On the Privacy Effect of Data Enhancement via the Lens of Memorization

Anonymous CVPR submission

Paper ID 2529

Abstract

Machine learning poses severe privacy concerns as it has been shown that the learned models can reveal sensitive information about their training data. Many works have investigated the effect of widely-adopted data augmentation (DA) and adversarial training (AT) techniques, termed data enhancement in the paper, on the privacy leakage of machine learning models. Such privacy effects are often measured by membership inference attacks (MIAs), which aim to identify whether a particular example belongs to the training set or not. We propose to investigate privacy from a new perspective called memorization. Through the lens of memorization, we find that previously deployed MIAs produce misleading results as they are less likely to identify samples with higher privacy risks as members compared to samples with low privacy risks. To solve this problem, we deploy a recent attack that can capture individual samples' memorization degrees for evaluation. Through extensive experiments, we unveil non-trivial findings about the connections between three essential properties of machine learning models, including privacy, generalization gap, and adversarial robustness. We demonstrate that, unlike existing results, the generalization gap is shown not highly correlated with privacy leakage. Moreover, stronger adversarial robustness does not necessarily imply that the model is more susceptible to privacy attacks.

1. Introduction

It has been shown in several studies [3, 4, 34, 41] that machine learning models especially deep neural networks (DNNs) raise severe privacy concerns, as they tend to memorize sensitive information about the training data. To quantitatively evaluate the privacy leakage that a machine learning model reveals about its training data, a basic approach that has been intensively used is the so-called *membership inference* [32]. That is, given the access to a target model, the goal of the adversary is to determine whether a particular data point was used for training this target model (being a member) or not (being a non-member). Such membership information can reveal quite sensitive information about the individuals such as the health conditions [28] and serve as the basis for stronger types of privacy attacks [4].

Several studies show that the attack success rates of membership inference attacks (MIAs) are highly correlated with the generalization gap, i.e., the difference between training and test accuracies [21,31,32,35,39]. Such correlation is also observed when applying different data enhancement methods including Data Augmentation (DA) and Adversarial Training (AT). [35] show that applying AT can make the model more vulnerable to MIAs, and they conclude that one main reason is that the generalization gap becomes larger after applying AT than standard training. The DA methods, on the other hand, are widely believed to be effective in reducing the privacy leakage [30,32,39] as they are usually helpful in avoiding overfitting. Label Smoothing [36], as a particular DA method, however, is recognized to increase privacy leakage while reducing the generalization gap simultaneously [14, 20].

However, the results shown in the aforementioned works might be misleading as the deployed MIAs for measuring the privacy leakage have the following limitations: 1) It has been criticized in several works [2, 14, 29] that the previous MIAs often have quite high false positive rates (FPR), i.e., many non-members are falsely identified as members. However, a good attack should obtain meaningful attack rates under low FPR regions as it is more realistic for practical applications such as computer security [22, 24]. As an example shown by [2], if an attack with overall 50.05% accuracy can reliably identify just 0.1% members without any false alarm, i.e., FPR=0, and judge the remaining samples by random guess with 50% accuracy, it puts much more risk to the model than another attack which guesses any sample with a chance of 50.05% being correct. In this case, and the latter has a high FPR. 2) We find that the previous MIAs are inconsistent with the privacy risks on individual data points, even though they could have high overall success rates (See Sec. 4.3). Specifically, they have more difficulties in identifying training samples with high privacy risks as members compared to the samples with low privacy risks, which is at odds with the intuition that samples with higher privacy

risks should be more easily identified.

We propose to address the above limitations by taking a new perspective called *memorization* [10, 11]. A data point is said to be *memorized* if the output of the model is quite sensitive to this individual data point, e.g., the prediction confidence of the learned model on this particular data point could be quite low unless it appears in the training set [10]. The concept of memorization fundamentally captures the privacy risk under the framework of differential privacy (DP) [8, 9], which is considered to be a strong privacy definition. Empirically, we find that a recent attack called Likelihood Ratio Attack (LiRA) [2] is effective in reflecting the memorization degree, as we show in Sec. 4.3. LiRA also demonstrates much better performance under the low FPR regions compared with traditional MIAs [2]. Therefore, we adopt LiRA to reinvestigate the privacy effects of both DA and AT. Through extensive experiments, we unveil several non-trivial findings (see Sec. 6 for details), which urge the community to rethink the relations among three important properties of machine learning models, including privacy leakage, generalization gap, and adversarial robustness. The major findings include:

- Unlike the previous studies [17,21,31,32,35,39] showing that the generalization gap and privacy leakage are highly correlated, our results demonstrate a much weaker correlation.
- Applying AT can increase the memorization degrees of training samples, thereby resulting in more privacy leakage. In addition, it shows that stronger adversarial robustness does not necessarily come with a cost on privacy leakage.

To the best of our knowledge, this is the first systematic evaluation of DA and AT via the lens of memorization.

2. Related Work

Data Enhancement and Privacy It has been empirically observed in [30,32] that DA is effective in mitigating MIAs. [20] further show that it is difficult to use DA to achieve substantial mitigation effects against MIAs while achieving better generalization gaps. In addition, Label Smoothing is shown to be able to increase both the privacy leakage and the test accuracy simultaneously [14, 20]. As for AT, it is also important to investigate its privacy effect, as it is recognized to be one of the most effective ways to achieve adversarial robustness, which is another important issue in the security community. [35] conduct a systematic investigation using various AT methods and find that all of them can make the model more vulnerable to MIAs. However, the MIAs used in these studies are limilted in reflecting the privacy risks of individual samples from the memorization perspective. In addition, they all do not report results by the metric under low FPR regions.

Augmented Information Improves Privacy Attack It has been shown in several studies [5, 18, 40] that exploiting the information of augmented data would help to improve the attack success rate. They can be classified into two types: augmentation-unaware and augmentation-aware attacks. The former assumes that the adversary does not have knowledge of the augmented data but simply uses random augmentation to probe the model. It has been shown that, by querying the model multiple times, using the random augmented data generated with Gaussian noise, the attack success rate can be improved [18]. The latter assumes a stronger scenario where the particular augmented data used in training is known to the adversary. [5] show that the attack success rate can be significantly improved with the knowledge of the augmented data. Moreover, [40] show that the augmentation-aware attack can obtain a higher success rate on models trained with some data augmentation than the ones without augmentation.

3. Preliminaries

We consider feed-forward DNNs under the usual supervised setting. Suppose we have a training set $D_{\text{tr}} = \{(x, y)|(x, y) \in \mathcal{X} \times \mathcal{Y}\}$, where x is the feature vector (image) and y is the corresponding label. We denote the DNN model with parameter θ as $f_\theta : \mathcal{X} \to \mathcal{Y}$. During training, the data augmentation $T : \mathcal{X} \times \mathcal{Y} \to \mathcal{P}$ is applied to the training data to improve its diversity, where $\mathcal{P}$ is the set of all the probability measures defined on the power set $2^{\mathcal{X}\times\mathcal{Y}}$. Together, the optimal parameter θ^* of the model is fitted by:

$$\theta^* = \arg\min_{\theta} \sum_{(x,y)\in D_{\text{tr}}} \mathbb{E}_{(\tilde{x},\tilde{y})\sim T(x,y)}[L(f_\theta(\tilde{x}), \tilde{y})], \quad (1)$$

where $L(\cdot, \cdot)$ is the loss function.

3.1. Data Enhancement

Since DA and AT both involve the process of adding certain examples into the training set to enhance the performances, they are known as data enhancement techniques. In this paper, we investigate eight popular DA methods and two AT methods, as described below.

DA techniques

1) Random Cropping and Flipping [33]: sample new features by randomly cropping and horizontally flipping patches from the original feature in the training set.

2) Label Smoothing [36]: replace the hard labels with the soft continuous labels by uniformly assigning probabilities to other classes. Therefore, the probability of the augmented label is $\tilde{p}_i = 1 - \frac{(n-1)\epsilon}{n}$ for $i = y$ and it is $\tilde{p}_i = \frac{\epsilon}{n}$ for $i \neq y$, where $\epsilon \in (0, 1)$ and n denotes the number of the classes.

3) DisturbLabel [38]: change a portion of the ground-truth (GT) labels to incorrect labels, namely, $\tilde{y} = \epsilon y +$

$(1-\epsilon)y_f$, where ϵ is randomly sampled from $\{0,1\}$ and $y_f \in \{1,2,...,n\} \setminus \{y\}$ denotes the incorrect label.
4) Gaussian Augmentation [6]: add Gaussian noise to each feature. The new feature $\tilde{x} = x + \epsilon$, where $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$.
5) Cutout [7]: mask out a random square area of size $M \times M$ from each feature.
6) Mixup [42]: blend two features x_0, x_1 by a random ratio γ and creates a new feature $\tilde{x} = \gamma x_0 + (1-\gamma)x_1$. The corresponding label is $\tilde{y} = \gamma y_0 + (1-\gamma)y_1$.
7) Jitter [23]: randomly change the brightness, contrast, saturation and hue of each image.
8) Distillation [16]: train an auxiliary DNN $\hat{f}$ with the original training set and use the auxiliary DNN's soft outputs and *temperature* T as GT labels of the training features when training the target DNN. The temperature T determines the flatness of the soft labels.

AT techniques
1) PGD-AT [26] : use PGD attack to generate adversarial examples x_{adv} based on original features and replaces the original feature with the adversarial examples at each iteration of the training, i.e., $\tilde{x} = x_{\text{adv}}$.
2) TRADES [43]: use PGD attack to generate adversarial examples x_{adv}, too. It differs from PGD-AT in that its loss function consists of two components: $L(f_\theta(x), y) + L(f_\theta(x), f_\theta(x_{\text{adv}}))/\lambda$. The first component is the same as the loss of the standard training while the second component encourage the model to treat x and x_{adv} equally. The two components is weighted by λ. In the framework of Eq. (1), the discrete distribution of the transformation $T(x,y)$ can be represented as $\Pr(x,y) = \frac{\lambda}{\lambda+1}$ and $\Pr(x, f_\theta(x_{\text{adv}})) = \frac{1}{\lambda+1}$.
3) AWP [37]: use a regularization to explicitly flatten the weight loss landscape of PGD-AT by a double-perturbation mechanism.
4) TRADES-AWP [37]: incorporate the regularization mechanism of AWP into TRADES method.

3.2. Membership Inference Attack

The goal of MIA is to identify whether a specific data sample was used in training a particular model or not. MIA has become one of the most widely investigated privacy attacks due to its simplicity. Many existing MIA approaches [19, 25, 31, 32, 39] are able to achieve high attack accuracy by exploiting the fact that machine learning models often behave differently to the data used or not used for training. For example, the model is often more confident to the training data than test data. Thus, by setting a threshold to certain features such as the loss, confidence score, entropy, etc, the attack can achieve high accuracy in distinguishing members from non-members. For a comprehensive overview of MIAs, we refer the readers to [17].

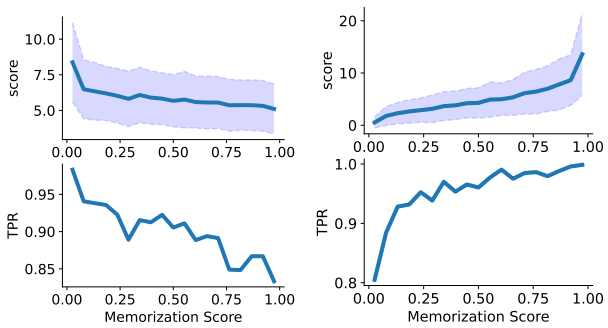


Figure 1. Feature score (top) and TPR (bottom) in terms of memorization score for a target model trained on CIFAR-100 using (a) MaxPreCA and (b) LiRA. For comparison, the feature score of the top left panel is scaled into the same scale as the top right panel.

In this paper we assume a *black-box* setting where the adversary only has query access to the outputs of the target model on given samples. Most MIAs follow [32] to train a number of so-called shadow models, which are trained similarly as the target model in order to mimic its behavior. Without loss of generality, here we assume the output is the prediction confidence and the shadow models are trained using the same data enhancement method as the target model.

4. Consistency of MIAs and Memorization

In what follows we first explain how to define memorization and then investigate the consistency of MIAs and memorization.

4.1. Memorization

A data point is said to be memorized by the model if it has a high impact on the model's behavior. In order to ensure that such impact is solely caused by this particular sample, one often needs to use the *leave-one-out setting*. Namely, except this particular sample all other settings are exactly the same. As an example, [10] defines a memorization score which measures how much information about the label of an individual data sample is being memorized by the model. Specifically, given the training set D_{tr} and the learning algorithm $\mathcal{A}$, for an arbitrary sample $(x,y) \in D_{\text{tr}}$, its memorization score is defined as:

$$\begin{aligned}&\text{mem}(\mathcal{A}, D_{\text{tr}}, (x,y)) \\ &= \Pr_{f_\theta \sim \mathcal{A}(D_{\text{tr}})}[f_\theta(x) = y] - \Pr_{f_\theta \sim \mathcal{A}(D_{\text{tr}} \setminus (x,y))}[f_\theta(x) = y],\end{aligned} \tag{2}$$

where $D_{\text{tr}} \setminus (x,y)$ denotes the data set D_{tr} with the sample (x,y) being removed. This definition is shown effective as it is able to assign atypical examples or outliers with high memorization scores and typical or easy samples with low memorization scores on various datasets including CIFAR-

100 and ImageNet datasets [11]. This complies perfectly with the intuition that an atypical example or outlier is often at a higher privacy risk as the model will behave quite differently on it when it is in or out of the training set.

4.2. Traditional MIAs Obtain Low Consistency with Memorization Scores

Most traditional MIAs exploit the fact that models are overconfident to the training data [15]. We use the widely adopted maximum predication confidence-based attack [31] (called MaxPreCA in this paper) as an example, whose main idea is to classify samples with the maximum predication confidence higher than a given threshold as members, otherwise as non-members. To investigate if this attack can capture the privacy risk of individual data points, in Fig. 1(a) we demonstrate its feature score, i.e., the maximum prediction confidence (top panel) and the True Positive Rate (TPR) (bottom panel) versus the memorization score [11]. We first divided all the samples into 20 bins according to their memorization score. The top panel shows the averaged feature scores (the solid blue line) of the samples in different bins along with their standard deviation (purple shadow). The bottom panel shows TPR calculated by using the optimal threshold for all samples. It is obvious that the higher the memorization score is, the less probable it gets identified as members correctly. Therefore, the attack result has a low consistency with memorization scores.

4.3. LiRA Obtains High Consistency with Memorization Scores

The main difference between LiRA and traditional MIAs is that LiRA considers the distribution of the model's prediction on an individual data point when it is in or out of the training set. It requires training a number of shadow models such that for each sample (x, y), half models include it in the training set and the other half models do not, denoted as IN and OUT models, respectively. Denote $f_\theta(x)_y$ as the model's confidence of x on label y. Denote the sets of scaled confidences of sample (x, y) computed using IN and OUT models as $\mathcal{Q}_{\text{in}}$ and $\mathcal{Q}_{\text{out}}$, respectively:

$$\begin{aligned} \mathcal{Q}_{\text{in}} &= \{\phi(f_\theta(x)_y) : (x, y) \in D_{\text{tr}}\} \\ \mathcal{Q}_{\text{out}} &= \{\phi(f_\theta(x)_y) : (x, y) \notin D_{\text{tr}}\}, \end{aligned} \tag{3}$$

where $\phi(p) = \log\left(\frac{p}{1-p}\right)$. $\mathcal{Q}_{\text{in}}$ and $\mathcal{Q}_{\text{out}}$ are used to fit two Gaussian distributions, denoted as IN distribution $\mathcal{N}\left(\mu_{\text{in}}, \sigma^2_{\text{in}}\right)$ and OUT distribution $\mathcal{N}\left(\mu_{\text{out}}, \sigma^2_{\text{out}}\right)$, respectively. Given an arbitrary sample, a standard likelihood-ratio test is performed to determine which distribution it more likely belongs to, where the likelihood ratio is:

$$\Lambda = \frac{p\left(\phi(f_\theta(x)_y) \mid \mathcal{N}\left(\mu_{\text{in}}, \sigma^2_{\text{in}}\right)\right)}{p\left(\phi(f_\theta(x)_y) \mid \mathcal{N}\left(\mu_{\text{out}}, \sigma^2_{\text{out}}\right)\right)}. \tag{4}$$

A sample will be classified as member if Λ is higher than a threshold. Compared to traditional MIAs, LiRA is shown effective under low FPR regions [2]. As for the memorization, in Fig. 1(b) we demonstrate the feature score, i.e., Λ (top panel) and the True Positive Rate (TPR) (bottom panel) versus the memorization score [11] using LiRA. Clearly, we can see that the higher a sample's memorization score is, the more likely it is correctly detected as a member, which is a reasonable result. Therefore, in what follows we will use LiRA as a tool to investigate how different DA and AT methods affect the privacy.

5. Evaluating Privacy Effects of Data Enhancement

We now proceed to evaluate the privacy effects of both DA and AT. Through the experiments we aim to investigate the relations of privacy, adversarial robustness, and generalization gap, as they are all crucial properties for a machine learning model.

5.1. Experimental Settings

We used 32 3080 GPUs to perform the experiments. The code for the experiments is implemented by Pytorch [27] and submitted along with the paper.

Dataset Following [2, 12, 20], we used the CIFAR-10 and CIFAR-100 datasets [1] for MIA evaluations. Both CIFAR-10 and CIFAR-100 contain 60,000 natural images with resolution of 32×32 from 10 and 100 categories, respectively.

MIA Settings We used LiRA to measure the privacy leakage of different DA and AT methods. For each data enhancement on each dataset, we trained 128 models with roughly 30,000 data points randomly selected from 60,000 data points. For each individual data point, we guaranteed that there were 64 IN models and 64 OUT models. All models used the same training recipes except the data enhancement strategies. Then we randomly chose one of the trained models as the target model and the remaining 127 models as the shadow models. After that, we did evaluation on all the 60,000 data points. To perform MIA, we queried the target model for ten times on each data point, including the original image, four shifted (± 4 pixels) variants, and their flipped versions. For each DA and AT method, we repeated the evaluation on ten randomly selected target models and then reported the mean and standard deviation for each metric.

Hyper-Parameters We used ResNet-18 [13] in all the experiments. Due to computational resource constraints, we did not conduct the same experiments on other architectures. Each model was optimized by stochastic gradient descent with an initial learning rate of 0.1 and a momentum of 0.9 for 100 epochs on a single GPU. Multi-step decay which scales the learning rate by 0.1 was used on the 75th and 90th epochs. The batch size is set to 256.

Method	Training Acc	Test Acc	TPR @ 0.1% FPR	TPR @ 0.001% FPR	Log-scale AUC	MIA Balanced Acc
Base	100.0 ± 0.0	92.8 ± 0.2	8.20 ± 0.45	2.45 ± 0.93	0.815 ± 0.007	63.34 ± 0.26
Smooth	100.0 ± 0.0	92.9 ± 0.3	5.22 ± 0.66	0.14 ± 0.07	0.734 ± 0.012	62.28 ± 0.86
Disturblabel	99.9 ± 0.0	92.7 ± 0.3	5.88 ± 0.83	0.70 ± 0.45	0.775 ± 0.013	61.69 ± 0.24
Noise	100.0 ± 0.0	92.6 ± 0.2	$\mathbf{8.33} \pm 0.26$	$\mathbf{2.79} \pm 0.68$	$\mathbf{0.819} \pm 0.004$	$\mathbf{63.56} \pm 0.24$
Cutout	100.0 ± 0.0	93.1 ± 0.4	7.71 ± 0.39	$\mathbf{2.48} \pm 1.03$	0.811 ± 0.010	63.23 ± 0.26
Mixup	99.7 ± 0.1	93.0 ± 0.2	5.17 ± 0.51	1.31 ± 0.40	0.779 ± 0.008	60.05 ± 0.53
Jitter	100.0 ± 0.0	92.7 ± 0.2	$\mathbf{8.24} \pm 0.35$	$\mathbf{2.97} \pm 0.76$	$\mathbf{0.819} \pm 0.004$	$\mathbf{63.41} \pm 0.31$
Distillation	99.9 ± 0.0	93.2 ± 0.2	7.04 ± 0.33	2.19 ± 0.70	0.805 ± 0.005	61.57 ± 0.39
PGD-AT	99.2 ± 0.1	82.2 ± 0.2	$\mathbf{23.78} \pm 0.89$	$\mathbf{10.52} \pm 2.30$	$\mathbf{0.897} \pm 0.005$	$\mathbf{78.82} \pm 0.37$
TRADES	96.2 ± 0.2	80.0 ± 0.4	$\mathbf{17.88} \pm 1.56$	$\mathbf{8.14} \pm 1.12$	$\mathbf{0.881} \pm 0.006$	$\mathbf{77.21} \pm 0.65$
AWP	93.2 ± 2.0	82.6 ± 0.9	$\mathbf{10.58} \pm 3.48$	$\mathbf{3.06} \pm 1.81$	$\mathbf{0.828} \pm 0.045$	$\mathbf{72.13} \pm 3.76$
TRADES-AWP	91.9 ± 0.5	80.5 ± 0.2	$\mathbf{12.43} \pm 0.89$	$\mathbf{3.48} \pm 1.36$	$\mathbf{0.848} \pm 0.006$	$\mathbf{74.86} \pm 0.80$

Table 1. Attack success rates of different data enhancement on CIFAR-10. The 2nd and 3rd columns show the training and test accuracies of each method, respectively. The 4th - 7th columns show four metrics to evaluate the extent of privacy leakage. We highlight the MIA success rates for different DA and AT methods that are larger than that for Base.

Method	Training Acc	Test Acc	TPR @ 0.1% FPR	TPR @ 0.001% FPR	Log-scale AUC	MIA Balanced Acc
Base	100.0 ± 0.0	70.3 ± 0.3	34.17 ± 1.05	17.24 ± 2.93	0.922 ± 0.002	83.17 ± 0.24
Smooth	100.0 ± 0.0	72.2 ± 0.4	$\mathbf{39.21} \pm 1.25$	$\mathbf{19.88} \pm 3.86$	$\mathbf{0.932} \pm 0.004$	$\mathbf{86.35} \pm 0.22$
Disturblabel	98.0 ± 0.2	69.9 ± 0.3	19.53 ± 0.64	6.54 ± 2.58	0.879 ± 0.007	76.65 ± 0.28
Noise	100.0 ± 0.0	69.7 ± 0.3	33.83 ± 0.91	$\mathbf{18.31} \pm 2.78$	$\mathbf{0.923} \pm 0.003$	$\mathbf{83.26} \pm 0.13$
Cutout	100.0 ± 0.0	70.3 ± 0.3	$\mathbf{34.71} \pm 1.58$	$\mathbf{17.25} \pm 5.02$	$\mathbf{0.923} \pm 0.005$	$\mathbf{83.53} \pm 0.22$
Mixup	99.7 ± 0.1	71.2 ± 0.4	32.73 ± 1.13	$\mathbf{19.18} \pm 2.48$	0.922 ± 0.003	82.39 ± 0.50
Jitter	100.0 ± 0.0	70.3 ± 0.3	$\mathbf{34.19} \pm 0.90$	$\mathbf{18.37} \pm 3.62$	$\mathbf{0.924} \pm 0.003$	$\mathbf{83.35} \pm 0.17$
Distillation	99.8 ± 0.0	72.6 ± 0.3	28.70 ± 0.83	14.58 ± 2.29	0.911 ± 0.002	79.46 ± 0.14
PGD-AT	99.5 ± 0.0	51.3 ± 0.3	$\mathbf{68.63} \pm 0.88$	$\mathbf{47.85} \pm 4.26$	$\mathbf{0.972} \pm 0.001$	$\mathbf{93.62} \pm 0.10$
TRADES	98.0 ± 0.3	49.0 ± 0.5	$\mathbf{60.23} \pm 0.87$	$\mathbf{37.45} \pm 5.15$	$\mathbf{0.963} \pm 0.002$	$\mathbf{92.19} \pm 0.23$
AWP	85.3 ± 0.8	54.4 ± 0.3	$\mathbf{39.92} \pm 2.40$	$\mathbf{17.34} \pm 4.22$	$\mathbf{0.931} \pm 0.006$	$\mathbf{88.10} \pm 0.38$
TRADES-AWP	95.9 ± 0.6	51.3 ± 0.4	$\mathbf{57.51} \pm 2.02$	$\mathbf{35.57} \pm 3.73$	$\mathbf{0.960} \pm 0.002$	$\mathbf{91.92} \pm 0.36$

Table 2. Attack success rates of different data enhancement on CIFAR-100. The same conventions are used as in Tab. 1.

The hyper-parameters of each DA method was set to achieve relatively high test accuracy by searching (see *Supplementary Materials* for details). Unless other specified, for all AT methods, we set the maximal perturbation ϵ under infinite norm to be 8. We set the step size to be $\epsilon/8$ and the number of iterative steps to be 10. In addition, following the default setting of each method, the regularization parameter λ was set to be $1/6$ for TRADES, the perturbation intensity γ was set to be $1e^{-2}$ for AWP and $5e^{-3}$ for TRADES-AWP.

5.2. Evaluation Results

Tab. 1 and 2 show the training and test accuracies and the MIA results by multiple queries on CIFAR-10 and CIFAR-100, respectively. We denote Random Cropping and Flipping by the *Base* method. Different from most previous studies [20, 35, 40], we evaluated the privacy leakage of seven DA methods from Label Smoothing to Distillation and four AT methods (Sec. 3) combined with Base. This is a practical setting as Random Cropping and Flipping has now become a default setting in computer vision field and it often brings considerable improvements on test accuracy. It has been criticized in [29] that the models with low test accuracies are not practically useful to evaluate the privacy leakage. See the results of different DA methods without Base in *Supplementary Materials*, where the test accuracy of Base indeed exceeds those of other DA methods by at least 8.4% on CIFAR-10 and 12.6% on CIFAR-100. Unless otherwise specified, all DA and AT methods also use Base as default.

With LiRA, we evaluated all models using four metrics: TPR @ 0.1% FPR, TPR @ 0.001% FPR, Log-scale Area Under the Curve (AUC), and the Balanced Accuracy. The numbers after $\pm$ denote the standard deviations. As mentioned earlier, it is more reasonable to use the metric of TPR under low FPR regions for evaluating privacy leak-

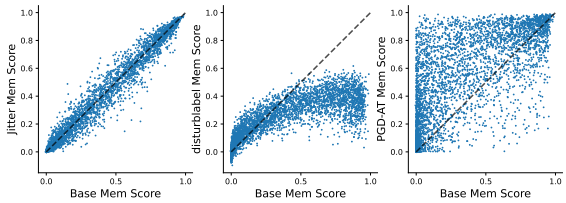


Figure 2. Memorization scores of 5,000 randomly selected samples using Jitter (top), Disturblabel (middle) and PGD-AT model (bottom) v.s., Base model.

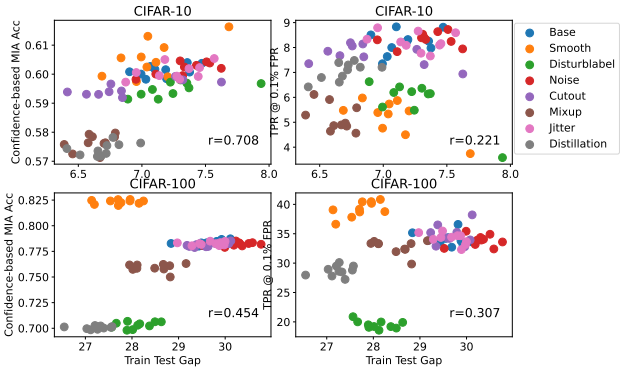
age. However, on the one hand, we empirically found that the results of TPR @ 0.001% FPR were unstable since their standard deviations were relatively large. On the other hand, 0.001% FPR might be too strict as it has been shown in [11] that both CIFAR-10 and CIFAR-100 contain quite a few pairs of hard samples that are very similar. These samples will inevitably be misclassified with high confidence as members when their counterparts are in the member set, thereby causing some false positive cases and resulting in FPRs that exceed the 0.001% tolerance. Therefore, we mainly use TPR @ 0.1% FPR as the attack success rate for the following analysis.

6. Analysis

In what follows we first verify the effectiveness of our attack results, and then analyze the relations between privacy, generalization gap and adversarial robustness.

6.1. Attack Results and Memorization Degrees

To verify whether the attack results shown in Tab. 1 and 2 indeed reflect the degree of memorization, in Fig. 2 we compare the memorization scores of 5,000 randomly selected samples computed using the same method as in [11] for three cases on CIFAR-100: Jitter, Disturblabel, and PGD-AT v.s. Base. Wherein the attack success rates are similar to, lower than, and higher than Base, respectively. Clearly, the corresponding changes in the memorization scores are consistent with the attack success rates. The memorization scores of the samples for Base and Jitter are similar (lying around the diagonal), which explains the similar attack success rate against the two methods. The memorization scores of many samples for Disturblabel are lower than for Base, especially the samples with high memorization scores for Base. Therefore, one reason why Disturblabel reduces the privacy leakage is that it can reduce the memorization scores of many atypical samples. The memorization scores for PGD-AT are in general higher than those for Base (the points are distributed on the upper of the diagonal). Thus, one major reason why AT causes a higher privacy leakage is that it memorizes many training samples that are not memorized by non-AT models.



(a) (b)

Figure 3. Attack success rate versus train-test gap of different DA models on CIFAR-10 and CIFAR-100 using (a) MaxPreCA and (b) LiRA, respectively. r stands for the Pearson correlation coefficient.

6.2. Privacy and Generalization Gap

Reducing the generalization gap does not necessarily reduce the vulnerability to MIAs. Many previous studies have shown that the attack success rates of MIAs are highly correlated with the generalization gap, i.e., the degree of overfitting [21, 31, 32, 35, 39]. To verify whether such high correlation is still true from the memorization perspective, in Fig. 3 we demonstrate the attack success rate in terms of train-test accuracy gap of all DA models for both CIFAR-10 and CIFAR-100 datasets using MaxPreCA and LiRA, respectively. It is easily observed that compared to the traditional attack results, our results demonstrate a more scattered distribution. We also compute the Pearson correlation coefficient r for each plot. As shown there, the Pearson correlation coefficients r of our results are significantly lower than the traditional results, e.g., for CIFAR-10, our r is only 0.221, which is much lower than 0.708 using the traditional attack. Hence, *via the lens of memorization, the generalization gap and privacy leakage appear less correlated than that of the previous results.*

It is easy to understand why many traditional attack results are sensitive to the generalization gap, as their success rate depends heavily on how different the model behaves for training and test samples. We remark that there is a distinction between memorization and overfitting: memorization is only necessary but not sufficient for overfitting [10], i.e., memorizing some training samples does not always cause overfitting. In fact, it has been both theoretically proved and empirically verified in [10, 11] that the memorizing certain long-tailed samples will help in decreasing the generalization gap. As a consequence, traditional attack might underestimate the privacy leakage for non-overfitted models, making the correlation coefficient unnecessarily high. This issue can be alleviated in our setting as we measure the

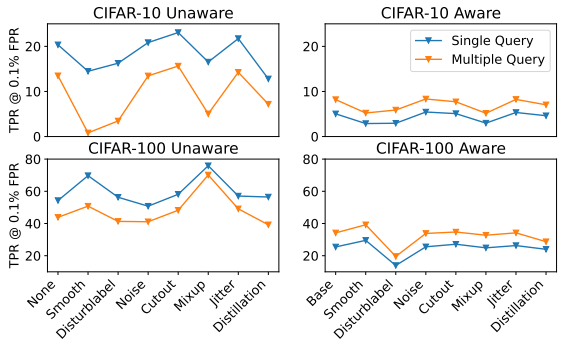


Figure 4. Attack success rates of single query and multiple queries in two cases: augmentation-unaware (left) and augmentation-aware (right). We evaluated different DA methods on CIFAR-10 and CIFAR-100 datasets, respectively. *None* in the figure stands for models trained without any DA.

Dataset	Method	Adversarial Acc	TPR @ 0.1% FPR
CIFAR-10	PGD-AT	38.8 ± 0.4	23.78 ± 0.89
	TRADES	45.2 ± 0.3	17.88 ± 1.56
	AWP	45.9 ± 0.1	10.58 ± 3.48
	TRADES-AWP	48.8 ± 0.2	12.43 ± 0.89
CIFAR-100	PGD-AT	16.9 ± 0.1	68.63 ± 0.88
	TRADES	19.7 ± 0.4	60.23 ± 0.87
	AWP	23.9 ± 0.1	39.92 ± 2.40
	TRADES-AWP	23.3 ± 0.2	57.51 ± 2.02

Table 3. The accuracies on adversarial examples (Adversarial Acc) and privacy leakage of different AT models on CIFAR-10 and CIFAR-100. The accuracies are evaluated using PGD with $\epsilon = 8$ and 20 iteration steps.

privacy leakage via the memorization perspective, the root cause of privacy leakage. We remark that even though [39] also pointed out that overfitting is not the only reason for causing vulnerability to privacy attacks, they did not explicitly identify what are other factors and their attack results still demonstrate a higher correlation compared with ours.

Data augmentation is not necessary an effective defense for MIAs. By inspecting the attack results for DA models, the privacy effects vary significantly across different DA methods. For example, Distillation and Disturblabel are shown effective in reducing the vulnerability to the privacy attack. Mixup, Cutout, Jitter and Gaussian noise methods do not seem to have big impacts on the attack success rate. The main reason is that applying DA does not always reduce the memorization scores of training samples, e.g., the Jitter model shown in Fig. 2. Moreover, among all DA methods, Label Smoothing have drawn a lot of attention as it has been shown in [14, 20] that applying Label Smoothing will make the model more susceptible to MIAs. To verify this, we computed the balanced accuracy using the traditional attack MaxPreCA [31]. As shown in the left plots of Fig. 3, Label Smoothing does increase the attack accuracies compared to Base for both datasets. However, from the memorization perspective, it did not demonstrate the same tendency. By inspecting the right plots of Fig. 3 we note that Label Smoothing demonstrates an inconsistent behavior on different datasets. On CIFAR-100 the privacy leakage is higher than Base while on CIAFR-10 the privacy leakage is lower. We conjecture that the privacy effect of Label Smoothing might be dependent on the complexity of datasets. Hence, the claim of Label Smoothing would consistently amplify the privacy leakage is not true. Overall, we conclude that it is difficult to give a general claim about whether DA can help to mitigate the privacy attack or not. We remind that extra attentions should be paid when relying on DA as a defense technique against MIAs.

Multiple queries can only enhance the attack if the augmentation method is known. As stated in Sec. 1, previous studies have shown that using augmented data to conduct multiple queries would enhance the attack success rate. To investigate this, we queried the target model using ten augmented counterparts generated by Base method for each data point. We then targeted all the DA models trained on Base as the augmentation-aware case. The augmentation-unaware case was then evaluated by targeting the DA models without using Base. As shown in Fig. 4, multiple queries did help improve the attack success rate for the augmentation-aware case, whereas for the augmentation-unaware case, they resulted in an opposite effect, i.e., lowering the attack success rate.

6.3. Privacy and Adversarial Robustness

Applying AT will make the model memorize more training samples, thereby causing more privacy leakage. As shown in Tab. 1 and 2, applying AT significantly increases privacy leakage compared to the non-AT models. For example, the TPR @ 0.1% FPR increases from 34.17% to 60.23% for Base to TRADES on CIFAR-100. One reason is that applying AT will force the model to fit all the adversarial examples found in the ℓ_∞ ball around each training sample, which often increases the influence of each sample on the trained model, thereby resulting in a higher privacy risk. To visualize the effect of applying AT, in Fig. 5 we choose three examples in CIFAR-100 with different memorization scores and draw their corresponding distributions of normalized confidence ϕ evaluated by IN and OUT models using Base and all four types of AT models. Clearly, if the IN and OUT distributions of a particular sample are more separated, it implies that the sample is at a higher privacy risk. We can see that for samples that has low privacy risks (e.g., Raccoon and Train), applying AT would make the distribution more separable. Note that there is a bot-

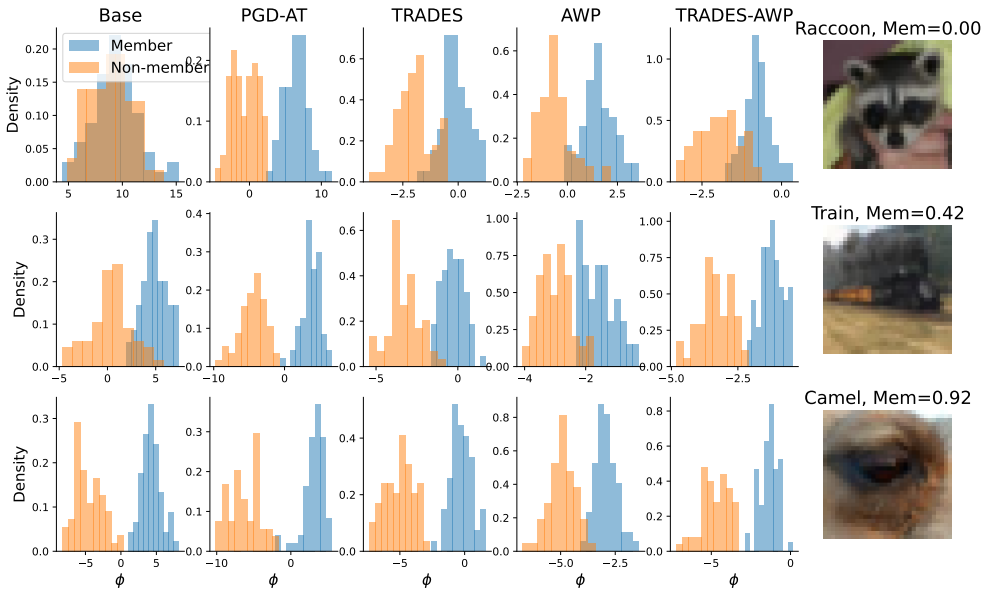


Figure 5. The distributions of normalized confidence ϕ of three samples with different memorization scores using Base and four AT models. Each raw corresponds to a sample.

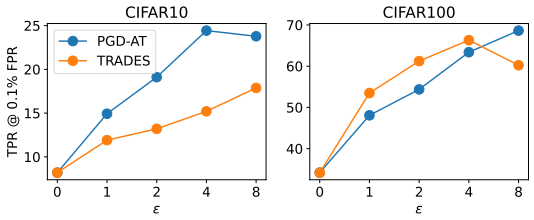


Figure 6. Attack success rates of PGD-AT and TRADES under different ϵ on CIFAR-10 and CIFAR-100 datasets.

tleneck of such effect: for samples that already at a high privacy risk (e.g., Camel with a high memorization score), applying AT would not make much difference as the distributions are very separated for all models. overall, we conclude that one major reason why AT causes a higher privacy leakage is that it memorizes many training samples that are not memorized by non-AT models.

Better adversarial robustness does not necessarily make the model more vulnerable to privacy attacks. To further investigate the relation between adversarial robustness and privacy leakage, in Tab. 3 we compare adversarial robustness and attack success rate using different AT methods on both CIFAR-10 and CIFAR-100. We can clearly see that compared to TARDES and PGD-AT, both AWP and TRADES-AWP achieve higher adversarial accuracies, while the attack success rates are lower. Thus, improving the adversarial robustness does not necessarily come with a cost on privacy. In addition to the attack results using different AT methods, it is also interesting to see how does the attack result changes along with varying parameters. Since ϵ is a critical parameter for AT, in Fig. 6 we compare the attack results of different ϵ using both PGD-AT and TRADES models on CIFAR-10 and CIFAR-100. We can see that overall the attack success rate tends to increase along with ϵ (at least for $\epsilon < 8$). In addition, we also observe a similar bottleneck effect (see Fig. A1 in *Supplementary Materials*). Thus, for samples with low memorization scores, increasing ϵ will increase the privacy risk while for samples with high memorization scores, it can hardly make much difference.

7. Conclusion

In this paper, we reinvestigate the privacy effect of applying data augmentation and adversarial training to machine learning models via a new perspective, namely the degree of memorization. Such reinvestigation is quite necessary as we found that the attacks deployed in previous studies for measuring the privacy leakage produces misleading results: the training samples with low privacy risks are more prone to be identified as members compared to the ones with high privacy risks. Through a systematic evaluation, we reveal some findings are in conflict with previous results, e.g., the generalization gap and privacy leakage are shown less correlated than previous results and Label Smoothing does not always amplify the privacy leakage. Moreover, we also show that improving the adversarial robustness (via adversarial training) does not necessarily make the model more vulnerable to privacy attacks. Our results call for more investigations on the privacy of machine learning models from the memorization perspective.

References

[1] Krizhevsky Alex, Hinton Geoffrey, et al. Learning multiple layers of features from tiny images. 2009. 4

[2] Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramèr. Membership inference attacks from first principles. In *IEEE Symposium on Security and Privacy (SP)*, pages 1897–1914. IEEE, 2022. 1, 2, 4

[3] Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *USENIX Security Symposium (USENIX)*, pages 267–284, 2019. 1

[4] Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom B. Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. Extracting training data from large language models. In *USENIX Security Symposium (USENIX)*, pages 2633–2650, 2021. 1

[5] Christopher A. Choquette-Choo, Florian Tramèr, Nicholas Carlini, and Nicolas Papernot. Label-only membership inference attacks. In *Int. Conf. Mach. Learn. (ICML)*, pages 1964–1974. PMLR, 2021. 2

[6] Jeremy M. Cohen, Elan Rosenfeld, and J. Zico Kolter. Certified adversarial robustness via randomized smoothing. In *Int. Conf. Mach. Learn. (ICML)*, pages 1310–1320, 2019. 3

[7] Terrance Devries and Graham W. TaylorW. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017. 3

[8] Cynthia Dwork. Differential privacy. In *ICALP, pp. 1–12*, 2006. 2

[9] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam D. Smith. Calibrating noise to sensitivity in private data analysis. *J. Priv. Confidentiality*, 7(3):17–51, 2016. 2

[10] Vitaly Feldman. Does learning require memorization? a short tale about a long tail. In *Annual ACM SIGACT Symposium on Theory of Computing (STOC)*, pages 954–959, 2020. 2, 3, 6

[11] Vitaly Feldman and Chiyuan Zhang. What neural networks memorize and why: Discovering the long tail via influence estimation. *Adv. Neural Inform. Process. Syst. (NeurIPS)*, 33:2881–2891, 2020. 2, 4, 6

[12] Ganesh Del Grosso, Hamid Jalalzai, Georg Pichler, Catuscia Palamidessi, and Pablo Piantanida. Leveraging adversarial examples to quantify membership information leakage. *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2022. 4

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 770–778, 2016. 4

[14] Dominik Hintersdorf, Lukas Struppek, and Kristian Kersting. To trust or not to trust prediction scores for membership inference attacks. *arXiv preprint arXiv:2111.09076*, 2022. 1, 2, 7

[15] Dominik Hintersdorf, Lukas Struppek, and Kristian Kersting. To trust or not to trust prediction scores for membership inference attacks. In Luc De Raedt, editor, *IJCAI*, pages 3043–3049, 2022. 4

[16] Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 3

[17] Hongsheng Hu, Zoran Salcic, Gillian Dobbie, and Xuyun Zhang. Membership inference attacks on machine learning: A survey. *ACM Computing Surveys (CSUR)*, 2021. 2, 3

[18] Bargav Jayaraman, Lingxiao Wang, Katherine Knipmeyer, Quanquan Gu, and David Evans. Revisiting membership inference under realistic assumptions. *Proceedings on Privacy Enhancing Technologies*, 2021(2), 2021. 2

[19] Jinyuan Jia, Ahmed Salem, Michael Backes, Yang Zhang, and Neil Zhenqiang Gong. Memguard: Defending against black-box membership inference attacks via adversarial examples. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 259–274, 2019. 3

[20] Yigitcan Kaya and Tudor Dumitras. When does data augmentation help with membership inference attacks? In *Int. Conf. Mach. Learn. (ICML)*, pages 5345–5355, 2021. 1, 2, 4, 5, 7

[21] KlasLeino and Matt Fredrikson. Stolen memories: Leveraging model memorization for calibrated {White-Box} membership inference. In *USENIX security symposium (USENIX)*, pages 1605–1622, 2020. 1, 2, 6

[22] Jeremy Z. Kolter and Marcus A. Maloof. Learning to detect and classify malicious executables in the wild. *Journal of Machine Learning Research*, 7(12), 2006. 1

[23] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inform. Process. Syst. (NeurIPS)*, 25, 2012. 3

[24] Aleksandar Lazarevic, Levent Ert"oz, Vipin Kumar, Aysel Ozgur, and Jaideep Srivastava. A comparative study of anomaly detection schemes in network intrusion detection. In *Proceedings of the SIAM international conference on data mining*, pages 25–36, 2003. 1

[25] Yunhui Long, Vincent Bindschaedler, Lei Wang, Diyue Bu, Xiaofeng Wang, Haixu Tang, Carl A. Gunter, and Kai Chen. Understanding membership inferences on well-generalized learning models. *arXiv preprint arXiv:1802.04889*, 2018. 3

[26] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *Int. Conf. Learn. Represent. (ICLR)*, 2018. 3

[27] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Adv. Neural Inform. Process. Syst. (NeurIPS)*, pages 8024–8035, 2019. 4

[28] Amir Hossein Poorjam, Yordan P. Raykov, Reham Badawy, Jesper Rindom Jensen, Mads Graesboll Christensen, and Max A. Little. Quality control of voice recordings in remote parkinson's disease monitoring using the infinite hidden markov model. In *Proc. Int. Conf. Acoust., Speech, Signal Process., pp. 805-809*, 2019. 1

[29] Shahbaz Rezaei and Xin Liu. On the difficulty of membership inference attacks. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 7892–7900, 2021. 1, 5

[30] Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, Yann Ollivier, and Hervé Jégou. White-box vs black-box: Bayes optimal strategies for membership inference. In *Int. Conf. Mach. Learn. (ICML)*, pages 5558–5567, 2019. 1, 2

[31] Ahmed Salem, Yang Zhang, Mathias Humbert, Pascal Berrang, Mario Fritz, and Michael Backes. Ml-leaks: Model and data independent membership inference attacks and defenses on machine learning models. *Network and Distributed Systems Security Symposium*, 2019. 1, 2, 3, 4, 6, 7

[32] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *IEEE symposium on security and privacy (SP)*, pages 3–18, 2017. 1, 2, 3, 6

[33] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2

[34] Congzheng Song, Thomas Ristenpart, and Vitaly Shmatikov. Machine learning models that remember too much. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 587–601, 2017. 1

[35] Liwei Song, Reza Shokri, and Prateek Mittal. Privacy risks of securing machine learning models against adversarial examples. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 241–257, 2019. 1, 2, 5, 6

[36] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 2818–2826, 2016. 1, 2

[37] Dongxian Wu, Shu-Tao Xia, and Yisen Wang. Adversarial weight perturbation helps robust generalization. In *Adv. Neural Inform. Process. Syst. (NeurIPS)*, 2020. 3

[38] Lingxi Xie, Jingdong Wang, Zhen Wei, Meng Wang, and Qi Tian. Disturblabel: Regularizing cnn on the loss layer. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 4753–4762, 2016. 2

[39] Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy risk in machine learning: Analyzing the connection to overfitting. In *IEEE computer security foundations symposium (CSF)*, pages 268–282, 2018. 1, 2, 3, 6, 7

[40] Da Yu, Huishuai Zhang, Wei Chen, Jian Yin, and Tie-Yan Liu. How does data augmentation affect privacy in machine learning? In *AAAI*, volume 35, pages 10746–10753, 2021. 2, 5

[41] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Commun. ACM*, 64(3):107–115, 2021. 1

[42] Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *Int. Conf. Learn. Represent. (ICLR)*, 2018. 3

[43] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P. Xing, Laurent El Ghaoui, and Michael I. Jordan. Theoretically principled trade-off between robustness and accuracy. In *Int. Conf. Mach. Learn. (ICML)*, pages 7472–7482, 2019. 3

Method	Training Acc	Test Acc	TPR @ 0.1% FPR	TPR @ 0.001% FPR	Log-scale AUC	MIA Balanced Acc
None	100.0 ± 0.0	82.9 ± 0.5	20.35 ± 4.31	9.44 ± 3.24	0.885 ± 0.013	76.25 ± 2.18
None + Smooth	100.0 ± 0.0	83.7 ± 0.5	14.48 ± 3.03	2.37 ± 1.79	0.839 ± 0.024	72.91 ± 1.60
None + Disturblabel	100.0 ± 0.0	84.1 ± 0.6	16.26 ± 1.03	3.43 ± 2.48	0.853 ± 0.016	72.34 ± 0.89
None + Noise	100.0 ± 0.0	82.4 ± 0.7	$\mathbf{20.84} \pm 3.69$	8.59 ± 2.88	$\mathbf{0.886} \pm 0.011$	$\mathbf{76.90} \pm 2.52$
None + Cutout	100.0 ± 0.0	84.0 ± 0.6	$\mathbf{23.07} \pm 0.80$	$\mathbf{10.53} \pm 1.85$	$\mathbf{0.894} \pm 0.004$	$\mathbf{77.42} \pm 0.37$
None + Mixup	100.0 ± 0.0	83.7 ± 0.4	16.53 ± 1.42	4.84 ± 1.58	$\mathbf{0.867} \pm 0.006$	$\mathbf{76.53} \pm 0.99$
None + Jitter	100.0 ± 0.0	82.0 ± 1.3	$\mathbf{21.73} \pm 5.85$	8.77 ± 3.64	$\mathbf{0.886} \pm 0.020$	$\mathbf{77.63} \pm 2.61$
None + Distillation	100.0 ± 0.0	84.4 ± 0.4	12.79 ± 1.81	5.16 ± 1.18	0.851 ± 0.010	69.25 ± 0.99

Table A1. Attack success rates of different DA methods without using Base on CIFAR-10. The 2nd and 3rd columns show the training and test accuracies of each method, respectively. The 4th - 7th columns show four metrics to evaluate the extent of privacy leakages. We highlight the MIA success rates for different DA methods that are larger than that for None.

Method	Training Acc	Test Acc	TPR @ 0.1% FPR	TPR @ 0.001% FPR	Log-scale AUC	MIA Balanced Acc
None	100.0 ± 0.0	54.0 ± 0.9	54.26 ± 10.72	30.68 ± 11.03	0.954 ± 0.014	92.96 ± 2.28
None + Smooth	100.0 ± 0.0	53.7 ± 2.0	$\mathbf{69.71} \pm 3.51$	$\mathbf{41.81} \pm 9.55$	$\mathbf{0.972} \pm 0.004$	$\mathbf{96.76} \pm 0.33$
None + Disturblabel	100.0 ± 0.0	55.5 ± 0.5	$\mathbf{56.30} \pm 1.22$	$\mathbf{37.37} \pm 4.58$	$\mathbf{0.959} \pm 0.003$	90.96 ± 0.23
None + Noise	100.0 ± 0.0	53.5 ± 1.2	50.75 ± 8.61	28.61 ± 10.75	0.950 ± 0.011	91.79 ± 1.70
None + Cutout	100.0 ± 0.0	54.1 ± 0.8	$\mathbf{58.14} \pm 5.60$	$\mathbf{36.64} \pm 7.94$	$\mathbf{0.961} \pm 0.007$	92.87 ± 1.04
None + Mixup	100.0 ± 0.0	49.3 ± 0.7	$\mathbf{75.88} \pm 1.09$	$\mathbf{51.04} \pm 6.80$	$\mathbf{0.978} \pm 0.002$	$\mathbf{96.13} \pm 0.06$
None + Jitter	100.0 ± 0.0	53.1 ± 0.9	$\mathbf{57.05} \pm 11.22$	30.21 ± 14.71	0.955 ± 0.016	$\mathbf{93.47} \pm 2.18$
None + Distillation	100.0 ± 0.0	57.7 ± 1.7	$\mathbf{56.45} \pm 3.95$	$\mathbf{35.61} \pm 7.31$	$\mathbf{0.959} \pm 0.006$	90.39 ± 0.88

Table A2. Attack success rates of different DA methods without using Base on CIFAR-100. The same conventions are used as in Tab. A1.

A. The Hyper-Parameters of Each Data Augmentation

As stated in the paper, the hyper-parameter of each DA method was set to achieve relatively high test accuracy by trying various values. Here we report the values we tried and the final values used when training 128 shadow models for each DA method.

Random Cropping and Flipping First, the images with resolution of 32×32 were padded with zeros of 4 pixels on each end. Then the padded images with the resolution of 36×36 were randomly cropped out to form inputs with the resolution of 32×32. Finally, the inputs were randomly flipped horizontally. Unless otherwise specified, all other DA and AT methods also use this as default.

Label Smoothing We tried ϵ including 0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, and 0.8. Finally, we chose 0.2 on CIFAR-10 and 0.3 on CIFAR-100.

Disturblabel We tried ϵ including 0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.425, 0.45, 0.5, 0.525, 0.55, 0.575, and 0.6. Finally, we chose 0.05 on CIFAR-10 and 0.3 on CIFAR-100.

Gaussian Augmentation We tried σ including 0.025, 0.01, 0.05, 0.075, 0.1, 0.125, 0.15, 0.175, 0.2, 0.225, 0.25, 0.275, 0.3, 0.325, and 0.35. Finally, we chose σ to be 0.01 on both datasets.

Cutout We tried M including 4, 8, 12, 16, and 20. Finally, we chose M to be 8 on both datasets.

Mixup γ used in Mixup is sampled from a *beta* distribution $\gamma \sim \mathrm{Be}(\alpha, \alpha)$. We tried α including 0.5, 0.1, 0.25, 1, 2, 4, 8, 16, 32, 64, 128, and 256. Finally, we chose α to be 0.5 on both datasets.

Jitter We used the ColorJitter function in Pytorch directly. We tried the parameters corresponding to brightness, contrast, saturation and hue including 0.05, 0.1, 0.2, 0.15, 0.25, 0.3, 0.35, 0.4, 0.45, and 0.5. Finally, we chose 0.05 on both datasets.

Distillation We tried T including 1, 2, 3, 5, and 10. Finally, we chose T to be 3 on both datasets.

B. Membership Inference Attack Results on Models without Using Base

The training and test accuracies and MIA results of all DA models trained without using Base are demonstrated in Tables A1 and A2. Here single query was used because it obtained higher attack success rates than multiple queries, as shown in Figure 4 in the paper. Here *None* stands for models trained without any DA (only the original image data). The test accuracies of models trained without using Base are much lower than that of models trained using Base.

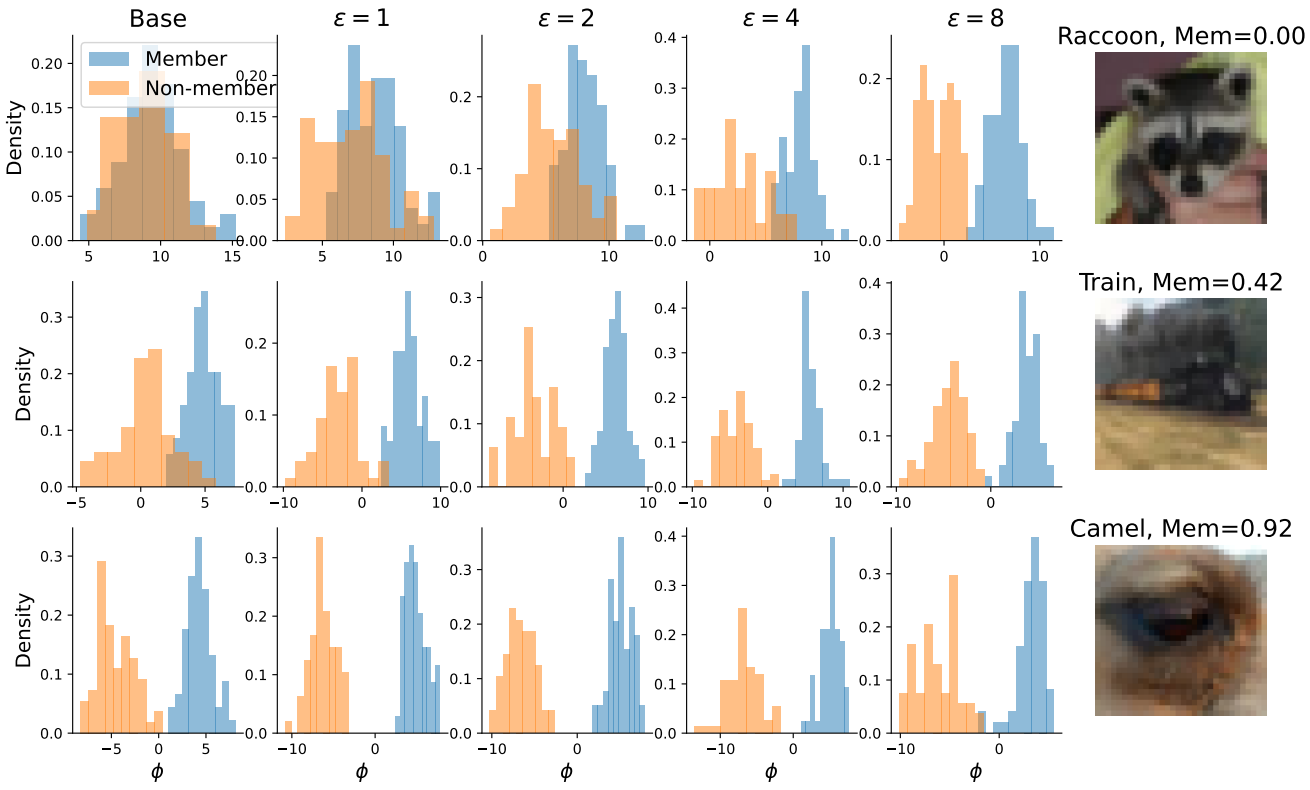


Figure A1. The distributions of normalized confidence ϕ of three samples with different memorization scores using Base and PGD-AT under four different ϵ. Each raw corresponds to a sample.