Gradient sparsity regularization for training unlearning-compatible models

Sobhan Hemati[1(✉)], Soufiane Lamghari[1], Masoud Asgharian[1] ,[2], Xu Li[1], and Hongliang Li[1]

[1] Huawei Noah's Ark Lab, Montreal, Canada
sobhan.hemati1@huawei.com
[2] Department of Mathematics and Statistics, McGill University, Montreal, Canada

Abstract. In this paper, we approach the machine unlearning (MU) problem from a novel perspective: Can we design a learning strategy that simplifies the later MU problem, i.e., leads to unlearning-compatible models? Motivated by recent findings that the cost to reverse SGD (unlearning error) is proportional to the accumulated loss curvature and that the generalization error (a measure of acquired subject-specific knowledge) is bounded by loss sharpness, we focus on the loss sharpness as the key factor to train unlearning-compatible models. By studying the theoretical implications of defining the sharpness over different perturbation norms, we propose a variation of the sharpness, quantified by the l_1 norm of the gradient, which minimizing it leads to a flatter learning trajectory and, as a result, smaller unlearning and generalization errors. Considering these results, to have unlearning-compatible models, we propose to regularize the training loss with l_1 norm of the gradient, which encourages gradient sparsity during training. Our experiments show that only by using our regularizer in the training stage, the MU performance of various baselines is considerably improved, suggesting the effectiveness of our approach in simplifying the MU problem by training unlearning-compatible models. Moreover, we demonstrate that the benefits of our approach become even more significant when dealing with poisoned data in backdoor attack scenarios. Our investigations show that in such a scenario, gradient sparsity not only simplifies model cleaning but also damps the gradient spikes, which are typically caused by poisonous data and encourage the model to automatically learn less from such poisonous data points.

1 Introduction

The "right to be forgotten" enforced by recent law regulations including the General Data Protection Regulation (GDPR) (27) and the California Consumer Privacy Act (CCPA) (12), enforce that at any time, any user can request to have their data or any knowledge extracted from their data to be removed from any service. Additionally, to mitigate the adverse effect of poisonous data (19) in backdoor attacks (11), we may need to remove the influence of a set of data

points (the forget set) from a trained model, while maintaining model performance on the remaining data (the retain set). This problem is known as machine unlearning (MU) (1). The exact MU is to retrain the model on the retain set from scratch. However, this approach is computationally inefficient as it wastes the model knowledge for the desirable retain set. To mitigate this, approximate MU techniques have been proposed to efficiently erase the effect of the forget set and preserve knowledge from the retain set as much as possible (13; 25; 29). Despite these efforts, a considerable gap still remains in their unlearning performance compared to the exact approach. This performance gap and the natural connection between learning and unlearning have motivated us to study the role of the learning method in the complexity of the subsequent MU task. Theoretical insights on confounding factors in the cost to reverse SGD (unlearning error) in (25) and our experimental investigations show that the performance of a given MU technique can be dependent on the learning scheme. These pushed us to raise the following question: **Can we design a theoretically backed learning scheme that simplifies the unlearning task, meaning that leads to unlearning-compatible models?** Motivated by:

1. The cost for reversing SGD is proportional to the average of the maximum eigenvalues of the Hessian matrix (loss curvature) through learning steps (25).
2. That the loss sharpness provides an upper bound on generalization error (6).

drive us to investigate role of the loss landscape sharpness[3] for developing an unlearning-compatible learning strategy. Intuitively, given that unlearning is the reverse operation of learning, with a flatter learning trajectory, it is easier to reverse SGD (25) (go back through the learning trajectory). Additionally, with flatter learning trajectories, models generalize better (6), i.e., learn less subject-specific information, which implies easier MU. Having these in mind, to build an unlearning-compatible learning strategy, we attempt to train the model through the flattest learning trajectory. We show that the l_1 norm of the gradient, which defines sharpness over l_∞ perturbations, quantifies the maximum sharpness compared with other forms of sharpness definitions. As a result, minimizing it leads to the flattest learning trajectory, the smallest generalization error and subsequently maximum simplification of the unlearning. To achieve this, we propose adding the l_1 norm of the gradient as a regularizer to the main loss, which encourages gradient sparsity. Additionally, the advantage of gradient sparsity for unlearning is even more dominant in the case of poisonous data. This is because poisonous data usually causes spikes in the gradient, and with the help of gradient sparsity loss, these spikes get heavily damped. We show that gradient sparsity can significantly mitigate the adverse effects of poisonous data.

[3] To simplify our theoretical analysis and sake of algorithmic efficiency, in this paper, we use sharpness as a notion of curvature.

2 Problem statement, evaluation metrics, and related work

Let $\boldsymbol{\theta} \in \mathbb{R}^k$ denote the model parameters and $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$ represent the training set where x_i and y_i are input and labels, respectively. In the MU, the objective is to erase the effect of a subset of training data points named forget set $\mathcal{D}_f$ where $\mathcal{D}_f \subseteq \mathcal{D}$, from a trained model while the knowledge over remaining data points (everything except forget set in training set), i.e., retain set, $\mathcal{D}_r$ where $\mathcal{D}_r = \mathcal{D} \setminus \mathcal{D}_f$ is maintained. In the approximate MU, the main goal is that the unlearned model, $\boldsymbol{\theta}_u$, mimics the model performance after exact MU on $\mathcal{D}_f$ and $\mathcal{D}_r$ with significantly smaller time and computational complexity. Given the definition of $\mathcal{D}_f$, which can be a random subset of training data points, a specific class label, or a set of poisonous data points, the MU problem is known as random forgetting, class forgetting, or a defence against backdoor attack, respectively. To evaluate MU methods, different strategies have been proposed (9; 10; 14; 25). Here, we follow the evaluation metrics proposed by **(author?)** (14) as they offer a comprehensive view of the performance of MU methods. Here we briefly introduce these metrics:

- *Forget accuracy (FA)*: FA measures the data influence removal of the MU method, and it is defined as the accuracy of the unlearned model on $\mathcal{D}_f$.
- *Retain accuracy (RA)*: RA quantifies the knowledge protection ability of the MU method, and it is calculated as the accuracy of the unlearned model on $\mathcal{D}_r$.
- *Test accuracy (TA)*: TA measures the generalization performance of the unlearned model, and it is the accuracy of $\boldsymbol{\theta}_u$ on the test set. For the class forgetting scenario, the test set does not include any data points from the forgetting class.
- *Membership inference attack (MIA)*: MIA itself is considered a privacy attack where the objective is to determine if a set of data points has been used for model training. Here, MIA-Efficacy, is calculated using confidence-based MIA predictor (23) and it is defined as, $\frac{TN}{|\mathcal{D}_f|}$ where TN is the number of samples in $\mathcal{D}_f$ that have been correctly predicted as non-member samples. MIA-Efficacy quantifies the knowledge removal ability of the MU method over $\mathcal{D}_f$, and the higher the MIA accuracy, the better the forgetting over $\mathcal{D}_f$.

To have better unlearning, the objective is for the FA, RA, TA, and MIA metrics to be as close as possible to those of the model unlearned with the exact method, i.e., retraining from scratch. To summarize these four metrics, similar to (14), we use the average Gap (Avg. Gap), which is the average of the absolute values of differences for each of the four metrics against the exact method. The lower the Avg. Gap, the better the performance of the MU method.

The MU problem has been approached from different points of view (2; 7; 10; 15; 16; 21; 22). The simplest way to perform MU is to take advantage of the catastrophic forgetting phenomenon in neural networks (28) through Finetuning

(FT) (9; 29). To this end, the model is finetuned on $\mathcal{D}_r$ for multiple steps. Considering MU as a reverse process of learning, another natural idea to do MU is to employ gradient ascent (GA) on the $\mathcal{D}_f$ (10; 25). Knowing that knowledge is stored in model parameters, perturbing those in charge of maintaining knowledge of $\mathcal{D}_f$ is another popular line of work. The perturbation for forgetting can be used in different forms. As some examples, in (30) authors employ noisy data to inject perturbation. In (8), the perturbation is performed in a multiplicative manner on the gradients, and (7) employs the Fisher information matrix to dampen parameters whose contribution to $\mathcal{D}_f$ is larger than $\mathcal{D}_r$. Another form of perturbation proposed in the literature for unlearning is pruning (14), where the authors encouraged weight sparsity and showed that this step simplifies the forgetting. Knowledge distillation is another approach that has been used to transfer $\mathcal{D}_f$ free knowledge to the target model (18). Finally, recent works have exploited loss the landscape structure to develop more effective unlearning methods (4; 20; 24).

3 Method: The unlearning-compatible learning strategy

In section 1, we discussed two main observations from the literature that the cost to reverse SGD is proportional to the accumulated loss curvature (25) and the generalization error (a quantification of subject-specific information the model has learned) is bounded by loss sharpness (6). These two main points motivate us to investigate the role of loss landscape sharpness for developing an unlearning-compatible learning strategy more deeply. Here, first, we state these connections in the form of propositions more formally, and subsequently build the theory behind our proposed learning method that simplifies the unlearning problem.

Proposition 1 *Assuming the SGD is used for training the model with parameters $\boldsymbol{\theta} \in \mathbb{R}^k$ and loss ℓ for t steps, the weight distance between the model after unlearning using gradient ascent and the gold-standard retrained model is bounded by the following expression (25).*

$$e = \eta^2 \frac{\|\boldsymbol{\theta}_t - \boldsymbol{\theta}_0\|_2}{t} \kappa_{avg} \frac{t^2 - t}{2} \tag{1}$$

where η is the learning rate, t is the training step, $\boldsymbol{\theta}_t$ and $\boldsymbol{\theta}_0$ are model parameters at steps t and 0 (initialization), $\nabla^2_{\boldsymbol{\theta}}\ell$ is Hessian matrix, and κ_{avg} is $\kappa_{avg} := \sum_{i=1}^{t} \kappa^i_{max}$ where κ^i_{max} the largest eigenvalue of Hessian matrix at step i.

Considering proposition 1 and results in (6) that a flatter learning trajectory leads to lower unlearning error (cost to reverse SGD) and also generalization error (learned subject-specific information) respectively, to achieve an unlearning-compatible models, we investigate the role of defining sharpness over perturbation norms in simplifying the unlearning problem and obtain unlearning-compatible models.

Proposition 2 (Loss sharpness for different perturbation norms.) *Let p-sharpness be* $\mathcal{S}_p(\theta) := \max_{\|\epsilon\|_p \le \rho} \Big(\ell(\theta+\epsilon) - \ell(\theta)\Big)$*, for a small* ρ *and differentiable* ℓ*, among perturbation norms* $p = \{1, 2, \infty\}$*, the* sharpness *under* $p = \infty$ *perturbations is the largest (worst-case), and its first-order approximation is* $\rho\|\nabla\ell(\theta)\|_1$ *(a gradient sparsity measure). More precisely,*

$$\begin{aligned} &\mathcal{S}_1(\theta) \le \mathcal{S}_2(\theta) \le \mathcal{S}_\infty(\theta), \\ &\mathcal{S}_\infty(\theta) = \max_{\|\epsilon\|_\infty \le \rho} \big(\ell(\theta+\epsilon) - \ell(\theta)\big) = \rho\,\|\nabla\ell(\theta)\|_1 + o(\rho) \approx \rho\,\|\nabla\ell(\theta)\|_1. \end{aligned} \tag{2}$$

Proof. *First-order Taylor expansion gives* $\ell(\theta+\epsilon) - \ell(\theta) = \epsilon^\top \nabla\ell(\theta) + o(\|\epsilon\|)$*, Therefore,*

$$\mathcal{S}_p(\theta) = \max_{\|\epsilon\|_p \le \rho} \epsilon^\top \nabla\ell(\theta) \;+\; o(\rho).$$

Let q *be the dual exponent of* p *(with* $p = 1 \Leftrightarrow q = \infty$ *and* $p = \infty \Leftrightarrow q = 1$*). By Hölder,*

$$\epsilon^\top \nabla\ell(\theta) \le \|\epsilon\|_p \|\nabla\ell(\theta)\|_q \le \rho\|\nabla\ell(\theta)\|_q,$$

So

$$\max_{\|\epsilon\|_p \le \rho} \epsilon^\top \nabla\ell(\theta) \le \rho\|\nabla\ell(\theta)\|_q. \tag{3}$$

This bound is tight (achievable), hence

$$\max_{\|\epsilon\|_p \le \rho} \epsilon^\top \nabla\ell(\theta) = \rho\|\nabla\ell(\theta)\|_q. \tag{4}$$

Instantiating Eq. 4 for $p = 1, 2$*, and* $p = \infty$*, yields* $\rho\,\|\nabla\ell(\theta)\|_\infty$*,* $\rho\,\|\nabla\ell(\theta)\|_2$*, and* $\rho\,\|\nabla\ell(\theta)\|_1$ *firs-order approximations for* $\mathcal{S}_1(\theta)$*,* $\mathcal{S}_2(\theta)$*, and* $\mathcal{S}_\infty(\theta)$*. Finally, since for any* $\nabla\ell(\theta)$*,*

$$\|\nabla\ell(\theta)\|_1 \ge \|\nabla\ell(\theta)\|_2 \ge \|\nabla\ell(\theta)\|_\infty. \tag{5}$$

we conclude

$$\mathcal{S}_1(\theta) \;\le\; \mathcal{S}_2(\theta) \;\le\; \mathcal{S}_\infty(\theta). \tag{6}$$

□

Proposition 2 shows that among three perturbation norms namely ℓ_1, ℓ_2, and ℓ_∞, the maximum sharpness achieved with ℓ_∞ perturbation where its first order approximation is $\|\nabla\ell(\boldsymbol{\theta})\|_1$, i.e., gradient sparsity. It is trivial that minimizing $\|\nabla\ell(\boldsymbol{\theta})\|_1$ as the maximum sharpness (worst case) leads to a flatter learning trajectory compared to other sharpness forms. In proposition 3, we derive the generalization errors for these three perturbation norms, and draw a comparison among them.

Proposition 3 (Generalization error for different perturbation norms) *Let $\ell_{\mathcal{S}}(\theta)$ and $\ell_{\mathcal{D}}(\theta)$ denote the population and empirical bounded losses ($\ell(\cdot) \in [0,1]$) where with probability at least $1-\delta$ over the draw of $\mathcal{D} \sim \mathcal{S}^n$ (the dataset $\mathcal{D}$ consists of n independent samples drawn from $\mathcal{S}$) and the Gaussian perturbation $\epsilon \sim \mathcal{N}(0, \sigma^2 I_k)$ on model parameters does not decrease the loss i.e., $\ell_{\mathcal{S}}(\theta) \leq \mathbb{E}_{\epsilon}\big[\ell_{\mathcal{S}}(\theta+\epsilon)\big]$. Defining the PAC-Bayes complexity term (as in (6)) for $\sigma > 0$ and a fix $\rho > 0$:*

$$\mathcal{C}(\theta, \sigma; n, k, \delta) := \sqrt{\frac{\frac{k}{4}\log\Big(1 + \frac{\|\theta\|_2^2}{k\sigma^2}\Big) + \frac{1}{4} + \log\frac{n}{\delta} + 2\log(6n+3k)}{n-1}}. \tag{7}$$

If we set tail weight $\delta' := \frac{1}{\sqrt{n}}$, $t := \log\frac{1}{\delta'} = \frac{1}{2}\log n$, then for each norm, it is possible to choose $\sigma_p(\rho)$ so that $\Pr(\|\epsilon\|_p \leq \rho) \geq 1-\delta'$, then for $p \in \{1, 2, \infty\}$, the following three bounds hold simultaneously:

$$\ell_{\mathcal{S}}(\theta) \leq \underbrace{\ell_{\mathcal{D}}(\theta) + \max_{\|\epsilon\|_p \leq \rho} \big(\ell_{\mathcal{D}}(\theta+\epsilon) - \ell_{\mathcal{D}}(\theta)\big)}_{\substack{\textit{empirical loss + sharpness in } \ell_p \\ \textit{form proposition 2, worst-case} \\ \textit{is minimized by gradient sparsity}}} + \underbrace{F_p(\|\theta\|_2^2, \rho^2, n, k, \delta)}_{\substack{\textit{Additive PAC-Bayes term} \\ \textit{smallest for } \ell_\infty\textit{-sharpness}}}, \tag{8}$$

where for the $k \geq 5$

$$\begin{aligned} F_\infty(\cdot) \leq F_2(\cdot) \leq F_1(\cdot), \\ F_p(\|\theta\|_2^2, \rho^2, n, k, \delta) := \delta' + \mathcal{C}\big(\theta, \sigma_p(\rho); n, k, \delta\big). \end{aligned} \tag{9}$$

so for fixed ρ, n, k, δ, the additive PAC-Bayes term is smallest for ℓ_∞-sharpness and largest for ℓ_1-sharpness.

Proof. *To show this, we start from PAC-Bayes bound on sharpness for ℓ_2 norm in (6) and using concentration inequalities in (17; 26) we derive generalization errors for sharpness defined over ℓ_1 and ℓ_∞ perturbation norms. The proof is provided in Sec. 11.2 in the supplementary material.* □

Proposition 3 reveals that the additive PAC-Bayes term is smallest for the ℓ_∞-sharpness. Additionally, from proposition 2 we know that gradient sparsity regularization minimizes the worst-case sharpness obtained by defining sharpness in ℓ_∞ and as a result leads to the flattest learning trajectory compared with other studied perturbation norms. These results from propositions 2 and 3 indicate that minimizing gradient sparsity leads to the smallest generalization error (see Eq. 8) and subsequently learning the least subject-specific information. Furthermore, from proposition 1, we know that the flatter the learning trajectory, the easier to unlearn (reverse SGD). Considering these findings, to achieve the flattest learning trajectory (to have unlearning-compatible models), we propose to add gradient sparsity as a regularization term to the main loss. More precisely, our proposed unlearning-compatible loss represented by $\ell_{\mathcal{D}}^{unc}(\boldsymbol{\theta})$ is

$$\ell_{\mathcal{D}}^{unc}(\boldsymbol{\theta}) = \ell_{\mathcal{D}}(\boldsymbol{\theta}) + \rho \|\nabla \ell_{\mathcal{D}}(\boldsymbol{\theta})\|_1, \tag{10}$$

where ρ is the regularization parameter controlling the maximum perturbation norm of sharpness in proposition 2. In the following, we empirically validate our theoretical findings and show that using the loss function proposed in Eq. 10 for training simplifies the unlearning problem. In proposition 4, in the supplementary material, we derived a closed-form expression for the Hessian of $\ell_{\mathcal{D}}^{unc}(\boldsymbol{\theta})$ in terms of $\ell_{\mathcal{D}}$ for the quadratic loss function $\ell(\boldsymbol{\theta}) = \frac{1}{2}\boldsymbol{\theta}^\top H \boldsymbol{\theta}$ where H is the hessian matrix and explicitly see how this gradient regularization affect eigenvalues of H.

4 Results

In this section, we provide experimental results that validate the ability of our proposed gradient regularization method in improving the unlearning-compability of models. For the details of experimental setups, please see Sec. 11.1 in the supplementary material.

4.1 Gradient sparsity improves unlearning-compatibility

To study how employing gradient l_1 regularization contributes to unlearning-compatibility, we consider the following experiment. We train a ResNet18 model on CIFAR-10 in two scenarios, namely with and without gradient l_1 regularization. Next, setting data points from a class as a forget set, we employ the FT method, which is fine-tuning the model after excluding the forget set as the most natural unlearning technique. For the case that the gradient regularization is used, we train the model for a range of different values for ρ and monitor the unlearning metrics. Figure 1 shows the result of this experiment, where clearly applying the FT unlearning technique to models trained with gradient sparsity regularization leads to lower Avg. Gap and, equivalently, better unlearning performance. To see more closely how each of the four unlearning metrics is affected, in Figure 2, we report them for both retrain (gold standard) and FT methods. Clearly, for the forget accuracy and MIA efficacy, employing gradient sparsity significantly reduces the gap between FT and retrain, while for the other two metrics, namely retain and test accuracy, the gap is kept within an acceptable range, resulting in an overall smaller Avg. Gap and better MU performance.

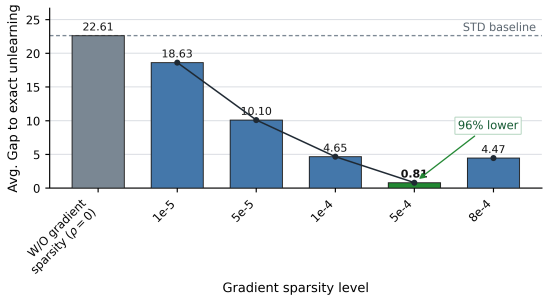
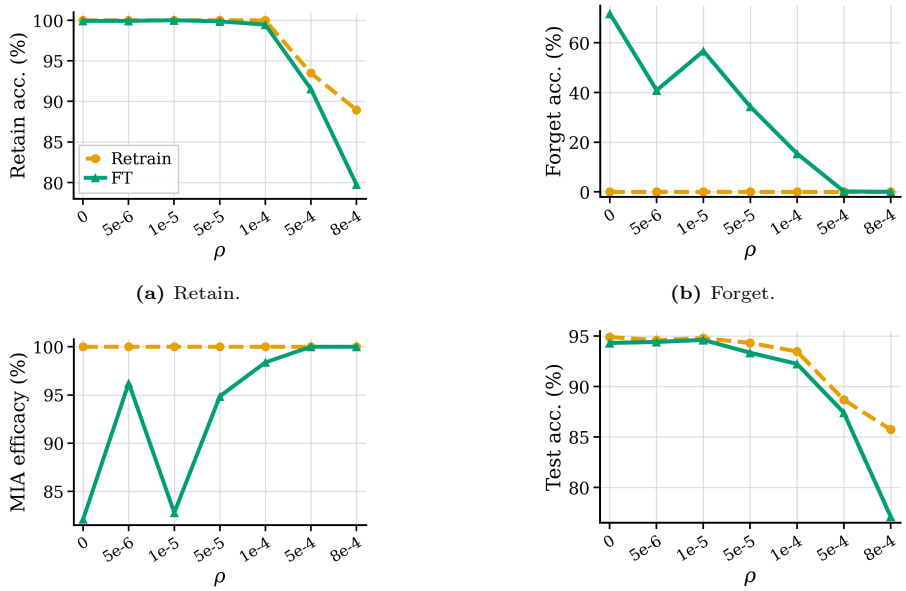


Fig. 1: Avg. Gap (lower is better) of four metrics between FT unlearning and the retrain (exact method) baseline across different levels of gradient sparsity (ρ).



(a) Retain. **(b)** Forget. **(c)** MIA. **(d)** Test.

Fig. 2: Metric-wise comparisons between retraining and FT across gradient sparsity (ρ) levels; closer alignment indicates better unlearning.

To validate the contribution of gradient sparsity in simplifying the MU problem for other MU methods, we report the above experiments for a range of MU baselines including weight sparsity (WS) (14), GA (10; 25), IU (13), FF (9), and FT (9; 29). The experiments are conducted in two MU scenarios, i.e., class and random forgetting. For the class-forgetting a full class is considered as the

Table 1: MU performance of multiple benchmarks for ResNet-18 trained on CIFAR-10 with gradient sparsity (**GS**) and without it, i.e., standard training (STD).

MU	RA		FA		MIA-Efficacy		TA		Avg. Gap	
	STD	**GS**	STD	**GS**	STD	**GS**	STD	**GS**	STD	**GS**
Random-forgetting										
Retrain	100.0	100.0	94.6	94.51	12.73	11.82	94.23	94.14	0.0	0.0
WS	88.36 (11.64)	96.78 (3.22)	86.22 (8.38)	94.36 (0.16)	15.44 (2.71)	10.64 (1.18)	85.47 (8.76)	90.46 (3.68)	7.87	**2.06**
GA	98.88 (1.12)	99.41 (0.59)	98.31 (3.71)	99.11 (4.6)	2.76 (9.98)	1.44 (10.38)	93.07 (1.16)	94.23 (0.09)	3.99	**3.91**
IU	99.38 (0.62)	96.4 (3.6)	99.18 (4.58)	96.24 (1.73)	2.04 (10.69)	6.76 (5.07)	93.93 (0.3)	89.18 (4.96)	4.05	**3.84**
FF	86.42 (13.58)	86.63 (13.37)	86.02 (8.58)	86.02 (8.42)	20.64 (7.91)	18.69 (6.73)	80.98 (13.25)	80.74 (13.82)	10.83	**10.59**
FT	99.91 (0.09)	88.5 (5.1)	98.96 (4.36)	86.8 (1.64)	2.91 (9.82)	20.07 (2.53)	94.25 (0.02)	83.81 (4.19)	3.57	**3.37**
Class-forgetting										
Retrain	100.0	100.0	0.0	0.0	100.0	100.0	94.9	94.59	0.0	**0.0**
WS	87.1 (12.9)	87.39 (6.1)	0.0 (0.0)	0.0 (0.0)	100.0 (0.0)	100.0 (0.0)	84.89 (10.01)	84.08 (4.6)	5.73	**2.68**
GA	11.11 (88.89)	11.11 (88.89)	0.0 (0.0)	0.0 (0.0)	0.0 (100.0)	100.0 (0.0)	11.11 (83.79)	11.11 (83.71)	68.17	**43.15**
IU	99.47 (0.53)	93.71 (6.29)	90.02 (90.02)	6.58 (6.58)	28.33 (71.67)	97.29 (2.71)	94.67 (0.23)	86.8 (7.47)	40.61	**5.76**
FF	99.51 (0.49)	99.34 (0.66)	11.04 (11.04)	0.02 (0.02)	100.0 (0.0)	100.0 (0.0)	94.87 (0.03)	93.7 (0.24)	2.89	**0.23**
FT	99.92 (0.08)	91.61 (1.88)	71.84 (71.84)	0.13 (0.13)	82.04 (17.96)	100.0 (0.0)	94.34 (0.56)	87.46 (1.22)	22.61	**0.81**

Table 2: MU performance of multiple benchmarks for VGG16 trained on CIFAR-10 gradient sparsity regularization (**GS**) and without it, i.e., standard training (STD).

MU	RA		FA		MIA-Efficacy		TA		Avg. Gap	
	STD	**GS**	STD	**GS**	STD	**GS**	STD	**GS**	STD	**GS**
Random data forgetting										
Retrain	99.99	100.0	92.71	93.18	11.71	13.02	92.81	92.9	0.0	0.0
WS	63.36 (36.62)	10.02 (0.1)	62.87 (29.84)	9.84 (0.87)	58.09 (46.38)	31.22 (30.6)	62.71 (30.1)	10.0 (0.0)	35.74	**7.89**
GA	99.35 (0.64)	9.99 (0.13)	99.18 (6.47)	9.96 (0.98)	1.6 (10.11)	58.38 (3.44)	93.1 (0.29)	10.17 (0.17)	4.38	**1.18**
IU	99.33 (0.66)	10.64 (0.53)	98.87 (6.16)	10.02 (1.04)	1.82 (9.89)	60.31 (1.51)	92.88 (0.07)	10.87 (0.87)	4.19	**0.99**
FF	89.64 (10.35)	9.82 (0.29)	88.98 (3.73)	9.4 (0.42)	18.02 (6.31)	69.49 (7.67)	83.23 (9.58)	9.81 (0.19)	7.49	**2.14**
FT	99.57 (0.41)	99.62 (0.38)	98.44 (5.73)	98.84 (5.67)	3.84 (7.87)	3.47 (9.56)	92.66 (0.15)	91.95 (0.95)	**3.54**	4.14
Class forgetting										
Retrain	99.99	100.0	0.0	0.0	100.0	100.0	93.56	93.77	0.0	0.0
WS	60.31 (39.68)	85.17 (14.83)	89.56 (89.56)	0.0 (0.0)	14.67 (85.33)	100.0 (0.0)	59.5 (34.06)	83.08 (10.69)	62.16	**6.38**
GA	92.15 (7.84)	10.11 (1.0)	44.0 (44.0)	0.22 (0.22)	64.67 (35.33)	0.22 (0.22)	84.95 (8.61)	10.03 (1.08)	23.94	**0.63**
IU	99.34 (0.66)	10.06 (1.05)	99.33 (99.33)	0.0 (0.0)	1.56 (98.44)	0.67 (0.67)	93.07 (0.49)	10.16 (0.95)	49.73	**0.67**
FF	97.99 (2.0)	9.95 (1.16)	99.33 (99.33)	0.0 (0.0)	2.0 (98.0)	0.0 (0.0)	90.73 (2.83)	9.86 (1.25)	50.54	**0.6**
FT	96.8 (3.19)	99.54 (0.46)	97.56 (97.56)	50.29 (50.29)	6.22 (93.78)	88.07 (11.93)	89.65 (3.91)	92.83 (0.93)	49.61	**15.9**

forget set, while in the random-forgetting, 10% of data points were randomly selected as the forget set. The results of this experiment are presented in Table 1, where clearly gradient sparsity regularization reduced the average GAP for all baselines. In Tables 2, 3, 4 and 5 we repeated the same evaluation protocol for ResNet-18-CIFAR-100, VGG16-CIFAR-10, ViT(s)-SVHN , and ResNet-18-ImageNet model&dataset pairs. The results show that in most cases, training models with our proposed gradient l_1 norm regularization improves the unlearning performance of different MU methods across different model architectures and datasets.

4.2 Gradient sparsity improves unlearning-compatibility better than other forms of gradient regularization

Proposition 1, motivated us that minimizing the training loss with a flatter learning trajectory can reduce the unlearning error (quantified by Eq. 1) and

Table 3: MU performance of multiple benchmarks for ResNet-18 trained on CIFAR-100 with gradient sparsity regularization (**GS**) and without it, i.e., standard training (STD).

MU	RA		FA		MIA-Efficacy		TA		Avg. Gap	
	STD	**GS**	STD	**GS**	STD	**GS**	STD	**GS**	STD	**GS**
Random data forgetting										
Retrain	99.97	99.98	76.18	76.42	49.18	46.22	74.82	75.26	0.0	0.0
WS	17.21 (82.76)	24.07 (75.91)	17.07 (59.11)	23.6 (52.82)	87.82 (38.64)	80.16 (33.94)	16.65 (58.17)	23.49 (51.77)	59.67	**53.61**
GA	97.5 (2.47)	97.54 (2.44)	97.69 (21.51)	97.76 (21.34)	6.31 (42.87)	5.38 (40.84)	75.47 (0.65)	76.02 (0.76)	16.87	**16.34**
IU	97.52 (2.45)	97.1 (2.88)	97.38 (21.2)	96.51 (20.09)	6.69 (42.49)	8.36 (37.86)	75.14 (0.32)	73.49 (1.77)	16.61	**15.65**
FF	97.44 (2.53)	97.5 (2.48)	97.58 (21.4)	97.69 (21.27)	6.96 (42.22)	6.51 (39.71)	74.9 (0.08)	75.37 (0.11)	16.56	**15.89**
FT	99.93 (0.04)	99.87 (0.11)	97.4 (21.22)	97.38 (20.96)	11.07 (38.11)	10.31 (35.91)	75.77 (0.95)	75.66 (0.4)	15.08	**14.34**
Class forgetting										
Retrain	99.98	99.1	0.0	0.0	100.0	100.0	74.28	74.54	0.0	0.0
WS	99.78 (0.2)	98.91 (0.19)	79.11 (79.11)	46.89 (46.89)	75.78 (24.22)	84.44 (15.56)	74.53 (0.25)	73.43 (1.11)	25.94	**15.94**
GA	81.98 (18.0)	79.79 (19.31)	9.11 (9.11)	4.22 (4.22)	96.89 (3.11)	97.56 (2.44)	59.99 (14.29)	59.77 (14.77)	11.13	**10.19**
IU	97.42 (2.56)	97.12 (1.98)	84.89 (84.89)	35.11 (35.11)	58.89 (41.11)	86.67 (13.33)	74.88 (0.6)	73.71 (0.83)	32.29	**12.81**
FF	94.43 (5.55)	6.27 (92.83)	98.22 (98.22)	0.0 (0.0)	4.89 (95.11)	100.0 (0.0)	68.2 (6.08)	5.19 (69.35)	51.24	**40.54**
FT	99.79 (0.19)	98.67 (0.43)	79.11 (79.11)	46.0 (46.0)	78.22 (21.78)	87.56 (12.44)	74.61 (0.33)	73.33 (1.21)	25.35	**15.02**

Table 4: MU performance of multiple benchmarks applied on ViT small trained on SVHN with gradient sparsity regularization (**GS**) and without it, i.e., standard training (STD).

MU	RA		FA		MIA-Efficacy		TA		Avg. Gap	
	STD	**GS**	STD	**GS**	STD	**GS**	STD	**GS**	STD	**GS**
Random data forgetting										
Retrain	91.27	94.11	86.62	88.98	29.29	22.04	84.05	86.69	0.0	0.0
WS	92.12 (0.85)	92.17 (1.94)	89.44 (2.82)	89.6 (0.62)	24.04 (5.25)	24.51 (2.47)	84.1 (0.05)	85.42 (1.27)	2.24	**1.58**
GA	93.71 (2.44)	93.9 (0.21)	94.02 (7.4)	93.93 (4.95)	17.4 (11.89)	17.69 (4.35)	85.36 (1.31)	87.22 (0.53)	5.76	**2.51**
IU	93.71 (2.44)	93.9 (0.21)	94.0 (7.38)	93.93 (4.95)	17.42 (11.87)	17.67 (4.37)	85.36 (1.31)	87.22 (0.53)	5.75	**2.52**
FF	93.62 (2.35)	93.89 (0.22)	93.93 (7.31)	93.82 (4.84)	17.51 (11.78)	17.82 (4.22)	85.34 (1.29)	87.1 (0.41)	5.68	**2.42**
FT	92.13 (0.86)	92.18 (1.93)	89.42 (2.8)	89.6 (0.62)	24.04 (5.25)	24.53 (2.49)	84.11 (0.06)	85.43 (1.26)	2.24	**1.58**
Class forgetting										
Retrain	94.72	93.41	0.0	0.0	100.0	100.0	87.16	86.49	0.0	0.0
WS	92.84 (1.88)	92.95 (0.46)	12.01 (12.01)	6.76 (6.76)	99.71 (0.29)	100.0 (0.0)	84.4 (2.76)	86.85 (0.36)	4.24	**1.89**
GA	93.67 (1.05)	87.97 (5.44)	94.75 (94.75)	87.47 (87.47)	16.14 (83.86)	26.87 (73.13)	85.47 (1.69)	81.16 (5.33)	45.34	**42.62**
IU	93.75 (0.97)	45.84 (47.57)	91.02 (91.02)	13.65 (13.65)	24.0 (76.0)	88.84 (11.16)	85.73 (1.43)	41.58 (44.91)	42.36	**29.1**
FF	93.46 (1.26)	87.93 (5.48)	94.39 (94.39)	87.18 (87.18)	17.0 (83.0)	28.36 (71.64)	85.46 (1.7)	81.05 (5.44)	45.09	**42.22**
FT	92.85 (1.87)	92.97 (0.44)	12.08 (12.08)	6.94 (6.94)	99.71 (0.29)	100.0 (0.0)	84.4 (2.76)	86.86 (0.37)	4.25	**1.94**

improve the unlearning-compatibility. In propositions 2 and 3, we showed that the gradient ℓ_1 norm quantifies the worst-case sharpness and, as a result, encouraging gradient sparsity leads to the flattest learning trajectory and subsequently smaller unlearning and generalization error compared with sharpness definitions. These results suggest that gradient ℓ_1 norm regularization should simplify the unlearning task better than other gradient norms and achieve the most unlearning-compatible models. To validate this theoretical finding in practice, we repeat the experiment in the Figure 1 for model trained with $\|\nabla\ell(\boldsymbol{\theta})\|_2$ regularization, which is the result of defining sharpness over $\|\cdot\|_2$ perturbation norm. The result of this experiment is reported in Figure 3, which shows that $\|\nabla\ell(\boldsymbol{\theta})\|_1$ (gradient sparsity) has increased the unlearning-compatibility more than $\|\nabla\ell(\boldsymbol{\theta})\|_2$ regularizer, which supports our theoretical results.

Table 5: Class-forgetting MU performance on ImageNet (ResNet-18) with gradient sparsity regularization (**GS**) and without it, i.e., standard training (STD).

MU	RA		FA		MIA-Efficacy		TA		Avg. Gap	
	STD	**GS**	STD	**GS**	STD	**GS**	STD	**GS**	STD	**GS**
Retrain	71.75	71.21	0.0	0.0	100.0	100.0	69.49	67.63	0.0	0.0
GA	65.93 (5.82)	59.05 (12.15)	14.90 (14.90)	0.0 (0.0)	87.46 (12.54)	99.62 (0.38)	64.62 (4.87)	57.28 (10.35)	9.53	**5.72**
FT	72.45 (0.70)	70.21 (0.99)	36.40 (36.40)	28.92 (28.92)	68.61 (31.39)	64.23 (35.77)	69.80 (0.31)	66.08 (1.55)	17.20	**16.81**

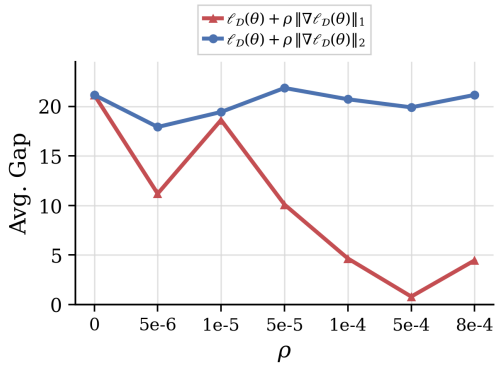
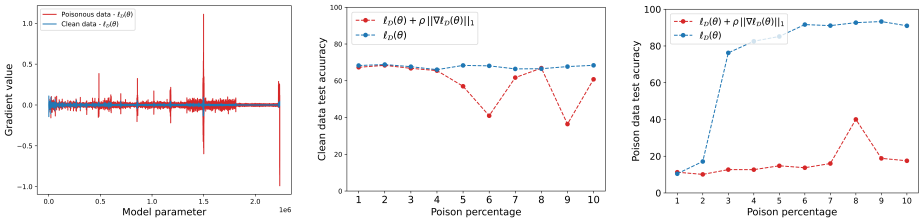


Fig. 3: Comparison of gradient ℓ_1 and ℓ_2 regularization in improving unlearning-compatibility; Avg. Gap (lower is better) of four metrics between FT unlearning and the retrain (exact method) baseline across different (ρ) values for models trained with gradient ℓ_1 and ℓ_2 regularization.

5 Gradient sparsity mitigates the adverse effect of poisonous data in backdoor attack beyond unlearning

We showed that our proposed gradient sparsity regularization can improve the unlearning-compability. Clearly, this is helpful in the case of a backdoor attack as it simplifies the process of removing poisonous data from the model. Here, we demonstrate that the advantage of the proposed gradient sparsity regularization under scenarios where the data is poisoned (backdoor attack) is even more pronounced. This is because poisonous data points cause huge spikes in the gradients as they often lead to a shift in the behaviour of the model over some already learned representations. To show this, the MobileNetV2 model is trained (from scratch for 10 epochs) on a poisoned CIFAR-10 dataset where 3% of the training set from one class is poisoned by the backdoor attack proposed by (11). After training, we visualized the gradient values over the clean and poisoned portions of the test set in Figure 4a. It is clear that gradient values over the poisoned portion of the dataset (red graph) contain larger magnitudes with aggressive spikes. Given this, we argue that the proposed gradient sparsity is even more helpful as it acts as a dampening module that suppresses the gradient spikes caused by poisonous data. This characteristic reduces the poisonous information received by the model, and subsequently, the model shows more ro-



(a) **(b)** **(c)**

Fig. 4: (a): Gradient values per parameter for the MobileNetV2 over clean and poisoned test sets. The model is trained on CIFAR-10 using standard loss $\ell_{\mathcal{D}}(\boldsymbol{\theta})$, where 3% of the training set is poisoned. (b) and (c): Test performance comparison of MobileNetV2 trained on poisoned CIFAR-10 for different poison percentages using standard training, $\ell_{\mathcal{D}}(\boldsymbol{\theta})$, and our proposed gradient sparsity regularization, $\ell_{\mathcal{D}}(\boldsymbol{\theta}) + \rho \|\nabla \ell_{\mathcal{D}}(\boldsymbol{\theta})\|_1$.

Table 6: Performance comparison of MobileNetV2 trained on poisoned CIFAR-10 using the standard training and our proposed gradient sparsity regularization on test data.

Method	Clean test acc. ↑	Poison test acc.↓
$\ell_{\mathcal{D}}(\boldsymbol{\theta})$	**68.19**	86.62
$\ell_{\mathcal{D}}(\boldsymbol{\theta}) + \rho \|\nabla \ell_{\mathcal{D}}(\boldsymbol{\theta})\|_1$	68.1	**24.63**

bustness to backdoor attacks by learning less from poisonous data. To validate this, we evaluate the performance of the poisoned MobileNetV2 model where 3% of the training set from one class is poisoned, trained with and without gradient sparsity on the clean and poisoned portion of the test set in the Table 6. In this Table, Clean test acc. and Poison test acc. represent the accuracy of the model on the clean and poisonous partitions of the dataset, respectively. Ideally, we want to maintain maximum Clean test acc. while Poison test acc. is as low as possible, meaning that the model is affected less by poisonous data while the knowledge on clean data is still preserved. As you can see from Table 6, comparing the model trained with our gradient sparsity regularization against standard training, while their accuracies are on par over clean data, for the poisonous data, the accuracy of the model trained with gradient sparsity is much lower (24.63% against 86.62%). This suggests that our proposed gradient sparsity can encourage the model to learn less from poisonous data points (by damping gradient spikes) without significant harm to clean data accuracy. To see the ability of our proposed regularization in filtering poisonous signals, we repeated this experiment for different percentages of poisonous data points and the results are reported in Figure 4b and 4b for accuracy over clean and poisonous data. As can be seen from Figure 4c, for different levels of poisonousness in the data, our proposed regularizer pushes the model to learn less malicious knowledge, while from Figure 4b, the ability to learn from clean data is almost preserved.

6 Memory and computational load

So far, we showed that gradient ℓ_1 regularization and simplify unlearning task and also gives us additional protection against poisonous data. However, atleast in our current implementation, this improved unlearning compatibility and robustness against poisonous data is not for free. More precisely, compared to standard training, optimizing $\ell_{\mathcal{D}}(\theta) + \rho\|\nabla_\theta \ell_{\mathcal{D}}(\theta)\|_1$ requires *double backprop* per step to calculate a Hessian–vector product where one reverse-mode pass is needed to compute $\nabla_\theta \ell_{\mathcal{D}}(\theta)$ with graph retention and a second reverse-mode pass to backpropagate through the gradient-norm term. As a result, the per-step, the proposed loss needs one forward pass (F) and two backward passes ($2B$) that yield $\approx F{+}2B$ compared with $F{+}B$ for standard training. Peak memory usage also increases because of additional activations and retaining the autograd state required for second backprop. The time/memory overhead in Table 7. Although the exact implementation of our proposed regularizer is more expensive than standard training, there are wide range of tricks that can be used to reduce the memory and computational load. For example, we can replace the exact gradient-norm term by a finite-difference approximation along the maximizing dual direction. More precisely, with $v = \operatorname{sign}(g)$ (where $g = \nabla_\theta \ell_{\mathcal{D}}(\theta)$) we can write:

$$\|g\|_1 = \max_{\|v\|_\infty \le 1} v^\top g \approx \frac{\ell_{\mathcal{D}}(\theta + \varepsilon v) - \ell_{\mathcal{D}}(\theta)}{\varepsilon}, \tag{11}$$

which yields a two-backward pass update using $\theta' = \theta + \varepsilon \operatorname{sign}(g)$ without the need of keeping the computational graph. This scheme totally removes the memory overhead. Other tricks include partial regularization (i.e., apply on a parameter subset, e.g., classifier head or every T steps). Finally, one can extend the trajectory-based sharpness proxies similar to the one proposed in the recent work "sharpness aware minimization for free" paper (3) that can substantially reduce the overhead of our regularizer. Exploring these ideas to improve the efficiency of our proposed gradient regularization and their consequences on unlearning-compatibility is beyond the scope of this paper, and we leave it for future work.

Table 7: Peak memory & compute overhead vs. standard training.

Training	**Compute/step**	**Peak mem.**
Standard	$F{+}B$ (1.0×)	1.0×
Grad sparsity	$F{+}2B$ (≈1.5–2.0×)	≈1.2–2.0×

7 Experimental validation of the finite-difference approximation.

We implemented **the finite-difference approximation** and compared it with the exact GS. Table 8 shows that on CIFAR-10 class forgetting, the approxima-

tion preserves the main unlearning benefit of GS while avoiding the higher-order graph required by exact GS.

Table 8: Comparison of exact and finite-difference GS on CIFAR-10 class forgetting. Lower GAP is better.

$\rho/10^{-4}$	Exact GS GAP ↓	Finite-Diff GS GAP ↓
0	22.61	22.61
1	4.65	9.04
5	**0.81**	1.49

8 Comparison with simple regularization techniques

One question that may be asked is whether such simplification in MU can be achieved using simple regularization methods like weight decay and dropout. To answer this question, we note that such regularization techniques are already used in the standard training pipeline in all experiments in this paper. For example, in Figure 1, for both cases, i.e., $\rho = 0$ and $\rho \neq 0$, the standard regularization techniques are part of the training pipeline. This suggests that our proposed gradient sparsity encourages unlearning-compatibility beyond the off-the-shelf standard regularizers.

9 Results on image generation task

To evaluate GS beyond image classification, we train standard and GS-regularized DDPMs and apply the same SalUn (5) unlearning method that supports unlearning for generative models to forget class 0 (airplane). Table 9 reports forget accuracy, where lower is better, and FID, where lower indicates better retained-class generation quality. Under the same unlearning budget, the GS-trained DDPM achieves stronger forgetting while maintaining comparable or better FID.

Table 9: Image generation unlearning: Class-0 forgetting with SalUn unlearning. Lower forgotten-class accuracy and lower FID are better.

Dataset	Training	Forget Acc. ↓	FID ↓
CIFAR-10	STD DDPM	0.854	37.89
	GS DDPM	**0.706**	**35.21**

10 Conclusion

In this paper, motivated by the natural connection between learning and unlearning, we raised the following question: **Can we design a learning scheme that simplifies the unlearning task?** Inspired by the findings that the cost to reverse SGD is proportional to the accumulated loss curvature through learning steps (25), and that the loss sharpness provides an upper bound on generalization error (a quantification of the amount of learned subject-specific information) (6), to maximize the unlearning-compatibility, we proposed to train the model through a flatter learning trajectory. To this end, we studied the role of perturbation norm in the definition of sharpness and theoretically showed that defining sharpness over ℓ_∞ perturbation leads to maximum sharpness quantified by $\|\nabla \ell_{\mathcal{D}}(\boldsymbol{\theta})\|_1$ and as a result minimizing it achieve smallest generalization and unlearning error. With this result, we proposed our unlearning-compatible loss $\ell_{\mathcal{D}}^{unc}(\boldsymbol{\theta}) = \ell_{\mathcal{D}}(\boldsymbol{\theta}) + \rho \|\nabla \ell_{\mathcal{D}}(\boldsymbol{\theta})\|_1$ which adds the ℓ_1 norm of the gradient as a regularizer to the training loss. Our experimental results show that such gradient sparsity regularization consistently improves the unlearning performance for multiple baselines, meaning that the unlearning-compatibility of the models has increased. Additionally, we showed that in case the training data contains poisonous data points, the $\|\nabla \ell_{\mathcal{D}}(\boldsymbol{\theta})\|_1$ acts as a filter and damps the signal (gradient spikes) from such data points. This increases the model's robustness to poisonous data points when it is faced with a backdoor attack, which highlights the advantage of the proposed regularization beyond unlearning.

Bibliography

[1] Cao, Y., Yang, J.: Towards making systems forget with machine unlearning. In: 2015 IEEE Symposium on Security and Privacy. pp. 463–480. IEEE (2015)

[2] Chen, M., Gao, W., Liu, G., Peng, K., Wang, C.: Boundary unlearning: Rapid forgetting of deep networks via shifting the decision boundary. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7766–7775 (2023)

[3] Du, J., Zhou, D., Feng, J., Tan, V., Zhou, J.T.: Sharpness-aware training for free. Advances in Neural Information Processing Systems **35**, 23439–23451 (2022)

[4] Fan, C., Jia, J., Zhang, Y., Ramakrishna, A., Hong, M., Liu, S.: Towards llm unlearning resilient to relearning attacks: A sharpness-aware minimization perspective and beyond. arXiv preprint arXiv:2502.05374 (2025)

[5] Fan, C., Liu, J., Zhang, Y., Wong, E., Wei, D., Liu, S.: Salun: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation. In: International Conference on Learning Representations. vol. 2024, pp. 53643–53673 (2024)

[6] Foret, P., Kleiner, A., Mobahi, H., Neyshabur, B.: Sharpness-aware minimization for efficiently improving generalization. arXiv preprint arXiv:2010.01412 (2020)

[7] Foster, J., Schoepf, S., Brintrup, A.: Fast machine unlearning without retraining through selective synaptic dampening. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 12043–12051 (2024)

[8] Gao, X., Ma, X., Wang, J., Sun, Y., Li, B., Ji, S., Cheng, P., Chen, J.: Verifi: Towards verifiable federated unlearning. IEEE Transactions on Dependable and Secure Computing (2024)

[9] Golatkar, A., Achille, A., Soatto, S.: Eternal sunshine of the spotless net: Selective forgetting in deep networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9304–9312 (2020)

[10] Graves, L., Nagisetty, V., Ganesh, V.: Amnesiac machine learning. In: Proceedings of the AAAI conference on artificial intelligence. vol. 35, pp. 11516–11524 (2021)

[11] Gu, T., Dolan-Gavitt, B., Garg, S.: Badnets: Identifying vulnerabilities in the machine learning model supply chain. arXiv preprint arXiv:1708.06733 (2017)

[12] Harding, E.L., Vanto, J.J., Clark, R., Hannah Ji, L., Ainsworth, S.C.: Understanding the scope and impact of the california consumer privacy act of 2018. Journal of Data Protection & Privacy **2**(3), 234–253 (2019)

[13] Izzo, Z., Smart, M.A., Chaudhuri, K., Zou, J.: Approximate data deletion from machine learning models. In: International Conference on Artificial Intelligence and Statistics. pp. 2008–2016. PMLR (2021)

[14] Jia, J., Liu, J., Ram, P., Yao, Y., Liu, G., Liu, Y., Sharma, P., Liu, S.: Model sparsity can simplify machine unlearning. Advances in Neural Information Processing Systems **36**, 51584–51605 (2023)
[15] Koh, P.W., Liang, P.: Understanding black-box predictions via influence functions. In: International conference on machine learning. pp. 1885–1894. PMLR (2017)
[16] Kurmanji, M., Triantafillou, P., Hayes, J., Triantafillou, E.: Towards unbounded machine unlearning. Advances in neural information processing systems **36**, 1957–1987 (2023)
[17] Laurent, B., Massart, P.: Adaptive estimation of a quadratic functional by model selection. Annals of statistics pp. 1302–1338 (2000)
[18] Li, Y., Chen, C., Zheng, X., Zhang, J.: Federated unlearning via active forgetting. arXiv preprint arXiv:2307.03363 (2023)
[19] Liu, Y., Fan, M., Chen, C., Liu, X., Ma, Z., Wang, L., Ma, J.: Backdoor defense with machine unlearning. In: IEEE INFOCOM 2022-IEEE conference on computer communications. pp. 280–289. IEEE (2022)
[20] Malekmohammadi, S., Xiong, L., et al.: Sharpness-aware parameter selection for machine unlearning. arXiv preprint arXiv:2504.06398 (2025)
[21] Mu, S., Klabjan, D.: Rewind-to-delete: Certified machine unlearning for nonconvex functions. arXiv preprint arXiv:2409.09778 (2024)
[22] Neel, S., Roth, A., Sharifi-Malvajerdi, S.: Descent-to-delete: Gradient-based methods for machine unlearning. In: Algorithmic Learning Theory. pp. 931–962. PMLR (2021)
[23] Song, L., Shokri, R., Mittal, P.: Privacy risks of securing machine learning models against adversarial examples. In: Proceedings of the 2019 ACM SIGSAC conference on computer and communications security. pp. 241–257 (2019)
[24] Tang, H., Khanna, R.: Sharpness-aware machine unlearning. arXiv preprint arXiv:2506.13715 (2025)
[25] Thudi, A., Deza, G., Chandrasekaran, V., Papernot, N.: Unrolling sgd: Understanding factors influencing machine unlearning. arXiv preprint arXiv:2109.13398 (2021)
[26] Vershynin, R.: High-dimensional probability: An introduction with applications in data science, vol. 47. Cambridge university press (2018)
[27] Voigt, P., Von dem Bussche, A.: The eu general data protection regulation (gdpr). A Practical Guide, 1st Ed., Cham: Springer International Publishing **10**(3152676), 10–5555 (2017)
[28] Wang, L., Zhang, X., Su, H., Zhu, J.: A comprehensive survey of continual learning: Theory, method and application. IEEE transactions on pattern analysis and machine intelligence **46**(8), 5362–5383 (2024)
[29] Warnecke, A., Pirch, L., Wressnegger, C., Rieck, K.: Machine unlearning of features and labels. arXiv preprint arXiv:2108.11577 (2021)
[30] Xu, R.Z., Hong, S.Y., Chi, P.W., Wang, M.H.: A revocation key-based approach towards efficient federated unlearning. In: 2023 18th Asia Joint Conference on Information Security (AsiaJCIS). pp. 17–24. IEEE (2023)