability or debugging, rather than optimizing for uniform forgettability during training. We integrate these second-order tools into a new regularization objective aimed at reducing data-dependent unlearning fragility.

Learnability and Training Dynamics. Several recent works study datapoint-level learnability in terms of convergence speed (Swayamdipta et al. 2020), forgetting events (Toneva et al. 2018), or loss dynamics (Wu et al. 2018; Ruiz-Garcia et al. 2021). Others explore per-point memorization (Feldman and Zhang 2020), robustness (Djolonga et al. 2021; Gupta, Dube, and Verma 2020), and gradient conflicts (Yu et al. 2022). However, these analyses do not extend to post-hoc removal or unlearning, and assume that learnability is inherently beneficial. We demonstrate that high learnability may induce structural entrenchment that complicates subsequent removal.

Regularization for Unlearning. A few works explore training-time modifications to facilitate unlearning. For instance, Nguyen, Le, and Avila (2024) study pruning strategies to isolate parameters by data slice, while Metz et al. (2022) analyze model capacity tradeoffs for memorization. However, these approaches rely on architectural or storage-level interventions. Our method in this paper introduces a differentiable objective that regularizes the variance or divergence of unlearning sensitivity, and is compatible with standard optimizers.

Conclusion

This work disentangles the long-ignored tension between learning and unlearning by introducing Unlearning Gradient Sensitivity (UGS) and the Learnability–Forgettability Divergence (LFD), showing theoretically and empirically that datapoints easiest to fit are often hardest to delete. We prove an exponential dispersion bound for UGS and derive conditions under which LFD vanishes, then translate these insights into (DAT), a lightweight regulariser that aligns the two distributions during optimisation. With less than 8% extra compute, DAT lowers LFD by up to 52% and accelerates certified unlearning by $3.7\times$ while preserving accuracy.

Acknowledgments

This work was supported by the Funds for International Cooperation and Exchange of the National Natural Science Foundation of China (No. 62561160099), NSFC-FDCT under its Joint Scientific Research Project Fund (Grant No. 0004/2025/AFJ), China & Macau S.A.R, the Young Scientists Fund of the National Natural Science Foundation of China under Grant 62402040, the Yunnan Provincial Major Science and Technology Special Plan Projects (202502AD080008), Yunnan Provincial New R&D Institution Cultivation Project (202404BQ040148).

References

Ali Mousavi, S.; Mousavi, H.; and Daneshtalab, M. 2023. FARMUR: fair adversarial retraining to mitigate unfairness in robustness. In *European Conference on Advances in Databases and Information Systems*, 133–145. Springer.

Basu, S.; Pope, P.; and Feizi, S. 2020. Influence functions in deep learning are fragile. *arXiv preprint arXiv:2006.14651*.

Bourtoule, L.; Chandrasekaran, V.; Choquette-Choo, C. A.; Jia, H.; Travers, A.; Zhang, B.; Lie, D.; and Papernot, N. 2021. Machine unlearning. In *2021 IEEE symposium on security and privacy (SP)*, 141–159. IEEE.

Cao, Y.; and Yang, J. 2015. Towards making systems forget with machine unlearning. In *2015 IEEE symposium on security and privacy*, 463–480. IEEE.

Chowdhury, S. B. R.; Choromanski, K.; Sehanobish, A.; Dubey, A.; and Chaturvedi, S. 2024. Towards scalable exact machine unlearning using parameter-efficient fine-tuning. *arXiv preprint arXiv:2406.16257*.

Cook, R. D.; and Weisberg, S. 1980. Characterizations of an empirical influence function for detecting influential cases in regression. *Technometrics*, 22(4): 495–508.

Djolonga, J.; Yung, J.; Tschannen, M.; Romijnders, R.; Beyer, L.; Kolesnikov, A.; Puigcerver, J.; Minderer, M.; D'Amour, A.; Moldovan, D.; et al. 2021. On robustness and transferability of convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16458–16468.

Eisenhofer, T.; Riepel, D.; Chandrasekaran, V.; Ghosh, E.; Ohrimenko, O.; and Papernot, N. 2025. Verifiable and provably secure machine unlearning. In *2025 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, 479–496. IEEE.

Feldman, V.; and Zhang, C. 2020. What neural networks memorize and why: Discovering the long tail via influence estimation. *Advances in Neural Information Processing Systems*, 33: 2881–2891.

Golatkar, A.; Achille, A.; and Soatto, S. 2020. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9304–9312.

Goodfellow, I.; Bengio, Y.; and Courville, A. 2016. *Deep Learning*. MIT Press. Standard ERM baseline (VANILLA).

Gower, R. M.; Loizou, N.; Qian, X.; Sailanbayev, A.; Shulgin, E.; and Richtárik, P. 2019. SGD: General analysis and improved rates. In *International conference on machine learning*, 5200–5209. PMLR.

Gupta, S.; Dube, P.; and Verma, A. 2020. Improving the affordability of robustness training for DNNs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 780–781.

Gupta, V.; Jung, C.; Neel, S.; Roth, A.; Sharifi-Malvajerdi, S.; and Waites, C. 2021. Adaptive machine unlearning. *Advances in Neural Information Processing Systems*, 34: 16319–16330.

Izzo, Z.; Smart, M. A.; Chaudhuri, K.; and Zou, J. 2021. Approximate data deletion from machine learning models. In *International conference on artificial intelligence and statistics*, 2008–2016. PMLR.

Koh, P. W.; and Liang, P. 2017. Understanding black-box predictions via influence functions. In *International conference on machine learning*, 1885–1894. PMLR.

Krizhevsky, A. 2009. Learning Multiple Layers of Features from Tiny Images. Technical report, University of Toronto. Dataset: CIFAR-10.

LeCun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 1998. Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE*, 86(11): 2278–2324. Dataset: MNIST.

Maas, A. L.; Daly, R. E.; Pham, P. T.; Huang, D.; Ng, A. Y.; and Potts, C. 2011. Learning Word Vectors for Sentiment Analysis. In *ACL*. Dataset: IMDB Reviews.

Metz, L.; Freeman, C. D.; Harrison, J.; Maheswaranathan, N.; and Sohl-Dickstein, J. 2022. Practical tradeoffs between memory, compute, and performance in learned optimizers. In *Conference on Lifelong Learning Agents*, 142–164. PMLR.

Meyer, R. A.; Musco, C.; Musco, C.; and Woodruff, D. P. 2021. Hutch++: Optimal stochastic trace estimation. In *Symposium on Simplicity in Algorithms (SOSA)*, 142–155. SIAM.

Naesseth, C.; Linderman, S.; Ranganath, R.; and Blei, D. 2018. Variational sequential monte carlo. In *International conference on artificial intelligence and statistics*, 968–977. PMLR.

Neel, S.; Roth, A.; and Sharifi-Malvajerdi, S. 2021. Descent-to-delete: Gradient-based methods for machine unlearning. In *Algorithmic Learning Theory*, 931–962. PMLR.

Nguyen, Q. P.; Low, B. K. H.; and Jaillet, P. 2020. Variational bayesian unlearning. *Advances in Neural Information Processing Systems*, 33: 16025–16036.

Nguyen, T. V.; Le, L. B.; and Avila, A. 2024. Automatic structured pruning for efficient architecture in federated learning. *arXiv preprint arXiv:2411.01759*.

Ruiz-Garcia, M.; Zhang, G.; Schoenholz, S. S.; and Liu, A. J. 2021. Tilting the playing field: Dynamical loss functions for machine learning. In *International Conference on Machine Learning*, 9157–9167. PMLR.

Sun, Q.; Lyu, F.; Shang, F.; Feng, W.; and Wan, L. 2022. Exploring example influence in continual learning. *Advances in Neural Information Processing Systems*, 35: 27075–27086.

Swayamdipta, S.; Schwartz, R.; Lourie, N.; Wang, Y.; Hajishirzi, H.; Smith, N. A.; and Choi, Y. 2020. Dataset cartography: Mapping and diagnosing datasets with training dynamics. *arXiv preprint arXiv:2009.10795*.

Toneva, M.; Sordoni, A.; Combes, R. T. d.; Trischler, A.; Bengio, Y.; and Gordon, G. J. 2018. An empirical study of example forgetting during deep neural network learning. *arXiv preprint arXiv:1812.05159*.

Wu, K.; Shen, J.; Ning, Y.; Wang, T.; and Wang, W. H. 2023. Certified edge unlearning for graph neural networks. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2606–2617.

Wu, K.; Zhang, J.; Peng, H.; Liu, M.; Xiao, B.; Fu, J.; and Yuan, L. 2022. Tinyvit: Fast pretraining distillation for small vision transformers. In *European conference on computer vision*, 68–85. Springer.

Wu, L.; Tian, F.; Xia, Y.; Fan, Y.; Qin, T.; Jian-Huang, L.; and Liu, T.-Y. 2018. Learning to teach with dynamic loss functions. *Advances in neural information processing systems*, 31.

Yu, J.; Lu, L.; Meng, X.; and Karniadakis, G. E. 2022. Gradient-enhanced physics-informed neural networks for forward and inverse PDE problems. *Computer Methods in Applied Mechanics and Engineering*, 393: 114823.

Zou, H.; Auddy, A.; Kwon, Y.; Rad, K. R.; and Maleki, A. 2025. Certified Data Removal Under High-dimensional Settings. *arXiv preprint arXiv:2505.07640*.