Towards Source-Free Machine Unlearning

Supplementary Material

Supplementary Overview:

Contents

1. Proof for Lemma 1 in more details

Proof. From the definition of $\Psi(\mathrm{H})$:

$$\Psi(\mathrm{X}) = \mathbb{E}_{\delta w \sim \mathcal{N}(\mathbf{0},\mathbf{I})} \left[(\frac{1}{2}\delta w^\top \mathrm{X} \delta w + \nabla_r^\top \delta w - \delta \mathcal{L}_r)^2 \right] \tag{1}$$

By neglecting higher order terms in the taylor approximation we can say, $\delta \mathcal{L}_r \approx \frac{1}{2}\delta w^\top H_r \delta w + \nabla_r^\top \delta w$. Substituting $\delta \mathcal{L}_r$ from Equation 1:

$$\Psi(\mathrm{X}) = \mathbb{E}_{\delta w \sim \mathcal{N}(\mathbf{0},\mathbf{I})} \left[(\frac{1}{2}\delta w^\top \mathrm{X} \delta w - \frac{1}{2}\delta w^\top \mathrm{H}_r \delta w)^2 \right] \tag{2}$$

$$= \mathbb{E}_{\delta w \sim \mathcal{N}(\mathbf{0},\mathbf{I})} \left[(\frac{1}{2}\delta w^\top (\mathrm{X} - \mathrm{H}_r) \delta w)^2 \right] \tag{3}$$

$$= \mathbb{E}_{\delta w \sim \mathcal{N}(\mathbf{0},\mathbf{I})} \left[(\frac{1}{2}\delta w^\top \mathrm{M} \delta w)^2 \right] \tag{4}$$

where, we define $\mathrm{M} = (\mathrm{X} - \mathrm{H}_r)$. We will now prove the following:

$$\mathbb{E}_{\delta w \sim \mathcal{N}(\mathbf{0},\mathbf{I})} \left[(\frac{1}{2}\delta w^\top \mathrm{M} \delta w)^2 \right] = \frac{1}{2}\mathrm{trace}(\mathrm{M}^2) + \frac{1}{4}\mathrm{trace}(\mathrm{M})^2 \tag{5}$$

Proof of Expectation

We aim to prove the following equation:

$$\mathbb{E}_{\delta w \sim \mathcal{N}(\mathbf{0},\mathbf{I})} \left[\left(\frac{1}{2}\delta w^\top \mathrm{M} \delta w \right)^2 \right] = \frac{1}{2}\mathrm{trace}(\mathrm{M}^2) + \frac{1}{4}\mathrm{trace}(\mathrm{M})^2, \tag{6}$$

where $\delta w \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is a Gaussian random vector with zero mean and identity covariance, and M is a symmetric matrix.

Step 1: Reformulation of the Expectation

Let $X = \delta w^\top \mathrm{M} \delta w$. Then, the left-hand side can be expressed as:

$$\mathbb{E} \left[\left(\frac{1}{2}\delta w^\top \mathrm{M} \delta w \right)^2 \right] = \frac{1}{4}\mathbb{E}[X^2], \tag{7}$$

where $X^2 = (\delta w^\top \mathrm{M} \delta w)^2$. Substituting $X = \delta w^\top \mathrm{M} \delta w$, we expand X^2:

$$X^2 = (\delta w^\top \mathrm{M} \delta w)^2 = \sum_{i,j} \sum_{k,l} \mathrm{M}_{ij} \mathrm{M}_{kl} \delta w_i \delta w_j \delta w_k \delta w_l, \tag{8}$$

where M_{ij} denotes the (i, j)-th entry of M, and δw_i is the i-th component of δw.

Step 2: Expectation of $\delta w_i \delta w_j \delta w_k \delta w_l$

Since $\delta w \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, the components δw_i are independent Gaussian random variables with mean 0 and variance 1. The expectation $\mathbb{E}[\delta w_i \delta w_j \delta w_k \delta w_l]$ depends on the indices i, j, k, l. Using properties of Gaussian random variables, we have:

$$\mathbb{E}[\delta w_i \delta w_j \delta w_k \delta w_l] = \begin{cases} 1, & \text{if } i = j,\ k = l,\ i \neq k, \\ 1, & \text{if } i = k,\ j = l,\ i \neq j, \\ 1, & \text{if } i = l,\ j = k,\ i \neq j, \\ 1, & \text{if } i = j = k = l, \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

This result follows from the Wick formula for moments of Gaussian random variables.

Step 3: Substituting into the Expectation

Return to the expectation of X^2:

$$\mathbb{E}[X^2] = \sum_{i,j}\sum_{k,l} \mathrm{M}_{ij}\mathrm{M}_{kl}\mathbb{E}[\delta w_i \delta w_j \delta w_k \delta w_l]. \tag{10}$$

Using the cases derived above, the non-zero contributions arise in the following scenarios:

- **Case 1:** $i = j, k = l, i \neq k$**:** The contribution is:

$$\sum_{i \neq k} \mathrm{M}_{ii}\mathrm{M}_{kk} = \mathrm{trace}(\mathrm{M})^2. \tag{11}$$

- **Case 2:** $i = k, j = l, i \neq j$**:** The contribution is:

$$\sum_{i,j} \mathrm{M}_{ij}^2 = \mathrm{trace}(\mathrm{M}^2). \tag{12}$$

- **Case 3:** $i = l, j = k, i \neq j$**:** This is identical to the second case, contributing:

$$\mathrm{trace}(\mathrm{M}^2). \tag{13}$$

- **Case 4:** $i = j = k = l$**:** The contribution is:

$$\sum_{i} \mathrm{M}_{ii}^2 = \mathrm{trace}(\mathrm{M}^2). \tag{14}$$

Combining these terms, the total expectation is:

$$\mathbb{E}[X^2] = 2\mathrm{trace}(\mathrm{M}^2) + \mathrm{trace}(\mathrm{M})^2. \tag{15}$$

Step 4: Final Simplification

Substituting back into the original equation:

$$\mathbb{E}\left[\left(\frac{1}{2}\delta w^\top \mathrm{M} \delta w\right)^2\right] = \frac{1}{4}\mathbb{E}[X^2] = \frac{1}{4}\left(2\mathrm{trace}(\mathrm{M}^2) + \mathrm{trace}(\mathrm{M})^2\right). \tag{16}$$

Simplify to obtain:

$$\mathbb{E}\left[\left(\frac{1}{2}\delta w^\top \mathrm{M} \delta w\right)^2\right] = \frac{1}{2}\mathrm{trace}(\mathrm{M}^2) + \frac{1}{4}\mathrm{trace}(\mathrm{M})^2. \tag{17}$$

So, clearly the minimizer of $\Psi(\mathrm{X})$ is at $\mathrm{M} = 0$ or $\mathrm{X} = \mathrm{H}_r$.

However it is the ideal case, where we do not approximate $\delta\mathcal{L}_r$. In our algorithm, we are minimizing and approximate objective $\tilde{\Psi}(\mathrm{X})$. Also from the definition we can say $\tilde{\mathrm{f}}_i(X) = \mathrm{f}_i(\mathrm{X}) + (\delta\mathcal{L}_r(w_i) - \delta\mathcal{L}_f(w_i))$. Since we assume that $|\delta\mathcal{L}_r(w_i) - \delta\mathcal{L}_f(w_i)| \leq \epsilon\ \forall i$, we can derive the following inequality:

$$(\mathrm{f}_i(\mathrm{X}) - \epsilon) \leq \tilde{\mathrm{f}}_i(X) \leq (\mathrm{f}_i(\mathrm{X}) + \epsilon) \tag{18}$$

$$\implies \frac{1}{m}\sum_{i=1}^{m}(\mathrm{f}_i(\mathrm{X}) - \epsilon))^2 \leq \tilde{\Psi}(\mathrm{X}) \leq \frac{1}{m}\sum_{i=1}^{m}(\mathrm{f}_i(\mathrm{X}) + \epsilon))^2 \tag{19}$$

Now we know:

$$\begin{aligned}
\frac{1}{m}\sum_{i=1}^{m}(\mathrm{f}_i(\mathrm{X}) + \epsilon))^2 &= \mathbb{E}_{\delta w \sim \mathcal{N}(\mathbf{0},\mathbf{I})}\left[(\frac{1}{2}\delta w^\top \mathrm{M}\delta w + \epsilon)^2\right] &(20)\\
&= \mathbb{E}_{\delta w \sim \mathcal{N}(\mathbf{0},\mathbf{I})}\left[(\frac{1}{2}\delta w^\top \mathrm{M}\delta w)^2\right] + 2\epsilon\mathbb{E}_{\delta w \sim \mathcal{N}(\mathbf{0},\mathbf{I})}\left[(\frac{1}{2}\delta w^\top \mathrm{M}\delta w)\right] + \epsilon^2 &(21)\\
&\leq \frac{1}{2}\mathrm{trace}(\mathrm{M}^2) + \frac{1}{4}\mathrm{trace}(\mathrm{M})^2 + \frac{1}{2}\mathrm{trace}(2\epsilon\mathrm{M}) &(22)\\
&= \frac{1}{2}\mathrm{trace}(\mathrm{M}^2 + 2\epsilon\mathrm{M}) + \frac{1}{4}\mathrm{trace}(\mathrm{M})^2 &(23)
\end{aligned}$$

Similarly expanding the lower bound also, we get the following inequality:

$$\frac{1}{2}\mathrm{trace}(\mathrm{M}^2 - 2\epsilon\mathrm{M}) + \frac{1}{4}\mathrm{trace}(\mathrm{M})^2 \leq \tilde{\Psi}(\mathrm{X}) \leq \frac{1}{2}\mathrm{trace}(\mathrm{M}^2 + 2\epsilon\mathrm{M}) + \frac{1}{4}\mathrm{trace}(\mathrm{M})^2 \tag{24}$$

$$\tag{25}$$

By seperately taking derivatives of the upper and lower bounds above, if we set it to 0, we get the following bound on the minimizer M.

$$-\frac{2\epsilon}{(2+d)}\mathrm{I}_d \leq \mathrm{M} \leq \frac{2\epsilon}{(2+d)}\mathrm{I}_d \tag{26}$$

where, $\mathrm{I}_d \in \mathbb{R}^{d\times d}$ is the identity matrix.

Details:

The objective function is defined as:

$$f(\mathrm{M}) = \frac{1}{2}\mathrm{Tr}(\mathrm{M}^2 + 2\epsilon\mathrm{M}) + \frac{1}{4}(\mathrm{Tr}(\mathrm{M}))^2, \tag{27}$$

where M is a $d \times d$ matrix, and ϵ is a scalar parameter. To find the optimal M, we compute the gradient of $f(\mathrm{M})$ with respect to M and solve $\nabla_{\mathrm{M}} f(\mathrm{M}) = 0$.

The gradient of each term in $f(\mathrm{M})$ is as follows:

- The gradient of $\frac{1}{2}\mathrm{Tr}(\mathrm{M}^2)$ is:
$$\nabla_{\mathrm{M}}\left(\frac{1}{2}\mathrm{Tr}(\mathrm{M}^2)\right) = \mathrm{M}. \tag{28}$$
- The gradient of $\epsilon\mathrm{Tr}(\mathrm{M})$ is:
$$\nabla_{\mathrm{M}}\left(\epsilon\mathrm{Tr}(\mathrm{M})\right) = \epsilon\mathrm{I}_d, \tag{29}$$
where I_d is the $d \times d$ identity matrix.
- The gradient of $\frac{1}{4}(\mathrm{Tr}(\mathrm{M}))^2$ is:
$$\nabla_{\mathrm{M}}\left(\frac{1}{4}(\mathrm{Tr}(\mathrm{M}))^2\right) = \frac{1}{2}\mathrm{Tr}(\mathrm{M})\mathrm{I}_d. \tag{30}$$

Combining these terms, the total gradient is:

$$\nabla_{\mathrm{M}} f(\mathrm{M}) = \mathrm{M} + \epsilon\mathrm{I}_d + \frac{1}{2}\mathrm{Tr}(\mathrm{M})\mathrm{I}_d. \tag{31}$$

Setting $\nabla_{\mathrm{M}} f(\mathrm{M}) = 0$, we have:

$$\mathrm{M} + \epsilon\mathrm{I}_d + \frac{1}{2}\mathrm{Tr}(\mathrm{M})\mathrm{I}_d = 0. \tag{32}$$

Rearranging, this becomes:

$$\mathrm{M} = -\epsilon\mathrm{I}_d - \frac{1}{2}\mathrm{Tr}(\mathrm{M})\mathrm{I}_d. \tag{33}$$

Let $\mathrm{Tr}(\mathrm{M}) = \tau$. Substituting τ into the equation, we get:

$$\mathrm{M} = -\epsilon\mathrm{I}_d - \frac{1}{2}\tau\mathrm{I}_d. \tag{34}$$

Taking the trace on both sides:

$$\tau = \mathrm{Tr}(\mathrm{M}) = \mathrm{Tr}\left(-\epsilon\mathrm{I}_d - \frac{1}{2}\tau\mathrm{I}_d\right). \tag{35}$$

Since $\mathrm{Tr}(\mathrm{I}_d) = d$ for a $d \times d$ matrix:

$$\tau = -\epsilon d - \frac{1}{2}\tau d. \tag{36}$$

Rearranging to isolate τ:

$$\tau\left(1 + \frac{d}{2}\right) = -\epsilon d. \tag{37}$$

Solving for τ:

$$\tau = \frac{-\epsilon d}{1 + \frac{d}{2}} = \frac{-2\epsilon d}{2 + d}. \tag{38}$$

Substituting τ Back into M Substitute $\tau = \frac{-2\epsilon d}{2+d}$ into $\mathrm{M} = -\epsilon\mathrm{I}_d - \frac{1}{2}\tau\mathrm{I}_d$:

$$\mathrm{M} = -\epsilon\mathrm{I}_d - \frac{1}{2}\left(\frac{-2\epsilon d}{2 + d}\right)\mathrm{I}_d. \tag{39}$$

Simplify:

$$\mathrm{M} = -\epsilon\mathrm{I}_d + \frac{\epsilon d}{2 + d}\mathrm{I}_d. \tag{40}$$

Combine terms:

$$\mathrm{M} = \left(-\epsilon + \frac{\epsilon d}{2 + d}\right)\mathrm{I}_d. \tag{41}$$

Simplify further:

$$\mathrm{M} = -\epsilon\left(1 - \frac{d}{2 + d}\right)\mathrm{I}_d = -\epsilon\left(\frac{2}{2 + d}\right)\mathrm{I}_d. \tag{42}$$

The optimal M is:

$$\mathrm{M} = -\frac{2\epsilon}{2 + d}\mathrm{I}_d, \tag{43}$$

where d is the dimension of the matrix M.
This inequality implies that the if the solution of optimization **??** is $\hat{\mathrm{H}}_r$, then

$$\mathrm{H}_r - \frac{2\epsilon}{(2 + d)}\mathrm{I}_d \preceq \hat{\mathrm{H}}_r \preceq \mathrm{H}_r + \frac{2\epsilon}{(2 + d)}\mathrm{I}_d \tag{44}$$

As a result we can conclude:

$$\|\hat{\mathrm{H}}_r - \mathrm{H}_r\| = \|\Delta\mathrm{H}_r\|_F \leq \frac{2\epsilon\|\mathrm{I}_d\|_F}{(2 + d)} \tag{45}$$

Since $\|\mathrm{I}_d\|_F = \sqrt{d}$, we conclude the proof.

□

2. Additional Experiments

We conducted experiments on the CIFAR-10, CIFAR-100, StanfordDogs, and Caltech-256 datasets using our proposed method for both linear classifier and mixed linear network cases. For all experiments, 500 perturbations were applied. The “Performance Gap” row represents the difference in performance between the methods Unlearned (+) and Unlearned (-). Unlearned (+) refers to unlearning using the remaining data samples, while Unlearned (-), our proposed method, performs unlearning without relying on the remaining data samples.

2.1. Linear Classifier Experiments

For the linear classifier experiments, a ResNet-18 model, pretrained on the ImageNet dataset and excluding the penultimate layer, was used to generate activations for our linear classifier. 10% of the data was selected to be forgotten. As shown in Tab. 1, the Unlearned (-) method achieves unlearning performance that is significantly close to the Unlearned (+) method. This result demonstrates that our method can perform well even without access to the remaining dataset, provided that the theoretical assumptions hold.

Method	Test Data	Remaining Data	Forget Data	MIA
CIFAR-10				
Retrained	72%	74%	72%	50%
Unlearned (+)	70.3%	72.4%	70.2%	50.2%
Unlearned (-)	70%	71%	68%	51.5%
Performance Gap	**0.3%**	**1.4%**	**1.8%**	**1.3%**
CIFAR-100				
Retrained	56.0%	61.4%	56.2%	50.4%
Unlearned (+)	49.8%	59.7%	48.7%	51.4%
Unlearned (-)	51.6%	59.6%	49.2%	51.8%
Performance Gap	**1.8%**	**0.1%**	**0.5%**	**0.4%**
StanfordDogs				
Retrained	59.3%	67.8%	60.2%	50.2%
Unlearned (+)	54.6%	65.1%	53.5%	51.7%
Unlearned (-)	55.0%	68.3%	54.5%	50.9%
Performance Gap	**0.4%**	**3.2%**	**1.0%**	**1.2%**
Caltech-256				
Retrained	54.4%	61.1%	54.8%	50%
Unlearned (+)	47.6%	57.3%	45.0%	51.3%
Unlearned (-)	49.4%	57.9%	48.4%	50.6%
Performance Gap	**1.8%**	**0.6%**	**3.4%**	**0.7%**

Table 1. Linear classifier experiments on CIFAR-10, CIFAR-100, StanfordDogs, and Caltech-256 datasets. A ResNet-18 model, pretrained on ImageNet and excluding the penultimate layer, was used to generate activations for the linear classifier. 10% of the data was selected for unlearning. The ”Performance Gap” row indicates the difference between the methods Unlearned (+) and Unlearned (-), where Unlearned (+) utilizes the remaining data samples, and Unlearned (-) (our proposed method) operates without access to the remaining data samples.

2.2. Mixed-Linear Network Experiments

For the mixed linear network experiments, a ResNet-18 model pretrained on the ImageNet dataset was used as the base model. The last few layers were linearized for training on the datasets. 15% of the data was selected for unlearning. Unlearned (+) refers to the unlearning process that utilizes the remaining data samples, while Unlearned (-) (our proposed method) performs unlearning without access to the remaining data samples. As shown in Tab. 2, the Unlearned (-) and Unlearned (+) methods exhibit very similar performance. This result demonstrates that our method (Unlearned (-)) can achieve significantly strong results even without access to the remaining samples for the mixed linear network case as well.

Method	Test Data	Remaining Data	Forget Data	MIA
CIFAR-10				
Retrained	86.3%	93.6%	87.7%	50.2%
Unlearned (+)	85.6%	91.9%	85.5%	50.0%
Unlearned (-)	84.5%	93.5%	86.7%	51.2%
Performance Gap	**1.1%**	**1.6%**	**1.2%**	**1.2%**
CIFAR-100				
Retrained	63.1%	68.9%	63.3%	50.2%
Unlearned (+)	61.9%	67.9%	62.2%	50.7%
Unlearned (-)	61.4%	70.1%	62.2%	51.7%
Performance Gap	**0.5%**	**2.2%**	**0.0%**	**1.0%**
StanfordDogs				
Retrained	73.6%	76.8%	72.1%	50.6%
Unlearned (+)	69.0%	76.1%	70.8%	50.2%
Unlearned (-)	70.1%	77.6%	71.2%	51.3%
Performance Gap	**1.1%**	**1.5%**	**0.6%**	**1.1%**
Caltech-256				
Retrained	60.3%	66.9%	60.5%	50.0%
Unlearned (+)	61.3%	65.6%	61.8%	49.8%
Unlearned (-)	58.4%	66.2%	60.7%	51.0%
Performance Gap	**2.9%**	**1.6%**	**1.1%**	**1.2%**

Table 2. Mixed linear network experiments on CIFAR-10, CIFAR-100, StanfordDogs, and Caltech-256 datasets. A ResNet-18 model, pretrained on the ImageNet dataset, was used as the base model. The last few layers were linearized for training with the datasets. 15% of the data was selected for unlearning. The ”Performance Gap” row indicates the difference between the methods Unlearned (+) and Unlearned (-). Unlearned (+) performs unlearning using the remaining data samples, while Unlearned (-), our proposed method, achieves competitive results even without access to the remaining data samples.