Forget with Precision: Assessing Machine Unlearning Approaches

Max Krähenmann [1] Leo Neubecker [1] Virgilio Strozzi [1] Igor Martinelli [1]

Abstract

The emerging field of Machine Unlearning aims to induce a model to forget a portion of the dataset on which it has been trained. This work explores various unlearning approaches, utilizing combinations of finetuning, label poisoning, pruning, and re-initialization, resulting in a total of eight distinct methods. These methods undergo evaluation on the ResNet-18 architecture trained on CIFAR-10 and AgeDB for considerations of computational efficiency, accuracy, forget quality. For the last criteria, a novel metric $\hat{\epsilon}$ inspired by the Likelihood-Ratio attack, with the goal of assessing the unlearning effectiveness of the model, is developed. Furthermore, two novel unlearning approaches, namely Pruning Complex and Activation Reset, are introduced. Among all the tested methods, Pruning Complex exhibits the best forget quality under the $\hat{\epsilon}$ metric.

1. Introduction

The field of deep learning has seen rapid advancements, leading to the emergence of Machine Unlearning. This process involves selectively removing data from trained models, crucial for maintaining the integrity of machine learning models in dynamic data environments. This paper focuses on developing efficient Machine Unlearning methods that balance targeted data removal and model performance preservation. Our contributions are:

1. **Defining Machine Unlearning:** We present a comprehensive framework for understanding Machine Unlearning within the context of deep learning, underscoring its significance in the modern data-centric world.
2. **Innovative Unlearning Techniques:** We employ known methods including Poisoning, Selective Pruning and Finetuning and Hybrid approaches, while also defining new methods such as Selective Pruning Complex and Activation Reset for efficient data removal.
3. **Experimental Analysis and Insights:** We assess our methods using the CIFAR-10 and AgeDB datasets and ResNet-18, focusing on computational efficiency, effectiveness of data removal, and utility of the model under the accuracy metric.

2. Related Work

The concept of Machine Unlearning in deep learning has significant overlap with fields such as privacy-preserving data processing, efficient model training, and dynamic data management. This section briefly synthesizes the major research themes that inform our approach.

Machine Learning and Privacy: With the advent of large-scale data processing, protecting privacy in machine learning has become crucial. Foundational studies like (Shokri et al., 2017) have exposed vulnerabilities in model privacy, setting the stage for our exploration of Machine Unlearning as a privacy-preserving technique.

Efficient Model Training and Updating: Addressing the computational burden of training large-scale models, research in efficient training methods (e.g., (He et al., 2016)) has influenced our Machine Unlearning strategies. We aim to apply these efficiency principles to minimize resource use while maintaining model integrity during unlearning.

Data Deletion and Model Adaptability: The evolving nature of datasets and privacy norms necessitates adaptable machine learning models. Research in data deletion and adaptability methods, such as label poisoning and finetuning (e.g., (Graves et al., 2021)), directly informs our unlearning techniques.

Differential Privacy in unlearning: The role of differential privacy in quantifying data privacy is integral to evaluating Machine Unlearning. Techniques developed in this area (e.g., (Carlini et al., 2022)) offer metrics to assess our unlearning methods' effectiveness in preserving privacy.

Practical Insights: Our project also gains insights from practical challenges and community-driven research, such as the NeurIPS 2023 Machine Unlearning competition ((Triantafillou et al., 2023)). These real-world applications and collaborative efforts enrich our understanding and development of effective Machine Unlearning methodologies.

[1] Department of Computer Science, ETH Zürich, Zürich, CH.

Student Project of the Deep Learning Course at ETH Zürich, Switzerland, 2023.

3. Methods

3.1. Formalizing Machine Unlearning

In order to measure the quality of our unlearning algorithms, we first must produce a way to evaluate their results. We refer to (Eleni Triantafillou, 2023) and their definition of differential privacy (DP). Specifically, we consider k-group level differential privacy. It is defined in the following way:

Definition 3.1. Group-level Differential privacy. A training algorithm $A : \mathcal{D} \rightarrow \mathcal{R}$ is (ϵ, δ, k) group-level DP if for all pairs of datasets $D, D' \in \mathcal{D}$ that differ by addition or removal of up to k training examples and all output regions $R \subseteq \mathcal{R}$:

$$Pr[A(D) \in R] \leq e^{\epsilon} Pr[A(D') \in R] + \delta$$

Machine Unlearning can then be defined as:

Definition 3.2. Machine Unlearning. For a fixed dataset D, forget set $S \subseteq D$, and a randomized learning algorithm A, an unlearning algorithm U is (ϵ, δ)-unlearning with respect to (D, S, A) if for all regions $R \subseteq \mathcal{R}$, we have that

$$Pr[A(D \setminus S) \in R] \leq e^{\epsilon} Pr[U(A(D), S, D) \in R] + \delta$$

and

$$Pr[U(A(D), S, D) \in R] \leq e^{\epsilon} Pr[A(D \setminus S) \in R] + \delta$$

Which implies that when having small values of ϵ and δ the distributions of the retained model is indistinguishable from the unlearned model.

We then interpret Machine Unlearning as a hypothesis test with the null hypothesis that A is trained on $D \setminus S$ and the alternative hypothesis that A is trained on D and subsequently unlearned with algorithm U. By computing the false positive rate (FPR) and the false negative rate (FNR) we can then estimate ϵ for fixed δ with the following formula:

$$\hat{\epsilon} = \max \left\{ \log \frac{1 - \delta - F\hat{P}R}{F\hat{N}R}, \log \frac{1 - \delta - F\hat{N}R}{F\hat{P}R} \right\}$$

where $F\hat{P}R$ and $F\hat{N}R$ are empirical estimates of the true FPR and FNR under an instantiated membership inference attack.

3.2. Attack and $\hat{\epsilon}$ value

The attack that we decided to use for membership inference is inspired by the Likelihood-Ratio attack (Carlini et al., 2022). First we get the hinge loss for each individual model and sample of the forget set, which requires that we have access to the un-normalized output logits. The hinge loss is computed as the subtraction of the maximum of the other logits from the target logit and is needed because we need a statistic that behaves gaussian. We then fit two gaussians for each sample, first on the distribution of the hinge loss of retrained models, then on the distribution induced by the unlearned models. Finally for each hinge loss (of a sample) we calculate its likelihood under the two gaussians, and take as attack score the likelihood ratio. In Figure 1 we can see how the distribution of attack scores for one sample could look like, in blue and red we have attack scores belonging to retrained and unlearned models respectively.

We then proceed to compute the Detection Error Tradeoff curve (Martin et al., 1997). For a certain boundary B_i, this will give us the FPR_i and the FNR_i. The way we aggregate these rates for each sample is the following:

- If both FPR_i and FNR_i are 0, the per-attack ϵ is set as ∞, since the distributions are perfectly separable
- If either FPR_i or FNR_i are 0, the rates are discarded as this is most likely an artifact of the small amount of models used
- Otherwise the $\hat{\epsilon}_i$ is computed as described above.

Finally, to obtain a final score for $\hat{\epsilon}$, we take the maximum of each sample and take the median over the samples as some values might be infinite.

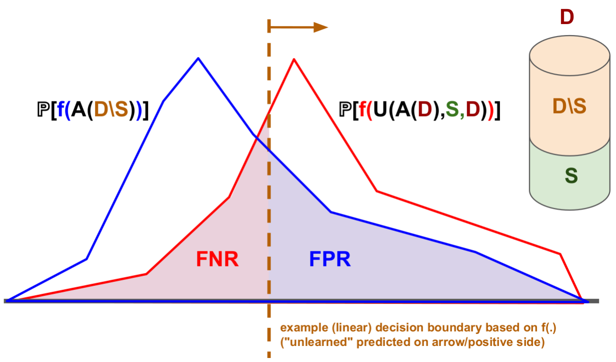
D
ℙ[f(A(D\S))]
ℙ[f(U(A(D),S,D))]
D\S
S
FNR
FPR
example (linear) decision boundary based on f(.)
("unlearned" predicted on arrow/positive side)

Figure 1. FPR and FNR based on a boundary

3.3. Dataset and Model

The dataset we choose to focus on is CIFAR-10, since it is a simple and known classification dataset where it is possible to train model with high accuracy. Additionally we also use the AgeDB Dataset to further test some of the simpler unlearning methods.

The datasets are split randomly into retain set and forget set. The forget set has a size of approximately 2% of the original dataset. For the AgeDB dataset we also take care that forget and retain sets are subject exclusive (images of one person only appear in one of the two sets).

The model that we employ is ResNet-18, just like in the Kaggle competition (Triantafillou et al., 2023). The ResNet-18 implementation is taken from (Mountchicken, 2022) and huggingface-hub for the two datasets respectively. In case of AgeDB we initialize the weights with the pretrained IMAGENET1K_V1 weights from pytorch.

3.4. Training

We train $N = 80$ models on the retain set and one model on the union of retain and forget set, the unlearning approaches is then applied to this last model to generate additionally N unlearned models. We then can perform the attack explained in Section 3.2 to derive the $\hat{\epsilon}$ value.

For CIFAR-10 the training is conducted over 100 epochs with SGD and parameters set as follows: learning rate 0.1, batch size 128, momentum 0.9, and weight decay 5×10^{-4}. We employ a multi-step learning rate scheduler that divides the learning rate by 10 at epochs 35, 70, and 90. Additionally we augment the data with `torchvision`'s AutoAugment. This strategy allows us to achieve an accuracy of approximately 99% on the retained set and 95% on the held-out test set.

In case of the AgeDB dataset we first pretrain on the IMDB-WIKI dataset for 100 epochs and then finetune N models on the AgeDB for 15 epochs, with batch size of 128. We use Adam optimizer with default parameters and constant learing rate of 0.001 and 0.001 for pretrain and finetune stages respectively. The images are first preprocessed such that we have a complete facial coverage and are then augmented with a random horizontal flip and crop. We finally achieve a mean average error of 6.73 for the age regression task.

3.5. Unlearning

For each of the approaches, we impose the restriction that each approach should not take more than 5% of the time required for retraining completely the model. This directly translates to a maximum of 5 epochs when using the retained/full dataset. The rationale behind this constraint is to ensure a substantial gain over the time required for full model retraining.

We now present the different approaches employed which, given a model, the retain set and the forget set, return an unlearned version.

3.5.1. Finetuning on the Retain Set

In this approach, 5 epochs are performed on the retain set with a low learning rate 0.001.

3.5.2. Label Poisoning

This approach focuses on randomly assigning labels to the elements of the forget set (or adding gaussian noise in case of the regression task). We employ two variants of label poisoning, one were the unlearning is only applied over the forget set for 1 epoch (2 on the regression task) with learning rate 0.0007, and a second one where the whole dataset is used for 5 epoch with learning rate 0.002.

3.5.3. Two stage Hybrid

Employing Label poisoning can potentially reduces the accuracy at a high degree, thus in the hybrid method we first perform label poisoning and then mitigate the accuracy drop by finetuning again. We propose to first perform 1 epoch on the poisoned forget set with a higher learning rate 0.01, and then recover performance over 5 epochs of training on the retain set with learning rate 0.001.

3.5.4. Selective Pruning

The concept underlying selective pruning is to identify and prune the weights in the model associated with knowledge from the forget set. This process aims to intuitively guide the model to forget information related to those specific samples. To achieve this, we fully train a model on the forget set, deliberately overfitting it until it reaches 100% accuracy. By comparing the weights of this overfitted model to those of the original model, we can identify the most similar weights. We compute the absolute difference between the weights, set a threshold corresponding to a quantile of 0.30, and reinitialize the weights that fall within the threshold. The reinitialized weights use Kaiming initialization (He et al., 2015). The resulting model is then trained for 5 epochs on the retained set with a learning rate of 0.001.

Additionally, we introduce two strategies inspired by selective pruning: *Selective Pruning Last Layer* and *Selective Pruning Complex*. In the former, selective pruning is exclusively applied to the last linear layer of ResNet-18. This choice is motivated by the intuition that the last layer is responsible for combining all the previous features, making it a crucial area for inducing forgetting. The latter strategy involves overfitting two models: one on the forget set and another on a random fraction of the retain set, both until they reach close to 100% accuracy. The objective is to derive a more complex forgetting strategy by reinitializing only the weights similar to both the model overfitted on the forget set and the model overfitted on the fraction of the retain set. We assume that these weights are challenging to forget when fine-tuning on the retain set since they are intrinsic on both sets, but their knowledge also contains the forget set. In both strategies, the resulting model subsequently undergoes fine-tuning for 5 epochs on the retain set with a learning rate of 0.001.

3.5.5. Activation reset

The idea of this method is to assess whether a specific kernel in a convolutional layer exhibits an average higher activation when processing a sample from the forget set compared to one from the retain set. This is achieved by measuring the mean activation of each kernel at each layer for all samples in the forget set. The same process is repeated for a subset of the retain set, equivalent in size to the forget set. Subsequently, these values are compared, and if 80% of the activation values for the forget set in each kernel, hence their neuron activations output, are higher than their respective counterparts in the retain set, the weights are reinitialized using Kaiming initialization (He et al., 2015). This approach is motivated by the assumption that the weights of these kernels are learned to respond specifically to samples from the forget set, necessitating reinitialization to facilitate forgetting. Additionally, the final Multi-Layer Perceptron (MLP) layer is also reinitialized with the same technique since its inputs are altered. Following this initialization, the model undergoes fine-tuning for 5 epochs on the retain set with a learning rate of 0.001.

4. Results

We show the full results of all the methods in Table 1 and partially in Table 2. We also provide a baseline which is obtained by splitting the logits of the models trained on the retain set into two subsets and trying our attack on these. This should represent results that are very close to the optimal for the $\hat{\epsilon}$ value, since these logits should be indistinguishable by design.
By comparing the different methods in Table 1, we notice that Pruning Complex is the one with lowest $\hat{\epsilon}$ value, for all the values of delta as exhibited in Figure 2. All the approaches shows a high accuracy on the retain set, while some forgets and lose more performance on the forget set as expected, with a lowest peak with Poison Full. It is interesting to notice that a low accuracy on the forget set does not imply that the $\hat{\epsilon}$ value is small and that the model has forgotten the samples. We also remind that an optimal unlearned model would generally exhibit a high and similar test set and forget set accuracy, which is mostly exhibited in the Hybrid, Pruning Last and Activation Reset methods.

We also notice that Finetune, Poison and Hybrid methods evaluated on the AgeDB dataset (see Table 2) display negligible difference in forget quality and high difference on the test set metric. The opposite is true when evaluated on CIFAR-10, this can indicate that these unlearning approaches don't generalize well across different datasets and tasks, due to the difference in output distributions.

Table 1. Evaluation on CIFAR-10: $\hat{\epsilon}$ values and accuracy on the retain set ($D \setminus S$), forget set (S) and test set, with fixed $\delta = 0.05$.

Approach	$\hat{\epsilon}$	$D \setminus S$	S	Test
Baseline	0.88	99.92	94.61	94.99
Finetune	3.74	99.99	99.51	94.58
Poison	4.16	99.72	99.55	94.21
Poison Full	4.20	99.99	87.41	94.16
Hybrid	2.83	96.09	96.23	94.58
Pruning	3.22	98.21	97.47	92.60
Pruning Last	2.30	99.99	94.43	**95.04**
Pruning Complex	**2.20**	99.99	91.10	94.31
Activation Reset	3.58	99.99	93.96	94.76

Table 2. Evaluation on AgeDB: $\hat{\epsilon}$ values and mean averge error on forget set (S) and test set with fixed $\delta = 0.05$.

Approach	$\hat{\epsilon}$	S	Test
Baseline	0.40	7.86	6.73
Finetune	**3.93**	6.28	**6.18**
Poison	4.06	8.10	8.73
Hybrid	4.00	7.22	6.85

5. Conclusion

In this work we explore the new domain of Machine Unlearning, inspired by the challenges posed in the Kaggle competition (Triantafillou et al., 2023). Drawing from established definitions in the field of Differential Privacy, we devise a custom membership inference attack, loosely inspired by the Likelihood-Ratio attack. Our experimentation involved training multiple ResNet18 models on the CIFAR-10 and AgeDB datasets, subjecting them to various unlearning algorithms. Notably, we implement a range of approaches combining Finetuning, Poisoning, and Pruning, alongside novel methods such as Activation Reset and Pruning Complex. The latter, in particular, demonstrates the most promising results under the $\hat{\epsilon}$ metric, while maintaining comparable accuracy on the test set.

Unfortunately, due to the challenges of working with a more complex dataset like AgeDB, we didn't have time to evaluate all methods on it. Assessing the most promising approaches on AgeDB is a compelling avenue for future research, given its closer relation to real-world privacy concerns in machine learning. Another potential direction for future endeavors involves delving deeper into the fine-tuning of parameters within each unlearning method. Investigating whether the observed performance enhancements extend to diverse datasets would further enrich our understanding. Lastly, introducing a baseline for comparing models trained on the complete dataset with models trained on the retained set would provide a worst-case scenario for the $\hat{\epsilon}$ value.

References

Carlini, N., Chien, S., Nasr, M., et al. Membership inference attacks from first principles, 2022.

Eleni Triantafillou, P. K. Evaluation for the neurips machine unlearning competition. In *NeurIPS 2023*. NeurIPS, 2023.

Graves, L., Nagisetty, V., and Ganesh, V. Amnesiac machine learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 11516–11524, 2021.

He, K., Zhang, X., Ren, S., et al. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, 2015.

He, K., Zhang, X., Ren, S., et al. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Martin, A. F., Doddington, G. R., Kamm, T. M., Ordowski, M., and Przybocki, M. A. The det curve in assessment of detection task performance. In *EUROSPEECH*, 1997. URL https://api.semanticscholar.org/CorpusID:9497630.

Mountchicken. Resnet18-cifar10. https://github.com/Mountchicken/ResNet18-CIFAR10/tree/main, 2022.

Shokri, R., Stronati, M., Song, C., and Shmatikov, V. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, pp. 3–18. IEEE, 2017.

Triantafillou, E., Pedregosa, F., Hayes, J., et al. Neurips 2023 - machine unlearning, 2023. URL https://kaggle.com/competitions/neurips-2023-machine-unlearning.

A. Estimate of Epsilon

The results are showed for the models trained on CIFAR-10.

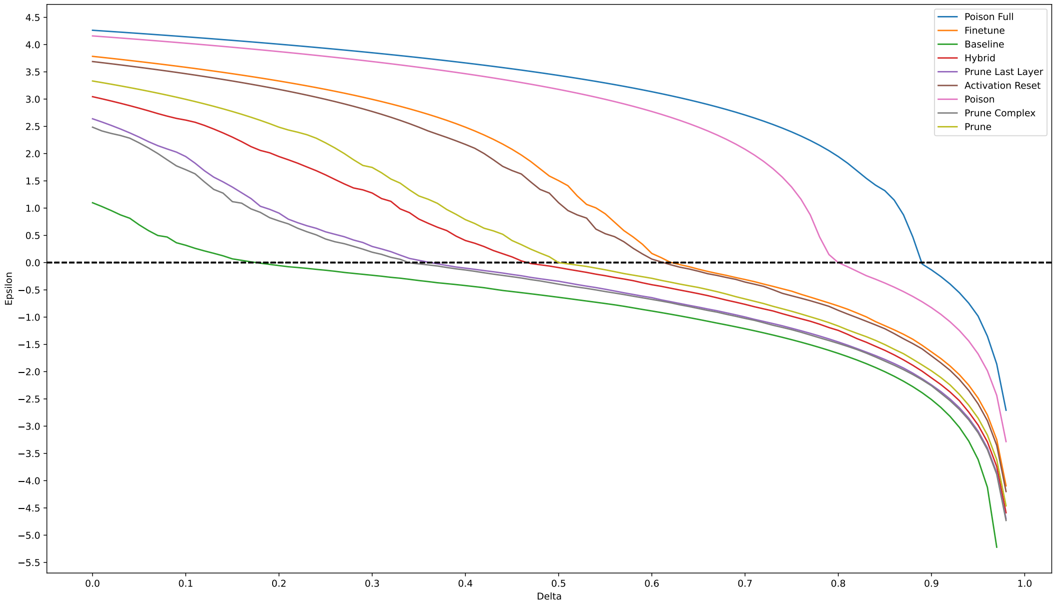

Figure 2. Plot of the $\hat{\epsilon}$ with different fixed values of δ for all the methods.

B. Membership inference attack

Histogram plots illustrating the results of the membership inference attack conducted on the various methods employed in this study to derive the $\hat{\epsilon}$ value, are presented in this Appendix. The results are showed only for the models trained on CIFAR-10. The x-axis represents the logits values obtained after applying the hinge loss, while the y-axis denotes the count of neural networks falling into each bin. The distinction between the model trained on the full dataset and the untrained model is visualized using different colors. Each histogram plot corresponds to a specific sample, and the attack involves 80 models trained on the retained set and 80 models after applying the unlearning technique.

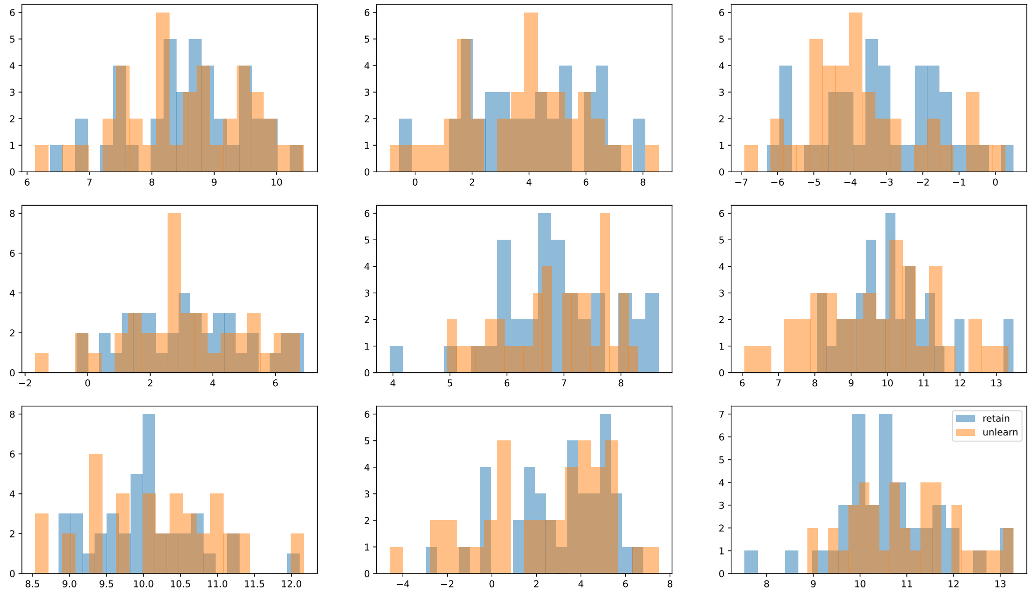

Figure 3. Histogram plot of the hinge losses in the attack on the baseline method.

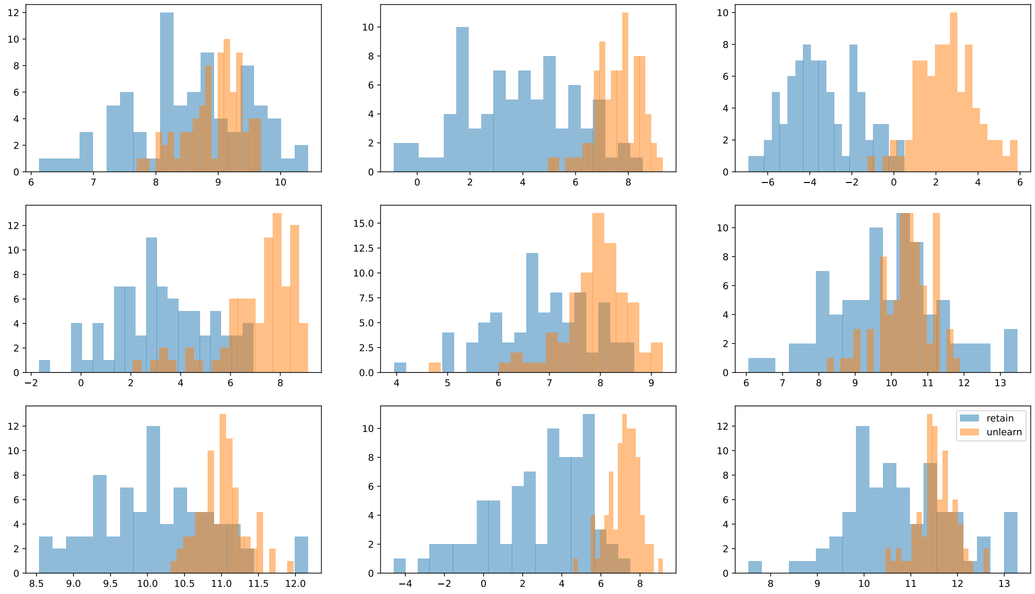

Figure 4. Histogram plot of the attack on the finetune method.

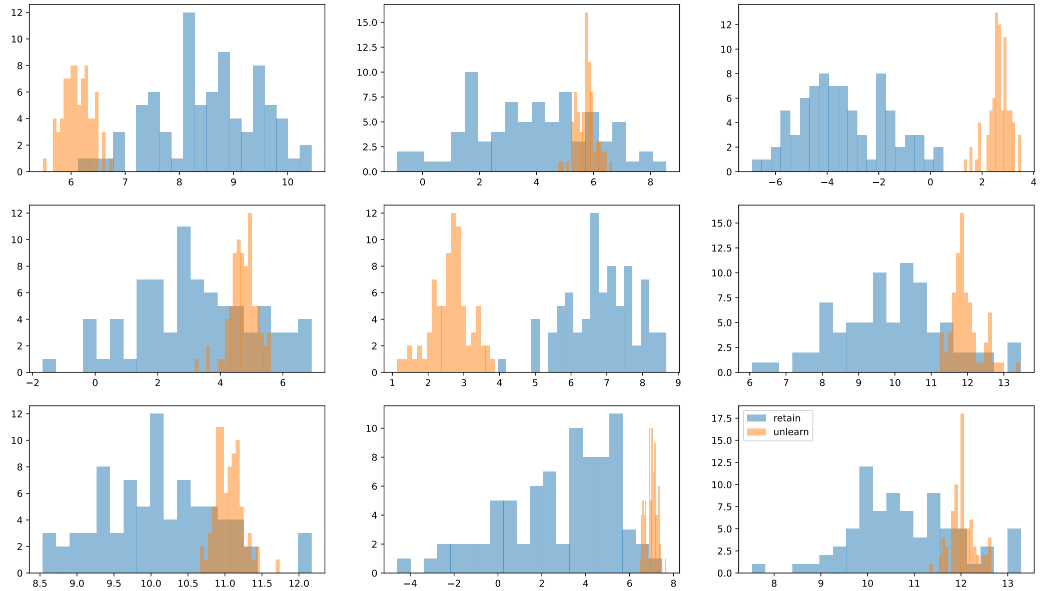

Figure 5. Histogram plot of the attack on the simple poison method.

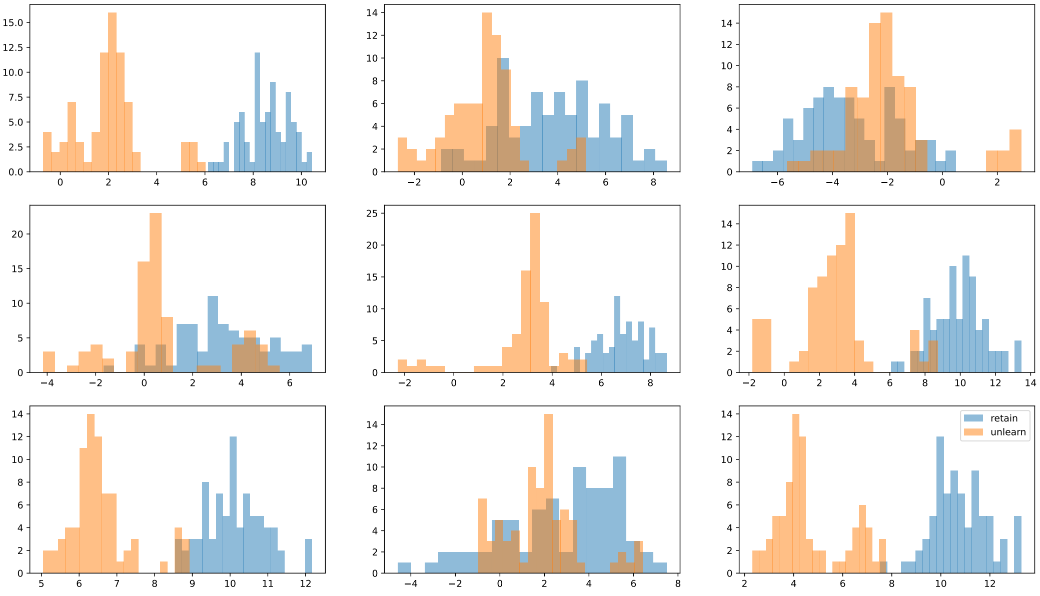

Figure 6. Histogram plot of the attack on the poison full method.

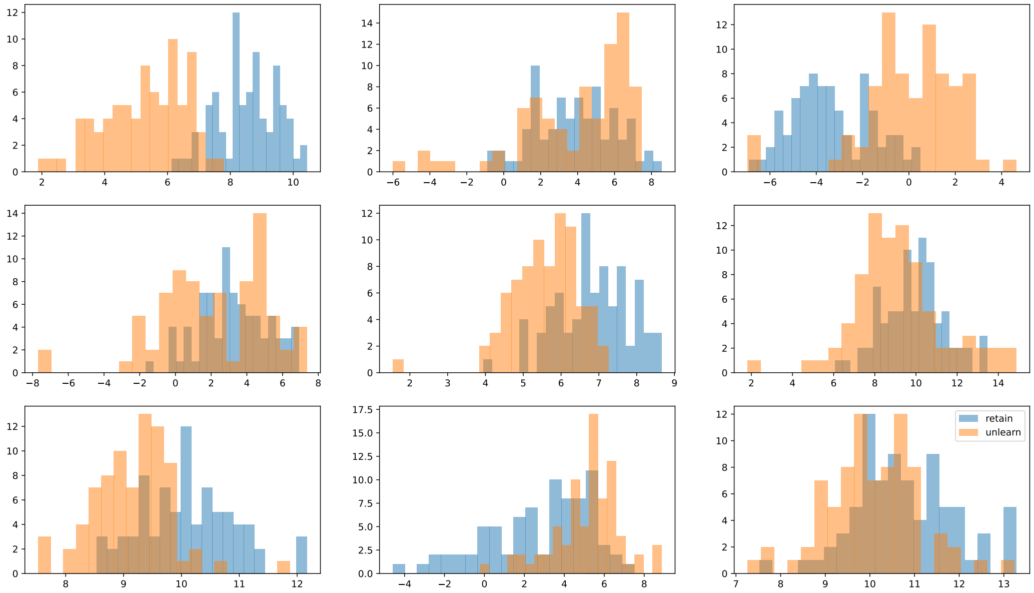

Figure 7. Histogram plot of the attack on the hybrid method.

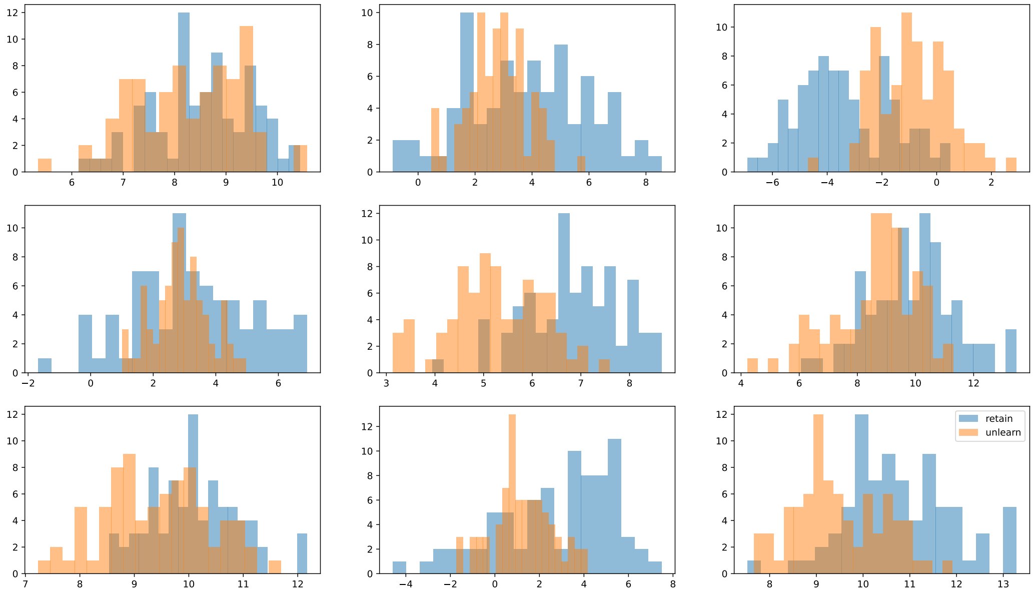

Figure 8. Histogram plot of the attack on the prune method.

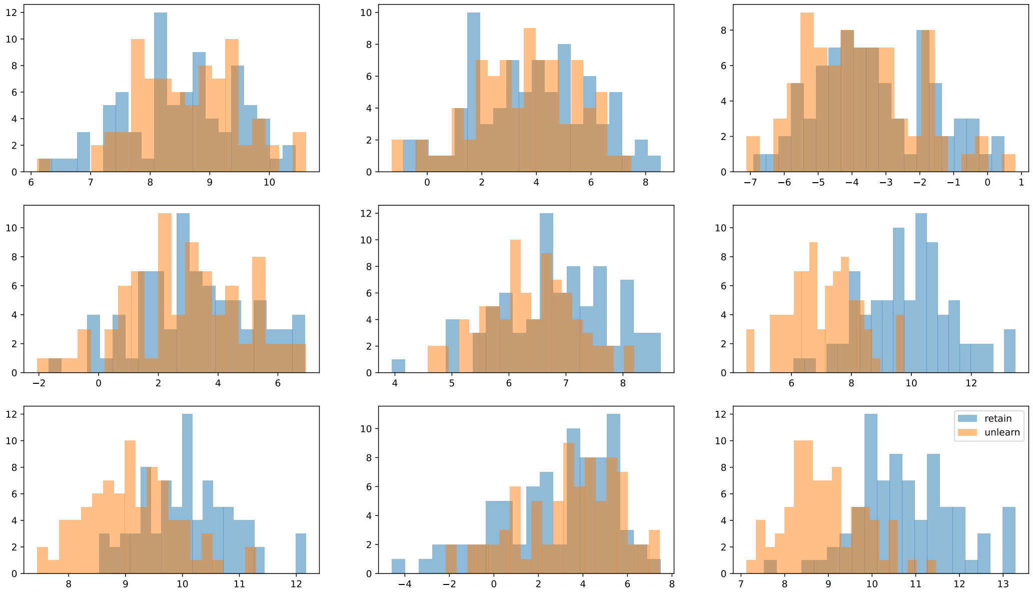

Figure 9. Histogram plot of the attack on the prune of the last layer method.

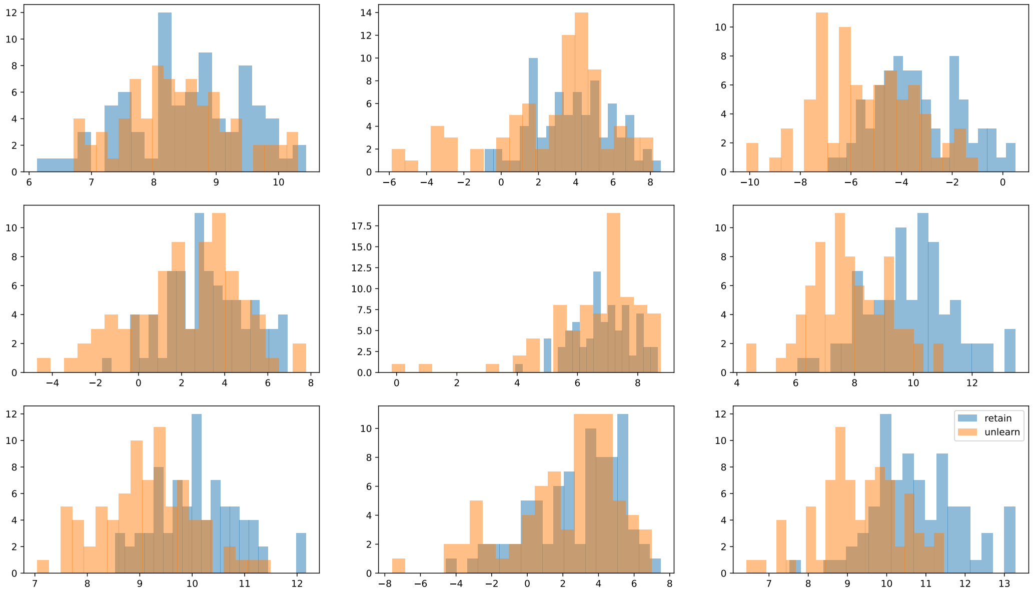

Figure 10. Histogram plot of the attack on the prune complex method.

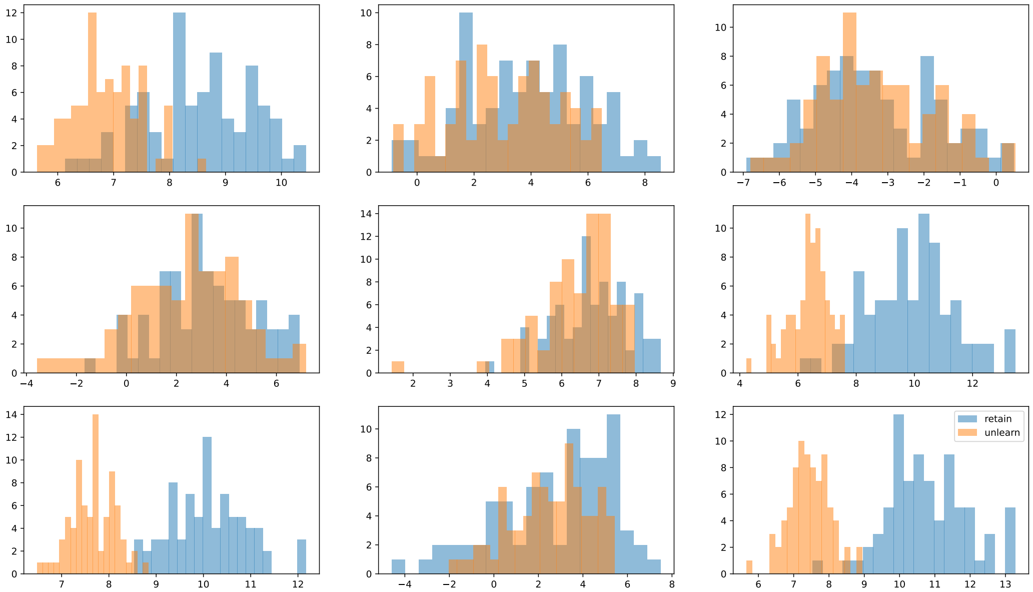

Figure 11. Histogram plot of the attack on the activation reset method.