

Privacy Risks of Securing Machine Learning Models against Adversarial Examples

Liwei Song
liweis@princeton.edu
Princeton University

Reza Shokri
reza@comp.nus.edu.sg
National University of Singapore

Prateek Mittal
pmittal@princeton.edu
Princeton University

ABSTRACT

The arms race between attacks and defenses for machine learning models has come to a forefront in recent years, in both the security community and the privacy community. However, one big limitation of previous research is that the security domain and the privacy domain have typically been considered *separately*. It is thus unclear whether the defense methods in one domain will have any unexpected impact on the other domain.

In this paper, we take a step towards resolving this limitation by combining the two domains. In particular, we measure *the success of membership inference attacks against six state-of-the-art adversarial defense methods that mitigate adversarial examples (i.e., evasion attacks)*. Membership inference attacks aim to infer an individual’s participation in the target model’s training set and are known to be correlated with target model’s overfitting and sensitivity with regard to training data. Meanwhile, adversarial defense methods aim to enhance the robustness of target models by ensuring that model predictions are unchanged for a small area around each training sample. Thus, adversarial defenses typically have a more fine-grained reliance on the training set and make the target model more vulnerable to membership inference attacks.

To perform the membership inference attacks, we leverage the conventional inference method based on prediction confidence and propose *two new inference methods that exploit structural properties of adversarially robust defenses*. Our experimental evaluation demonstrates that compared with the natural training (undefended) approach, *adversarial defense methods can indeed increase the target model’s risk against membership inference attacks*. When applying adversarial defenses to train the robust models, the membership inference advantage increases by up to 4.5 times compared to the naturally undefended models. Beyond revealing the privacy risks of adversarial defenses, we further investigate the factors that influence the membership information leakage and study the effect of potential countermeasures.

1 INTRODUCTION

Machine learning models, especially deep neural networks, have been deployed prominently in many real-world applications, such as image classification [26, 44], speech recognition [9, 21], natural language processing [2, 8], and game playing [32, 43]. However, since the machine learning algorithms were originally designed without considering potential adversarial threats, their security and privacy vulnerabilities have come to a forefront in recent years, together with the arms race between attacks and defenses [6, 22, 36].

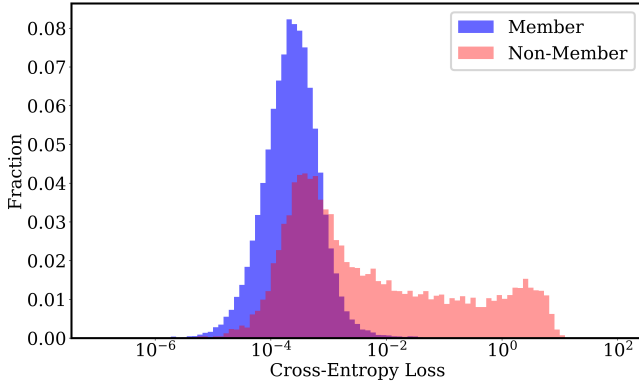
In the security domain, the adversary aims to induce misclassifications to the target machine learning model, with attack methods divided into two categories: evasion attacks and poisoning attacks

[22]. Evasion attacks, also known as adversarial examples, perturb inputs at the test time to induce wrong predictions by the target model [4, 7, 15, 35, 51]. In contrast, poisoning attacks target the training process by maliciously modifying part of training data to cause the trained model to misbehave on some test inputs [5, 25, 40]. In response to these attacks, the security community has designed new training algorithms to secure machine learning models against evasion attacks [16, 30, 31, 45, 56, 61] or poisoning attacks [23, 50].

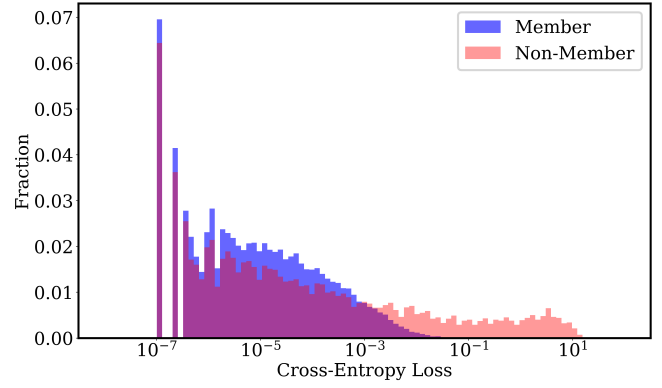
In the privacy domain, the adversary aims to obtain private information about the training data or the target model. Attacks targeting data privacy include: the adversary inferring whether input examples were used to train the target model with membership inference attacks [38, 42, 59], learning global properties of training data with property inference attacks [12], reconstructing training data with model inversion attacks [10, 11] or colluding model training attacks [47, 59]. Attacks targeting model privacy include: the adversary unconverging the model details with model extraction attacks [53], and inferring hyperparameters with hyperparameter stealing attacks [55]. In response to these attacks, the privacy community has designed defenses to prevent privacy leakage of training data [1, 19, 33, 41] or the target model [24, 28].

However, one important limitation of current machine learning defenses is that they typically focus solely on either the security domain or the privacy domain. It is thus unclear whether defense methods in one domain will have any unexpected impact on the other domain. In this paper, we take a step towards enhancing our understanding of machine learning models when both the security domain and privacy domain are combined together. In particular, we seek to understand the privacy risks of securing machine learning models by evaluating *membership inference attacks against adversarially robust deep learning models*, which aim to mitigate the threat of adversarial examples.

The membership inference attack aims to infer whether a data point is part of the target model’s training set or not, posing a serious privacy risk as the membership can reveal an individual’s sensitive information. For example, participation in a hospital’s health analytic training set means that an individual was once a patient in that hospital. It has been shown that the success of membership inference attacks is highly related to the target model’s overfitting and sensitivity as to training data [38, 42, 59]. Adversarially robust models aim to enhance the robustness of target models by ensuring that model predictions are unchanged for a small area (such as l_∞ ball) around each training example. Intuitively, adversarially robust models have the potential to overfit on the training set and increase the model sensitivity, resulting in an enhanced risk of membership inference attacks. As an example, Figure 1 shows the histogram of cross-entropy loss values of training data and test data for both naturally undefended and adversarially robust CIFAR10



(a) Adversarially robust model from Madry et al. [30], with 99% train accuracy and 87% test accuracy.



(b) Naturally undefended model, with 100% train accuracy and 95% test accuracy. Around 23% training and test examples have zero loss.

Figure 1: Histogram of CIFAR10 classifiers’ loss values of training data (members) and test data (non-members). We can see the larger divergence between the loss distribution over members and non-members on the robust model as compared to the natural model. This shows the privacy risk of securing deep learning models against adversarial examples.

classifiers provided by Madry et al. [30]. We can see that members (training data) and non-members (test data) can be distinguished more easily for the robust model, compared to the natural model.

To measure the membership inference risks of adversarially robust models, besides the conventional inference method based on prediction confidence, we propose two new inference methods that exploit the structural properties of adversarial defenses. We measure the privacy risks of robust models trained with six state-of-the-art adversarial defense methods, and find that adversarially robust models are indeed more susceptible to membership inference attacks than naturally undefended models. We further perform a comprehensive investigation to analyze the relation between privacy leakage and model properties, and study the effect of countermeasures such as temperature scaling and regularization.

In summary, we make the following contributions in this paper:

- (1) We propose two new membership inference attacks specific to adversarially robust models by exploiting adversarial examples’ predictions and verified worst-case predictions. With these two new methods, we can achieve higher inference accuracies than the conventional inference method based on prediction confidence of benign inputs.
- (2) We perform membership inference attacks on models trained with six state-of-the-art adversarial defense methods (3 empirical defenses [30, 45, 61] and 3 verifiable defenses [16, 31, 56]). We demonstrate that all methods indeed increase the model’s membership inference risk. By defining the membership inference advantage as the increase in inference accuracy over random guessing (multiplied by 2) [59], we show that robust machine learning models can incur a membership inference advantage 4.5×, 2.1×, 3.5× times the membership inference advantage of naturally undefended models, on Yale Face, Fashion-MNIST, and CIFAR10 datasets, respectively.
- (3) We further explore the factors that influence the membership inference performance of the adversarially robust

model, including its robustness generalization, the adversarial perturbation constraint, and the model capacity.

- (4) We study the effect of potential countermeasures, including temperature scaling and regularization, to reduce privacy leakage via membership inference attacks.

Some of our analysis was briefly discussed in a short workshop paper [48]. In this paper, we go further by proposing two new membership inference attacks and measuring four more adversarial defense methods, where we show that all adversarial defenses can increase the privacy risks of target models. We also perform a comprehensive investigation of factors that impact the privacy risks and discuss potential countermeasures.

2 BACKGROUND AND RELATED WORK: ADVERSARIAL EXAMPLES AND MEMBERSHIP INFERENCE ATTACKS

In this section, we first present the background and related work on adversarial examples and defenses, and then discuss membership inference attacks.

2.1 Adversarial Examples and Defenses

Let $F_\theta : \mathbb{R}^d \rightarrow \mathbb{R}^k$ be a machine learning model with d input features and k output classes, parameterized by weights θ . For an example $z = (\mathbf{x}, y)$ with the input feature \mathbf{x} and the ground truth label y , the model outputs a prediction vector over all class labels $F_\theta(\mathbf{x})$ with $\sum_{i=0}^{k-1} F_\theta(\mathbf{x})_i = 1$, and the final prediction will be the label with the largest prediction probability $\hat{y} = \arg\max_i F_\theta(\mathbf{x})_i$. For neural networks, the outputs of its penultimate layer are known as logits, and we represent them as a vector $g_\theta(\mathbf{x})$. The softmax function is then computed on logits to obtain final prediction vector.

$$F_\theta(\mathbf{x})_i = \frac{\exp(g_\theta(\mathbf{x})_i)}{\sum_{j=0}^{k-1} \exp(g_\theta(\mathbf{x})_j)} \quad (1)$$

Given a training set D_{train} , the natural training algorithm aims to make model predictions match ground truth labels by minimizing

the prediction loss over all training examples.

$$\min_{\theta} \frac{1}{|D_{\text{train}}|} \sum_{\mathbf{z} \in D_{\text{train}}} \ell(F_{\theta}, \mathbf{z}), \quad (2)$$

where $|D_{\text{train}}|$ denotes the size of training set, and ℓ computes the prediction loss. A widely-adopted loss function is the cross-entropy loss:

$$\ell(F_{\theta}, \mathbf{z}) = - \sum_{i=0}^{k-1} \mathbb{1}\{i = y\} \cdot \log(F_{\theta}(\mathbf{x})_i), \quad (3)$$

where $\mathbb{1}\{\cdot\}$ is the indicator function.

2.1.1 Adversarial examples: Although machine learning models have achieved tremendous success in many classification scenarios, they have been found to be easily fooled by adversarial examples [4, 7, 15, 35, 51]. Adversarial examples induce incorrect classifications to target models, and can be generated via imperceptible perturbations to benign inputs.

$$\operatorname{argmax}_i F_{\theta}(\tilde{\mathbf{x}})_i \neq y, \quad \text{such that } \tilde{\mathbf{x}} \in \mathcal{B}_{\epsilon}(\mathbf{x}), \quad (4)$$

where $\mathcal{B}_{\epsilon}(\mathbf{x})$ denotes the set of points around \mathbf{x} within the perturbation budget of ϵ . Usually a l_p ball is chosen as the perturbation constraint for generating adversarial examples i.e., $\mathcal{B}_{\epsilon}(\mathbf{x}) = \{\mathbf{x}' \mid \|\mathbf{x}' - \mathbf{x}\|_p \leq \epsilon\}$. We consider the l_{∞} -ball adversarial constraint throughout the paper, as it is widely adopted by most adversarial defense methods [16, 30, 31, 37, 45, 56, 61].

The solution to Equation (4) is called an ‘‘untargeted adversarial example’’ as the adversarial goal is to achieve any incorrect classification. In comparison, a ‘‘targeted adversarial example’’ ensures that the model prediction is a specified incorrect label y' , which is not equal to y .

$$\operatorname{argmax}_i F_{\theta}(\tilde{\mathbf{x}})_i = y', \quad \text{such that } \tilde{\mathbf{x}} \in \mathcal{B}_{\epsilon}(\mathbf{x}). \quad (5)$$

Unless otherwise specified, an adversarial example in this paper refers to an untargeted adversarial example.

To provide adversarial robustness under the perturbation constraint \mathcal{B}_{ϵ} , instead of natural training algorithm shown in Equation (2), a robust training algorithm is adopted by adding an additional robust loss function.

$$\min_{\theta} \frac{1}{|D_{\text{train}}|} \sum_{\mathbf{z} \in D_{\text{train}}} \alpha \cdot \ell(F_{\theta}, \mathbf{z}) + (1 - \alpha) \cdot \ell_R(F_{\theta}, \mathbf{z}, \mathcal{B}_{\epsilon}), \quad (6)$$

where α is the ratio to trade off natural loss and robust loss, and ℓ_R measures the robust loss, which can be formulated as maximizing prediction loss ℓ' under the constraint \mathcal{B}_{ϵ} .

$$\ell_R(F_{\theta}, \mathbf{z}, \mathcal{B}_{\epsilon}) = \max_{\tilde{\mathbf{x}} \in \mathcal{B}_{\epsilon}(\mathbf{x})} \ell'(F_{\theta}, (\tilde{\mathbf{x}}, y)) \quad (7)$$

ℓ' can be same as ℓ or other appropriate loss functions.

However, it is usually hard to find the exact solution to Equation (7). Therefore, the adversarial defenses propose different ways to approximate the robust loss ℓ_R , which can be divided into two categories: empirical defenses and verifiable defenses.

2.1.2 Empirical defenses: Empirical defense methods approximate robust loss values by generating adversarial examples \mathbf{x}_{adv} at each training step with state-of-the-art attack methods and computing their prediction loss. Now the robust training algorithm can be expressed as following.

$$\min_{\theta} \frac{1}{|D_{\text{train}}|} \sum_{\mathbf{z} \in D_{\text{train}}} \alpha \cdot \ell(F_{\theta}, \mathbf{z}) + (1 - \alpha) \cdot \ell'(F_{\theta}, (\mathbf{x}_{adv}, y)) \quad (8)$$

Three of our tested adversarial defense methods belong to this category, which are described as follows.

PGD-Based Adversarial Training (PGD-Based Adv-Train) [30]: Madry et al. [30] propose one of the most effective empirical defense methods by using the projected gradient descent (PGD) method to generate adversarial examples for maximizing cross-entropy loss ($\ell' = \ell$) and training purely on those adversarial examples ($\alpha = 0$). The PGD attack contains T gradient descent steps, which can be expressed as

$$\tilde{\mathbf{x}}^{t+1} = \Pi_{\mathcal{B}_{\epsilon}(\mathbf{x})}[\tilde{\mathbf{x}}^t + \eta \cdot \text{sign}(\nabla_{\tilde{\mathbf{x}}} \ell(F_{\theta}, (\tilde{\mathbf{x}}^t, y)))] \quad (9)$$

where $\tilde{\mathbf{x}}^0 = \mathbf{x}$, $\mathbf{x}_{adv} = \tilde{\mathbf{x}}^T$, η is the step size value, ∇ denotes the gradient computation, and $\Pi_{\mathcal{B}_{\epsilon}(\mathbf{x})}$ means the projection onto the perturbation constraint $\mathcal{B}_{\epsilon}(\mathbf{x})$.

Distributional Adversarial Training (Dist-Based Adv-Train) [45]: Instead of strictly satisfying the perturbation constraint with projection step $\Pi_{\mathcal{B}_{\epsilon}(\mathbf{x})}$ as in PGD attacks, Sinha et al. [45] generate adversarial examples by solving the Lagrangian relaxation of cross-entropy loss:

$$\max_{\tilde{\mathbf{x}}} \ell(F_{\theta}, (\tilde{\mathbf{x}}, y)) - \gamma \|\tilde{\mathbf{x}} - \mathbf{x}\|_p, \quad (10)$$

where γ is the penalty parameter for the l_p distance. A multi-step gradient descent method is adopted to solve Equation (10). The model will then be trained on the cross-loss entropy ($\ell' = \ell$) of adversarial examples only ($\alpha = 0$).

Sinha et al. [45] derive a statistical guarantee for l_2 distributional robustness with strict conditions requiring the loss function ℓ to be smooth on \mathbf{x} , which are not satisfied in our setting. We mainly use widely-adopted ReLU activation functions for our machine learning models, which result in a non-smooth loss function. Also, we generate adversarial examples with l_{∞} distance penalties by using the algorithm proposed by Sinha et al. [45] in Appendix E, where there is no robustness guarantee. Thus, we categorize the defense method as empirical.

Difference-based Adversarial Training (Diff-Based Adv-Train)

[61]: Instead of using the cross-entropy loss of adversarial examples, with insights from a toy binary classification task, Zhang et al. [61] propose to use the difference (e.g., Kullback-Leibler (KL) divergence) between the benign output $F_{\theta}(\mathbf{x})$ and the adversarial output $F_{\theta}(\mathbf{x}_{adv})$ as the loss function ℓ' , and combine it with natural cross entropy loss ($\alpha \neq 0$).

$$\ell'(F_{\theta}, (\mathbf{x}_{adv}, y)) = d_{kl}(F_{\theta}(\mathbf{x}_{adv}), F_{\theta}(\mathbf{x})), \quad (11)$$

where d_{kl} computes the KL divergence. Adversarial examples are also generated with PGD-based attacks, except that now the attack goal is to maximize the output difference,

$$\tilde{\mathbf{x}}^{t+1} = \Pi_{\mathcal{B}_{\epsilon}(\mathbf{x})}[\tilde{\mathbf{x}}^t + \eta \cdot \text{sign}(\nabla_{\tilde{\mathbf{x}}} d_{kl}(F_{\theta}(\tilde{\mathbf{x}}^t), F_{\theta}(\mathbf{x})))] \quad (12)$$

2.1.3 Verifiable defenses: Although empirical defense methods are effective against state-of-the-art adversarial examples [3], there is no *guarantee* for such robustness. To obtain a guarantee for robustness, verification approaches have been proposed to compute an upper bound of prediction loss ℓ' under the adversarial perturbation constraint \mathcal{B}_ϵ . If the input can still be predicted correctly in the verified worst case, then it is certain that there is no misclassification existing under \mathcal{B}_ϵ .

Thus, verifiable defense methods take the verification process into consideration during training by using the verified worst case prediction loss as robust loss value ℓ_R . Now the robust training algorithm becomes

$$\min_{\theta} \frac{1}{|D_{\text{train}}|} \sum_{\mathbf{z} \in D_{\text{train}}} \alpha \cdot \ell(F_{\theta}, \mathbf{z}) + (1 - \alpha) \cdot \mathcal{V}(\ell'(F_{\theta}, (\tilde{\mathbf{x}}, y)), \mathcal{B}_\epsilon), \quad (13)$$

where \mathcal{V} means verified upper bound computation of prediction loss ℓ' under the adversarial perturbation constraint \mathcal{B}_ϵ . In this paper, we consider the following three verifiable defense method. **Duality-Based Verification (Dual-Based Verify)** [56]: Wong and Kolter [56] compute the verified worst-case loss by solving its dual problem with convex relaxation on non-convex ReLU operations and then minimize this overapproximated robust loss values only ($\alpha = 0, \ell' = \ell$). They further combine this duality relaxation method with the random projection technique to scale to more complex neural network architectures [57], like ResNet [20].

Abstract Interpretation-Based Verification (Abs-Based Verify) [31]: Mirman et al. [31] leverage the technique of abstract interpretation to compute the worse-case loss: an abstract domain (such as interval domain, zonotope domain [13]) is used to express the adversarial perturbation constraint \mathcal{B}_ϵ at the input layer, and by applying abstract transformers on it, the maximum verified range of model output is obtained. They adopt a softplus function on the logits $g_{\theta}(\tilde{\mathbf{x}})$ to compute the robust loss value and then combine it with natural training loss ($\alpha \neq 0$)

$$\ell'(F_{\theta}, (\tilde{\mathbf{x}}, y)) = \log(\exp(\max_{y' \neq y} g_{\theta}(\tilde{\mathbf{x}})_{y'} - g_{\theta}(\tilde{\mathbf{x}})_y) + 1) \quad (14)$$

Interval Bound Propagation-Based Verification (IBP-Based Verify) [16]: Gowal et al. [16] share a similar design as Mirman et al. [31]: they express the constraint \mathcal{B}_ϵ as a bounded interval domain (one specified domain considered by Mirman et al. [31]) and propagate this bound to the output layer. The robust loss is computed as a cross-entropy loss of verified worse-case outputs ($\ell' = \ell$) and then combined with natural prediction loss ($\alpha \neq 0$) as the final loss value during training.

2.2 Membership Inference Attacks

For a target machine learning model, the membership inference attacks aim to determine whether a given data point was used to train the model or not [18, 29, 34, 38, 42, 59]. The attack poses a serious privacy risk to the individuals whose data is used for model training, for example in the setting of health analytics.

Shokri et al. [42] design a membership inference attack method based on training an inference model to distinguish between predictions on training set members versus non-members. To train the inference model, they introduce the *shadow training technique*: (1) the adversary first trains multiple “shadow models” which simulate

the behavior of the target model, (2) based on the shadow models’ outputs on their own training and test examples, the adversary obtains a labeled (member vs non-member) dataset, and (3) finally trains the inference model as a neural network to perform membership inference attack against the target model. The input to the inference model is the prediction vector of the target model on a target data record.

A simpler inference model, such as a linear classifier, can also distinguish significantly vulnerable members from non-members. Yeom et al. [59] suggest comparing the prediction confidence value of a target example with a threshold (learned for example through shadow training). Large confidence indicates membership. Their experiment results show that such a simple inference model is reasonably effective and achieve membership inference accuracy close to that of the shadow training method on target models. In this paper, we mainly follow this confidence-thresholding membership inference approach.

3 MEMBERSHIP INFERENCE ATTACKS AGAINST ROBUST MODELS

In this section, we first present some insights on why training models to be robust against adversarial examples make them more susceptible to membership inference attacks. We then formally present our membership inference attacks.

Throughout the paper, we use “*natural (default) model*” and “*robust model*” to denote the machine learning model with natural training algorithm and robust training algorithm, respectively. We also call the unmodified inputs and adversarially perturbed inputs as “*benign examples*” and “*adversarial examples*”. When evaluating the model’s classification performance, “*train accuracy*” and “*test accuracy*” are used to denote the classification accuracy of benign examples from training and test sets; “*adversarial train accuracy*” and “*adversarial test accuracy*” represent the classification accuracy of adversarial examples from training and test sets; “*verified train accuracy*” and “*verified test accuracy*” measure the classification accuracy under the verified worst-case predictions from training and test sets. Finally, an input example is called “*secure*” when it is correctly classified by the model for all adversarial perturbations within the constraint \mathcal{B}_ϵ , “*insecure*” otherwise.

The performance of membership inference attacks is highly related to how much overfitted they are on their training data [42, 59]. An extremely simple attack algorithm can infer membership based on whether or not an input is correctly classified. In this case, it is clear that a large gap between the target model’s train and test accuracy leads to a significant membership inference attack accuracy (as most members are correctly classified, but not the non-members). Tsipras et al. [54] and Zhang et al. [61] show that robust training might lead to a drop in test accuracy (i.e., more overfitting). This is shown based on both empirical and theoretical analysis on toy classification tasks. Moreover, the overfitting phenomenon can be worse for a robust model when evaluating its accuracy on adversarial examples [39, 46]. Thus, compared with the natural models, **the robust models might leak more membership information, due to exhibiting a larger generalization error, in both the benign or adversarial settings.**

The performance of membership inference attack is related to the target model’s sensitivity with regard to training data [29]. The sensitivity measure is the influence of one data point on the target model’s performance by computing its prediction difference, when trained with and without this data point. Intuitively, when the target model is more sensitive on a training point, it is more likely that the membership inference adversary can guess its membership correctly. The robust training algorithms aim to ensure that model predictions remain unchanged for a small area (such as the l_∞ ball) around any data point. However, in practice, they guarantee this for the training examples, thus, magnifying the influence of the training data on the model. Therefore, compared with the natural training, **the robust training algorithms might make the model more susceptible to membership inference attacks, by increasing its sensitivity to its training data.**

To validate the above insights, let’s take the natural and the robust CIFAR10 classifiers provided by Madry et al. [30] as an example. From Figure 1, we have seen that compared to the natural model, the robust model has a larger divergence between the prediction loss of training data and test data. Our fine-grained analysis in Appendix A further reveals that the large divergence of robust model is highly related to its robustness performance. Moreover, the robust model incurs a significant generalization error in the adversarial setting, with 96% adversarial train accuracy, and only 47% adversarial test accuracy. Finally, we will experimentally show in Section 5.2.1 that the robust model is indeed more sensitive with regard to training data.

3.1 Membership Inference Performance

Table 1: Notations for membership inference attacks against robust machine learning models.

Symbol	Description
F	Target machine learning model.
\mathcal{B}_ϵ	Adversarial perturbation constraint when training a robust model.
D_{train}	Model’s training set.
D_{test}	Model’s test set.
\mathbf{x}	Benign (unmodified) input example.
y	Ground truth label for the input \mathbf{x} .
\mathbf{x}_{adv}	Adversarial example generated from \mathbf{x} .
\mathcal{V}	Robustness verification to compute verified worst-case predictions.
\mathcal{I}	Membership inference strategy.
A_{inf}	Membership inference accuracy.
$ADVT_{inf}$	Membership inference advantage compared to random guessing.

In this part, we describe the membership inference attack and its performance formally, with notations listed in Table 1. For a neural network model F (we skip its parameter θ for simplicity) that is robustly trained with the adversarial constraint \mathcal{B}_ϵ , the membership inference attack aims to determine whether a given input example $\mathbf{z} = (\mathbf{x}, y)$ is in its training set D_{train} or not. We denote the inference strategy adopted by the adversary as $\mathcal{I}(F, \mathcal{B}_\epsilon, \mathbf{z})$, which codes members as 1, and non-members as 0.

We use the fraction of correct membership predictions, as the metric to evaluate membership inference accuracy. We use a test set D_{test} which does not overlap with the training set, to represent

non-members. We sample a random data point (\mathbf{x}, y) from either D_{train} or D_{test} with an equal 50% probability, to test the membership inference attack. We measure the membership inference accuracy as follows.

$$A_{inf}(F, \mathcal{B}_\epsilon, \mathcal{I}) = \frac{\sum_{\mathbf{z} \in D_{\text{train}}} \mathcal{I}(F, \mathcal{B}_\epsilon, \mathbf{z})}{2 \cdot |D_{\text{train}}|} + \frac{\sum_{\mathbf{z} \in D_{\text{test}}} 1 - \mathcal{I}(F, \mathcal{B}_\epsilon, \mathbf{z})}{2 \cdot |D_{\text{test}}|}, \quad (15)$$

where $|\cdot|$ measures the size of a dataset.

According to Equation (15), a random guessing strategy will lead to a 50% membership inference accuracy. We also use the notion of membership inference advantage proposed by Yeom et al. [59] to measure how much improvement our inference strategy obtains compared to random guessing.

$$ADVT_{inf} = 2 \times (A_{inf} - 0.5) \quad (16)$$

3.2 Exploiting the Model’s Predictions on Benign Examples

We adopt a confidence-thresholding inference strategy due to its simplicity and effectiveness [59]: an input (\mathbf{x}, y) is inferred as member if its prediction confidence $F(\mathbf{x})_y$ is larger than a preset threshold value. We denote this inference strategy as \mathcal{I}_B since it relies on the benign examples’ predictions. We have the following expressions for this inference strategy and its inference accuracy.

$$\begin{aligned} \mathcal{I}_B(F, \mathcal{B}_\epsilon, (\mathbf{x}, y)) &= \mathbb{1}\{F(\mathbf{x})_y \geq \tau_B\} \\ A_{inf}(F, \mathcal{B}_\epsilon, \mathcal{I}_B) &= \frac{1}{2} + \frac{1}{2} \cdot \left(\frac{\sum_{\mathbf{z} \in D_{\text{train}}} \mathbb{1}\{F(\mathbf{x})_y \geq \tau_B\}}{|D_{\text{train}}|} \right. \\ &\quad \left. - \frac{\sum_{\mathbf{z} \in D_{\text{test}}} \mathbb{1}\{F(\mathbf{x})_y \geq \tau_B\}}{|D_{\text{test}}|} \right), \end{aligned} \quad (17)$$

where $\mathbb{1}\{\cdot\}$ is the indicator function and the last two terms are the values of complementary cumulative distribution functions of training examples’ and test examples’ prediction confidences, at the point of threshold τ_B , respectively. In our experiments, we evaluate the worst case inference risks by choosing τ_B to achieve the highest inference accuracy, i.e., maximizing the gap between two complementary cumulative distribution function values. In practice, an adversary can learn the threshold via the shadow training technique [42].

This inference strategy \mathcal{I}_B does not leverage the adversarial constraint \mathcal{B}_ϵ . In this paper, we propose two new membership inference strategies by taking \mathcal{B}_ϵ into consideration, which are tailored to robust models.

3.3 Exploiting the Model’s Predictions on Adversarial Examples

Our first new inference strategy is to generate an (untargeted) adversarial example \mathbf{x}_{adv} for input (\mathbf{x}, y) under the constraint \mathcal{B}_ϵ , and use a threshold on the model’s prediction confidence on \mathbf{x}_{adv} . We have following expression for this strategy \mathcal{I}_A and its inference

accuracy.

$$\begin{aligned} \mathcal{I}_A(F, \mathcal{B}_\epsilon, (\mathbf{x}, y)) &= \mathbb{1}\{F(\mathbf{x}_{adv})_y \geq \tau_A\} \\ A_{inf}(F, \mathcal{B}_\epsilon, \mathcal{I}_A) &= \frac{1}{2} + \frac{1}{2} \cdot \left(\frac{\sum_{\mathbf{z} \in D_{train}} \mathbb{1}\{F(\mathbf{x}_{adv})_y \geq \tau_A\}}{|D_{train}|} \right. \\ &\quad \left. - \frac{\sum_{\mathbf{z} \in D_{test}} \mathbb{1}\{F(\mathbf{x}_{adv})_y \geq \tau_A\}}{|D_{test}|} \right) \end{aligned} \quad (18)$$

In our experiments, we use the PGD attack method shown in Equation (9) with the same perturbation constraint \mathcal{B}_ϵ as in the robust training process to obtain \mathbf{x}_{adv} . We choose the preset threshold τ_A to achieve highest inference accuracy, i.e., maximizing the gap between two complementary cumulative distribution functions of prediction confidence on adversarial train and test examples.

3.3.1 Targeted adversarial examples. We extend the attack to exploiting targeted adversarial examples. Targeted adversarial examples contain information about distance of the benign input to each label’s decision boundary, and are expected to leak more membership information than the untargeted adversarial example which only contains information about distance to a nearby label’s decision boundary. We adapt the PGD attack method to find targeted adversarial examples (Equation (5)) by iteratively minimizing the targeted cross-entropy loss.

$$\tilde{\mathbf{x}}^{t+1} = \Pi_{\mathcal{B}_\epsilon(\mathbf{x})}(\tilde{\mathbf{x}}^t - \eta \cdot \text{sign}(\nabla_{\tilde{\mathbf{x}}} \ell(F_\theta, (\tilde{\mathbf{x}}^t, y')))) \quad (19)$$

The confidence thresholding inference strategy does not apply for targeted adversarial examples because there exist $k - 1$ targeted adversarial examples (we have $k - 1$ incorrect labels) for each input. Instead, following Shokri et al. [42], we train a binary inference classifier for each class label to perform the membership inference attack. For each class label, we first choose a fraction of training and test points and generate corresponding targeted adversarial examples. Next, we compute model predictions on the targeted adversarial examples, and use them to train the membership inference classifier. Finally, we perform inference attacks using the remaining training and test points.

3.4 Exploiting the Verified Worst-Case Predictions on Adversarial Examples

Our attacks above generate adversarial examples using the *heuristic* strategy of projected gradient descent. Next, we leverage *verification* techniques \mathcal{V} used by the verifiably defended models [16, 31, 56] to obtain the input’s worst-case predictions under the adversarial constraint \mathcal{B}_ϵ . We use the input’s worst-case prediction confidence to predict its membership. The expressions for this strategy \mathcal{I}_V and its inference accuracy are as follows.

$$\begin{aligned} \mathcal{I}_V(F, \mathcal{B}_\epsilon, (\mathbf{x}, y)) &= \mathbb{1}\{\mathcal{V}(F(\tilde{\mathbf{x}})_y, \mathcal{B}_\epsilon) \geq \tau_V\} \\ A_{inf}(F, \mathcal{B}_\epsilon, \mathcal{I}_V) &= \frac{1}{2} + \frac{1}{2} \cdot \left(\frac{\sum_{\mathbf{z} \in D_{train}} \mathcal{V}(F(\tilde{\mathbf{x}})_y, \mathcal{B}_\epsilon) \geq \tau_V}{|D_{train}|} \right. \\ &\quad \left. - \frac{\sum_{\mathbf{z} \in D_{test}} \mathcal{V}(F(\tilde{\mathbf{x}})_y, \mathcal{B}_\epsilon) \geq \tau_V}{|D_{test}|} \right), \end{aligned} \quad (20)$$

where $\mathcal{V}(F(\tilde{\mathbf{x}})_y, \mathcal{B}_\epsilon)$ returns the verified worst-case prediction confidence for all examples $\tilde{\mathbf{x}}$ satisfying the adversarial perturbation constraint $\tilde{\mathbf{x}} \in \mathcal{B}_\epsilon(\mathbf{x})$, and τ_V is chosen in a similar manner as our previous two inference strategies.

Note that different verifiable defenses adopt different verification methods \mathcal{V} . Our inference strategy uses the same verification method which is used in the target model’s verifiably robust training process.

4 EXPERIMENT SETUP

In this section, we describe the datasets, neural network architectures, and corresponding adversarial perturbation constraints that we use in our experiments. Throughout the paper, we focus on the l_∞ perturbation constraint: $\mathcal{B}_\epsilon(\mathbf{x}) = \{\mathbf{x}' \mid \|\mathbf{x}' - \mathbf{x}\|_\infty \leq \epsilon\}$. The detailed architectures are summarized in Appendix B. We will make our code publicly available to facilitate reproducible experiments and verification.

Yale Face. The extended Yale Face database B is used to train face recognition models, and contains gray scale face images of 38 subjects under various lighting conditions [14, 27]. We use the cropped version of this dataset, where all face images are aligned and cropped to have the dimension of 168×192 . In this version, each subject has 64 images with the same frontal poses under different lighting conditions, among which 18 images were corrupted during the image acquisition, leading to 2,414 images in total [27]. In our experiments, we select 50 images for each subject to form the training set (total size is 1,900 images), and use the remaining 514 images as the test set.

For the model architecture, we use a convolutional neural network (CNN) with the convolution kernel size 3×3 , as suggested by Simonyan et al. [44]. The CNN model contains 4 blocks with different numbers of output channels (8, 16, 32, 64), and each block contains two convolution layers. The first layer uses a stride of 1 for convolutions, and the second layer uses a stride of 2. There are two fully connected layers after the convolutional layers, each containing 200 and 38 neurons. When training the robust models, we set the l_∞ perturbation budget (ϵ) to be $8/255$.

Fashion-MNIST. This dataset consists of a training set of 60,000 examples and a test set of 10,000 examples [58]. Each example is a 28×28 grayscale image, associated with a class label from 10 fashion products, such as shirt, coat, sneaker.

Similar to Yale Face, we also adopt a CNN architecture with the convolution kernel size 3×3 . The model contains 2 blocks with output channel numbers (256, 512), and each block contains three convolution layers. The first two layers both use a stride of 1, while the last layer uses a stride of 2. Two fully connected layers are added at the end, with 200 and 10 neurons, respectively. When training the robust models, we set the l_∞ perturbation budget (ϵ) to be 0.1. **CIFAR10.** This dataset is composed of 32×32 color images in 10 classes, with 6,000 images per class. In total, there are 50,000 training images and 10,000 test images.

We use the wide ResNet architecture [60] to train a CIFAR10 classifier, following Madry et al. [30]. It contains 3 groups of residual layers with output channel numbers (160, 320, 640) and 5 residual units for each group. Similarly, two fully connected layer with 200 and 10 neurons are added at the end. When training the robust models, we set the l_∞ perturbation budget (ϵ) to be $8/255$.

5 MEMBERSHIP INFERENCE ATTACKS AGAINST EMPIRICALLY ROBUST MODELS

In this section we discuss membership inference attacks against 3 empirical defense methods: PGD-based adversarial training (PGD-Based Adv-Train) [30], distributional adversarial training (Dist-Based Adv-Train) [45], and difference-based adversarial training (Diff-Based Adv-Train) [61]. We train the robust models against the l_∞ adversarial constraint on the Yale Face dataset, the Fashion-MNIST dataset, and the CIFAR10 dataset, with neural network architecture as described in Section 4. Following previous work [3, 30, 31, 56], the perturbation budget ϵ values are set to be $8/255$, 0.1 , and $8/255$ on three datasets, respectively. For the empirically robust model, as explained in Section 2.1, there is no verification process to obtain robustness guarantee. Thus the membership inference strategy \mathcal{I}_V does not apply here.

We first present an overall analysis that compares membership inference accuracy for natural models and robust models using multiple inference strategies across multiple datasets. We then present a deeper analysis of membership inference attacks against the PGD-based adversarial training defense.

5.1 Overall Results

Table 2: Membership inference attacks against natural and empirically robust models [30, 45, 61] on the Yale Face dataset with a l_∞ perturbation constraint $\epsilon = 8/255$. Based on Equation (16), the natural model has an inference advantage of 11.70%, while the robust model has an inference advantage up to 37.72%.

Training method	train acc	test acc	adv-train acc	adv-test acc	inference acc (\mathcal{I}_B)	inference acc (\mathcal{I}_A)
Natural	100%	98.25%	4.95%	4.09%	55.85%	54.53%
PGD-Based Adv-Train [30]	99.89%	96.69%	99.05%	77.63%	61.69%	68.86%
Dist-Based Adv-Train [45]	99.58%	93.77%	84.16%	56.23%	62.23%	64.25%
Diff-Based Adv-Train [61]	99.53%	93.77%	99.42%	83.85%	58.06%	65.73%

Table 3: Membership inference attacks against natural and empirically robust models [30, 45, 61] on the Fashion-MNIST dataset with a l_∞ perturbation constraint $\epsilon = 0.1$. Based on Equation (16), the natural model has an inference advantage of 14.24%, while the robust model has an inference advantage up to 28.84%.

Training method	train acc	test acc	adv-train acc	adv-test acc	inference acc (\mathcal{I}_B)	inference acc (\mathcal{I}_A)
Natural	100%	92.18%	4.29%	4.11%	57.12%	50.99%
PGD-Based Adv-Train [30]	99.93%	90.88%	97.53%	68.74%	58.32%	64.42%
Dist-Based Adv-Train [45]	97.98%	90.62%	70.22%	53.13%	57.35%	59.91%
Diff-Based Adv-Train [61]	99.35%	90.92%	90.57%	72.71%	57.02%	58.91%

Table 4: Membership inference attacks against natural and empirically robust models [30, 45, 61] on the CIFAR10 dataset with a l_∞ perturbation constraint $\epsilon = 8/255$. Based on Equation (16), the natural model has an inference advantage of 14.86%, while the robust model has an inference advantage up to 51.30%.

Training method	train acc	test acc	adv-train acc	adv-test acc	inference acc (\mathcal{I}_B)	inference acc (\mathcal{I}_A)
Natural	100%	95.01%	0%	0%	57.43%	50.85%
PGD-Based Adv-Train [30]	99.99%	87.25%	96.07%	46.59%	74.89%	75.65%
Dist-Based Adv-Train [45]	100%	90.10%	43.61%	27.34%	67.16%	64.69%
Diff-Based Adv-Train [61]	99.50%	87.99%	77.01%	47.28%	61.18%	67.08%

The membership inference attack results against natural models and empirically robust models [30, 45, 61] are presented in Table 2, Table 3 and Table 4, where “acc” stands for accuracy, while “adv-train acc” and “adv-test acc” report adversarial accuracy under PGD attacks as shown in Equation (9).

According to these results, **all three empirical defense methods will make the model more susceptible to membership inference attacks**: compared with natural models, robust models increase the membership inference advantage by up to 3.2 \times , 2.0 \times , and 3.5 \times , for Yale Face, Fashion-MNIST, and CIFAR10, respectively.

We also find that **for robust models, membership inference attacks based on adversarial example’s prediction confidence (\mathcal{I}_A) have higher inference accuracy than the inference attacks based on benign example’s prediction confidence (\mathcal{I}_B) in most cases. On the other hand, for natural models, inference attacks based on benign examples’ prediction confidence lead to higher inference accuracy values**. This happens because our inference strategies rely on the difference between confidence distribution of training points and that of test points. For robust models, most of training points are (empirically) secure against adversarial examples, and adversarial perturbations do not significantly decrease the confidence on them. However, the test set contains more insecure points, and thus adversarial perturbations will enlarge the gap between confidence distributions of training examples and test examples, leading to a higher inference accuracy. For natural models, the use of adversarial examples will decrease the confidence distribution gap, since almost all training points and test points are not secure with adversarial perturbations. The only exception is Dist-Based Adv-Train CIFAR10 classifier, where inference accuracy with strategy \mathcal{I}_B is higher, which can be explained by the poor robustness performance of the model: more than 56% training examples are insecure. Thus, adversarial perturbations will decrease the confidence distribution gap between training examples and test examples in this specific scenario.

5.2 Detailed Membership Inference Analysis of PGD-Based Adversarial Training

In this part, we perform a detailed analysis of membership inference attacks against PGD-based adversarial training defense method [30]

by using the CIFAR10 classifier as an example. We first perform a sensitivity analysis on both natural and robust models to show that the robust model is more sensitive with regard to training data compared to the natural model. We then investigate the relation between privacy leakage and model properties, including robustness generalization, adversarial perturbation constraint and model capacity. We finally show that the predictions of targeted adversarial examples can further enhance the membership inference advantage.

5.2.1 Sensitivity Analysis. In the sensitivity analysis, we remove sample CIFAR10 training points from the training set, perform retraining of the models, and compute the performance difference between the original model and retrained model.

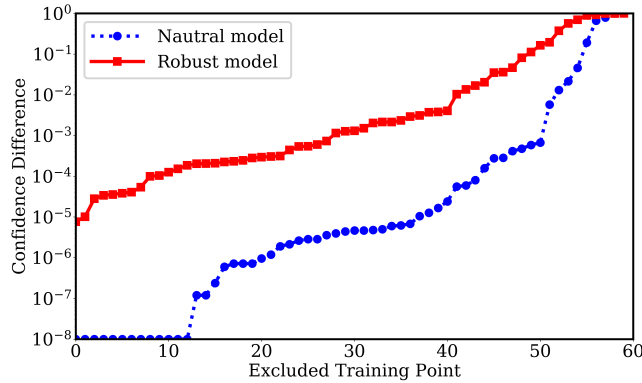


Figure 2: Sensitivity analysis of both robust [30] and natural CIFAR10 classifiers. x-axis denotes the excluded training point id number during the retraining process, and y-axis denotes the difference in prediction confidence between the original model and the retrained model (measuring model sensitivity). The robust model is more sensitive to the training data compared to the natural model.

We excluded 10 training points (one for each class label) and retrained the model. We computed the sensitivity of each excluded point as the difference between its prediction confidence in the retrained model and the original model. We obtained the sensitivity metric for 60 training points by retraining the classifier 6 times. Figure 2 depicts the sensitivity values for the 60 training points (in ascending order) for both robust and natural models. We can see that compared to the natural model, **the robust model is indeed more sensitive to the training data, thus leaking more membership information.**

5.2.2 Privacy risk with robustness generalization. We perform the following experiment to demonstrate the relation between privacy risk and robustness generalization. Recall that in the approach of Madry et al. [30], adversarial examples are generated from *all* training points during the robust training process. In our experiment, we modify the above defense approach to (1) leverage adversarial examples from a *subset* of the CIFAR10 training data to compute the robust prediction loss, and (2) leverage the remaining subset of training points as benign inputs to compute the natural prediction loss.

Table 5: Mixed PGD-based adversarial training experiments [30] on CIFAR10 dataset with a l_∞ perturbation constraint $\epsilon = 8/255$. During the training process, part of the training set, whose ratio is denoted by adv-train ratio, is used to compute robust loss, and the remaining part of the training set is used to compute natural loss.

Adv-train ratio	train acc	test acc	adv-train acc	adv-test acc	inference acc (\mathcal{I}_B)	inference acc (\mathcal{I}_A)
0	100%	95.01%	0%	0%	57.43%	50.85%
1/2	100%	87.78%	75.85%	43.23%	67.20%	66.36%
3/4	100%	86.68%	88.34%	45.66%	71.07%	72.22%
1	99.99%	87.25%	96.07%	46.59%	74.89%	75.65%

The membership inference attack results are summarized in Table 5, where the first column lists the ratio of training points used for computing robust loss. We can see that **as more training points are used for computing the robust loss, the membership inference accuracy increases**, due to the larger gap between adv-train accuracy and adv-test accuracy.

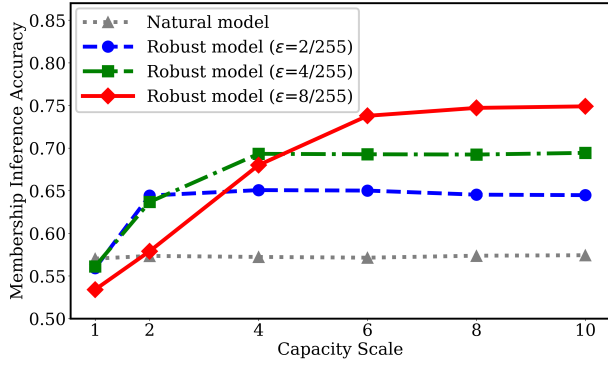
5.2.3 Privacy risk with model perturbation budget. Next, we explore the relationship between membership inference and the adversarial perturbation budget ϵ , which controls the maximum absolute value of adversarial perturbations during robust training process.

Table 6: Membership inference attacks against robust CIFAR10 classifiers [30] with varying adversarial perturbation budgets.

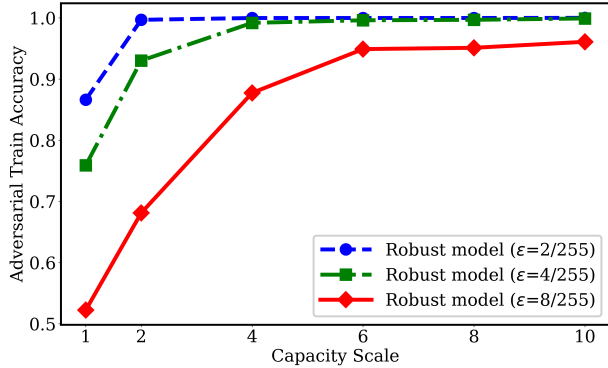
Perturbation budget (ϵ)	train acc	test acc	adv-train acc	adv-test acc	inference acc (\mathcal{I}_B)	inference acc (\mathcal{I}_A)
2/255	100%	93.74%	99.99%	82.20%	64.48%	66.54%
4/255	100%	91.19%	99.89%	70.03%	69.44%	72.43%
8/255	99.99%	87.25%	96.07%	46.59%	74.89%	75.65%

We performed the robust training [30] for three CIFAR10 classifiers with varying adversarial perturbation budgets, and show the result in Table 6. Note that a model trained with a larger ϵ is more robust since it can defend against larger adversarial perturbations. From Table 6, we can see that **more robust models leak more information about the training data.** With a larger ϵ value, the robust model relies on a larger l_∞ ball around each training point, leading to a higher membership inference attack accuracy.

5.2.4 Privacy risk with model capacity. Madry et al. [30] have observed that compared with natural training, robust training requires a significantly larger model capacity (e.g., deeper neural network architectures and more convolution filters) to obtain high robustness. In fact, we can think of the robust training approach as adding more “virtual training points”, which are within the l_∞ ball around original training points. Thus the model capacity needs to be large enough to fit well on the larger “virtual training set”.



(a) Membership inference attacks against models with different model capacities.



(b) Adversarial train accuracy for models with different model capacities.

Figure 3: Membership inference accuracy and adversarial train accuracy for CIFAR10 classifiers [30] with varying model capacities. The model with a capacity scale of s contains 3 groups of residual layers with output channel numbers ($16s$, $32s$, $64s$), as described in Section 4.

Here we investigate the influence of model capacity by varying the capacity scale of wide ResNet architecture [60] used in CIFAR10 training, which is proportional to the output channel numbers of residual layers. We perform membership inference attacks for the robust models, and show the results in Figure 3. The attacks are based on benign inputs’ predictions (strategy \mathcal{I}_B) and the gray line measures the privacy leakage for the natural models as a baseline.

First, we can see that **as the model capacity increases, the model has a higher membership inference accuracy**, along with a higher adversarial train accuracy. Second, **when using a larger adversarial perturbation budget ϵ , a larger model capacity is also needed**. When $\epsilon = 2/255$, a capacity scale of 2 is enough to fit the training data, while for $\epsilon = 8/255$, a capacity scale of 8 is needed.

5.2.5 Inference attacks using targeted adversarial examples. Next, we investigate membership inference attacks using *targeted* adversarial examples. For each input, we compute 9 targeted adversarial examples with each of the 9 incorrect labels as targets using Equation (19). We then compute the output prediction vectors for all

adversarial examples and use the shadow-training inference method proposed by Shokri et al. [42] to perform membership inference attacks. Specifically, for each class label, we learn a dedicated inference model (binary classifier) by using the output predictions of targeted adversarial examples from 500 training points and 500 test points as the training set for the membership inference. We then test the inference model on the remaining CIFAR10 training and test examples from the same class label. In our experiments, we use a 3-layer fully connected neural network with size of hidden neurons equal to 200, 20, and 2 respectively. We call this method “model-infer (targeted)”.

For untargeted adversarial examples or benign examples, a similar class label-dependent inference model can also be obtained by using either untargeted adversarial example’s prediction vector or benign example’s prediction vector as features of the inference model. We call these methods “model-infer (untargeted)” and “model-infer (benign)”. We use the same 3-layer fully connected neural network as the inference classifier.

Finally, we also adapt our confidence-thresholding inference strategy to be class-label dependent by choosing the confidence value according to prediction confidence values from 500 training points and 500 test points, and then testing on remaining CIFAR10 points from the same class label. Based on whether the confidence value is from the untargeted adversarial input or the benign input, we call the method as “confidence-infer (untargeted)” and “confidence-infer (benign)”.

Table 7: Comparison of membership inference attacks against the robust CIFAR10 classifier [30]. Inference attack strategies include combining predictions of targeted adversarial examples, untargeted adversarial examples, and benign examples with either training an inference neural network model or thresholding the prediction confidence.

Class label	confidence-infer (benign)	model-infer (benign)	confidence-infer (untargeted)	model-infer (untargeted)	model-infer (targeted)
0	70.88%	71.49%	72.21%	72.70%	74.42%
1	63.57%	64.42%	67.52%	67.69%	68.88%
2	80.16%	76.74%	79.71%	80.16%	83.58%
3	90.43%	90.49%	87.64%	87.83%	90.57%
4	82.30%	82.17%	81.83%	81.57%	84.47%
5	81.34%	79.84%	81.57%	81.34%	83.02%
6	75.34%	70.92%	77.66%	76.97%	79.94%
7	69.54%	67.61%	72.92%	72.82%	72.98%
8	69.16%	69.57%	74.36%	74.40%	75.33%
9	68.13%	66.34%	71.86%	72.06%	73.32%

The membership inference attack results using the above five strategies are presented in Table 7. We can see that the **targeted adversarial example based inference strategy “model-infer (targeted)” always has the highest inference accuracy**. This is because the targeted adversarial examples contain information about distance of the input to each label’s decision boundary, while

untargeted adversarial examples contain information about distance of the input to only a nearby label’s decision boundary. Thus targeted adversarial examples leak more membership information. As an aside, we also find that our confidence-based inference methods obtain nearly the same inference results as training neural network models, showing the effectiveness of the confidence-thresholding inference strategies.

6 MEMBERSHIP INFERENCE ATTACKS AGAINST VERIFIABLY ROBUST MODELS

In this section we perform membership inference attacks against 3 verifiable defense methods: duality-based verification (Dual-Based Verify) [56], abstract interpretation-based verification (Abs-Based Verify) [31], and interval bound propagation-based verification (IBP-Based Verify) [16]. We train the verifiably robust models using the network architectures as described in Section 4 (with minor modifications for the Dual-Based Verify method [56] as discussed in Appendix C), the l_∞ perturbation budget ϵ is set to be $8/255$ for the Yale Face dataset and 0.1 for the Fashion-MNIST dataset. We do not evaluate the verifiably robust models for the full CIFAR10 dataset as none of these three defense methods scale to the wide ResNet architecture.

6.1 Overall Results

The membership inference attack results against natural and verifiably robust models are presented in Table 8 and Table 9, where “acc” stands for accuracy, “adv-train acc” and “adv-test acc” measure adversarial accuracy under PGD attacks (Equation (9)), and “ver-train acc” and “ver-test acc” report the verified worst-case accuracy under the perturbation constraint \mathcal{B}_ϵ .

For the Yale Face dataset, all three defense methods leak more membership information. The IBP-Based Verify method even leads to an inference accuracy above 75%, higher than the inference accuracy of empirical defenses shown in Table 2, resulting a $4.5\times$ membership inference advantage (Equation (16)) than the natural model. The inference strategy based on verified prediction confidence (strategy \mathcal{I}_V) has the highest inference accuracy as the verification process enlarges prediction confidence between training data and test data.

On the other hand, for the Fashion-MNIST dataset, we fail to obtain increased membership inference accuracies on the verifiably robust models. However, we also observe much reduced benign train accuracy (below 90%) and verified train accuracy (below 80%), which means that **the model fits the training set poorly**. Similar to our analysis of empirical defenses, we can think the verifiable defense as adding more “virtual training points” around each training example to compute its verified robust loss. Since the verified robust loss is an upper bound on the real robust loss, the added “virtual training points” are in fact beyond the l_∞ ball. Therefore, the model capacity needed for verifiable defenses is even larger than that of empirical defense methods.

From the experiment results in Section 5.2.4, we have shown that if the model capacity is not large enough, the robust model will not fit the training data well. This explains why membership inference accuracies for verifiably robust models are limited in Table 9. However, enlarging the model capacity does not guarantee

that the training points will fit well for verifiable defenses because the verified upper bound of robust loss is likely to be looser with a deeper and larger neural network architecture. We validate our hypothesis in the following two subsections.

6.2 Varying Model Capacities

We use models with varying capacities to robustly train on the Yale Face dataset with the IBP-Based Verify defense [16] as an example.

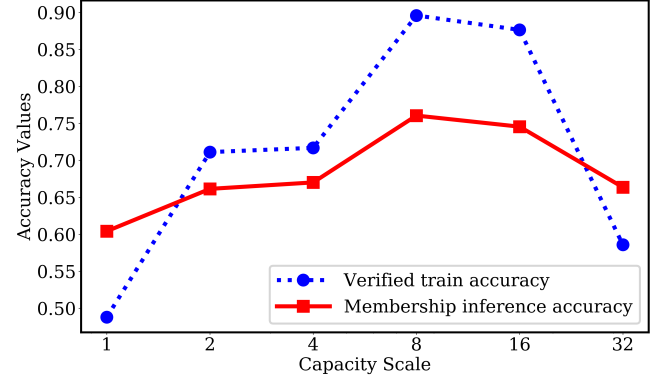


Figure 4: Verified train accuracy and membership inference accuracy using inference strategy \mathcal{I}_V for robust Yale Face classifiers [16] with varying capacities. The model with a capacity scale of s contains 4 convolution blocks with output channel numbers ($s, 2s, 4s, 8s$), as described in Section 4.

We present the results in Figure 4, where model capacity scale of 8 corresponds to the original model architecture, and we perform membership inference attacks based on verified worst-case prediction confidence \mathcal{I}_V . We can see that when model capacity increases, at the beginning, robustness performance gets improved, and we also have a higher membership inference accuracy. However, when the model capacity is too large, the robustness performance and the membership inference accuracy begin decreasing, since now the verified robust loss becomes too loose.

6.3 Reducing the Size of Training Set

In this subsection, we further prove our hypothesis by showing that when the size of the training set is reduced so that the model can fit well on the reduced dataset, the verifiable defense method indeed leads to an increased membership inference accuracy.

We choose the duality-based verifiable defense method [56, 57] and train the CIFAR10 classifier with a normal ResNet architecture: 3 groups of residual layers with output channel numbers (16, 32, 64) and only 1 residual unit for each group. The whole CIFAR10 training set have too many points to be robustly fitted with the verifiable defense algorithm: the robust CIFAR10 classifier [57] with $\epsilon = 2/255$ has the train accuracy below 70%. Therefore, we select a subset of the training data to robustly train the model by randomly choosing 1000 (20%) training images for each class label. We vary the perturbation budget value (ϵ) in order to observe when the model capacity is not large enough to fit on this partial CIFAR10 set using the verifiable training algorithm [56].

Table 8: Membership inference attacks against natural and verifiably robust models [16, 31, 56] on the Yale Face dataset with a l_∞ perturbation constraint $\epsilon = 8/255$. Based on Equation (16), the natural model has the inference advantage of 11.70%, while the robust model has the inference advantage up to 52.10%.

Training method	train acc	test acc	adv-train acc	adv-test acc	ver-train acc	ver-test acc	inference acc (I_B)	inference acc (I_A)	inference acc (I_V)
Natural	100%	98.25%	4.95%	4.09%	N.A.	N.A.	55.85%	54.53%	N.A.
Dual-Based Verify [56]	98.89%	92.80%	98.53%	83.66%	96.37%	68.87%	55.90%	60.06%	64.48%
Abs-Based Verify [31]	99.26%	83.27%	86.21%	50.97%	43.32%	18.09%	65.11%	65.70%	67.05%
IBP-Based Verify [16]	99.16%	85.80%	94.16%	69.26%	89.58%	36.77%	60.45%	66.25%	76.05%

Table 9: Membership inference attacks against natural and verifiably robust models [16, 31, 56] on the Fashion-MNIST dataset with a l_∞ perturbation constraint $\epsilon = 0.1$.

Training method	train acc	test acc	adv-train acc	adv-test acc	ver-train acc	ver-test acc	inference acc (I_B)	inference acc (I_A)	inference acc (I_V)
Natural	100%	92.18%	4.29%	4.11%	N.A.	N.A.	57.12%	50.99%	N.A.
Dual-Based Verify [56]	75.13%	74.29%	65.78%	65.37%	61.77%	61.45%	50.58%	50.43%	50.45%
Abs-Based Verify [31]	86.44%	85.47%	74.22%	73.30%	69.69%	68.89%	50.79%	50.70%	50.59%
IBP-Based Verify [16]	89.85%	86.26%	82.72%	78.57%	79.20%	74.17%	52.13%	52.01%	52.67%

Table 10: Membership inference attacks against natural and verifiably robust CIFAR10 classifiers [56] trained on a subset (20%) of the training data with varying l_∞ perturbation budgets.

Training method	Perturbation budgets (ϵ)	train acc	test acc	adv-train acc	adv-test acc	ver-train acc	ver-test acc	inference acc (I_B)	inference acc (I_A)	inference acc (I_V)
Natural	N.A.	99.83%	71.8%	N.A.	N.A.	N.A.	N.A.	71.50%	N.A.	N.A.
Dual-Based Verify [56]	0.25/255	100%	73.1%	99.99%	69.8%	99.99%	68.18%	76.13%	76.16%	76.04%
Dual-Based Verify [56]	0.5/255	99.98%	69.29%	99.98%	64.51%	99.97%	60.89%	77.06%	77.35%	77.09%
Dual-Based Verify [56]	0.75/255	100%	65.25%	99.95%	59.46%	99.85%	54.71%	77.99%	78.50%	78.20%
Dual-Based Verify [56]	1/255	99.78%	63.96%	99.44%	57.06%	98.61%	50.74%	76.30%	77.05%	77.16%
Dual-Based Verify [56]	1.25/255	98.46%	61.79%	97.30%	53.76%	95.36%	46.70%	74.07%	75.10%	75.41%
Dual-Based Verify [56]	1.5/255	96.33%	60.97%	94.27%	51.72%	90.19%	44.23%	71.08%	72.29%	72.69%

We show the obtained results in Table 10, where the natural model has a low test accuracy (below 75%) and high privacy leakage (inference accuracy is 71.50%) since we only use 20% training examples to learn the classifier. By using the verifiable defense method [56], **the verifiably robust models have increased membership inference accuracy values**, for all ϵ values. We can also see that when increasing the ϵ values, at the beginning, the robust model is more and more susceptible to membership inference attacks (inference accuracy increases from 71.50% to 78.50%). However, beyond a threshold of $\epsilon = 1/255$, the inference accuracy starts to decrease, since a higher ϵ requires a model with a larger capacity to fit well on the training data.

7 DISCUSSION: POTENTIAL COUNTERMEASURES

In this section, we discuss potential countermeasures that can reduce the risk of membership inference attacks while maintaining model robustness.

7.1 Temperature Scaling

Our membership inference strategies leverage the difference between the prediction confidence of the target model on its training set and test set. Thus, a straightforward mitigation method is to reduce this difference by applying temperature scaling on logits [17]. The temperature scaling method was shown to be effective

to reduce privacy risk for natural (baseline) models by Shokri et al. [42], while we are studying its effect for robust models here.

Temperature scaling is a post-processing calibration technique for machine learning models that divides logits by the temperature, T , before the softmax function. Now the model prediction probability can be expressed as

$$F(\mathbf{x})_i = \frac{\exp(g(\mathbf{x})_i/T)}{\sum_{j=0}^{k-1} \exp(g(\mathbf{x})_j/T)}, \quad (21)$$

where $T = 1$ corresponds to original model prediction. By setting $T > 1$, the prediction confidence $F(\mathbf{x})_y$ is reduced, and when $T \rightarrow \infty$, the prediction output is close to uniform and independent of the input, thus leaking no membership information while making the model useless for prediction.

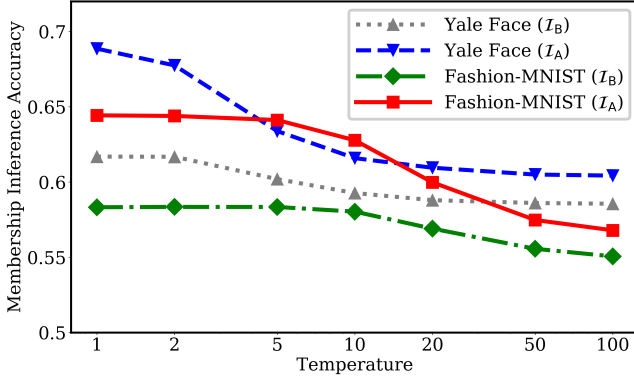


Figure 5: Membership inference accuracy on robust Yale Face and Fashion-MNIST classifiers [30] with varying temperature values [17].

We apply the temperature scaling technique on the robust Yale Face and Fashion-MNIST classifiers using the PGD-based adversarial training defense [30] and investigate its effect on membership inference. We present the membership inference results for varying temperature values (while maintaining the same classification accuracy) in Figure 5. We can see that increasing the temperature value decreases the membership inference accuracy.

7.2 Regularization to Improve Robustness Generalization

Regularization techniques such as parameter norm penalties and dropout [49], are typically used during the training process to solve overfitting issues for machine learning models. Shokri et al. [42] and Salem et al. [38] validate their effectiveness against membership inference attacks. Furthermore, Nasr et al. [33] propose to measure the performance of membership inference attack at each training step and use the measurement as a new regularizer.

The above mitigation strategies are effective regardless of natural or robust machine learning models. For the robust models, we can also rely on the regularization approach, which improves the model’s robustness generalization. This can mitigate membership inference attacks, since a poor robustness generalization leads to a severe privacy risk. We study the method proposed by Song et al.

[46] to improve model’s robustness generalization and explore its performance against membership inference attacks.

The regularization method in [46] performs domain adaptation (DA) [52] for the benign examples and adversarial examples on the logits: two multivariate Gaussian distributions for the logits of benign examples and adversarial examples are computed, and l_1 distances between two mean vectors and two covariance matrices are added into the training loss.

Table 11: Membership inference attacks against robust models [30], where the perturbation budget ϵ is $8/255$ for the Yale Face dataset, and 0.1 for the Fashion-MNIST dataset. When using DA, we modify the robust training algorithm by adding the regularization loss proposed by Song et al. [46].

Dataset	using DA [46]?	train acc	test acc	adv-train acc	adv-test acc	inference acc (I_B)	inference acc (I_A)
Yale Face	no	99.89%	96.69%	99.05%	77.63%	61.69%	68.86%
Yale Face	yes	99.32%	94.75%	99.26%	88.52%	60.73%	63.14%
Fashion MNIST	no	99.93%	90.88%	97.53%	68.74%	58.32%	64.42%
Fashion MNIST	yes	88.97%	86.98%	81.59%	78.65%	51.19%	51.49%

We apply this DA-based regularization approach on the PGD-based adversarial training defense [30] to investigate its effectiveness against membership inference attacks. We list the experimental results both with and without the use of DA regularization for Yale Face and Fashion-MNIST datasets in Table 11. We can see that the DA-based regularization can decrease the gap between adversarial train accuracy and adversarial test accuracy (robust generalization error), leading to a reduction in membership inference risk.

8 CONCLUSIONS

In this paper, we have connected both the security domain and the privacy domain for machine learning systems by investigating the membership inference privacy risk of robust training approaches (that mitigate the adversarial examples). To evaluate the membership inference risk, we propose *two new inference methods that exploit structural properties of adversarially robust defenses*, beyond the conventional inference method based on the prediction confidence of benign input. By measuring the success of membership inference attacks on robust models trained with six state-of-the-art adversarial defense approaches, we find that *all six robust training methods will make the machine learning model more susceptible to membership inference attacks, compared to the naturally undefended training*. Our analysis further reveals that the privacy leakage is related to target model’s robustness generalization, its adversarial perturbation constraint, and its capacity. We also study the effect of potential countermeasures against membership inference attacks, including temperature scaling and regularization to improve robustness generalization. The detailed analysis in our paper highlights the importance of thinking about security and privacy together. Specifically, the membership inference risk needs to be considered when designing approaches to defend against adversarial examples.

9 ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation under grants CNS-1553437, CNS-1704105, CIF-1617286 and EARS-1642962, by the Office of Naval Research Young Investigator Award, by the Army Research Office Young Investigator Prize, by Faculty research awards from Intel and IBM, and by the National Research Foundation, Prime Ministers Office, Singapore under its Strategic Capability Research Centres Funding Initiative.

REFERENCES

- [1] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep learning with differential privacy. In *ACM Conference on Computer and Communications Security (CCS)*. 308–318.
- [2] Daniel Andor, Chris Alberti, David Weiss, Aliaksei Severyn, Alessandro Presta, Kuzman Ganchev, Slav Petrov, and Michael Collins. 2016. Globally normalized transition-based neural networks. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*. 2442–2452.
- [3] Anish Athalye, Nicholas Carlini, and David Wagner. 2018. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International Conference on Machine Learning (ICML)*.
- [4] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrđić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. 2013. Evasion attacks against machine learning at test time. In *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD)*. 387–402.
- [5] Battista Biggio, Blaine Nelson, and Pavel Laskov. 2012. Poisoning attacks against support vector machines. In *International Conference on Machine Learning (ICML)*. 1467–1474.
- [6] Battista Biggio and Fabio Roli. 2018. Wild patterns: Ten years after the rise of adversarial machine learning. *Pattern Recognition* 84 (2018), 317–331.
- [7] Nicholas Carlini and David Wagner. 2017. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy (S&P)*. 39–57.
- [8] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research* 12, Aug (2011), 2493–2537.
- [9] Li Deng, Geoffrey Hinton, and Brian Kingsbury. 2013. New types of deep neural network learning for speech recognition and related applications: An overview. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 8599–8603.
- [10] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. 2015. Model inversion attacks that exploit confidence information and basic countermeasures. In *ACM Conference on Computer and Communications Security (CCS)*. 1322–1333.
- [11] Matthew Fredrikson, Eric Lantz, Somesh Jha, Simon Lin, David Page, and Thomas Ristenpart. 2014. Privacy in Pharmacogenetics: An End-to-End Case Study of Personalized Warfarin Dosing. In *USENIX Security Symposium*. 17–32.
- [12] Karan Ganju, Qi Wang, Wei Yang, Carl A Gunter, and Nikita Borisov. 2018. Property Inference Attacks on Fully Connected Neural Networks using Permutation Invariant Representations. In *ACM Conference on Computer and Communications Security (CCS)*. 619–633.
- [13] Timon Gehr, Matthew Mirman, Dana Drachler-Cohen, Petar Tsankov, Swarat Chaudhuri, and Martin Vechev. 2018. AI²: Safety and robustness certification of neural networks with abstract interpretation. In *IEEE Symposium on Security and Privacy (S&P)*. 3–18.
- [14] Athinodoros S Georgiades, Peter N Belhumeur, and David J Kriegman. 2001. From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Transactions on Pattern Analysis & Machine Intelligence* 6 (2001), 643–660.
- [15] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and Harnessing Adversarial Examples. In *International Conference on Learning Representations (ICLR)*.
- [16] Sven Gowal, Krishnamurthy Dvijotham, Robert Stanforth, Rudy Bunel, Chongli Qin, Jonathan Uesato, Timothy Mann, and Pushmeet Kohli. 2018. On the effectiveness of interval bound propagation for training verifiably robust models. In *NeurIPS Workshop on Security in Machine Learning (SECML)*.
- [17] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. On calibration of modern neural networks. *International Conference on Machine Learning (ICML)*.
- [18] J Hayes, L Melis, G Danezis, and E De Cristofaro. 2018. LOGAN: Membership Inference Attacks Against Generative Models. In *Proceedings on Privacy Enhancing Technologies (PoPETs)*.
- [19] Jamie Hayes and Olga Ohrimenko. 2018. Contamination Attacks and Mitigation in Multi-Party Machine Learning. In *Conference on Neural Information Processing Systems (NeurIPS)*. 6602–6614.
- [20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *IEEE conference on computer vision and pattern recognition (CVPR)*. 770–778.
- [21] Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath, et al. 2012. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal processing magazine* 29, 6 (2012), 82–97.
- [22] Ling Huang, Anthony D Joseph, Blaine Nelson, Benjamin IP Rubinstein, and JD Tygar. 2011. Adversarial machine learning. In *ACM Workshop on Artificial Intelligence and Security (AISec)*. 43–58.
- [23] Matthew Jagielski, Alina Oprea, Battista Biggio, Chang Liu, Cristina Nita-Rotaru, and Bo Li. 2018. Manipulating machine learning: Poisoning attacks and countermeasures for regression learning. In *IEEE Symposium on Security and Privacy (S&P)*.
- [24] Manish Kesarwani, Bhaskar Mukhoty, Vijay Arya, and Sameep Mehta. 2018. Model extraction warning in mlaas paradigm. In *Proceedings of the 34th Annual Computer Security Applications Conference (ACSAC)*. ACM, 371–380.
- [25] Pang Wei Koh and Percy Liang. 2017. Understanding black-box predictions via influence functions. In *International Conference on Machine Learning (ICML)*. 1885–1894.
- [26] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Conference on Neural Information Processing Systems (NeurIPS)*. 1097–1105.
- [27] Kuang-Chih Lee, Jeffrey Ho, and David J Kriegman. 2005. Acquiring linear subspaces for face recognition under variable lighting. *IEEE Transactions on Pattern Analysis & Machine Intelligence* 5 (2005), 684–698.
- [28] Taesung Lee, Benjamin Edwards, Ian Molloy, and Dong Su. 2019. Defending Against Model Stealing Attacks Using Deceptive Perturbations. In *Deep Learning and Security Workshop (DLS)*.
- [29] Yunhui Long, Vincent Bindschaedler, and Carl A Gunter. 2017. Towards Measuring Membership Privacy. *arXiv preprint arXiv:1712.09136* (2017).
- [30] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2018. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations (ICLR)*.
- [31] Matthew Mirman, Timon Gehr, and Martin Vechev. 2018. Differentiable abstract interpretation for provably robust neural networks. In *International Conference on Machine Learning*. 3575–3583.
- [32] Matej Moravčík, Martin Schmid, Neil Burch, Viliam Lisý, Dustin Morrill, Nolan Bard, Trevor Davis, Kevin Waugh, Michael Johanson, and Michael Bowling. 2017. Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. *Science* 356, 6337 (2017), 508–513.
- [33] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2018. Machine learning with membership privacy using adversarial regularization. In *ACM Conference on Computer and Communications Security (CCS)*.
- [34] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2019. Comprehensive Privacy Analysis of Deep Learning: Passive and Active White-box Inference Attacks against Centralized and Federated Learning. In *IEEE Symposium on Security and Privacy (S&P)*.
- [35] Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z Berkay Celik, and Ananthram Swami. 2016. The limitations of deep learning in adversarial settings. In *IEEE European Symposium on Security and Privacy (EuroS&P)*. 372–387.
- [36] Nicolas Papernot, Patrick McDaniel, Arunesh Sinha, and Michael Wellman. 2018. SoK: Security and Privacy in Machine Learning. In *IEEE European Symposium on Security and Privacy (EuroS&P)*. 399–414.
- [37] Aditi Raghunathan, Jacob Steinhardt, and Percy Liang. 2018. Certified defenses against adversarial examples. In *International Conference on Learning Representations (ICLR)*.
- [38] Ahmed Salem, Yang Zhang, Mathias Humbert, Mario Fritz, and Michael Backes. 2019. ML-leaks: Model and data independent membership inference attacks and defenses on machine learning models. In *Network and Distributed Systems Security Symposium (NDSS)*.
- [39] Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. 2018. Adversarially Robust Generalization Requires More Data. In *Conference on Neural Information Processing Systems (NeurIPS)*.
- [40] Ali Shafahi, W Ronny Huang, Mahyar Najibi, Octavian Suci, Christoph Studer, Tudor Dumitras, and Tom Goldstein. 2018. Poison Frogs! Targeted Clean-Label Poisoning Attacks on Neural Networks. In *Conference on Neural Information Processing Systems (NeurIPS)*.
- [41] Reza Shokri and Vitaly Shmatikov. 2015. Privacy-preserving deep learning. In *ACM Conference on Computer and Communications Security (CCS)*. 1310–1321.
- [42] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership inference attacks against machine learning models. In *IEEE Symposium on Security and Privacy (S&P)*. 3–18.

- [43] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. 2016. Mastering the game of Go with deep neural networks and tree search. *Nature* 529, 7587 (2016), 484.
- [44] Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*.
- [45] Aman Sinha, Hongseok Namkoong, and John Duchi. 2018. Certifying some distributional robustness with principled adversarial training. In *International Conference on Learning Representations (ICLR)*.
- [46] Chuanbiao Song, Kun He, Liwei Wang, and John E Hopcroft. 2019. Improving the Generalization of Adversarial Training with Domain Adaptation. In *International Conference on Learning Representations (ICLR)*.
- [47] Congzheng Song, Thomas Ristenpart, and Vitaly Shmatikov. 2017. Machine learning models that remember too much. In *ACM Conference on Computer and Communications Security (CCS)*. 587–601.
- [48] Liwei Song, Reza Shokri, and Prateek Mittal. 2019. Membership inference attacks against adversarially robust deep learning models. In *Deep Learning and Security Workshop (DLS)*.
- [49] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research* 15, 1 (2014), 1929–1958.
- [50] Jacob Steinhardt, Pang Wei W Koh, and Percy S Liang. 2017. Certified defenses for data poisoning attacks. In *Conference on Neural Information Processing Systems (NeurIPS)*. 3517–3529.
- [51] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2014. Intriguing properties of neural networks. In *International Conference on Learning Representations (ICLR)*.
- [52] Antonio Torralba, Alexei A Efros, et al. 2011. Unbiased look at dataset bias. In *IEEE conference on computer vision and pattern recognition (CVPR)*.
- [53] Florian Tramèr, Fan Zhang, Ari Juels, Michael K Reiter, and Thomas Ristenpart. 2016. Stealing Machine Learning Models via Prediction APIs. In *USENIX Security Symposium*. 601–618.
- [54] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. 2019. Robustness may be at odds with accuracy. In *International Conference on Learning Representations (ICLR)*.
- [55] Binghui Wang and Neil Zhenqiang Gong. 2018. Stealing hyperparameters in machine learning. In *IEEE Symposium on Security and Privacy (S&P)*.
- [56] Eric Wong and Zico Kolter. 2018. Provable defenses against adversarial examples via the convex outer adversarial polytope. In *International Conference on Machine Learning (ICML)*. 5283–5292.
- [57] Eric Wong, Frank Schmidt, Jan Hendrik Metzen, and J Zico Kolter. 2018. Scaling provable adversarial defenses. In *Conference on Neural Information Processing Systems (NeurIPS)*.
- [58] Han Xiao, Kashif Rasul, and Roland Vollgraf. 2017. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747* (2017).
- [59] Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. 2018. Privacy risk in machine learning: Analyzing the connection to overfitting. In *IEEE Computer Security Foundations Symposium (CSF)*. 268–282.
- [60] Sergey Zagoruyko and Nikos Komodakis. 2016. Wide Residual Networks. In *Proceedings of the British Machine Vision Conference (BMVC)*.
- [61] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P Xing, Laurent El Ghaoui, and Michael I Jordan. 2019. Theoretically principled trade-off between robustness and accuracy. In *International Conference on Machine Learning (ICML)*.

A FINE-GRAINED ANALYSIS OF PREDICTION LOSS OF THE ROBUST CIFAR10 CLASSIFIER

Here, we perform a fine-grained analysis of Figure 1a by separately visualizing the prediction loss distributions for test points which are secure and test points which are insecure. A point is deemed as secure when it is correctly classified by the model for all adversarial perturbations within the constraint \mathcal{B}_ϵ .

Note that only a few training points were not secure, so we focused our fine-grained analysis on the test set. Figure 6 shows that **insecure test inputs are very likely to have large prediction loss (low confidence value)**. Our membership inference strategies directly use the confidence to determine membership, so the

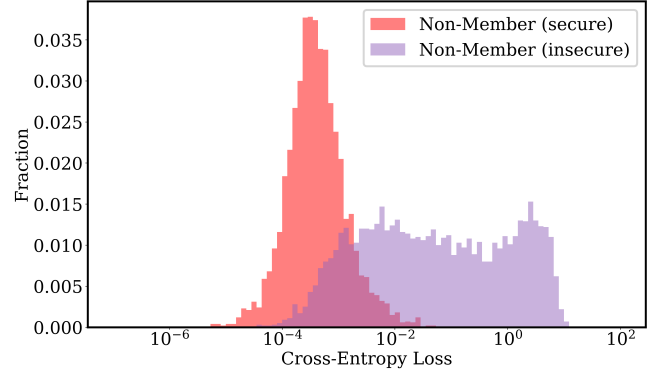


Figure 6: Histogram of the robust CIFAR10 classifier [30] prediction loss values of both secure and insecure test examples. An example is called “secure” when it is correctly classified by the model for all adversarial perturbations within the constraint \mathcal{B}_ϵ .

privacy risk has a strong relationship with robustness generalization, even when we purely rely on the prediction confidence of the benign unmodified input.

B MODEL ARCHITECTURE

We present the detailed neural network architectures in Table 12.

Table 12: Model architectures used on Yale Face, Fashion-MNIST and CIFAR10 datasets. “Conv $c \times w \times h + s$ ” represents a 2D convolution layer with c output channels, kernel size of $w \times h$, and a stride of s , “Res $c-n$ ” corresponds to n residual units [20] with c output channels, and “FC n ” is a fully connect layer with n neurons. All layers except the last FC layer are followed by ReLU activations, and the final prediction is obtained by applying the softmax function on last FC layer.

Yale Face	Fashion-MNIST	CIFAR10
Conv 8 $3 \times 3 + 1$	Conv 256 $3 \times 3 + 1$	Conv 16 $3 \times 3 + 1$
Conv 8 $3 \times 3 + 2$	Conv 256 $3 \times 3 + 1$	Res 160-5
Conv 16 $3 \times 3 + 1$	Conv 256 $3 \times 3 + 2$	Res 320-5
Conv 16 $3 \times 3 + 2$	Conv 512 $3 \times 3 + 1$	Res 640-5
Conv 32 $3 \times 3 + 1$	Conv 512 $3 \times 3 + 1$	FC 200
Conv 32 $3 \times 3 + 2$	Conv 512 $3 \times 3 + 2$	FC 10
Conv 64 $3 \times 3 + 1$	FC 200	
Conv 64 $3 \times 3 + 2$	FC 10	
FC 200		
FC 38		

C EXPERIMENT MODIFICATIONS FOR THE DUALITY-BASED VERIFIABLE DEFENSE

When dealing with the duality-based verifiable defense method [56, 57] (implemented in PyTorch), we find that the convolution with a kernel size 3×3 and a stride of 2 as described in Section 4 is not applicable. The defense method works by backpropagating the neural network to express the dual problem, while the convolution

with a kernel size 3×3 and a stride of 2 prohibits their backpropagation analysis as the computation of output size is not divisible by 2 (PyTorch uses a round down operation).

For the same reason, we also need to change the dimension of the Yale Face input to be 192×192 by adding zero paddings. In our experiments, we have validated that the natural models trained with the above modifications have similar accuracy and privacy performance as the natural models without modifications reported in Table 8 and Table 9.