# When Does Data Augmentation Help With Membership Inference Attacks?

**Yigitcan Kaya** [1] **Tudor Dumitraş** [1]

## Abstract

Deep learning models often raise privacy concerns as they leak information about their training data. This leakage enables membership inference attacks (MIA) that can identify whether a data point was in a model's training set. Research shows that some *data augmentation* mechanisms may reduce the risk by combatting a key factor increasing the leakage, overfitting. While many mechanisms exist, their effectiveness against MIAs and privacy properties have not been studied systematically. Employing two recent MIAs, we explore the lower bound on the risk in the absence of formal upper bounds. First, we evaluate 7 mechanisms and differential privacy, on three image classification tasks. We find that applying augmentation to increase the model's utility does not mitigate the risk and protection comes with a utility penalty. Further, we also investigate why popular label smoothing mechanism consistently amplifies the risk. Finally, we propose *loss-rank-correlation* (LRC) metric to assess how similar the effects of different mechanisms are. This, for example, reveals the similarity of applying high-intensity augmentation against MIAs to simply reducing the training time. Our findings emphasize the utility-privacy trade-off and provide practical guidelines on using augmentation to manage the trade-off.

## 1 Introduction

Deep learning has emerged as one of the cornerstones of large-scale machine learning. However, its fundamental dependence on data forces practitioners to collect and use private information (Shokri & Shmatikov, 2015). This situation has given rise to privacy concerns as deep learning models are shown to leak information about their data (Fredrikson et al., 2015; Yang et al., 2019; Carlini et al., 2018; Shokri et al., 2017). In particular, this enables membership inference attacks (MIAs) to find out whether a data point was in a model's training set or not. MIAs often represent a serious privacy risk; for example, learning that an individual was in a hospital's diagnosis training data also reveals that this individual was a patient there.

MIAs are possible when the adversary can distinguish between a model's predictions on training set and test set samples. Such leakage allows the adversary to infer which samples were used for training the model. Although differential privacy (DP) provides a formal upper bound for this risk, applying it often hurts the model's utility (Abadi et al., 2016), and its guarantees might be too conservative against known attacks (Jayaraman & Evans, 2019). Searching for the root causes of this vulnerability, researchers have proposed *overfitting* as one factor (Yeom et al., 2017).

Overfitting causes a rift between a model's performance on the training samples and its generalization performance on the test samples. On the other hand, data augmentation, i.e., generating new training samples from the existing ones, is known to shrink this *generalization gap* (Szegedy et al., 2016; Zhang et al., 2018; 2016). In consequence, prior results suggest that augmentation, as well as increasing the utility, may also reduce the risk (Shokri et al., 2017; Sablayrolles et al., 2019; Yu et al., 2020). As utility is a key factor for the adoption of security mechanisms in practice, this raises the intriguing prospect of a *free lunch* concerning MIAs in deep learning. The abundance of augmentation methods (Simonyan & Zisserman, 2014; Xie et al., 2016; DeVries & Taylor, 2017) and a lack of work studying whether there really is a free lunch motivate our study.

Our first contribution is to systematically analyze the effectiveness of data augmentation mechanisms against MIAs. Data augmentation conventionally refers to randomly generating new training features, such as cropping, however, we also evaluate techniques that modify the labels, such as label smoothing (Szegedy et al., 2016). Unlike the *theoretical upper bound for leakage* provided by DP, we explore the *lower bound* that three practical MIAs can achieve.

We evaluate 7 popular data augmentation mechanisms using modern convolutional neural networks, on three popular image classification tasks: Fashion-MNIST, CIFAR-10, and CIFAR-100. We find that (i) when augmentation is

[1] University of Maryland, Maryland, USA. Correspondence to: Yigitcan Kaya <cankaya@umiacs.umd.edu>.

*Proceedings of the 38$^{th}$ International Conference on Machine Learning*, PMLR 139, 2021. Copyright 2021 by the author(s).

solely applied for boosting the accuracy, with low-intensity, it fails to achieve substantial protection against MIAs; (ii) high-intensity augmentation, e.g., cropping 90% of an image, hurts the accuracy but it also reduces the risk; (iii) the popular label smoothing mechanism often increases the accuracy and the risk simultaneously. Further, we observe that the models trained with DP can still thwart MIAs even when they provide meaningless privacy guarantees. However, augmentation still seems to be more practical by providing similar protection while causing less accuracy damage. We believe our findings establish a guideline for practitioners on using augmentation against MIAs.

Our second contribution is to investigate our findings on label smoothing (*LS*), which is a popular mechanism in deep learning domains such as vision (Szegedy et al., 2016) and language (Vaswani et al., 2017). We find that *LS* causes models to overfit on smooth labels and leads to more uniform predictions on the training set than on the test set. While this discrepancy gives more leverage to MIAs, why *LS* boosts the accuracy is still under debate (Müller et al., 2019; Meister et al., 2020). Moreover, we show that combining *LS* with another mechanism still results in an amplified risk. These results imply that label smoothing is a hidden risk for practitioners as no other mechanism we evaluate consistently amplifies the risk and the accuracy.

Our third contribution is to design a simple black-box metric, *loss-rank-correlation* (LRC), for studying the similarities between different mechanisms. Building on Spearman's rank correlation coefficient (Spearman, 1904), LRC quantifies the similarity between two models by correlating their losses on the same set of samples. Our experiments suggest that LRC is a reliable tool for capturing similarities. For example, LRC reveals that Mixup (Zhang et al., 2018) and Gaussian augmentation yield significantly different models than other mechanisms. Moreover, LRC also shows that applying high-intensity augmentation to mitigate MIAs resembles reducing the number of iterations (epochs) a model is trained for. This brings the benefits of using augmentation defensively and the prospect of a free lunch into question. Reducing epochs may offer similar protection while being more practical, more computationally efficient.

For reproducibility and future research, we also release our source code at `https://github.com/yigitcankaya/augmentation_mia`.

## 2 Related Work

**Membership Inference Attacks (MIAs).** MIAs have been proposed to exploit the fundamental privacy flaws in deep learning. Shokri et al. (2017); Salem et al. (2018) propose MIAs based on training an inference model to distinguish between a model's predictions on training and test set samples. Yeom et al. (2017) propose a simpler, but equally effective, attack that infers membership by comparing the model's loss on a sample with the average training loss. Following up on Yeom et al.'s work, Sablayrolles et al. (2019) design more advanced attacks that compare with carefully tuned loss thresholds. These studies have also suggested that decreasing overfitting via data augmentation may be effective against MIAs. The lack of comprehensive evaluation on the effectiveness of augmentation, however, prevents it from being considered as a countermeasure. We use these attacks to evaluate various augmentation techniques on modern tasks and investigate their effectiveness.

**Overfitting in Deep Learning.** Arpit et al. (2017) and Zhang et al. (2016) have shown that deep learning models can memorize random data and labels with almost perfect accuracy, which highlights their capability to overfit to their training data. The divergence between the model's performance on its training set and its test set is called the *generalization gap*. Several studies have proposed data augmentation techniques for shrinking this gap and improving generalization, such as label smoothing (Szegedy et al., 2016), Mixup (Zhang et al., 2018), or random cropping (Simonyan & Zisserman, 2014). We investigate the ties between augmentation, generalization, and vulnerability to MIAs.

**Deep Learning with Differential Privacy (DP).** DP offers a formal framework to measure and limit an algorithm's privacy leakage (Dwork, 2008). Building on this framework, Abadi et al. (2016) have proposed the differentially private stochastic gradient descent (DP-SGD) algorithm for deep learning. DP-SGD, by clipping and adding noise to the parameter updates, ensures that the leakage stays within a privacy budget, $\varepsilon$. However, as DP focuses on the worst-case leakage, Jayaraman & Evans (2019) have shown that its guarantees might be too strict against practical MIAs. We use DP as the *gold standard* to compare against as its resilience against MIAs is backed by formal guarantees.

## 3 Experimental Setup

**Datasets.** We use three datasets for evaluation: Fashion-MNIST (Xiao et al., 2017), CIFAR-10 and CIFAR-100 (Krizhevsky et al., 2009). The Fashion-MNIST consists of $28 \times 28$ pixels, gray-scale images of fashion items drawn from 10 classes; containing 60,000 training and 10,000 validation images. The CIFAR-10 and CIFAR-100 consist of $32 \times 32$ pixels, colored natural images drawn from 10 and 100 classes, respectively; containing 50,000 training and 10,000 validation images. We scale the pixel values between 0 and 1 and train our baseline models without any augmentation. These tasks represent different levels of complexity and risk of overfitting; Fashion-MNIST being the easiest and CIFAR-100 being the most difficult as it has a large number of classes and few samples per class.

**Architectures and Hyperparameters.** We experiment with simple variants of VGG (Simonyan & Zisserman, 2014), a prototypical convolutional neural network (CNN) architecture. For Fashion-MNIST, we use 4-layer CNNs and for CIFAR-10 and CIFAR-100 we use 10-layer CNNs. We train our models for 35 epochs using the ADAM optimizer (Reddi et al., 2019). We set the $L_2$ weight decay coefficient to $10^{-6}$ and the batch size to 128. Finally, we repeat our experiments multiple times to compensate for the randomness in training deep learning models.

**Metrics.** To quantify a model's utility, we use its top-1 accuracy on the validation set, $Acc$, as a percentage. Accuracy of an MIA, $Inf$, is the probability that the adversary can guess correctly whether an input is from the training set or not. Following the prior work (Shokri et al., 2017), we apply the MIAs on a total of 10,000 data points—5,000 from the model's training and test sets, each. Note that, on this data set, a random guessing strategy will lead to 50% $Inf$. To quantify the success of an MIA, we use the adversary advantage metric (Yeom et al., 2017) as a percentage, i.e., $Adv = 2 \times (Inf - 50\%)$. Finally, to quantify a mechanism's impact on utility, we measure the relative accuracy drop (RAD) over the baseline model's accuracy.

## 4 Effectiveness of Data Augmentation

**Setting.** We consider the supervised learning setting and the standard feed-forward deep neural network (DNN) structure for classification. A DNN model, a classifier, is a function, $F$, that maps a feature vector $x$, e.g., a natural image, to an output vector $\hat{y}$, i.e., the vector of probabilities for $x$ belonging to each class. The model then classifies $x$ into the most likely class, i.e., $\mathrm{argmax}_i\, \hat{y}^i$. The DNN's parameters are learned on an often private training set, $\mathcal{S}$, containing multiple $(x, y)$ pairs; where $y$ is the ground-truth label of sample $x$. During training, the parameters are updated to minimize the *loss* $\mathcal{L}(\hat{y}, y)$, i.e., a measure of how off $\hat{y}$ is from $y$, on the samples in $\mathcal{S}$. After training, $F$ is evaluated on previously unseen samples in the test set, $\mathcal{D}$. Because of their high capacity, modern DNNs usually *overfit* on $\mathcal{S}$ and have a large *generalization gap*.

**Attacks.** Membership inference attacks (MIAs) aim to answer whether a target sample, $(x_t, y_t)$, was in $\mathcal{S}$ of $F_v$, the victim model, We use two black-box MIAs that rely only on knowing $F_v$'s loss on the target sample, $\mathcal{L}_t = \mathcal{L}(F_v(x_t), y_t)$. These attacks exploit the observation that overfitting causes $F_v$ to produce a distinct loss distribution on $\mathcal{S}$. They first find a *threshold*, $\tau$, then infer the sample's membership if $\mathcal{L}_t < \tau$. We opt for black-box attacks over white-box ones for two reasons: (i) they are more realistic for real-world scenarios and (ii) optimal MIAs only depend on the loss function, thus black-box attacks are comparable to white-box attacks (Sablayrolles et al., 2019).

The standard attack by Yeom et al. (2017) estimates $F_v$'s average loss using $N$ training samples the adversary knows of, i.e., $\tau = \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}(F_v(x_i), y_i)$, where $(x_i, y_i) \in \mathcal{S}$. The more powerful attack by Sablayrolles et al. (2019) estimates $\tau$ that achieves the highest $Adv$ using $N$ samples from $\mathcal{S}$ and $N$ samples from $\mathcal{D}$. We refer to the standard attack's adversary advantage as $Adv_{std}$ and the powerful attack's advantage as $Adv_{pow}$. We set $N = 100$ and, in Appendix, we also evaluate $N = 50$ and $N = 250$ as well. As expected, increasing $N$ leads to stronger, albeit less realistic, attacks; however, it does not change the overall trends we observe. Although augmentation modifies training features or the labels, we assume that the adversary only knows the unmodified $(x_t, y_t)$. In Appendix, we also evaluate *augmentation-aware* attacks, which is shown to be effective against random data augmentation mechanisms (Yu et al., 2020). We observe that these attacks often perform similarly or worse than the powerful attack, especially when applying augmentation with high intensities.

### 4.1 Data Augmentation Mechanisms

We individually apply and evaluate the following mechanisms that, either randomly or deterministically, modify $(x, y) \in \mathcal{S}$ based on their respective hyper-parameter.

**Soft Labels (*SL*) (Hinton et al., 2015)** supply probabilities of a sample's class memberships, e.g., (80% cat, 15% dog, 5% frog), compared to conventional *hard labels* that only indicate a binary membership, e.g., in the cat class. We use distillation from Hinton et al. (2015) to augment the given hard labels as soft labels. We first train a *teacher* model, $F_t$, with no augmentation. We then use the $F_t$'s output vectors, $\hat{y}$, as the soft labels, i.e., $\hat{y}_i = F_t(x_i, \mathcal{T}) | \forall x_i \in \mathcal{S}$. Here, the *temperature* parameter $\mathcal{T}$ determines the flatness of the soft labels; e.g., for $\mathcal{T}$=1, (95% cat, 4% dog, 1% frog); for $\mathcal{T}$=10, (55% cat, 35% dog, 10% frog). Shokri et al. (2017) suggest that soft labels may reduce the MIA risk. We train augmented models with $1 \leq \mathcal{T} \leq 1000$.

**Label Smoothing (*LS*) (Szegedy et al., 2016)** also augments hard labels by attaching probability estimates. However, unlike soft labels, it simply assigns uniform probabilities to other classes, e.g., (80% cat, 10% dog, 10% frog). This popular mechanism improves performance on many tasks (Müller et al., 2019). The parameter $\alpha$ controls the smoothing intensity, e.g., for $\alpha$=0.3, (80% cat, 10% dog, 10% frog); for $\alpha$=0.6, (60% cat, 20% dog, 20% frog). We train augmented models with $0.01 \leq \alpha \leq 0.995$

**DisturbLabel (*DL*) (Xie et al., 2016)** randomly replaces a portion of labels in the training set with incorrect values in each training iteration. The authors argue that this mechanism prevents overfitting by implicitly averaging over exponentially many models that are trained with different label sets. Here, the ratio of the labels replaced is controlled by the parameter $\theta$, e.g., for $\theta$=0.5, 50% of the training labels in an iteration will be wrong. We train augmented models with $0.01 \leq \theta \leq 0.99$

**Random Cropping (*RC*) (Simonyan & Zisserman, 2014)** takes an $x \in \mathcal{S}$ and pads it with $\mathcal{P}$ zeros on each end, e.g., a $32 \times 32$ pixel image becomes $(32+2\mathcal{P}) \times (32+2\mathcal{P})$. Then, depending on the model's input size, a random portion of the padded input is cropped out and used as the training sample, at each iteration. The parameter $\mathcal{P}$ controls how much of the original $x$ is kept in the augmented $x$, on average, e.g., for a 32 × 32 image, $\mathcal{P}$=10 crops out ∼30%, and $\mathcal{P}$=36 crops out ∼17%. Especially for visual tasks, random cropping has become a standard by leading to better features and, therefore, increased performance. We train augmented models with $1 \leq \mathcal{P} \leq 38$.

**Cutout (*CO*) (DeVries & Taylor, 2017)** occludes random regions of size $\mathcal{M} \times \mathcal{M}$ of the training images, at each iteration. The authors argue that this improves the robustness and reduces overfitting. The parameter $\mathcal{M}$ controls the size of the masked out region, on average, e.g., for a $32 \times 32$ image, $\mathcal{M}$=20 occludes ∼28%, and $\mathcal{M}$=50 occludes ∼93%. We train augmented models with $4 \leq \mathcal{M} \leq 52$.

**Gaussian Augmentation (*GA*) (Cohen et al., 2019)** simply adds noise drawn from $\mathcal{N}(0, \sigma^2 I)$ to each $x$, at each training iteration. Here, $\sigma^2$, the variance, controls the intensity of the added noise. The authors use Gaussian augmentation to improve robustness against adversarial noise. We train augmented models with $0.01 \leq \sigma^2 \leq 0.35$.

**Mixup (*MU*) (Zhang et al., 2018)** trains a model on the convex combination of randomly selected sample pairs, i.e., $\tilde{x} = \lambda x_i + (1 - \lambda)x_j$ and $\tilde{y} = \lambda y_i + (1 - \lambda)y_j$ where $(x_i, y_i), (x_j, y_j) \in \mathcal{S}$. Here, $\lambda \in [0, 1]$ is a random variable drawn from Beta$(\gamma, \gamma)$. The parameter $\gamma$ controls the interpolation between two samples, as $\gamma \to 0$ then $\lambda = 0 \vee \lambda = 1$, i.e., no interpolation; and as $\gamma \to \infty$ then $\lambda \to 0.5$, i.e., simple averaging. The authors show that Mixup improves the generalization in deep learning, reduces the memorization of corrupt labels, and increases robustness. We train augmented models with $0.1 \leq \gamma < 256$.

### 4.2 Higher Accuracy Does Not Mitigate the MIA Risk

In this section, we apply augmentation to boost a model's accuracy and to shrink the generalization gap. All baselines models achieve ∼100% training accuracy, therefore, the gains on the testing accuracy means a smaller generalization gap. For each mechanism, we find the hyper-parameter that achieves the highest testing accuracy. Table 1 presents the accuracies ($Acc$) of these augmented models and the success rates of the MIAs ($Adv$) against them. First, MIAs are less threatening on the simpler task, Fashion-MNIST, as models already achieve high $Acc$. This hints that the risk is even greater for large-scale modern tasks and further motivates our study.

Further, we observe that augmentation generally increases the $Acc$ over the baseline models up to 10%. However, no mechanism is able to reduce $Adv_{pow}$ by more than 50%. As a result, applying augmentation for $Acc$ still leaves the models vulnerable to MIAs. Moreover, for baseline models $Adv_{std}$ and $Adv_{pow}$ are similar; however, against augmented models, $Adv_{pow}$ is often higher than $Adv_{std}$. This indicates that using weaker MIAs gives a false sense of security and also explains that why prior work, such as (Shokri et al., 2017), finds that mechanisms such as *SL* might reduce the risk.

Finally, we see that *LS* amplifies the risk: on CIFAR-100 it boosts the $Acc$ by 10% but also causes 25% higher $Adv_{pow}$. In the cases where *LS* increases the $Adv$, the models have ∼100% training accuracy and a higher testing accuracy than the baseline models. This finding that a smaller generalization gap not translating to a lower risk suggests that overfitting might be a sufficient but not a necessary condition for MIAs. In Section 5, we further investigate this behavior of *LS*.

Table 1: **Applying data augmentation for accuracy.** In each cell, the first row presents the model's $Acc$ and the second row presents $Adv_{std}$ (*left*) and $Adv_{pow}$ (*right*). The segment indicated by ∅ contains the unaugmented baseline models. We highlight when a mechanism amplifies $Adv$.

| MECH. | FMNIST | CIFAR10 | CIFAR100 |
|---|---|---|---|
| ∅ | 93.5%<br>8.6 / 8.5 | 85.5%<br>27.7 / 28.1 | 54.3%<br>58.2 / 60.2 |
| *SL* | 93.8%<br>4.9 / 8.3 | 86.5%<br>16.8 / 24.4 | 57.6%<br>20.8 / 39.8 |
| *LS* | 94.1%<br>**10.0 / 17.9** | 86.3%<br>13.3 / 27.7 | 59.8%<br>**61.4 / 75.6** |
| *DL* | 94.0%<br>6.7 / 6.5 | 86.6%<br>20.5 / 25.5 | 57.0%<br>49.7 / **64.0** |
| *RC* | 94.2%<br>4.7 / 4.5 | 88.7%<br>18.4 / 18.4 | 59.7%<br>32.7 / 32.0 |
| *CO* | 94.1%<br>8.3 / 8.3 | 87.7%<br>25.6 / 26.1 | 59.1%<br>34.9 / 33.9 |
| *GA* | 93.5%<br>**8.7 / 9.9** | 85.1%<br>27.7 / **28.6** | 54.7%<br>**58.8 / 62.5** |
| *MU* | 94.2%<br>6.9 / 8.7 | 86.9%<br>16.1 / 23.7 | 57.8%<br>45.1 / 57.0 |

### 4.3 Applying Augmentation to Reduce the MIA Risk

The previous section shows that low-intensity augmentation to boost $Acc$ still leaves the models vulnerable. In this section, we evaluate how high-intensity augmentation, i.e., hyper-parameters are tuned higher, fares against MIAs. As this might hurt $Acc$, we evaluate our results under two utility scenarios: <10% and <25% relative accuracy drop (RAD) over the baseline models. In these scenarios, we select the hyper-parameters that achieve the smallest $\max(Adv_{std}, Adv_{pow})$, within the respective RAD limit. We focus on limiting the utility drop as this is a top concern for most practitioners for applying security measures.

Table 2 presents the results of applying augmentation defensively against MIAs. First, between two RAD settings, we see an overall drop in $Adv_{std}$ and $Adv_{pow}$, which suggests that mitigating MIAs requires high-intensity augmentation and comes with a utility penalty. Further, defeating MIAs is significantly easier on Fashion-MNIST than on more complex tasks. In RAD<10% setting, MIAs have negligible success against Fashion-MNIST models; whereas against most CIFAR-100 models, they still have moderate success. In RAD<25% setting, on the other hand, mechanisms such as *RC*, *DL*, and *CO* reduce $Adv_{pow}$ by more than 80% over the baseline. Moreover, there is only a minor difference between $Adv_{std}$ and $Adv_{pow}$, suggesting that the mechanisms provide real benefits in these settings. Overall, our findings highlight that augmentation, on complex tasks, cannot provide a *free lunch* of defeating MIAs while preserving $Acc$.

Turning our attention to individual mechanisms, we see *RC* and *CO* standing out as the most effective mechanisms. Even in RAD<10% setting, these mechanisms reduce $Adv$ by 85%-100% when tuned to randomly keep only ~50% of the original $x_i \in \mathcal{S}$, in each training iteration. As a result, it takes more than ~7 training iterations for the model to see the whole $x_i$. Similarly, when $\theta$>0.5, *DL* is also effective, especially in RAD<25% setting, i.e., only in less than 50% of the iterations a sample's label is not corrupted. This leads us to hypothesize that such disruptive augmentations are similar to reducing the number of training iterations, which we investigate in Section 6.

In Figure 1, we show the effect of increasing the intensity of *RC* on the losses a model produces on training and testing samples. Low-intensity augmentation (*left*) results in two distinct loss distributions, which is easily exploited by black-box MIAs for inferring the membership of samples. However, increasing the intensity (*middle* and *right*) brings two distributions closer and, consequently, leaves less leverage to MIAs for inference.

Finally, we see that *MU* and *SL* are moderately successful. *MU* reduces $Adv$ 55-85% when it is tuned to almost maximum intensity ($\gamma \in \{128, 256\}$). *SL* reduces $Adv$ 65-100%, with very high temperature values ($\mathcal{T}$>100) that prior work has not experimented with. As the least effective mechanisms against MIAs, we identify *GA*, which hurts $Acc$, and *LS*, which has a limited impact on $Acc$.

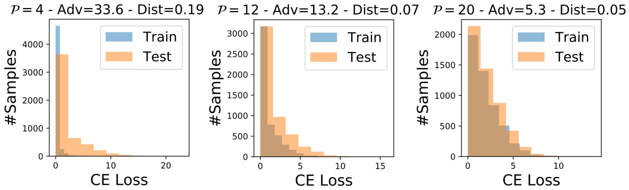

Figure 1: **The effect of increasing the cropping augmentation intensity ($\mathcal{P}$) on $\mathcal{S}$ and $\mathcal{D}$ loss distributions.** *Dist* refers to Jensen-Shannon distance between training and testing histograms represented as probability densities. All models are trained on CIFAR-100.

**Comparison With Differential Privacy (DP).** By providing information-theoretic privacy guarantees, DP has become a de-facto privacy standard. As the gold standard, we evaluate models trained using Differentially-Private Stochastic Gradient Descent (DP-SGD) by Abadi et al. (2016). Essentially, DP-SGD clips and adds noise to the gradients computed on $\mathcal{S}$ during training to limit the influence of a single sample on the model. The variance of this noise determines the *privacy budget*, $\varepsilon$, which defines an upper bound on the leakage. In general, decreasing $\varepsilon$ increases privacy, and, although there is no consensus, $\varepsilon$<1 is usually acceptable for privacy (Jayaraman & Evans, 2019).

We present the results for DP in Table 3 under different privacy budgets. We see even formally meaningless budgets ($\varepsilon$>100) reduce $Adv$ by 65-90% and decreasing $\varepsilon$ reduces $Adv$ further, as expected. This aligns with the findings of Jayaraman & Evans (2019) that thwarting MIAs might not require strong formal guarantees. For $\varepsilon$<1, we see negligible $Adv$ success; however, the $Acc$ penalty for CIFAR-100 is more than 85%. Overall, until a breakthrough in research on applying DP to complex tasks, augmentation might be more practical against known MIAs by causing less $Acc$ penalty in exchange for reducing $Adv$.

## 5 Why Label Smoothing Amplifies the Risk

The previous section shows that label smoothing (*LS*) can simultaneously increase $Acc$ and $Adv_{pow}$. In Figure 2, we present the trend between $Acc$ and $Adv_{pow}$ for models trained with *LS* on CIFAR-100. We see that *LS* can boost $Acc$ by up to 10% over the baseline; however, it can also cause up to 40% increase in $Adv_{pow}$. This adverse effect makes *LS* a hidden privacy risk for practitioners who use it to increase model utility. In this section, we aim to shed light on this phenomenon and how *LS* affects a model.

Table 2: **Applying augmentation against MIAs in RAD $< 10\%$ and $< 25\%$ settings.** (Same format as Table 1))

| MECH. | FMNIST | CIFAR10 | CIFAR100 |
|---|---|---|---|
| $\emptyset$ | 93.5%<br>8.6 / 8.5 | 85.5%<br>27.7 / 28.1 | 54.3%<br>58.2 / 60.2 |
| *SL (10)* | 88.8%<br>0.0 / 0.0 | 77.4%<br>5.5 / 8.1 | 49.7%<br>14.1 / 21.1 |
| *SL (25)* | 88.8%<br>0.0 / 0.0 | 65.6%<br>2.3 / 2.2 | 46.2%<br>10.7 / 12.9 |
| *LS (10)* | 84.6%<br>0.0 / 0.8 | 77.8%<br>12.5 / 13.6 | 55.1%<br>44.1 / 45.0 |
| *LS (25)* | 84.6%<br>0.0 / 0.8 | 77.8%<br>12.5 / 13.6 | 55.1%<br>44.1 / 45.0 |
| *DL (10)* | 90.3%<br>0.7 / 0.7 | 78.1%<br>4.7 / 4.2 | 52.2%<br>28.2 / 32.0 |
| *DL (25)* | 90.3%<br>0.7 / 0.7 | 78.1%<br>4.7 / 4.2 | 42.4%<br>9.1 / 7.0 |
| *RC (10)* | 86.1%<br>0.0 / 0.0 | 78.4%<br>3.2 / 3.1 | 52.8%<br>8.9 / 7.1 |
| *RC (25)* | 86.1%<br>0.0 / 0.0 | 67.3%<br>1.2 / 0.6 | 42.6%<br>4.1 / 2.9 |
| *CO (10)* | 84.6%<br>0.4 / 0.5 | 80.4%<br>6.0 / 5.1 | 49.3%<br>8.7 / 8.2 |
| *CO (25)* | 84.6%<br>0.4 / 0.5 | 66.5%<br>2.1 / 2.1 | 45.8%<br>7.5 / 5.0 |
| *GA (10)* | 89.9%<br>2.2 / 2.2 | 78.9%<br>24.1 / 26.4 | 48.9%<br>50.6 / 49.9 |
| *GA (25)* | 89.9%<br>2.2 / 2.2 | 75.8%<br>23.2 / 24.2 | 41.2%<br>44.2 / 42.3 |
| *MU (10)* | 92.4%<br>3.4 / 3.4 | 84.2%<br>11.9 / 9.7 | 49.8%<br>14.1 / 13.7 |
| *MU (25)* | 92.4%<br>3.4 / 3.4 | 84.2%<br>11.9 / 9.7 | 48.4%<br>13.1 / 11.9 |

Table 3: **DP-SGD against MIAs.** (Same format as Table 1)

| BUDGET | FMNIST | CIFAR10 | CIFAR100 |
|---|---|---|---|
| $\varepsilon \in (100, 1000]$ | 90.0%<br>2.7 / 2.9 | 69.2%<br>5.1 / 5.4 | 32.6%<br>6.0 / 5.1 |
| $\varepsilon \in (10, 100]$ | 88.7%<br>1.6 / 1.6 | 64.0%<br>2.2 / 2.7 | 24.0%<br>3.9 / 2.9 |
| $\varepsilon \in [10, 1)$ | 87.1%<br>0.0 / 0.0 | 55.6%<br>1.6 / 1.6 | 16.1%<br>14.2 / 11.2 |
| $\varepsilon \in (0, 1]$ | 83.2%<br>0.3 / 0.0 | 43.3%<br>0.2 / 0.0 | 6.8%<br>1.3 / 1.1 |

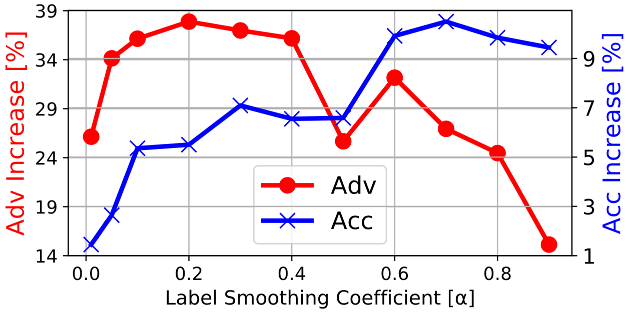

Figure 2: **The effect of *LS* on $Acc$ and $Adv_{pow}$.** The numbers are relative to the CIFAR-100 baseline in Table 1.

***LS* Leads to Smoother Predictions on $\mathcal{S}$ Than on $\mathcal{D}$.** Given a training sample, *LS* pushes the model to output smooth, uniform, probabilities for the classes other than the one the sample belongs to. For example, on CIFAR-100 with 100 classes, when $\alpha$=0.1, *LS* assigns 0.001 probability to each other class during training. In Figure 3, we visualize how *LS* statistically changes the *non-maximum* prediction probabilities on CIFAR-100. Given $\hat{y}$, the prediction vector, the non-maximum probabilities are $\{\hat{y}^i | i \in \mathcal{K} \setminus \{\text{argmax}\,\hat{y}\}\}$, where $\mathcal{K}$ is the set of all classes.

The top plot shows that the average non-maximum probability exactly follows $\alpha$ on $\mathcal{S}$, which is not the case on $\mathcal{D}$. Further, because *LS* assigns uniform probabilities, it also forces the standard deviation of the non-maximum probabilities to be zero. The bottom plot shows that, regardless of $\alpha$, this is the case on $\mathcal{S}$ and, again, not the case on $\mathcal{D}$. Our findings show that *LS* causes the models to overfit on smooth labels and leads to discernible statistics on $\mathcal{S}$. This gives MIAs more leverage to infer whether $(x_t, y_t) \in \mathcal{S}$ and, therefore, amplifies the risk.

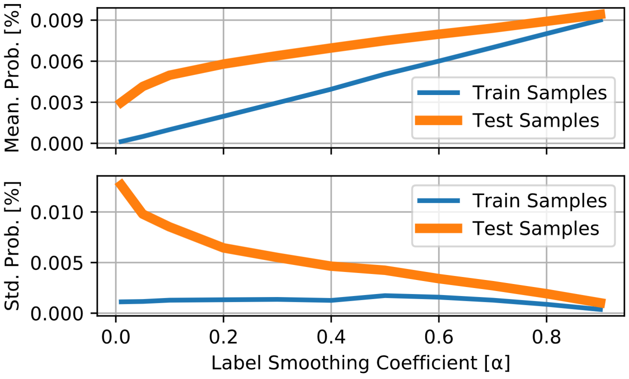

Figure 3: **The effect of *LS* on output smoothness.**

***LS* Erases Less Information on $\mathcal{D}$ Than on $\mathcal{S}$.** Knowledge distillation (Hinton et al., 2015) relies on the information a teacher model encodes into its non-maximum predictions. However, by forcing them to be uniform, *LS* hinders a model's ability to be distilled (Müller et al., 2019). Prior work calls this *information erasure* and shows that student models perform worse when they are distilled from teachers trained with $LS$. The previous section shows that $LS$ causes less uniform non-maximum predictions on $\mathcal{D}$ than on $\mathcal{S}$. As a result, we conjecture that information erasure is also less severe on $\mathcal{D}$ than on $\mathcal{S}$.

To test our hypothesis, we follow the experiment from Müller et al. with one addition: we distill the teachers trained with $LS$ using the samples outside their $\mathcal{S}$. We first randomly split the original $\mathcal{S}$ of CIFAR-100 into two equal parts and train teachers with $LS$ on the second part. We then use either the samples in the first part (unseen) or the second part (seen) to distill a teacher into smaller student models. To compensate for the randomness, we repeat this multiple times and report the average results.

In Figure 4, we plot the average $Acc$ differences between the first and the second students for different teacher $\alpha$ and distillation $\mathcal{T}$ values. Across the board, we see that the first students achieve up to $15\%$ higher $Acc$ than second students. This confirms our hypothesis and also provides more insight into why $LS$ increases the MIA success, i.e., it erases more information on $\mathcal{S}$ than on $\mathcal{D}$.

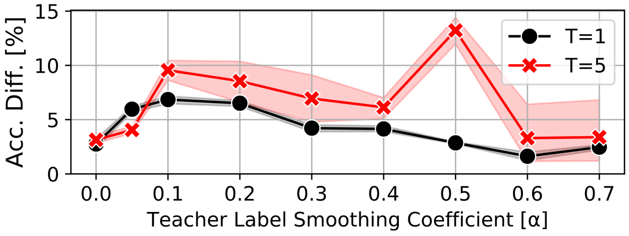

Figure 4: **The effect of *LS* on distillation performance.**

**Using *LS* Together With Other Mechanisms.** In this work, we evaluate each augmentation mechanism in isolation to study their individual effects. However, applying them in conjunction is often beneficial for boosting the accuracy (He et al., 2019). On the other hand, we find that $LS$ amplifies the MIA risk; whereas, mechanisms such as $RC$ mitigate it. As a result, a natural question to ask is whether using them together would amplify or mitigate the risk.

Table 4 presents the results of combining $LS$ with $RC$ to either maximize $Acc$ (*second* segment) or to minimize $Adv$ (*third* segment). Here, we use the same $\mathcal{P}$ values as in Tables 1 and 2. First, we see that combining can further boost $Acc$ by up to $8\%$, over only using $RC$. However, these combinations also inadvertently amplify $Adv$ by up to $45\%$. On the other hand, we see that combining to minimize $Adv$ fails to provide any major benefits. These combinations have more or less the same $Acc$ and $Adv$ as only using $RC$. Our results imply that it might not be feasible to combine $LS$ with other mechanisms for avoiding its privacy risks while still reaping its utility benefits.

Table 4: **Training with both *LS* and *RC* on CIFAR-100.**

| COMBINATION | $Acc$ | $Adv_{std}$ | $Adv_{pow}$ |
|---|---|---|---|
| Baselines (Only $RC$) | | | |
| $\mathcal{P} = 4, \alpha = 0.00$ | $59.7\%$ | $32.7\%$ | $32.0\%$ |
| $\mathcal{P} = 14, \alpha = 0.00$ | $52.8\%$ | $8.9\%$ | $7.1\%$ |
| $\mathcal{P} = 20, \alpha = 0.00$ | $42.5\%$ | $4.1\%$ | $2.9\%$ |
| Combine to Maximize $Acc$ | | | |
| $\mathcal{P} = 4, \alpha = 0.90$ | $64.7\%$ | **$37.6\%$** | **$38.4\%$** |
| $\mathcal{P} = 14, \alpha = 0.70$ | $56.4\%$ | **$9.3\%$** | **$10.4\%$** |
| $\mathcal{P} = 20, \alpha = 0.60$ | $45.2\%$ | $2.8\%$ | **$4.3\%$** |
| Combine to Minimize $Adv$ | | | |
| $\mathcal{P} = 4, \alpha = 0.01$ | $60.2\%$ | **$33.5\%$** | **$34.8\%$** |
| $\mathcal{P} = 14, \alpha = 0.10$ | $53.9\%$ | $6.6\%$ | **$9.8\%$** |
| $\mathcal{P} = 20, \alpha = 0.90$ | $39.9\%$ | $1.9\%$ | **$3.4\%$** |

**Impact of *LS* on Non-Image Tasks.** Our experiments focus on image classification as data augmentation is ubiquitous in this domain. Here, we ask whether $LS$ amplifies the MIA risk on non-image tasks too. Specifically, we evaluate Purchase-100 and Texas-100 tasks[1], which are common benchmarks in the MIA literature (Jayaraman & Evans, 2019; Shokri et al., 2017). Table 5 shows that, on Texas-100, $LS$ has its detrimental effect as it amplifies both $Acc$ and $Adv$ simultaneously. On the other hand, on Purchase-100, we cannot conclusively claim $LS$ amplifies the MIA risk as it causes a drop in $Acc$. The higher $Adv$ in this task might also stem from a larger generalization gap. Overall, these experiments show that $LS$ can be a privacy risk on other domains as well, depending on the task.

Table 5: **Impact of *LS* on non-image tasks.** All models are trained till they reach $\sim 100\%$ training accuracy on a basic fully connected architecture.

| $\alpha$ | 0.00 | 0.01 | 0.05 | 0.10 | 0.20 | 0.30 |
|---|---|---|---|---|---|---|
| Purchase-100 | | | | | | |
| $Adv$ | $14.6\%$ | $17.4\%$ | $19.6\%$ | $20.5\%$ | $18.9\%$ | $18.4\%$ |
| $Acc$ | $89.5\%$ | $87.8\%$ | $86.4\%$ | $84.9\%$ | $84.9\%$ | $83.8\%$ |
| Texas-100 | | | | | | |
| $Adv$ | $55.0\%$ | $58.4\%$ | $63.2\%$ | $64.6\%$ | $64.6\%$ | $63.5\%$ |
| $Acc$ | $59.1\%$ | $60.2\%$ | $61.3\%$ | $62.0\%$ | $62.4\%$ | $62.1\%$ |

## 6 Measuring the Mechanism Similarities

In Section 4, we evaluated 7 different mechanisms against MIAs. Although they augment $\mathcal{S}$ in different ways, we ob-

[1] https://github.com/privacytrustlab/datasets

served that mechanisms such as *RC*, *CO*, or *DL* have a similar impact on $Acc$ and $Adv$. In this section, we aim to develop a simple black-box metric to quantify how similar the overfitting patterns of the two models are. We then use this metric to investigate whether different augmentation mechanisms change the models in distinct ways.

**Loss-Rank-Correlation (LRC) Metric.** In Figure 5, we present how augmentation affects a model's cross-entropy (CE) loss distribution on 5,000 samples from $\mathcal{S}$. The model trained with no augmentation produces $\sim 0$ loss on all samples, which indicates overfitting. Both *RC* and *CO*, on the other hand, prevent the model from producing $\sim 0$ loss on approximately 1,250 samples. This observation leads us to ask whether these mechanisms prevent overfitting on different sets of samples. Preventing overfitting on different sets of samples would imply that these mechanisms affect models in distinct ways.

To answer this question, we design the *loss-rank-correlation* (LRC) metric. To our best knowledge, there is not an established efficient and model agnostic metric to quantify model similarity. We aim to fill this gap using LRC. Further, by only requiring query access, LRC is suitable for reasoning about practical MIAs.

To compute the LRC between two models trained on the same $\mathcal{S}$, we first draw a set of 5,000 samples from $\mathcal{S}$. We then compute Spearman's rank correlation coefficient (Spearman, 1904) between the loss values of two models on this set, which gives us the LRC score. We opt for ranking correlation as it takes the order of elements into account. For example, if two models have similar overfitting patterns and produce $\sim$ 0 loss on the same samples, then the LRC between them will be close to one. Note that a model's LRC score with itself is always one.

LRC aims to answer whether different augmentation mechanisms have distinct effects on the model.

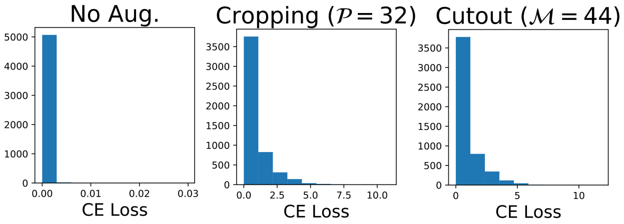

Figure 5: **The effect of augmentation on loss distribution for $x \in \mathcal{S}$.** All models are trained on CIFAR-10.

**LRC Captures Model Similarity.** In this section, we measure the LRC scores between CIFAR-10 models trained with different *RC* intensities, controlled by hyperparameter $\mathcal{P}$. Intuitively, when the intensities are closer, we would expect the resulting models to be more similar. The scores in Figure 6 reflect this intuition, e.g., LRC is greater between $\mathcal{P}$=14 and $\mathcal{P}$=20 (0.83) than between $\mathcal{P}$=14 and $\mathcal{P}$=26 (0.71). Because we train multiple models for each $\mathcal{P}$ and report the averages, the scores on the main diagonal are less than one. This experiment supports that LRC is a reliable tool to capture how similar the two models are.

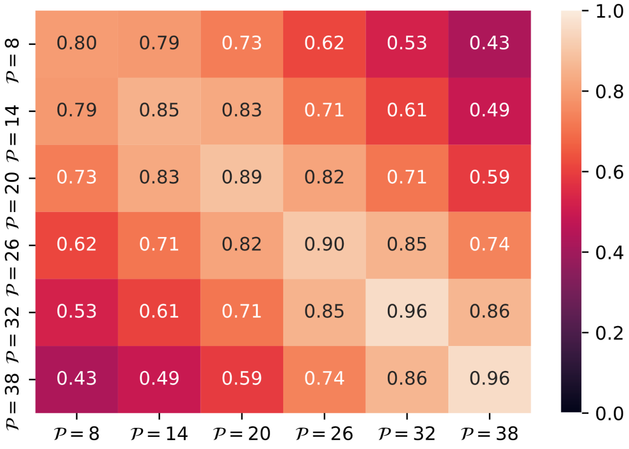

Figure 6: **LRC scores between models trained with *RC*.**

**Identifying Similar Mechanisms.** In Figure 7, we present the LRC scores between CIFAR-10 models trained with different mechanisms, and differential privacy. We select the models in RAD<10% settings where we apply high-intensity augmentation to mitigate MIAs. First, we see that *GA* and *MU* result in models significantly dissimilar from the rest with lower LRC scores across the board. Turning our attention to similar mechanisms, we see $\{$*SL*, *LS*$\}$ and $\{$*RC*, *CO*$\}$ pairs standing out. This is expected considering that these pairs augment the training samples similarly, i.e., changing the hard labels or removing random portions of samples. This also further highlights the ability of LRC to capture similarities. We believe quantifying the similarities between mechanisms and finding diverse combinations for combatting overfitting is an important research direction.

**High-Intensity Augmentation Behaves Like Short Training.** In the initial training epochs, deep learning models learn simple, almost linear, classifiers that are less prone to overfitting (Nakkiran et al., 2019). However, as the iterations progress, the model's complexity goes up as well as its tendency to overfit. To evaluate how *simple* models fare against MIAs, we train CIFAR-10 models for fewer, i.e., $\mathcal{I}\in\{3, 4, 7\}$, epochs. Compared to the baseline model, i.e., $\mathcal{I}$=35, these models have 17%, 9%, 5% lower $Acc$, respectively. However, they are also much more resilient to MIAs with 90%, 85%, 65% less $Adv$. Our results show that high-intensity augmentation that defeats MIAs also harms $Acc$, similar to reducing $\mathcal{I}$. As a result, we hypothesize that the models resulting from high-intensity augmentation might be similar to the models trained for fewer epochs.

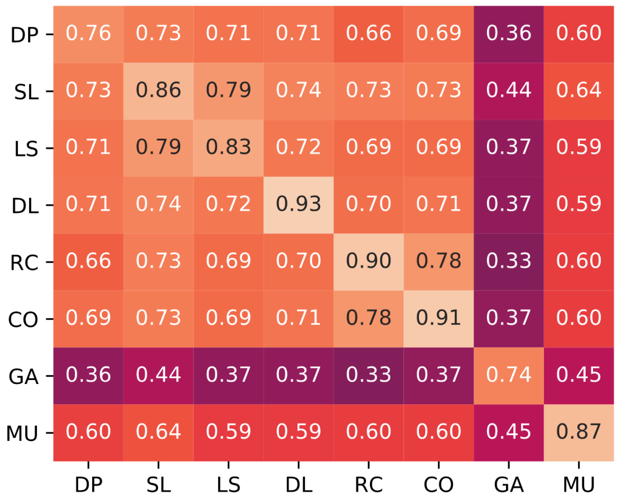

Figure 7: **LRC scores between different mechanisms.**

In Figure 8, we present the LRC scores between the simple models and the models with increasing augmentation intensities. For example, the LRC score between $\mathcal{P}$=16 and $\mathcal{I}$=7 is 0.79; whereas between $\mathcal{P}$=16 and $\mathcal{I}$=3 is it 0.64. On the other hand, the LRC score between $\mathcal{P}$=32 and $\mathcal{I}$=7 is 0.70; whereas between $\mathcal{P} = 32$ and $\mathcal{I}$=3 is it 0.79. This suggests that increasing the intensity resembles training for fewer epochs and vice versa. Further, the high LRC scores across the board further supports our hypothesis that high-intensity augmentation is similar to short training. Overall, these findings pose an intriguing dilemma: low-intensity augmentation boosts the accuracy but leaves the models vulnerable; whereas, high-intensity augmentation hurts the accuracy and alleviates MIAs but it might not be different than simply training the models for fewer epochs.

## 7 Conclusions

We conduct a systematical study on the effectiveness of data augmentation in mitigating membership inference attacks (MIAs) against deep learning models. We first find that applying augmentation only for accuracy cannot defeat MIAs and applying for mitigation cannot avoid hurting the accuracy. We then identify the effective, e.g., random cropping, and ineffective, e.g., Gaussian augmentation, mechanisms and propose practical guidelines. Further, we reveal and investigate the tendency of label smoothing mechanism to improve both the generalization and the attack performance. Finally, we propose a simple methodology to quantify how similar two models are, which shows that augmentation that mitigates MIAs might not be different than simply training for fewer iterations. We hope that our work will be as a bridge between the practical and formal solutions to the privacy leakage problem.

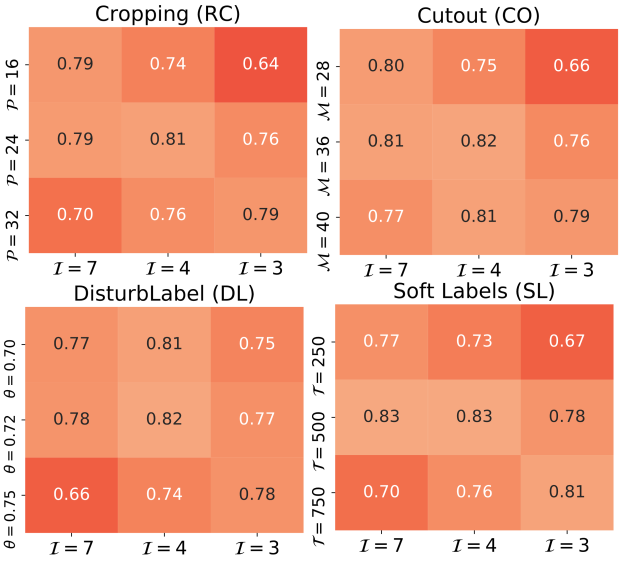

Figure 8: **Comparing augmentation with short training.**

## 8 Acknowledgements

This research was partially supported by the Department of Defense and by the Intelligence Advanced Research Projects Agency (IARPA). The content of this paper does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

## References

Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., and Zhang, L. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 308–318. ACM, 2016.

Arpit, D., Jastrzębski, S., Ballas, N., Krueger, D., Bengio, E., Kanwal, M. S., Maharaj, T., Fischer, A., Courville, A., Bengio, Y., et al. A closer look at memorization in deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 233–242. JMLR. org, 2017.

Carlini, N., Liu, C., Kos, J., Erlingsson, Ú., and Song, D. The secret sharer: Measuring unintended neural network memorization & extracting secrets. *arXiv preprint arXiv:1802.08232*, 2018.

Cohen, J., Rosenfeld, E., and Kolter, Z. Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning*, pp. 1310–1320. PMLR, 2019.

DeVries, T. and Taylor, G. W. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

Dwork, C. Differential privacy: A survey of results. In *International conference on theory and applications of models of computation*, pp. 1–19. Springer, 2008.

Fredrikson, M., Jha, S., and Ristenpart, T. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pp. 1322–1333. ACM, 2015.

He, T., Zhang, Z., Zhang, H., Zhang, Z., Xie, J., and Li, M. Bag of tricks for image classification with convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 558–567, 2019.

Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Jayaraman, B. and Evans, D. Evaluating differentially private machine learning in practice. In *28th USENIX Security Symposium (USENIX Security 19). Santa Clara, CA: USENIX Association*, 2019.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

Meister, C., Salesky, E., and Cotterell, R. Generalized entropy regularization or: There's nothing special about label smoothing. In Jurafsky, D., Chai, J., Schluter, N., and Tetreault, J. R. (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pp. 6870–6886. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.acl-main.615. URL `https://doi.org/10.18653/v1/2020.acl-main.615`.

Müller, R., Kornblith, S., and Hinton, G. E. When does label smoothing help? In *Advances in Neural Information Processing Systems*, pp. 4696–4705, 2019.

Nakkiran, P., Kaplun, G., Kalimeris, D., Yang, T., Edelman, B. L., Zhang, F., and Barak, B. Sgd on neural networks learns functions of increasing complexity. *arXiv preprint arXiv:1905.11604*, 2019.

Reddi, S. J., Kale, S., and Kumar, S. On the convergence of adam and beyond. *arXiv preprint arXiv:1904.09237*, 2019.

Sablayrolles, A., Douze, M., Schmid, C., Ollivier, Y., and Jégou, H. White-box vs black-box: Bayes optimal strategies for membership inference. In *International Conference on Machine Learning*, pp. 5558–5567. PMLR, 2019.

Salem, A., Zhang, Y., Humbert, M., Berrang, P., Fritz, M., and Backes, M. Ml-leaks: Model and data independent membership inference attacks and defenses on machine learning models. *arXiv preprint arXiv:1806.01246*, 2018.

Shokri, R. and Shmatikov, V. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, pp. 1310–1321. ACM, 2015.

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.

Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Spearman, C. The proof and measurement of association between two things. *The American Journal of Psychology*, 15(1):72–101, 1904. ISSN 00029556. URL `http://www.jstor.org/stable/1412159`.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30, pp. 5998–6008. Curran Associates, Inc., 2017. URL `https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf`.

Xiao, H., Rasul, K., and Vollgraf, R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Xie, L., Wang, J., Wei, Z., Wang, M., and Tian, Q. Disturblabel: Regularizing cnn on the loss layer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4753–4762, 2016.

Yang, Z., Chang, E.-C., and Liang, Z. Adversarial neural network inversion via auxiliary knowledge alignment. *arXiv preprint arXiv:1902.08552*, 2019.

Yeom, S., Fredrikson, M., and Jha, S. The unintended consequences of overfitting: Training data inference attacks. *arXiv preprint arXiv:1709.01604*, 2017.

Yu, D., Zhang, H., Chen, W., Yin, J., and Liu, T.-Y. How does data augmentation affect privacy in machine learning?, 2020.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.

Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018. URL `https://openreview.net/forum?id=r1Ddp1-Rb`.