Reconciling Utility and Membership Privacy via Knowledge Distillation

Virat Shejwalkar
University of Massachusetts Amherst
vshejwalkar@cs.umass.edu

Amir Houmansadr
University of Massachusetts Amherst
amir@cs.umass.edu

***Abstract*—Large capacity machine learning models are prone to membership inference attacks in which an adversary aims to infer whether a particular data sample is a member of the target model's training dataset. Such membership inferences can lead to serious privacy violations as machine learning models are often trained using privacy-sensitive data such as medical records and controversial user opinions. Recently, defenses against membership inference attacks are developed, in particular, based on differential privacy and adversarial regularization; unfortunately, such defenses highly impact the classification accuracy of the underlying machine learning models.**

In this work, we present a new defense against membership inference attacks that preserves the utility of the target machine learning models significantly better than prior defenses. Our defense, called *distillation for membership privacy* (DMP), leverages knowledge distillation to train machine learning models with membership privacy. We analyze the key requirements for membership privacy and provide a novel criterion to select data used for knowledge transfer, in order to improve membership privacy of the final models. DMP works effectively against the attackers with either a whitebox or blackbox access to the target model.

We evaluate DMP's performance through extensive experiments on different deep neural networks and using various benchmark datasets. We show that DMP provides a significantly better tradeoff between inference resistance and classification performance than state-of-the-art membership inference defenses. For instance, a DMP-trained DenseNet provides a classification accuracy of 65.3% for a 54.4% blackbox membership inference attack accuracy, while an adversarially regularized DenseNet provides a classification accuracy of only 53.7% for a (much worse) 68.7% blackbox membership inference attack accuracy.

I. Introduction

The recent breakthroughs in deep learning and computing infrastructure, and the availability of large amounts of data have facilitated the adoption of machine learning (ML) in various domains ranging from recommendation systems to critical health-care management. The quality and quantity of data plays an instrumental role in the performance of machine learning models. Many companies providing ML-as-a-Service computing platforms (e.g., Google API, Amazon AWS, etc.) enable novice data owners to train ML models for different applications. Such models are then released either as a prediction API and accessed in a blackbox fashion, or as a set of parameters and accessed in a whitebox fashion.

The data used for training ML models often contains sensitive user information such as clinical records, location traces, personal photos, etc. [12], [13], [54]; therefore, an ML model trained using sensitive data may pose privacy threats to the data owners by leaking the sensitive information. This has been demonstrated through various inference attacks [24], [49], [17], [23], [10], [32], most notably the *membership inference attack* [50] which is the focus of our work. An adversary with a blackbox or whitebox access to the target model can mount the membership inference attack to determine if a given target sample belonged to the training set of the target model or not [35], [29]. The attack performance significantly improves with a whitebox access to the trained models [36]. Membership inference attacks are able to distinguish the members from non-members by *learning* the behavior of the target model on member versus non-member inputs. They use different features of the target model for this classification, including the entropy of the predictions [50], the prediction loss, and gradients of the input loss with respect to the model parameters [36]. Membership inference attacks are particularly more effective against large neural networks [50], [29], [17], [18], [47] because such models can better memorize their training samples.

Recent work has investigated several defenses against membership inference attacks. *Differential privacy* (DP) based defenses add noise to learning objective or outputs of the model [39], [7], [9], [11], [41]. These defenses aim to provide the worst case privacy guarantees, i.e., privacy to any dataset, and therefore, add very large amount of noise which significantly hurts the utility of the trained models [41], [39]. Furthermore, DP defenses are shown to provide unacceptable privacy and utility tradeoffs [25], therefore questioning their use in practice. Sablayrolles et al. [46] showed that membership privacy is a weaker notion of privacy than DP and argued that membership inference resistance of models improves with generalization. On the same lines, Nasr et al. [35] propose *adversarial regularization* targeted to defeat membership leakage by improving the target model's generalization. However, as we demonstrate, the adversarial regularization and other state-of-the-art regularizations, including label smoothing and confidence penalty, fail to provide acceptable membership privacy-utility tradeoffs. In summary, the *existing defenses against membership inference attacks offer poor tradeoffs between model utility and membership privacy.*

Our contributions. In this work, we demonstrate a defense against membership inference that significantly improves the tradeoffs between privacy and utility compared to the existing defenses. That is, for a given degree of membership inference resistance, our defense provides significantly higher classification performances for the target model, when compared to the existing defenses. Our defense mechanism, called *distillation for membership privacy* (DMP), leverages *knowledge distillation* [6], [20]. Distillation, a *knowledge transfer* technique, is primarily used to reduce the sizes of trained models to make them deployable on resource-constrained devices such as mobile phones. Our intuition behind the use of knowledge transfer for membership privacy is the absence of direct access of the final trained model to the privacy-sensitive training data during training.

The objective of our defense, DMP, is to train a machine learning model with acceptable tradeoffs,[1] which the current defenses do not offer. The first stage of DMP is the *pre-distillation* phase in which DMP trains an *unprotected* model using the sensitive (private) training data without any privacy guarantees. Next, during the *distillation* phase, DMP transfers the knowledge of the unprotected model into predictions of a non-private reference data drawn from the same distribution as the sensitive dataset. The final stage of DMP is the *post-distillation* phase which outputs a *protected* model trained on the reference data and its predictions. In conventional distillation, the capacity of the protected model is smaller than that of the unprotected model. However, in DMP we do not impose this restriction and simply use the same architecture for both the models.

Privacy protection in DMP: DMP is a *meta-regularizer* in that it is agnostic to the properties of the unprotected model. A naive use of distillation may not improve membership privacy, because conventional knowledge transfer techniques [6], [20], [39] do not follow any specific properties in selecting the reference data. For instance, Hinton et al. [20] use a subset of the training data for distillation, which provides good accuracy but *increases* overfitting. This implies that using arbitrary reference data for distillation *cannot* provide the requisite inference resistance, as we show later (Figure 4). Therefore, a major challenge for our DMP technique is selecting the reference data that amplifies its privacy protection. We address this by providing a novel criterion for selecting DMP's reference data. We assume a posterior distribution of the parameters learned on a given training data [46] and argue that to protect membership privacy of a member of the private training data, the output distributions of the models trained with and without the sample should be statistically close. We show the intractability of the corresponding objective and provide a more practical approach to select the reference data based on the distribution entropy of predictions of given data. We also show the effectiveness of the final reference data selection objective in improving the inference resistance. Additionally, we show that using high temperatures in softmax layer of the *unprotected model* and/or smaller reference data sizes reduces sensitive membership information leaked to the *protected model* and strengthens its membership inference resistance.

Utility preservation in DMP: To provide superior tradeoffs, DMP's objective is to preserve the utility of the target model while providing membership inference resistance. To do so, DMP trains the protected model using Kullback-Leibler divergence as the loss function. This forces the protected model to imitate the behavior of the original, unprotected model on the reference data, therefore, strongly preserves its classification accuracy [57], [6], [20].

Empirical validation: We evaluate DMP extensively on several benchmark classification tasks and show that *DMP significantly outperforms existing defenses in terms of the tradeoffs*. For example, for CIFAR-100 classification with DenseNet (L=100, k=12), training, test, and inference accuracies of the DMP-trained model are 66.7%, 63.1%, and 53%, respectively, which are significantly better than the adversarially regularized model (77.8%, 58.4%, and 61.9%, respectively) and the unprotected model (99%, 65.2%, and, 72.2%, respectively), in terms of the tradeoffs. For a deeper DenseNet (L=140, k=19), to reduce the generalization error by 26% over the unprotected model, DMP incurs 0.2% accuracy loss while an adversarially regularized model incurs a 27% accuracy loss. We also show that DMP achieves better tradeoffs than the state-of-the-art regularization techniques.

Note that, although DMP does not provide worst case guarantees as DP, our experiments show that DMP and DP trained models with the same generalization error exhibit equal susceptibility to membership inference. Moreover, not just the average but also the worse case susceptibility is equivalent. However, DMP trained models have superior utility for a given empirical membership inference risk. For instance, test and inference accuracies of AlexNet trained on CIFAR-10 using DMP are 65% and 51.3%, while using DP-SGD are 51.7% and 51.7%. Similar to DP-SGD, our comparison with PATE [41], [39] shows that at low ϵ, PATE produces students with poor utility: at modest $\epsilon = 42.9$, PATE student has 33.9% accuracy compared to 79.6% baseline, while DMP-trained θ_{p} with equivalent membership inference risk has 76.9% accuracy. These findings are similar to that of [25]. This, in conjunction with theoretical results of [46], [60] and their empirical validations by attacks in [50], [35], [36], [60], implies that DMP training produces models with the improved tradeoffs.

II. Preliminaries

A. Machine learning setting

In this work, we focus on supervised learning and classification problems. Let X be a d-dimensional feature space and Y be the $\mathbf{c}$-dimensional output space, where $\mathbf{c}$ denotes the total number of prediction classes. The objective of machine learning is to learn a parameter vector θ, which represents a mapping $f_\theta : X \rightarrow Y$. f_θ outputs a $\mathbf{c}$-dimensional vector with each dimension representing the probability of input belonging to the corresponding class. Let $\Pr(\mathbf{X}, \mathbf{Y})$ be the underlying

[1] Here onward tradeoffs imply the tradeoffs between membership inference risk and classification performance, unless stated otherwise.

distribution of all data points in the universe $X \times Y$, where $\mathbf{X}$ and $\mathbf{Y}$ are random variables for the feature vectors and the classes of data, respectively. Consider $\ell(\theta(\mathbf{x}), y)$ to be a loss function measuring the deviation of the model's prediction on input $\mathbf{x}$, and the actual label of $\mathbf{x}$, i.e., y. The objective of a machine learning model, θ, is to minimize the expected loss over all $(\mathbf{x}, y)$:

$$L(f_\theta) = \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim \Pr(\mathbf{X},\mathbf{Y})}[\ell(f_\theta(\mathbf{x}), y)]$$

This minimization is intractable because it is over the entire data population. Therefore, in practice, the loss functions is minimized over a finite set of training samples drawn from the population, i.e., $D_{\text{tr}} \subset (X, Y)$. The corresponding optimization problem is:

$$L_{D_{\text{tr}}}(f_\theta) = \frac{1}{|D_{\text{tr}}|} \sum_{(\mathbf{x},y)\in D_{\text{tr}}} \ell(f_\theta(\mathbf{x}), y) \tag{1}$$

$$\theta^* = \operatorname*{argmin}_{\theta} \;\; L_{D_{\text{tr}}}(f_\theta) + \lambda R(\theta) \tag{2}$$

where θ is the parameter vector, also called model parameters, of the mapping f_θ, $R(\theta)$ is a regularizer whose goal is to generalize the model, and λ is a hyperparameter.

B. Knowledge distillation

Knowledge distillation was introduced by Hinton et al. [20] with the purpose of *model compression*.To perform distillation, a large network, θ, is trained on some data, D. Then another data, D', is drawn from the same distribution as D, and the prediction vectors (called *soft labels*) of D' are obtained by querying θ. Finally, the soft labels and features of D' are used to train another neural network θ' with a smaller (compressed) size.

Usually, machine learning models use a *softmax layer* after their output layer, to produce probabilities over the classes. The functionality of the softmax layer is given by

$$F(X) = \left[\frac{e^{z_l(X)/T}}{\sum_{i=0}^{\mathbf{c}-1} e^{z_i(X)/T}}\right]_{l\in 0\ldots\mathbf{c}-1} \tag{3}$$

where the $z(X)$ vector denotes the $\mathbf{c}$-dimensional output of the last layer of neural network, and T is a parameter of softmax called *temperature*. To train the distilled network, θ', either of $z(X)$ or $F(X)$ can be used.

If the true labels, called *hard labels*, are available for features of D', the dataset is called *labeled* otherwise called *unlabeled*. If D' is labeled, the distilled network, θ', can be trained with both hard and soft labels. As shown in other domains [40], [57], [6], θ' achieves accuracy very close to or even better (in case of labeled D') than that of θ. We note that in knowledge distillation the size of second neural network is smaller than that of the network whose knowledge is distilled in D', however when these two networks have the same size the learning process is also known as *knowledge transfer*. Without loss of generality use the term 'knowledge distillation' in our work.

C. Membership inference attack setting

Membership inference is a serious privacy concern for machine learning models [19], [33], [50], [30], [46], [29]. Consider a machine learning model θ and a data sample $(\mathbf{x}, y)$. The goal of a membership inference adversary is to infer whether $(\mathbf{x}, y)$ belongs to the dataset used to train the model θ. The membership inference attack exploits the memorization of training data by large neural networks by inspecting various features of the target trained model. Therefore, the standard approach for the membership inference adversary is to train an inference model, h, whose goal is to classify data samples into members and non-members.

Let θ be the target model and $h : \mathcal{F}(X, Y, \theta) \to [0, 1]$ be the inference model. Given a data sample $(\mathbf{x}, y)$, the inference adversary computes $\mathcal{F}(X, Y, \theta)$, which is a combination of different features of θ related to $(\mathbf{x}, y)$. For instance, θ's prediction on $(\mathbf{x}, y)$ [50], [35], [33], θ's loss on $(\mathbf{x}, y)$ [46], the gradients of the loss [36], [33], etc. Based on this input feature vector $\mathcal{F}(X, Y, \theta)$, h outputs the probability that $(\mathbf{x}, y)$ is a member of θ's training set. Let $\Pr_D(\mathbf{X}, \mathbf{Y})$ and $\Pr_{\setminus D}(\mathbf{X}, \mathbf{Y})$ be the conditional probabilities of the members and non-members, respectively. For the above setting, the expected gain of the inference model can be computed as:

$$\begin{aligned} G_\theta(h) = 0.5 \times &\mathop{\mathbb{E}}_{(\mathbf{x},y)\sim \Pr_D(\mathbf{X},\mathbf{Y})}[\log(h(\mathcal{F}(\mathbf{x}, y, \theta)))] \\ + 0.5 \times &\mathop{\mathbb{E}}_{(\mathbf{x},y)\sim \Pr_{\setminus D}(\mathbf{X},\mathbf{Y})}[\log(1 - h(\mathcal{F}(\mathbf{x}, y, \theta)))] \end{aligned} \tag{4}$$

In practice [19], [33], [50], [30], [36], [35], the inference adversary knows only a (small) subset of the members D, i.e., she only knows $D^A \subset D$ and has access to enough non-members D'^A required to train h. Therefore, the adversary computes an empirical gain as:

$$\begin{aligned} G_{\theta,D^A,D'^A}(h) = &\frac{1}{|D^A|} \sum_{(\mathbf{x},y)\in D^A} [\log(h(\mathcal{F}(\mathbf{x}, y, \theta)))] \\ + &\frac{1}{|D'^A|} \sum_{(\mathbf{x},y)\in D'^A} [\log(1 - h(\mathcal{F}(\mathbf{x}, y, \theta)))] \end{aligned} \tag{5}$$

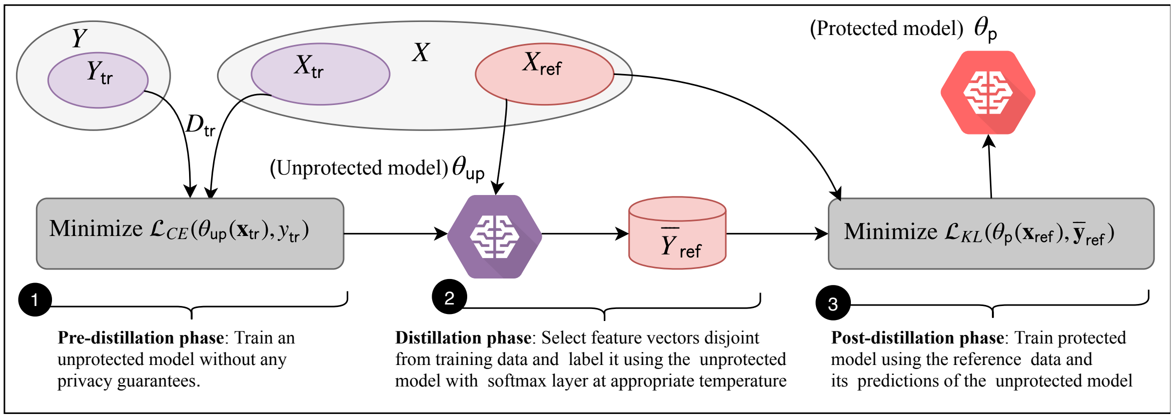
which is used to get the inference model:

$$h = \operatorname*{argmax}_{h} \;\; G_{\theta,D^A,D'^A}(h) \tag{6}$$

In (5), the two summations compute the empirical gain of inference model on the subset of members and non-members that the adversary has. Note that, *the empirical gain decreases if the features, $\mathcal{F}(\mathbf{x}, y, \theta)$, on the members and non-members are indistinguishable*.

III. Introducing Distillation For Membership Privacy (DMP)

We present Distillation For Membership Privacy (DMP), whose goal is to train ML models that are resilient to membership inference attacks. Our design of DMP is motivated by the poor privacy-utility tradeoffs provided by existing defenses against membership inference discussed in Section


Y
Y_{tr}
D_{tr}
X
X_{tr}
X_{ref}
(Protected model) θ_{p}
(Unprotected model) θ_{up}
Minimize $\mathcal{L}_{CE}(\theta_{\text{up}}(\mathbf{x}_{\text{tr}}), y_{\text{tr}})$
$\overline{Y}_{\text{ref}}$
Minimize $\mathcal{L}_{KL}(\theta_{\text{p}}(\mathbf{x}_{\text{ref}}), \overline{\mathbf{y}}_{\text{ref}})$
1
2
3

Pre-distillation phase: Train an unprotected model without any privacy guarantees.

Distillation phase: Select feature vectors disjoint from training data and label it using the unprotected model with softmax layer at appropriate temperature

Post-distillation phase: Train protected model using the reference data and its predictions of the unprotected model

Figure 1: The three stages of Distillation for Membership Privacy technique.

VII. DMP leverages knowledge distillation [20], introduced in Section II-B, to train high-utility ML models resistant to membership inference.

A. Notations

We start by introducing the notations used throughout the paper. We consider the data universe $(X \times Y)$ and its true underlying distribution $\Pr(\mathbf{X}, \mathbf{Y})$ as described in Section II-A. A *labeled* dataset consists of pairs of feature vectors and labels, i.e., it is a subset of $(X \times Y)$. On the other hand, an *unlabeled* dataset consists of only feature vectors, i.e., it is a subset of X.

We use $D_{\text{tr}} \subset (X \times Y)$ to refer to a *private* training dataset. We call an ML model trained using a private dataset D_{tr} as *unprotected model*, denoted by θ_{up}, due to its high susceptibility to membership inference attacks. On the other hand, we call an ML model *protected* and denote it by θ_{p} if it is trained in a way that resists membership inference attacks. As we described later, DMP trains protected models using a *non-sensitive reference dataset*, which is sampled from X but is disjoint from the private training data, D_{tr}. X_{ref} represents an unlabeled reference dataset, and $\overline{Y}_{\text{ref}}$ represents the soft labels (prediction vectors) of the θ_{up} on X_{ref}. Unless stated otherwise, we assume that all models use a softmax layer and θ^T implies that T is the temperature of the softmax layer of θ.

B. Main intuition of DMP

Sablayrolles et al. [46] show that a model θ trained on a sample z_1 provides (ϵ, δ) membership privacy to z_1 if the expected loss of the models not trained on z_1 is ϵ-close to the loss of θ on z_1, with probability at least $1-\delta$. They assume the posterior distribution of the parameters learned using a given training data $D = \{z_1, .., z_n\}$ to be:

$$\mathbb{P}(\theta|z_1, ..., z_n) \propto \exp(\sum_{i=1}^{n} \ell(\theta, z_i)) \tag{7}$$

Consider a neighboring dataset $D' = \{z_1, .., z'_j, .., z_n\}$ of D; a neighboring dataset is obtained by modifying at most one sample of D [14]. Using the assumption above, to provide membership privacy to z_j, the log of the ratio of probabilities of obtaining the same θ from D and D' should be bounded, i.e., the following ratio should be bounded:

$$\log\left|\frac{\mathbb{P}(\theta|D)}{\mathbb{P}(\theta|D')}\right| = |\ell(\theta, z_j) - \ell(\theta, z'_j)| \tag{8}$$

(8) implies that, if θ was indeed trained on z_j, then to provide membership privacy to z_j, the loss of θ on z_j should be same as on any non-member sample z'_j.

We build our defense on this intuition, i.e., we aim to train a model with statistically close losses on the members and non-members. To achieve this, we leverage the knowledge transfer paradigm, because, it restricts the direct access of θ_{p} to the private training data, and prevents leakage of sensitive membership information to θ_{p}. Furthermore, based on (7) and (8), we derive a condition to choose the reference data such the leakage can be reduced. In the pre-distillation phase, we train an unprotected model, θ_{up}, on D_{tr} without any privacy guarantees. But, unlike the conventional knowledge distillation, to transfer knowledge, *we use X_{ref} for which θ_{up} has low loss*. However, as loss of an unlabeled sample cannot be computed, we use entropy of the sample's output as a proxy for loss of θ_{up} on the sample.

Note that, due to memorization, θ_{up} has lower entropy on the members of D_{tr}. But, due to high dimensionality of the input feature space, there exist samples with low loss/entropy that are far from the members. Intuitively, such samples are easy to classify and none of the members of D_{tr} significantly affect their predictions, and therefore, these predictions do not leak membership information of any particular member. We make this intuition more clear in the following sections. Finally, DMP trains the protected model, θ_{p}, on the reference data predictions. The superior classification performance of DMP is due to training on $\theta_{\text{up}}(X_{\text{ref}})$ using KL-divergence loss which forces DMP-trained models to perfectly match the performance of θ_{up} on the test data.

Algorithm 1 Distillation for Membership Privacy

```
Input: $D_{\text{tr}}$, $X_{\text{ref}}$, $T_{\text{up}}$, $T_{\text{p}}$
Initialize $\theta_{\text{up}}$                                          ▷ Initialization
for $T_{\text{up}}$ epochs do
    Perform SGD with cross-entropy loss:
    $\underset{\theta_{\text{up}}}{\text{argmin}} \quad -\frac{1}{|D_{\text{tr}}|}\sum_{(\mathbf{x},y)\in(D_{\text{tr}})} \mathcal{L}_{\text{CE}}(\theta_{\text{up}}(\mathbf{x}), y)$
end for                                                        ▷ Pre-distillation
$\overline{Y}_{\text{ref}} = \{\overline{\mathbf{y}} = \theta_{\text{up}}(\mathbf{x}) \quad ...\forall\ \mathbf{x} \in X_{\text{ref}}\}$        ▷ Distillation
for $T_{\text{p}}$ epochs do
    Perform SGD to minimize KL divergence loss between $\theta_{\text{p}}(\mathbf{x})$ and $\theta_{\text{up}}(\mathbf{x})$
    $\underset{\theta_{\text{p}}}{\text{argmin}} \quad \frac{1}{|X_{\text{ref}}|}\sum_{\mathbf{x}\in X_{\text{ref}}} \mathcal{L}_{\text{KL}}(\theta_{\text{p}}(\mathbf{x}), \theta_{\text{up}}(\mathbf{x}))$
end for                                                        ▷ Post-distillation
Output: $\theta_{\text{p}}$
```

C. Details of the DMP technique

Here we present the details of three main phases of DMP technique, which Algorithm 1 summarizes.

1) Pre-distillation phase: In this phase, an unprotected model, θ_{up}, is trained on the sensitive, labeled training data, D_{tr}, using standard training techniques and without any privacy enforcement. In particular, we simply use the stochastic gradient descent (SGD) algorithm to train θ_{up} on D_{tr}:

$$\theta_{\text{up}} = \underset{\theta}{\text{argmin}} \quad -\frac{1}{|D_{\text{tr}}|}\sum_{(\mathbf{x},y)\in(D_{\text{tr}})}\sum_{i=0}^{\mathbf{c}-1}\mathbb{I}_{i=y}\ \log(\theta(\mathbf{x})) \tag{9}$$

where $\mathbb{I}_{i=y}$ outputs 1 when i is the true class of the sample $(\mathbf{x}, y)$ and 0 otherwise; $\mathbf{c}$ is the number of classes in the classification task.

2) Distillation phase: In this phase, we first obtain the reference data, X_{ref}, that is used to transfer the knowledge of θ_{up} in θ_{p}. The selection of X_{ref} is important to reduce the membership leakage, which we discuss in detail in Section IV-B. Note that, the unlabeled reference data, X_{ref}, cannot be used directly for any learning. We label X_{ref} using θ_{up} to get $\overline{Y}_{\text{ref}}$, i.e., $\overline{Y}_{\text{ref}} = \theta_{\text{up}}(X_{\text{ref}})$. Note that the last layer of θ_{up} is a softmax layer at temperature T. As shown later in Section IV-C, the temperature parameter should be chosen properly to increase the membership inference resistance. Also, we will show that using the reference data with low entropy predictions, $\overline{Y}_{\text{ref}}$, of the unprotected model is the key enabler of membership privacy. Low entropy predictions are characteristics of the members of D_{tr}, however, non-members with low entropy can also be obtained due to the high dimensional input feature space. We leave investigating the ways to produce such samples, e.g., using generative adversarial networks [22], to future work.

3) Post-distillation phase: In this phase, we train a protected model θ_{p} on $(X_{\text{ref}}, \overline{Y}_{\text{ref}})$ obtained in the distillation phase. The empirical risk for model θ_{p} on a reference sample $(\mathbf{x}, \overline{\mathbf{y}}) \in (X_{\text{ref}}, \overline{Y}_{\text{ref}})$ is defined using the Kullback-Leibler divergence; here, $\overline{\mathbf{y}} = \theta_{\text{up}}(\mathbf{x})$. The final θ_{p} is obtained by solving the empirical risk minimization problem given by (11).

$$\mathcal{L}_{\text{KL}}(\mathbf{x}, \overline{\mathbf{y}}) = \sum_{i=0}^{\mathbf{c}-1}\overline{\mathbf{y}}_i\ \log\Big(\frac{\overline{\mathbf{y}}_i}{\theta(\mathbf{x})_i}\Big) \tag{10}$$

$$\theta_{\text{p}} = \underset{\theta}{\text{argmin}}\ \frac{1}{|X_{\text{ref}}|}\sum_{(\mathbf{x},\overline{\mathbf{y}})\in(X_{\text{ref}},\overline{Y}_{\text{ref}})}\mathcal{L}_{\text{KL}}(\mathbf{x}, \overline{\mathbf{y}}) \tag{11}$$

Note that, (11) is minimized when $\theta_{\text{up}}(\mathbf{x}) = \theta_{\text{p}}(\mathbf{x})$ for all $(\mathbf{x}, \overline{\mathbf{y}}) \in (X_{\text{ref}}, \overline{Y}_{\text{ref}})$. Hence, we expect *$\theta_p$ to perfectly learn the behavior of θ_{up} on the non-member inputs*. Therefore, in theory, the performance of θ_{p} on the test data is close to that of θ_{up}, which is empirically observed in many previous works [20], [6], [45]. Therefore, due to the empirical risk minimization in (11), *the DMP-trained models do not lose classification performance on the test data while preserving membership privacy*. Next, we formalize and validate the properties of the reference data required for stronger membership privacy.

IV. Fine-Tuning The DMP Technique

The membership inference resistance of DMP trained models significantly depends on the reference data used for knowledge transfer. Therefore, in this section, we analyze the properties of the reference data that should be used to improve the efficacy of DMP. Specifically, using the assumption in (7), we derive and validate an empirical approach to choose the reference data in order to achieve strong membership inference resistance via DMP training.

A. Objective to select reference data

Consider the DMP training described in Algorithm 1. Next, consider two sets of training data D_{tr} and D'_{tr} such that $D'_{\text{tr}} \leftarrow D_{\text{tr}}\backslash z$, and a reference data X_{ref}. Then, the log of the ratio of the posterior probabilities of learning the exact same parameters θ_{p} using Algorithm 1 is given by (12), which we denote by $\mathcal{R}$.

$$\mathcal{R} = \Big|\log\Big(\frac{\mathbb{P}(\theta_{\text{p}}|D_{\text{tr}}, X_{\text{ref}})}{\mathbb{P}(\theta_{\text{p}}|D'_{\text{tr}}, X_{\text{ref}})}\Big)\Big| \tag{12}$$

Note that, $\mathcal{R}$ is an extension of (8) to the setting of DMP, where the final model is trained via the knowledge transferred using $(X_{\text{ref}}, \theta_{\text{up}}(X_{\text{ref}}))$, instead of directly training on D_{tr}. DMP can achieve stronger membership inference resistance for the member z by reducing $\mathcal{R}$. Note that, at the first glance, this condition is similar to that imposed by the differential privacy. However, unlike the differential privacy, this condition is concerned with the privacy only of the *given* training data, D_{tr}.

The predictions on the reference data are the source of the membership information leaked to θ_{p}. Therefore, the main aim of our analysis is to derive a practical approach of choosing reference samples such that $\mathcal{R}$ is reduced. Next, we modify $\mathcal{R}$ as:

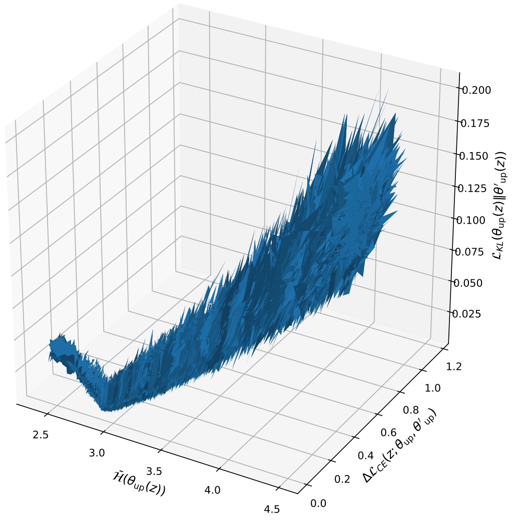


Figure 2: Empirical validation of simplification of (15) to (16): Increase in $\Delta\mathcal{L}_{\text{CE}}$ increases $\Delta\mathcal{L}_{\text{KL}}$, and that of (15) to (20): Increase in $\mathcal{H}(\theta_{\text{up}}(z))$ increases $\Delta\mathcal{L}_{\text{KL}}$.

$$\mathcal{R} = \Big| - \frac{1}{T} \sum_{\mathbf{x}\in X_{\text{ref}}} \mathcal{L}_{\text{KL}}((\mathbf{x}, \theta_{\text{up}}(\mathbf{x})); \theta_{\text{p}}) - \mathcal{L}_{\text{KL}}((\mathbf{x}, \theta'_{\text{up}}(\mathbf{x})); \theta_{\text{p}}) \Big| \tag{13}$$

$$\leq \frac{1}{T} \sum_{\mathbf{x}\in X_{\text{ref}}} \Big| \mathcal{L}_{\text{KL}}(\theta_{\text{up}}(\mathbf{x}) \| \theta_{\text{p}}(\mathbf{x})) - \mathcal{L}_{\text{KL}}(\theta'_{\text{up}}(\mathbf{x}) \| \theta_{\text{p}}(\mathbf{x})) \Big| \tag{14}$$

where θ_{up} and θ'_{up} are trained on D_{tr} and D'_{tr}, respectively. Note that, (13) holds due to the assumption in (7) and because DMP training minimizes KL-divergence between predictions of θ_{p} and θ_{up} on the reference data. (14) follows from (13) because $|a + b| \leq |a| + |b|$. Therefore, to minimize (8), (14) should be minimized. Hence, for better membership privacy, the reference data should be chosen to minimize the objective formulated in (15).

$$X^*_{\text{ref}} = \operatorname*{argmin}_{X_{\text{ref}}\in X} \Bigg(\frac{1}{T} \sum_{\mathbf{x}\in X_{\text{ref}}} \Big| \mathcal{L}_{\text{KL}}(\theta_{\text{up}}(\mathbf{x}) \| \theta_{\text{p}}(\mathbf{x})) - \mathcal{L}_{\text{KL}}(\theta'_{\text{up}}(\mathbf{x}) \| \theta_{\text{p}}(\mathbf{x})) \Big| \Bigg) \tag{15}$$

The objective in (15) minimizes when $\theta_{\text{up}}(\mathbf{x}) = \theta'_{\text{up}}(\mathbf{x})$, and is quite intuitive: It implies that, to provide strong membership privacy to z, the chosen reference data samples should be such that *the distributions of outputs of θ_{up} and θ'_{up} on the reference data should not be affected by the presence of z.*

B. An empirical approach to select the reference data

Solving (15) involves impractical repetitive training of protected and unprotected models. To avoid such training, in this section, we give a practical approach to choose the reference data and justify its utility.

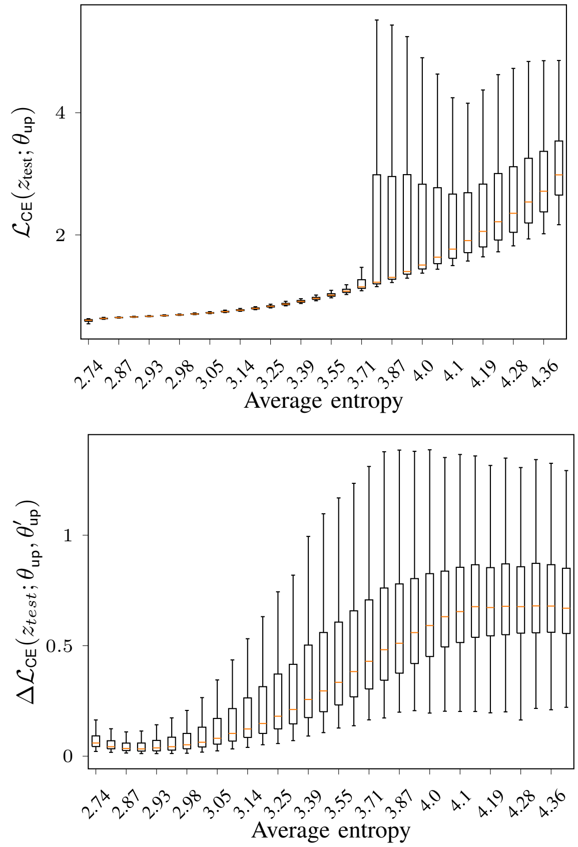


Figure 3: Empirical validation of the reductions: (16) → (19)→ (20). With increase in entropy of reference samples, the cross-entropy loss and difference in cross-entropy also increase.

As noted above, (15) implies that the output distributions of θ_{up} and θ'_{up} on the reference data should be statistically close. Therefore, to simplify the analysis, instead of KL divergence loss, we use the closely related *cross-entropy loss* and simplify (15) as follows:

$$X^*_{\text{ref}} = \operatorname*{argmin}_{X_{\text{ref}}\in X} \frac{1}{T} \sum_{\substack{(\mathbf{x},y)\\ \in(X_{\text{ref}},Y_{\text{ref}})}} \Big| \mathcal{L}_{\text{CE}}((\mathbf{x}, y); \theta'_{\text{up}}) - \mathcal{L}_{\text{CE}}((\mathbf{x}, y); \theta_{\text{up}}) \Big| \tag{16}$$

where $\mathcal{L}_{\text{CE}}$ is cross-entropy loss; for clarity of presentation, here onward, we denote $\mathcal{L}_{\text{CE}}$ by $\mathcal{L}$. We assume for the time being that $Y_{\text{ref}} \in Y$ are the true labels for X_{ref}. To understand this, note that DMP minimizes $\mathcal{L}_{\text{KL}}(\theta_{\text{up}}(\mathbf{x}) \| \theta_{\text{p}}(\mathbf{x}))$, hence, without loss of generality, we assume $\theta_{\text{p}}(\mathbf{x})$ tends to $\theta_{\text{up}}(\mathbf{x})$. Then, (16) simply becomes the KL-divergence between output distributions of θ_{up} and θ'_{up}. Figure 2 shows that, for any given reference sample, as the difference in cross-entropy losses $\Delta\mathcal{L}$ increases, the corresponding KL-divergence losses also increase, which validates (15) → (16).

Next, to avoid repetitive training, we simplify the term for each sample in (16) using the results by Koh et al. [27]. More specifically, [27] proposes a linear approximation to the difference in losses of a pair of models trained with and without a sample in a training data. If θ and θ_{-z} are two models trained with and without sample z, then the difference in loss on some test sample z_{test} is given by [27] as:

$$|\mathcal{L}(z_{\text{test}}, \theta_{-z}) - \mathcal{L}(z_{\text{test}}, \theta)| = |\nabla_\theta \mathcal{L}(z_{\text{test}}, \theta) H_\theta^{-1} \nabla_\theta \mathcal{L}(z, \theta)| \tag{17}$$

where H_θ is the Hessian matrix defined as $H_\theta = \frac{1}{n}\sum_{z\in D_{\text{tr}}} \nabla^2_\theta(z, \theta)$. Substituting (17) in (16) simplifies the objective in (15) to:

$$X^*_{\text{ref}} = \underset{X_{\text{ref}} \in X}{\text{argmin}} \ \frac{1}{T} \sum_{\mathbf{x}_p \in X_{\text{ref}}} |\nabla_\theta \mathcal{L}(z_{\text{p}}, \theta_{\text{up}}) H_\theta^{-1} \nabla_\theta \mathcal{L}(z, \theta_{\text{up}})| \tag{18}$$

Note that, for a given member z, $H_\theta^{-1} \nabla_\theta \mathcal{L}(z, \theta)$ in (18) remains constant and the minimization reduces to minimizing the gradient $\nabla_\theta \mathcal{L}(z_{\text{p}}, \theta_{\text{up}})$. The lower the loss $\mathcal{L}(z_{\text{p}}, \theta_{\text{up}})$, the smaller the gradient $\nabla_\theta \mathcal{L}(z_{\text{p}}, \theta_{\text{up}})$. Therefore objective (18) further simplifies as:

$$X^*_{\text{ref}} = \underset{X_{\text{ref}} \in X}{\text{argmin}} \ \frac{1}{T} \sum_{\mathbf{x}_p \in X_{\text{ref}}} \mathcal{L}_{\text{CE}}(z_{\text{p}}, \theta_{\text{up}}) \tag{19}$$

Note that, it is not possible to solve the objective in (19) as it is, because the loss cannot be computed due to the unavailability of the true labels of X_{ref}. However, as the loss involved in case of DMP is the cross-entropy loss, minimizing the loss is equivalent to minimizing the entropy of prediction $\theta_{\text{up}}(\mathbf{x}_p)$. This gives us the final objective as:

$$X^*_{\text{ref}} = \underset{X_{\text{ref}} \in X}{\text{argmin}} \ \frac{1}{T} \sum_{\mathbf{x}_p \in X_{\text{ref}}} \mathcal{H}(\theta_{\text{up}}(z_{\text{p}})) \tag{20}$$

where, $\mathcal{H}(\mathbf{v}) \triangleq \sum_i -\mathbf{v}_i \log(\mathbf{v}_i)$ is the entropy of $\mathbf{v}$. Based on the reductions from (15) to (19) and then to (20), we hypothesize that, **using the reference data with low entropy predictions of θ_{up} strengthens the membership resistance of θ_{p}, and vice versa.** Next, we empirically validate the reductions: (15) → (19)→ (20). Specifically, we show that reference data samples with lower entropy have lower cross-entropy loss, i.e., (19)→ (20). Then, we show that, the difference between cross-entropy losses of two models θ_{up} and θ'_{up}, trained on neighboring datasets, on a sample increases with the increase in cross-entropy loss of their prediction on the sample, i.e., (16) → (19). This, in combination with the reduction (15) → (16) demonstrated in Figure 2, completes the validation of (15) → (19).

We use Purchase-100 data and randomly pick D_{tr} of size 10k and train θ_{up}. Next, we perform leave-one-out training to train 100 models for 100 samples randomly removed from D_{tr} and denote by $\theta_{\text{up}}^{-z'}$ the model trained on $D_{\text{tr}} \backslash z'$. We then compute the cross-entropy losses $\mathcal{L}_{\text{CE}}$ of all the 101 models for the remaining 187,324 reference data D_{ref}.[2] We then arrange them in the increasing order of prediction entropies, $\mathcal{H}(\theta_{\text{up}}(z)) \forall z \in D_{\text{ref}}$, and divide them in bins of size 10k. We also compute $\Delta\mathcal{L}_{\text{CE}}(z; \theta_{\text{up}}, \theta_{\text{up}}^{-z'}) = |\mathcal{L}_{\text{CE}}(z; \theta_{\text{up}}) - \mathcal{L}_{\text{CE}}(z; \theta_{\text{up}}^{-z'})|$ for all the bins. Figure 3 shows $\mathcal{L}_{\text{CE}}(z; \theta_{\text{up}})$ and $\Delta\mathcal{L}_{\text{CE}}(z; \theta_{\text{up}}, \theta_{\text{up}}^{-z'})$ for all the samples in each bin averaged over 100 pairs of $(\theta_{\text{up}}, \theta_{\text{up}}^{-z'})$; x-axes denote $\mathcal{H}(\theta_{\text{up}}(z))$. It can be clearly seen that *with the increase in entropy of predictions, both $\mathcal{L}_{CE}$* (Figure 3 upper) *and $\Delta\mathcal{L}_{CE}$ increase* (Figure 3 lower). This validates the reductions from (15) to (19) and then to (20).

Note that, the reference samples in the bins with $\mathcal{H}(\theta_{\text{up}}(z)) \in [3.79, 4.1]$ have monotonically increasing median, as expected, but arbitrarily high *variance* for the cross-entropy loss. This is because these bins contain a few very difficult-to-classify samples for which θ_{up} is less accurate but more confident. This will reduce the entropy but increase the loss, as shown in the figure. Similarly, for the same samples, $\Delta\mathcal{L}_{\text{CE}}(z; \theta_{\text{up}}, \theta_{\text{up}}^{-z'})$ will also be higher, because both θ_{up} and $\theta_{\text{up}}^{-z'}$ have high confidence but low accuracy on the samples. In other words, the 100 pairs of $(\theta_{\text{up}}, \theta_{\text{up}}^{-z'})$ may always disagree on the outputs of such samples, which will lead to the higher variance of the difference in the cross-entropy losses as shown in the figure.

Next, we validate the main hypothesis. Figure 4 (lower) shows the decrease in the inference resistance and Figure 4 (upper) shows the increase in the classification performance of θ_{p} with the increase in entropies of the reference data used. The reason for this tradeoff is as follows: The higher entropy predictions contain more useful information [37], [20], which leads to θ_{p} with better classification performance. However, such predictions are also sensitive to the presence of a member in the private training data (as shown by $\mathcal{L}_{\text{KL}}(\theta_{\text{up}}(z_{\text{test}}) \| \theta_{\text{up}}(z_{\text{test}}))$ versus $\bar{\mathcal{H}}$ in Figure 2), and therefore, carry sensitive membership information, which leads to higher membership inference risk due to the final θ_{p}.

C. Hyperparameters of DMP

In this section, we discuss two important hyperparameters of DMP training: size of the reference data and softmax temperature used in unprotected model while generating soft predictions of the reference data. Appropriate settings of the two hyperparameters are important to attain good tradeoffs for the protected model, θ_{p}.

1) Temperature of the softmax layer of θ_{up}: As analyzed above, the lower the KL-divergence between $\theta_{\text{up}}(\mathbf{x})$ and $\theta'_{\text{up}}(\mathbf{x})$, the stronger the membership inference resistance of θ_{p}. This can be improved via appropriately setting the temperature of softmax layer in θ_{up} and θ'_{up}. At higher temperatures, for a fixed $\mathbf{x} \in X_{\text{ref}}$, softmax layer produces softer $\theta_{\text{up}}(\mathbf{x})$ and $\theta'_{\text{up}}(\mathbf{x})$ [20]. The softer the predictions $\theta_{\text{up}}(\mathbf{x})$ and $\theta'_{\text{up}}(\mathbf{x})$, the lower the KL-divergence between them. To understand this, note that $\mathcal{L}_{\text{KL}}(\theta_{\text{up}}(\mathbf{x}) \| \theta'_{\text{up}}(\mathbf{x}))\Big|_{T\to\infty} \to 0$, and the $\mathcal{L}_{\text{KL}}$ increases with reduction in T. Therefore, *once the reference data is fixed*, $\mathcal{R}$ in (12) can be further reduced by setting

[2] To compute losses, we use the true labels of the reference data.

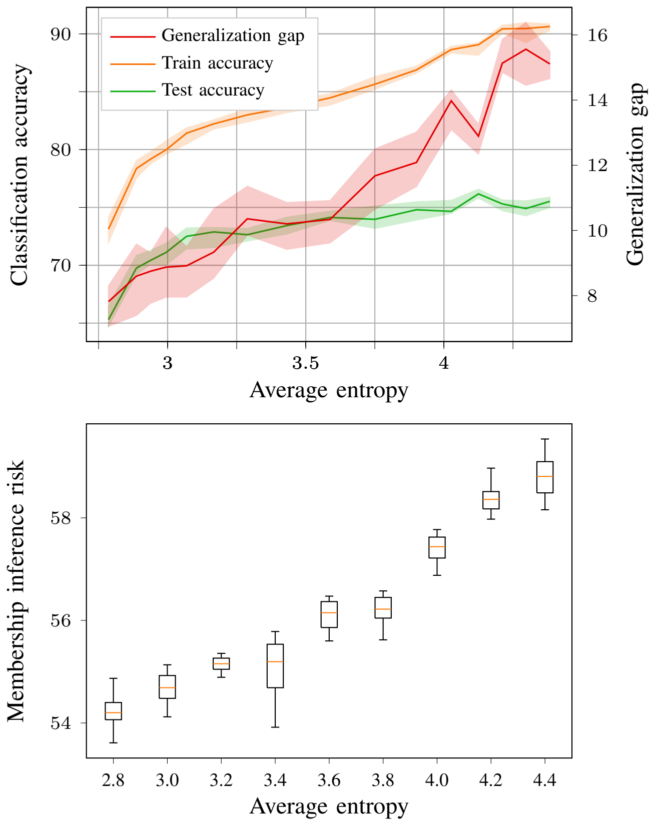


Figure 4: Effect of choice of the reference data on regularization performance of DMP and on membership inference risk due to final θ_{p}. As hypothesized in Section IV-B, with entropy of predictions of θ_{up} on reference data, both the generalization error and membership inference risk also increase.

appropriate high temperatures of softmax layer in θ_{up}, which reduces the difference in (14). Therefore, the higher temperatures improve the membership inference resistance, but at the cost of reductions in the classification performance of θ_{p}. This is demonstrated in Figure 7 and Table VII.

2) Size of reference data: DMP selects the reference data in order to reduce the objective given by (14) and consequently $\mathcal{R}$ in (12). From (14), it is clear that the larger the size of reference data, the higher the value of the objective. Therefore, with increase in reference data, membership privacy due to DMP decreases, if all the other parameters of training are kept constant. However, similar to the softmax layer temperature, size of the reference poses a tradeoff: Although the smaller size of reference data tightens the bound on $\mathcal{R}$ and improves membership privacy, it reduces the classification performance due to the reduced knowledge transferred. We demonstrate this tradeoff in Figure 6. Hence, size of the reference data should be chosen to meet the desired tradeoffs.

V. Experimental Setup

A. Datasets

CIFAR-100. CIFAR-100 is a popular benchmark dataset used to evaluate image recognition algorithms [28]. It contains 60,000 color (RGB) images (50000 for training and 10000 for testing), each of 32 × 32 pixels. The images are clustered into 100 classes based on objects in the images and each class has 500 training and 100 test images.

CIFAR-10. CIFAR-10 has 60,000 color (RGB) images (50,000 for training and 10,000 for testing), each of 32 × 32 pixels. The images are clustered into 10 classes based on the objects in the images and each class has 5,000 training and 1,000 test images. In DMP, the protected models learn on the predictions of unprotected model, which contain more useful information with larger number of classes. Therefore, we use this dataset to assess the efficacy of DMP when the number of classes is small. Due to insufficient amount of data to choose the reference data from according to the final objective in (20), for both of the CIFAR datasets, we use all the 25,000 data disjoint from the private training data for knowledge transfer.

Purchase-100. The Purchase-100 dataset contains the shopping records of several thousand online customers, extracted during Kaggle's "acquire valued shopper" challenge [1]. Each record in the dataset is the shopping history of a single customer. The dataset contains 600 different products, and each user has a binary record which indicates whether she has bought each of the products (a total of 197,324 data records). The records are clustered into 100 classes based on the similarity of the purchases, and the objective is to identify the class of each user's purchases. We use 10000 reference samples selected based on the tradeoffs shown in Figure 4.

Table I: Data sizes used in DMP training. D_{tr} and D_{ref} are the private training and reference data, respectively, and are disjoint. D, D' data are the adversary's knowledge of the members and non-members of D_{tr}. Here, D' and D_{ref} are disjoint.

Dataset	DMP training		Attack training	
	$\lvert D_{\mathsf{tr}}\rvert$	$\lvert D_{\mathsf{ref}}\rvert$	$\lvert D\rvert$	$\lvert D'\rvert$
Purchase-100	10000	10000	5000	5000
CIFAR-100	25000	25000	12500	8000
CIFAR-10	25000	25000	12500	8000

Table II: Temperature of the softmax layers for the different combinations of dataset and network architecture used to produce the results in Table III.

Combination acronym	Dataset	Architecture	$\lvert\theta\rvert$	T
P-FC	Purchase-100	Fully Connected	1.32M	1.0
C100-A	CIFAR-100	AlexNet	2.47M	4.0
C100-D12		DenseNet12	0.77M	4.0
C100-D19		DenseNet19	25.6M	1.0
C10-A	CIFAR-10	AlexNet	2.47M	1.0

B. Target model architectures

Unlike conventional distillation [20], DMP uses the same architecture for both unprotected and protected models. The details of the architectures for all the datasets is given in Table II. For Purchase-100, the fully connected network has hidden layers of sizes $\{1024, 512, 256, 128\}$. For CIFAR-100, we choose two DenseNet models to assess the efficacy of DMP for two models with equivalent performance, but significantly different capacities. In Table II, DenseNet12 corresponds to DenseNet-BC (L=100, k=12) and DenseNet19 corresponds to DenseNet-BC (L=190, k=40). For the comparison with PATE using CIFAR-10, we use the generator and discriminator architectures used in [48]. We measure the training (A_{train}) and test (A_{test}) accuracy of these models as the percentage of the training and test data for which the models produce correct labels. The generalization error (E_{gen}) is measured as the difference of training and test accuracy.

C. Membership inference attack model architectures

We use the state-of-the-art membership inference attack model proposed by Nasr et al. [36] to evaluate the strength of DMP and compare it with the other defenses. For a given input, we use its feature vector, label, and cross-entropy loss of the target model's prediction as the features for the blackbox membership inference. In addition, for the whitebox membership inference, we also use gradients of the loss with respect to last two layers of the target model and outputs of the last two layers of the target model as the features. Following previous works [50], [35], [36], we measure the whitebox (A_{wb}) and blackbox (A_{bb}) membership inference risks as the accuracy of the corresponding attack models. Attack model outputs *member* or *non-member* for a given record, therefore the attack accuracy is measured as the percentage of unknown test data for which the attack model correctly predicts membership. We use the same number of members and non-members in the test data.

VI. Experiments

Next, we present our evaluation of DMP. We implement DMP using PyTorch [2].

A. Comparison with regularization techniques

Membership inference can be prevented to a large extent, although not completely, by regularizing models, i.e., reducing the gap between train and test data accuracy of models [35], [50], [29]. Hence, we compare DMP with several state-of-the-art regularization schemes.

1) Comparison with adversarial regularization: In Table III, we compare the models with the best tradeoffs and the models with equivalent generalization errors, trained using DMP and adversarial regularization. The corresponding unprotected model baselines are shown in 'No defense' column. Table II describes the details of acronyms used for combinations of dataset and models. E_{gen} is generalization error, A_{test} is test accuracy of the target ML model, and A_{wb} and A_{bb} are whitebox and blackbox membership inference risks, respectively. The goal of an effective defense mechanism is to reduce E_{gen}, A_{wb}, and A_{bb} while keeping A_{test} high. It is clear that due to high E_{gen}, the unprotected models are highly susceptible to blackbox and whitebox membership inference attacks for all datasets and model architectures.

First, consider the best tradeoffs due to adversarial regularization: For the best tradeoffs due to adversarial regularization, we use the models with attack accuracy and classification performance within 10-15% of the unprotected baseline. For Purchase-100, classification accuracy reduces by 8.9% to reduce inference risk by 15%, while DMP incurs just 3.4% accuracy loss to reduce inference risk by 21%. For the simple Purchase-100 task, the adversarially regularized model provides acceptable tradeoffs, but for the more complex CIFAR-100 task, it reduces inference risk just by 11-13% and classification accuracy by 5-13%. DMP, on the other hand, maintains the classification accuracy within 2.5% of that of the baseline cases, while also reduces the inference risk by 16% (C10-A) to 35% (C100-A).

Next, we compare tradeoffs of the adversarially regularized models which have E_{gen} equivalent to the DMP trained models; compare the 'Equivalent E_{gen}' and 'DMP' columns. As expected, with equivalent generalization errors, both DMP and adversarial regularization incur similar membership inference risks. However, classification performance of the DMP trained models is far superior to adversarially regularized models: For CIFAR-10 and CIFAR-100 tasks, the DMP models are almost twice as accurate as the adversarially regularized models. These comparisons show that, **DMP reduces the membership inference risk significantly with negligible reduction in the classification accuracy and provides much better tradeoffs than adversarial regularization**. In Appendix B, we also show the indistinguishability of different statistics and features of the DMP-trained models, and compare it with the indistinguishability due to adversarial regularization. Indistinguishability of such features has been shown to be effective in mitigating the inference risk [35], [36].

A closer look at the optimization problem solved by the adversarial regularization in [35] suggests that it should produce models with the optimal classification performance for the given attack model used for regularization. However, this optimization is exactly the same as that of the generative adversarial networks (GANs) [22] with the generator replaced by the target model and the discriminator by the attack model. Therefore, similar to the poor generalization of the generator in GANs [4], [5], generalization of the target models produced using the adversarial regularization in its current form is poor. This is seen in our empirical results and also in the results given in the original adversarial regularization work [35].

2) Comparison with other regularization techniques: Next, we compare DMP with label smoothing [53], confidence penalty [43], and dropout [52]. We compare the models with equivalent classification performance and equivalent generalization errors. To compare their tradeoffs, we perform thorough evaluation across all the datasets detailed in Section V.

Table III: Comparisons of generalization error (E_{gen}), classification accuracy (A_{test}), and membership inference risks (A_{wb} for whitebox and A_{bb} for blackbox inference) between DMP and adversarial regularization. Training accuracy is the summation of E_{gen} and A_{test}. DMP significantly improves the tradeoffs over the adversarial regularization. Check Table II for experimental setup.

Dataset and model	No defense				Adversarial regularization								DMP			
					Best tradeoffs				Equivalent E_{gen}							
	E_{gen}	A_{test}	A_{wb}	A_{bb}	E_{gen}	A_{test}	A_{wb}	A_{bb}	E_{gen}	A_{test}	A_{wb}	A_{bb}	E_{gen}	A_{test}	A_{wb}	A_{bb}
P-FC	24.0	76.0	77.1	76.8	22.4	68.1	62.3	61.9	9.7	56.5	55.8	55.4	10.1	74.1	55.3	55.1
C100-A	63.2	36.8	90.3	91.3	50.9	31.6	79.3	78.3	6.9	19.7	54.3	54.0	6.5	35.7	55.7	55.6
C100-D12	33.8	65.2	72.2	71.8	19.4	58.4	61.9	61.7	5.5	26.5	51.4	51.3	3.6	63.1	53.7	53.0
C100-D19	34.4	65.5	82.3	81.6	30.8	53.7	69.5	68.7	7.2	33.9	54.2	53.4	7.3	65.3	54.7	54.4
C10-A	32.5	67.5	77.9	77.5	29.8	62.6	65.2	65.0	4.2	53.4	51.9	51.2	3.1	65.0	51.3	50.6

Table IV: Demonstration of superior tradeoffs due to DMP training compared to the regularization methods. Check Table III for the accuracies of the corresponding DMP-trained models. '–' denotes that the regularizer could not achieve E_{gen} equivalent to DMP.

Experimental setup			Equivalent A_{test}				Equivalent E_{gen}			
Dataset	Model	Regularization	E_{gen}	A_{test}	A_{wb}	A_{bb}	E_{gen}	A_{test}	A_{wb}	A_{bb}
Purchase-100	Fully Connected	WD	21.7	78.1	69.7	70.1	10.3	42.5	54.9	55.4
		WD + DR	22.1	77.4	77.1	76.8	9.1	42.1	56.4	56.8
		WD + LS	21.1	78.4	76.5	76.8	12.3	42.0	57.2	57.0
		WD + CP	22.9	76.9	70.1	70.5	12.5	43.4	56.4	56.4
CIFAR-100	DenseNet12	WD	31.0	67.8	72.9	72.9	4.0	26.3	49.9	49.7
		WD + DR	31.0	68.2	73.7	73.6	3.7	32.3	51.2	51.0
		WD + LS	31.6	68.0	70.3	70.1	2.7	13.0	51.0	51.4
		WD + CP	31.1	67.5	74.3	74.7	–	–	–	–
CIFAR-10	AlexNet	WD	31.0	68.9	73.2	73.3	4.1	45.9	52.4	52.5
		WD + DR	30.6	69.4	73.8	73.4	3.2	44.7	51.9	51.7
		WD + LS	29.9	69.9	74.8	75.0	4.8	53.2	53.8	53.0
		WD + CP	29.9	70.0	70.6	71.1	–	–	–	–

The results are shown in Table IV. We see from the 'Equivalent A_{test}' column that all regularization techniques improve the classification performance over the corresponding baselines shown in 'No defense' column of Table III. However, they reduce overfitting negligibly: The maximum reduction in E_{gen} due to the regularizations is 1.8% for Purchase-100, 3.8% for CIFAR-100, and 2.6% for CIFAR-10. This is because these techniques aim to produce models that generalize better to test data, which is evident from the improved classification perfomance of the corresponding models in Table IV. But, they do not reduce the memorization of the private training data by the models. Consequently, fail to reduce the membership inference risk: The maximum reduction in A_{wb} due to the regularizations is 7% for Purchase-100, 1.9% for CIFAR-100, and 6.8% for CIFAR-10. Note that, the confidence penalty and the label smoothing techniques reduce the inference risk, but not the generalization error. This is because the corresponding models have smoother output distributions, which are more indistinguishable than the output distributions of models without any privacy. This reduces the gap between the KL-divergence losses of the model on members and non-members, and therefore, reduces the inference risk (Section III-B).

The comparison between models with equivalent E_{gen} that are trained using DMP and the regularization techniques shows that, although the inference risk of such models is similar, the classification performance of the DMP-trained models is far superior. Therefore, **DMP training offers superior tradeoffs than all the existing regularization techniques**.

Table V: Tradeoffs of AlexNet trained on CIFAR-10 using DP-SGD and DMP. With ϵ, both the model accuracy and membership inference risk increase. For equivalent *low membership inference resistance*, the accuracy of DMP-trained models is 12.8% higher than DP-SGD-trained models.

Defense	Privacy budget (ϵ)	Training accuracy	Test accuracy	Attack accuracy
No defense	n/a	100	67.5	77.9
DMP	n/a	68.1	65.0	51.3
DP-SGD	>100	55.8	52.2	51.7
	50.2	37.2	36.9	50.2
	12.5	30.2	31.7	49.9
	6.8	27.8	29.4	50.0

B. Comparison with differentially private models

Comparison with DP-SGD. We compare DMP and DP-SGD [3] defenses in terms of the (empirically observed) tradeoffs between membership inference risk and classification performance of the final models. Recently, Jayaraman et al. [25] performed a thorough analysis of the tradeoffs of DP-SGD models trained on Purchase-100 and CIFAR-100 datasets. They show that the differentially private models offer poor tradeoffs on complex tasks when evaluated using membership and attribute inference attacks. Below, we confirm their findings for the CIFAR-10 dataset and show that DMP provides much better tradeoffs.

DMP and DP-SGD cannot be compared directly in terms of their theoretical privacy guarantees. Instead, we follow the methodology of [25] and compare the tradeoffs offered by the corresponding models, when evaluated using membership inference attacks. Table V shows the results for AlexNet trained on CIFAR-10 data, averaged over 3 runs of each experiment; δ is constant at 10^{-6}. We note that, DP-SGD incurs significant (35%) loss in classification performance at lower ϵ to provide strong membership privacy. With larger ϵ, the accuracy of DP-SGD-trained models increases, but at the cost of higher membership inference risk. This risk arises due to poor generalization at high privacy budgets which is sufficient for successful membership inference. More importantly, **for a given low membership inference risk ($\sim$ 51%), the DP-SGD-trained models incur a significantly higher classification performance loss (15.3%) than the DMP trained models (2.5%), compared to the baseline model.**

Comparison with PATE. Papernot et al. [41], [39] proposed PATE, a distributed training technique to produce differentially private models. PATE requires exorbitantly large amounts of data to train a teachers ensemble so that the student model, which is trained on the predictions of the ensemble using semi-supervised learning, has good classification performance. PATE works well on simple tasks such as MNIST and SVHN[3] with limited data, but for more complex tasks such as Glyph, it requires *65 million samples to train a good ensemble of teachers*. We compare DMP with PATE for CIFAR-10 task. In PATE, the student is trained via the unstable semi-supervised learning, which requires a good combination of generator and discriminator network architectures. Therefore, we use the generator and discriminator architectures from [48] which provided state-of-the-art performance.

We use the same data partitions as in Table I and train the ensembles of 5, 10, and 25 teachers. We use the confident-GNMax (GNMax) aggregation scheme [41] on the outputs of the three ensembles to label a subset of 25,000 reference samples. A subset of results are shown in Table VI, and the complete comparison is deferred to Appendix A due to space limitations. It is known that, the larger the number of labeled samples, the better the semi-supervised student [41], [48]. But, GNMax produces very small number of labels at low ϵ's, e.g, *GNMax generated 0 labels for $\epsilon < 10$.* At high ϵ GNMax generates enough labels to train a good student model, as shown in the Table VI. But, its DP guarantees are meaningless at such high ϵ's, and therefore, it is simply a knowledge transfer based semi-supervised learning, while DMP is knowledge transfer based supervised learning. DMP provides better accuracy than PATE for the same membership inference risk, because, PATE divides the private training data among teachers to produce an ensemble, the accuracy of which is strictly lower than the model DMP trains on the entire training data. Therefore, *PATE trains a student on the predictions less useful than the predictions used to train θ_p in DMP*. This is reflected in the significantly superior performance of DMP-trained model, which has training, test, and attack accuracies of 77.98%, 76.79%, and 50.8%, respectively.

Table VI: Comparison with PATE: The student architecture is of the discriminator in [48]. PATE does not produce enough labels at low ϵ and suffers huge losses in classification performance, and at high ϵ, it simply acts as a semi-supervised learning with knowledge transfer. DMP-trained model achieves training, test, and attack accuracies of 77.98%, 76.79%, and 50.8%, respectively.

# of Teachers	Queries answered	Privacy budget (ϵ)	Student accuracy Train	Student accuracy Test	Attack accuracy
5	49	195.9	31.4	33.9	49.1
	1163	11684	65.4	68.1	49.0
10	23	42.9	39.1	38.3	50.1
	1527	6535	63.9	65.2	49.8
25	108	183.5	53.8	55.7	49.0
	4933	1794.1	57.8	60.3	48.6

Discussion. DP-SGD and PATE provide theoretical differential, and therefore membership, privacy guarantees by updating the model parameters using the gradients of the loss on the private training data that are perturbed using calibrated DP noise. On the other hand, the membership inference resistance due to DMP is a result of bounding the ratio $\mathcal{R}$ given by (12). We note that, as Sablayrolles et al. [46] show, differential privacy is a stronger privacy notion than the membership privacy and that achieving membership privacy does not guarantee differential privacy. Therefore, bounding the ratio (12) suffices to achieve significant improvements in the membership inference resistance, while the KL-divergence based loss minimization involved in training of the protected model provides superior classification performance. DMP effectively combines these two requirements and provides improved tradeoffs as shown above.

C. Membership inference against highly susceptible classes

In all the above evaluations, we essentially studied the average membership inference risk a model poses to the members of its training data. As empirically shown in [36], [59], the membership inference susceptibility varies across different classes of a classification task. Therefore, it is important that DMP trained models preserve privacy of all the classes fairly,

[3] Even for SVHN, PATE [39] uses the extended SVHN data of size 630K.

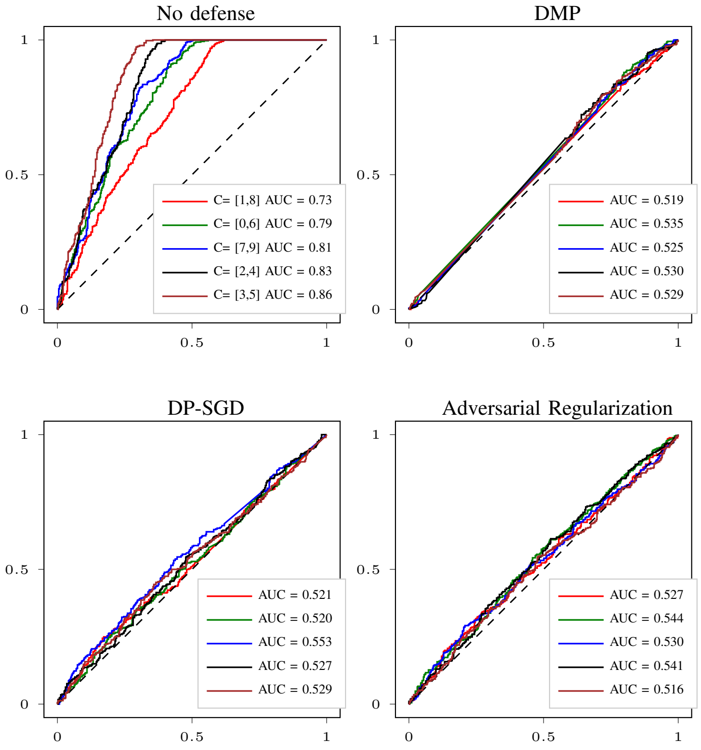


Figure 5: ROC curves (true positive versus false positive rates) for CIFAR-10 classes with varying susceptibility to membership inference due to the baseline AlexNet (upper). DMP trained AlexNet (lower) provides strong resistance to all the classes—including the ones that are most susceptible without a defense—and provides fair resistance across classes.

and do not trade the privacy of highly susceptible classes with that of less susceptible classes, because the latter ones are easy to protect.

To test the susceptibility of different classes, we plot Receiver Operating Characteristic curves in Figure 5 for different CIFAR-10 classes while keeping the experimental setting the same as before. First, as shown in Figure 5, the *classes are paired according to their membership inference susceptibility due to θ_{up} trained without any defense*; classes [1,8] are the least and [3,5] are the most susceptible. Figure 5 also shows the ROC curves of the same pairs when DMP, DP-SGD, and adversarial regularization defenses are used to train the protected model, θ_{p}. We note that the DMP trained model has very low area under curve (AUC) for all classes (average AUC = 0.528), including for the most susceptible classes (AUC = 0.529). This also implies that DMP defense does not tradeoff the privacy of susceptible members with that of resilient members, and provides privacy to all the classes in equitable fashion. Note that, although DP provides information theoretic privacy guarantees, in practice, DP-SGD-trained models with equivalent generalization error also exhibit the membership privacy disparity similar to the DMP trained models. For the adversarial regularization defense, we find the similar trend of susceptibility across various classes.

D. Hyperparameter selection in DMP

We demonstrate the impact of the two hyperparameters of DMP (Section IV-C) on its performance.

1) The temperature of the softmax layer: The softmax temperature, T, in θ_{up} plays an important role in the amount of knowledge transferred from the private to non-private model (Section IV-C). Our results in Table VII confirm our analytical understanding of the use of the softmax temperature: increasing the temperature for AlexNet with CIFAR-100 dataset reduces the classification accuracy of θ_{p}, but also strengthens the membership inference resistance. Therefore, the softmax temperature T should be chosen depending on the desired privacy-utility tradeoff. Table II shows the temperatures used in our experiments for different datasets and models.

Table VII: Effect of the softmax temperature on DMP: For a fixed X_{ref}, increase in the temperature of softmax layer of θ_{up} reduces $\Delta\mathcal{L}_{\text{KL}}$ in (14) and reduces the ratio $\mathcal{R}$ in (12), which strengthens the membership privacy.

Defense	Softmax T	Training Accuracy	Test Accuracy	Attack Accuracy
No defense	n/a	100	36.8	91.3
DMP	2	46.6	37.3	57.4
	4	42.2	35.7	55.6
	6	36.4	32.8	52.5
	8	12.1	12.3	51.7

2) The size of reference data: We analyzed in Section IV-C the effect of the size of the reference data on DMP-trained θ_{p}: The more the reference data, the looser the bound on $\mathcal{R}$ in (12), and therefore, weaker the membership resistance of the corresponding θ_{p}. To validate this, we quantify the classification accuracy and the membership inference risk of θ_{p} with increasing the amount of X_{ref}. We use Purchase-100 data and vary $|X_{\text{ref}}|$ as shown in Figure 6; we fix the softmax T of θ_{up} at 1.0. θ_{up} used here has train accuracy, test accuracy, and membership inference risk of 99.9%, 77.0% and 77.1%, respectively. Initially, the test accuracy of θ_{p} increases with $|X_{\text{ref}}|$ due to the useful knowledge transferred. But, beyond the test accuracy of θ_{up}, its predictions essentially inserts noise in the training data of θ_{p}, therefore the gain from increasing the size of reference data slows down. Although this noise marginalizes the increase in the test performance of θ_{p}, it also prevents θ_{p} from learning more about D_{tr} and prevents further inference risk. This is shown by the train accuracy and membership inference risk curves in Figure 6, respectively. Therefore, size of reference data should be selected based on the desired tradeoffs of the final model.

Finally, we note that if the correct labels Y_{ref} are available for X_{ref}, both the classification accuracy and membership risk due to θ_{p} improve. For instance, for C100-D12, DMP training with labeled X_{ref} increases the classification accuracy by 63.1% to 67.2% and reduces the inference risk by 53.7%

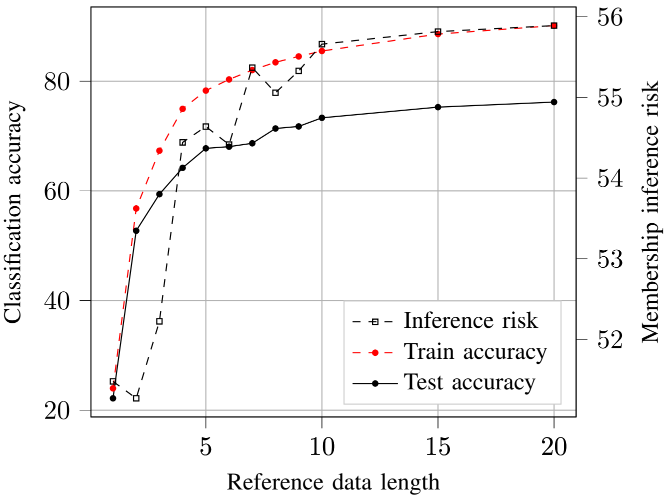


Figure 6: Classification accuracy and membership inference risk for different reference data sizes, $|X_{\textsf{ref}}|$. With increasing $|X_{\textsf{ref}}|$, (14) and thereofer the ratio $\mathcal{R}$ in (12) increases, which increases the membership inference risk due to $\theta_{\textsf{p}}$.

to 51.8%. Similarly, for P-FC, the classification accuracy increases from 74.3% to 77.2% and the membership risk reduces from 55.5% to 51.4%. Therefore, similar to data augmentation techniques, *DMP also serves as an efficient utility improvement technique in the presence of labeled reference data.*

VII. Related Work

Privacy preserving machine learning is an active area of research. Defenses based on trusted hardware and cryptographic primitives [8], [21], [31], [34] hinder a direct access to sensitive training data during training. However, the final models remain susceptible to various inference attacks through blackbox or whitebox accesses, especially for large capacity neural networks due to their large memorization capacities [16]. Such inference attacks include input inference [17], blackbox and whitebox membership inference [50], [36], [47], [29], attribute inference [10], parameter inference [55], [56], training data embedding attacks [51], and side-channel attacks [58]. In this paper, we focus on the membership inference attacks for adversaries with blackbox and whitebox access to the model.

Several recent defenses have been proposed against membership inference attacks [3], [41], [35], [39]. Unfortunately, the existing defenses do not provide acceptable tradeoffs between privacy and utility, i.e., they hurt the model's classification performance significantly to provide membership privacy. Defenses based on differential privacy (DP) [3], [41], [39], [26], [42] provide rigorous membership privacy guarantees, but as demonstrated by Jayaraman et al. [25], the resulting models are of no practical use. Furthermore, as [25], [29] shows—and we confirm in our work—with relaxed privacy budgets, DP defenses are also susceptible to the membership inference. The primary reason for the susceptibility is the high generalization error of such models, which is sufficient for membership inference [32], [44], [50], [36], [29]. Adversarial regularization [35] is a recent defense that is tailored to membership inference attacks, and claims to improve the tradeoffs. However, as shown in Section VI-A, the adversarial regularization defense fails to provide acceptable tradeoffs when evaluated against state-of-the-art membership inference attacks.

Knowledge distillation has been used in several privacy defenses [39], [26], [42], [38], [7], [57], which perform distillation using a noisy aggregate of predictions of models of multiple data holders. In particular, PATE [39], [41] combines knowledge distillation and DP [3]. In PATE, an input is labeled by an ensemble of *teacher models*, and the final *student model* is trained using the noisy aggregates of all labels. PATE requires exorbitantly large amounts of data to train a good teachers' ensemble. In the absence of such data, the aggregation in PATE does not produce sufficient number of labels at low privacy budgets, and therefore, cannot train accurate student models; we demonstrate this in Section VI-B. Even at higher privacy budgets, PATE-trained students do not attain the accuracy similar to DMP-trained models, because, the ensemble used to transfer knowledge in PATE is significantly less accurate than the unprotected model used in DMP. In effect, DMP provides better tradeoffs than PATE. DP defenses add large amounts of noise to provide privacy to *any data* with the underlying distribution, and in this process incur high accuracy losses [41]. However, due to the targeted motivation to provide membership privacy, DMP defense uses the novel knowledge transfer via easy-to-classify samples, whose predictions are not affected by the presence of any particular member in the private training data. Therefore, our approach differs from these defenses in that, we do not explicitly add DP noise, and instead, prevent the membership leakage through the predictions of the reference data.

Regularization alone is shown to be ineffective against membership inference attacks [32], [36], [29]. Long et al. [32] proposed a membership inference attack against well-generalized models that identifies the vulnerable *outliers* in the sensitive training data of the model, whose membership can be inferred. In DMP, such outliers can be protected by setting high softmax temperatures or selecting samples with low entropy predictions (Section IV), but at the cost of utility degradation. This is similar to previous defenses: in DP-SGD, privacy budget is reduced and in the adversarial regularization, high regularization factor is set to provide privacy to the outliers, and in practice, at relaxed privacy budgets or low regularization factors, these defenses also pose the membership inference risk to such outliers [25], [44]. Leino et al. [29] also proposed whitebox membership inference attacks against well-generalized models, including differentially private models. However, we note that the primary objective of our DMP defense is to produce models with superior tradeoffs, i.e., achieve superior classification performance for a given degree of membership privacy. We demonstrated the effectiveness of DMP in Section VI in producing such models with state-of-the-art classification accuracy for a given membership privacy.

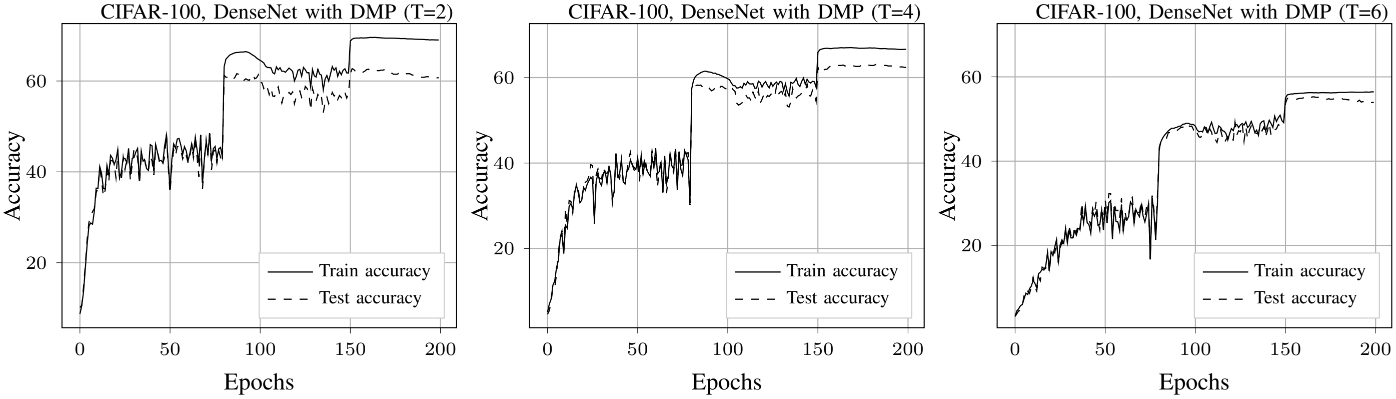


Figure 7: Impact of softmax temperature on training of θ_{p}: Increase in the temperature of softmax layer of θ_{up} reduces $\Delta\mathcal{L}_{\mathsf{KL}}$ in (14) and the ratio $\mathcal{R}$ in (12), which improves the membership privacy and generalization of θ_{p}, but at the cost of classification performance losses, as shown here.

To summarize, *all of the existing defenses rely on adding some explicit noise during the training or regularization of the model* in different ways. Because of such explicit noise additions, all these defenses suffer from significant utility degradations in terms of classification performance of the final models. By contrast, DMP provides membership inference resistance using a novel approach of selecting low entropy easy-to-classify samples for knowledge transfer. Knowledge transfer presents itself as a promising option for practical utility-membership privacy tradeoffs because of its proven ability to transfer the utility of the cumbersome model to the final model [20], [57], [6].

VIII. Conclusions

Motivated by the poor tradeoffs between model utility and resistance to membership inference attacks, we introduced distillation for membership privacy (DMP), an effective defense against membership inference attacks on machine learning models. DMP leverages knowledge transfer to train models resilient to membership inference and with high classification performance. We analyze the key requirements for membership inference resistance and provide a novel empirical approach to select the data used for knowledge transfer such that membership leakage during the transfer is reduced. DMP trains machine learning models that are resistant to whitebox and blackbox membership inference attacks while preserving the classification performance of the models significantly better than state-of-the-art membership inference defenses. We validate DMP's superior performance in terms of tradeoff between membership privacy and utility of the models through extensive experiments on different deep neural networks and using various benchmark datasets.

References

[1] "Acquire Valued Shoppers Challenge," https://www.kaggle.com/c/acquire-valued-shoppers-challenge/data, 2019, [Online; accessed 11-September-2019].

[2] "PyTorch Documentation," https://pytorch.org/, 2019, [Online; accessed 11-September-2019].

[3] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016.

[4] S. Arora, R. Ge, Y. Liang, T. Ma, and Y. Zhang, "Generalization and equilibrium in generative adversarial nets (gans)," *In Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 2017.

[5] S. Arora, A. Risteski, and Y. Zhang, "Do gans learn the distribution? some theory and empirics," *Georgia Institite of Technology Technical Report*, 2018.

[6] J. Ba and R. Caruana, "Do deep nets really need to be deep?" in *Advances in neural information processing systems*, 2014, pp. 2654–2662.

[7] R. Bassily, A. G. Thakurta, and O. D. Thakkar, "Model-agnostic private learning," in *Advances in Neural Information Processing Systems*, 2018, pp. 7102–7112.

[8] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth, "Practical secure aggregation for privacy-preserving machine learning," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017.

[9] M. H. Brendan, D. Ramage, K. Talwar, and L. Zhang, "Learning differentially private recurrent language models," *International Conference on Learning and Representation*, 2018.

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

[11] K. Chaudhuri, C. Monteleoni, and A. D. Sarwate, "Differentially private empirical risk minimization," *Journal of Machine Learning Research*, vol. 12, no. Mar, pp. 1069–1109, 2011.

[12] C.-L. Chi, W. N. Street, J. G. Robinson, and M. A. Crawford, "Individualized patient-centered lifestyle recommendations: An expert system for communicating patient specific cardiovascular risk information and prioritizing lifestyle options," *Journal of Biomedical Informatics 45, no. 6*, 2012.

[13] I. W. P. Consortium, "Estimation of the warfarin dose with clinical and pharmacogenetic data," *New England Journal of Medicine 360(8)*, 2009.

[14] Z. Ding, Y. Wang, G. Wang, D. Zhang, and D. Kifer, "Detecting violations of differential privacy," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2018, pp. 475–489.

[15] C. Dwork, A. Roth *et al.*, "The algorithmic foundations of differential privacy," *Foundations and Trends® in Theoretical Computer Science*, 2014.

[16] C. Dwork, A. Smith, T. Steinke, and J. Ullman, "Exposed! a survey of attacks on private data," 2017.

[17] M. Fredrikson, S. Jha, and T. Ristenpart, "Model inversion attacks that exploit confidence information and basic countermeasures," in

Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security. ACM, 2015.

[18] M. Fredrikson, E. Lantz, S. Jha, S. Lin, D. Page, and T. Ristenpart, "Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing." in *USENIX Security Symposium*, 2014.

[19] K. Ganju, W. Qi, W. Yang, C. A. Gunter, and N. Borisov, "Property inference attacks on fully connected neural networks using permutation invariant representations," *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018.

[20] H. Geoffrey and V. O. amd Dean Jeff, "Distilling the knowledge in a neural network," *NIPS 2014 Deep Learning Workshop*, 2014.

[21] R. Gilad-Bachrach, N. Dowlin, K. Laine, K. Lauter, M. Naehrig, and J. Wernsing, "Cryptonets: Applying neural networks to encrypted data with high throughput and accuracy," in *International Conference on Machine Learning*, 2016.

[22] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014.

[23] B. Hitaj, G. Ateniese, and F. Pérez-Cruz, "Deep models under the gan: information leakage from collaborative deep learning," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017.

[24] N. Homer, S. Szelinger, M. Redman, D. Duggan, W. Tembe, J. Muehling, J. V. Pearson, D. A. Stephan, S. F. Nelson, and D. W. Craig, "Resolving individuals contributing trace amounts of dna to highly complex mixtures using high-density snp genotyping microarrays," *PLoS genetics*, 2008.

[25] B. Jayaraman and D. Evans, "Evaluating differentially private machine learning in practice," in *USENIX Security Symposium*, 2019.

[26] H. Jihun, C. Yingjun, and B. Mikhail, "Learning privately from multiparty data," *Proceedings of The 33rd International Conference on Machine Learning*, 2016.

[27] P. W. Koh and P. Liang, "Understanding black-box predictions via influence functions," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 1885–1894.

[28] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," 2009.

[29] K. Leino and M. Fredrikson, "Stolen memories: Leveraging model memorization for calibrated white-box membership inference," *arXiv preprint arXiv:1906.11798*, 2019.

[30] N. Li, W. Qardaji, D. Su, Y. Wu, and W. Yang, "Membership privacy: a unifying framework for privacy definitions," *In Proceedings of ACM SIGSAC Conference on Computer and Communications Security*, 2013.

[31] Y. Lindell and B. Pinkas, "Privacy preserving data mining," in *Annual International Cryptology Conference*. Springer, 2000.

[32] Y. Long, V. Bindschaedler, L. Wang, D. Bu, X. Wang, H. Tang, C. A. Gunter, and K. Chen, "Understanding membership inferences on well-generalized learning models," *arXiv preprint arXiv:1802.04889*, 2018.

[33] L. Melis, C. Song, E. D. Cristofaro, and V. Shmatikov, "Exploiting unintended feature leakage in collaborative learning," *40th IEEE Symposium on Security and Privacy*, 2019.

[34] P. Mohassel and Y. Zhang, "Secureml: A system for scalable privacy-preserving machine learning," in *Security and Privacy (SP), 2017 IEEE Symposium on*. IEEE, 2017.

[35] M. Nasr, R. Shokri, and A. Houmansadr, "Machine learning with membership privacy using adversarial regularization," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2018, pp. 634–646.

[36] ——, "Comprehensive privacy analysis of deep learning: Stand-alone and federated learning under passive and active white-box inference attacks," *Security and Privacy (SP), 2019 IEEE Symposium on*, 2019.

[37] G. K. Nayak, K. R. Mopuri, V. Shaj, V. B. Radhakrishnan, and A. Chakraborty, "Zero-shot knowledge distillation in deep networks," in *International Conference on Machine Learning*, 2019, pp. 4743–4751.

[38] K. Nissim, S. Raskhodnikova, and A. Smith, "Smooth sensitivity and sampling in private data analysis," in *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*. ACM, 2007, pp. 75–84.

[39] N. Papernot, M. Abadi, U. Erlingsson, I. Goodfellow, and K. Talwar, "Semi-supervised knowledge transfer for deep learning from private training data," *International Conference on Learning and Representation*, 2017.

[40] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami, "Distillation as a defense to adversarial perturbations against deep neural networks," *Proceedings of the 37th IEEE Symposium on Security and Privacy*, 2016.

[41] N. Papernot, S. Song, I. Mironov, A. Raghunathan, K. Talwar, and Ú. Erlingsson, "Scalable private learning with pate," *arXiv preprint arXiv:1802.08908*, 2018.

[42] M. Pathak, S. Rane, and B. Raj, "Multiparty differential privacy via aggregation of locally trained classifiers," *In Advances in Neural Information Processing Systems*, 2010.

[43] G. Pereyra, G. Tucker, J. Chorowski, Ł. Kaiser, and G. Hinton, "Regularizing neural networks by penalizing confident output distributions," *arXiv preprint arXiv:1701.06548*, 2017.

[44] M. A. Rahman, T. Rahman, R. Laganiere, N. Mohammed, and Y. Wang, "Membership inference attack against differentially private deep learning model," *Transactions on Data Privacy 11, no. 1*, 2018.

[45] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "Fitnets: Hints for thin deep nets," *arXiv preprint arXiv:1412.6550*, 2014.

[46] A. Sablayrolles, M. Douze, C. Schmid, Y. Ollivier, and H. Jegou, "White-box vs black-box: Bayes optimal strategies for membership inference," in *International Conference on Machine Learning*, 2019, pp. 5558–5567.

[47] A. Salem, Y. Zhang, M. Humbert, M. Fritz, and M. Backes, "Ml-leaks: Model and data independent membership inference attacks and defenses on machine learning models," *NDSS*, 2019.

[48] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," in *Advances in neural information processing systems*, 2016, pp. 2234–2242.

[49] S. Sankararaman, G. Obozinski, M. I. Jordan, and E. Halperin, "Genomic privacy and limits of individual detection in a pool," *Nature genetics*, 2009.

[50] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *Security and Privacy (SP), 2017 IEEE Symposium on*, 2017.

[51] C. Song, T. Ristenpart, and V. Shmatikov, "Machine learning models that remember too much," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017.

[52] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

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

[54] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," *In Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014.

[55] F. Tramèr, F. Zhang, A. Juels, M. K. Reiter, and T. Ristenpart, "Stealing machine learning models via prediction apis," in *USENIX Security*, 2016.

[56] B. Wang and N. Z. Gong, "Stealing hyperparameters in machine learning," *Security and Privacy (SP), 2018 IEEE Symposium on*, 2018.

[57] J. Wang, W. Bao, L. Sun, X. Zhu, B. Cao, and P. S. Yu, "Private model compression via knowledge distillation," *In 33rd AAAI Conference on Artificial Intelligence*, 2019.

[58] L. Wei, Y. Liu, B. Luo, Y. Li, and Q. Xu, "I know what you see: Power side-channel attack on convolutional neural network accelerators," *arXiv preprint arXiv:1803.05847*, 2018.

[59] M. Yaghini, B. Kulynych, and C. Troncoso, "Disparate vulnerability: on the unfairness of privacy attacks against machine learning," *arXiv preprint arXiv:1906.00389*, 2019.

[60] S. Yeom, I. Giacomelli, M. Fredrikson, and S. Jha, "Privacy risk in machine learning: Analyzing the connection to overfitting," in *2018 IEEE 31st Computer Security Foundations Symposium (CSF)*. IEEE, 2018, pp. 268–282.

Appendix

A. Detailed comparison with PATE

In this section, we detail the experimental comparison between PATE [41], [39] and our DMP training algorithm on

Table VIII: Student with the discriminator architecture in [48] trained on CIFAR-10 using PATE: For $\epsilon < 10$, confident-GNMax does not answer any queries. Higher quality and quantity of labeled data is required to train a good student model using semi-supervised learning, which are obtained only when ϵ is significantly high. The corresponding DMP-trained model has 77.98% and 76.79% accuracies on the training and test data, and 50.8% membership inference accuracy.

5 Teachers				10 Teachers				25 Teachers			
Queries answered	Privacy bound ϵ	GNMax accuracy	Student accuracy	Queries answered	Privacy bound ϵ	GNMax accuracy	Student accuracy	Queries answered	Privacy bound ϵ	GNMax accuracy	Student accuracy
0	4.6	–	–	0	9	–	–	0	8.43	–	–
49	195.9	79.6	33.93	23	42.87	56.5	38.28	108	183.5	95.4	55.7
127	281.6	69.3	49.89	358	409.5	67.0	57.59	357	231.3	83.9	56.14
679	1283.7	70.3	58.04	1128	1092.5	66.13	60.94	1130	508.9	83.8	58.26
1163	11684	91.1	68.08	1527	6535	93.1	65.18	4933	1794.1	74.0	60.27

the CIFAR-10 classification task. The motivation of this comparison is to show that the DMP-trained models achieve significantly better tradeoffs between membership inference resistance and classification performance than the PATE trained models. As mentioned in Section VI-B, PATE relies on semi-supervised learning on a large pool of unlabeled data and a small number of labeled data. The labels are obtained using an ensemble of teachers that are trained on disjoint training datasets. The disjoint datasets are obtained by dividing the private training data equally among the teachers. Semi-supervised learning involves an unstable game between a generator G and a discriminator D, and the combination of the architectures of G and D should be chosen carefully for the training to be effective. Therefore, instead of AlexNet architecture, which is used in the rest of CIFAR-10 experiments, we use the discriminator architecture proposed in [48] as the CIFAR-10 classifier, along with the generator architecture in the work. This is because, the combination of G and D is empirically showed to provide state-of-the-art performance, and is also improved later by a few other works. For a fair comparison, we use the same partition of the CIFAR-10 data, given in Table I, as the private training and the unlabeled reference data for PATE and DMP. The accuracy of the baseline model trained on the entire private training data is 97.65% and 79.6% on training and test samples, respectively.

We divide the 25000 *training* samples in disjoint sets of sizes 5, 10 and 25, and then train three ensembles of teachers on the disjoint sets. That is, with the above partitioning, each of the teachers in the ensembles of sizes 5, 10, and 25 is trained on 5000, 2500 and 1000 samples, respectively. The accuracy, *without adding any noise to labels*, of the corresponding ensembles on the 25000 *reference* samples is 64.92%, 60.1% and 54.52%, respectively. We use the confident-GNMax (GNMax) aggregation scheme to add DP noise to the votes of teachers on the reference data and collect the final labels on the reference data. Note that, although we input all the 25000 reference samples to the aggregation scheme, not all the samples get labels in the end. Because, the GNMax aggregation scheme is similar to the sparse vector technique [15] and outputs a label only if the noisy version of the votes count of the label crosses a noisy version of a fixed threshold. Table VIII details the accuracy of the GNMax aggregation for different number of teachers and privacy levels (ϵ, δ). We keep the δ constant at 10^{-4}, because the order of the size of the reference data is 10^4 [41].

It can be seen from the results in Table VIII that, PATE aggregation cannot produce high quantity and/or quality labels for the unlabeled reference samples, especially at low ϵ values. This leads to the poor performance of the following semi-supervised training of the student. First, for reference note that, the combination of G and D we use achieves 67.3% accuracy with 1000 labeled samples and 75% accuracy with 4000 labeled samples when trained for 400 epochs. At low ϵ values that are important for meaningful DP guarantees, GNMax either does not produce any samples, e.g. for $\epsilon < 10$, or outputs insufficient samples to train a good student model. To achieve performance comparable to the baseline using the labels outputs by GNMax aggregation, the ϵ values need to be > 1000, which are unacceptable from DP standards.

On the other hand, DMP-trained model training, test, and attack accuracies of 77.98%, 76.79%, and 50.8%, respectively At low ϵ, PATE provides provable differential privacy, and therefore, the membership privacy, but the models obtained are cannot be used due to their poor classification performance. While at high ϵ, PATE produces good students but with meaningless DP guarantees. In other words, PATE at high ϵ's is simply a knowledge transfer based semi-supervised learning, while DMP is knowledge transfer based supervised learning. DMP does not divide data among teacher and therefore the predictions of the unprotected model used in DMP to train the protected model are more useful in terms of quality and quantity. Therefore, DMP-trained models achieve significantly better tradeoffs between membership privacy and classification performance than PATE trained models, and also that the DMP training is much more useful in practice than PATE.

B. *Indistinguishability due to DMP training*

In this section, we present the statistics of different features of the target models, trained with and without defenses, on the members and non-members of their training data. As discussed

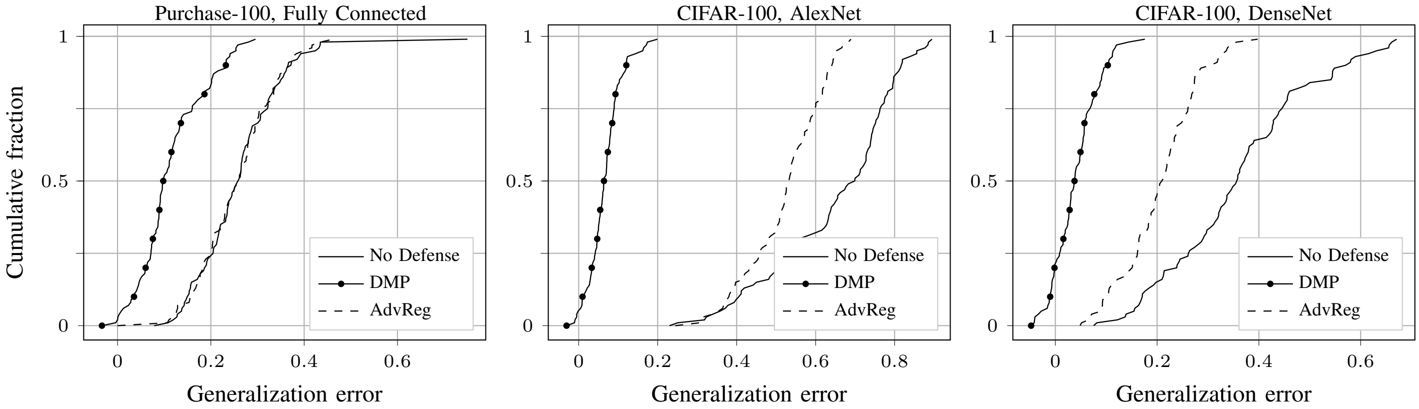


Figure 8: The empirical CDF of the generalization error of models trained with DMP and adversarial regularization (AdvReg), and without defense. The y-axis is the fraction of classes that have generalization error less than the corresponding value on x-axis. The error reduction using DMP is much larger (10-folds for CIFAR-100 dataset and by 2-folds for Purchase-100 dataset) than using AdvReg. Refer to 'Best tradeoffs' column in Table III for the specific accuracies due to adversarial regularization defense.

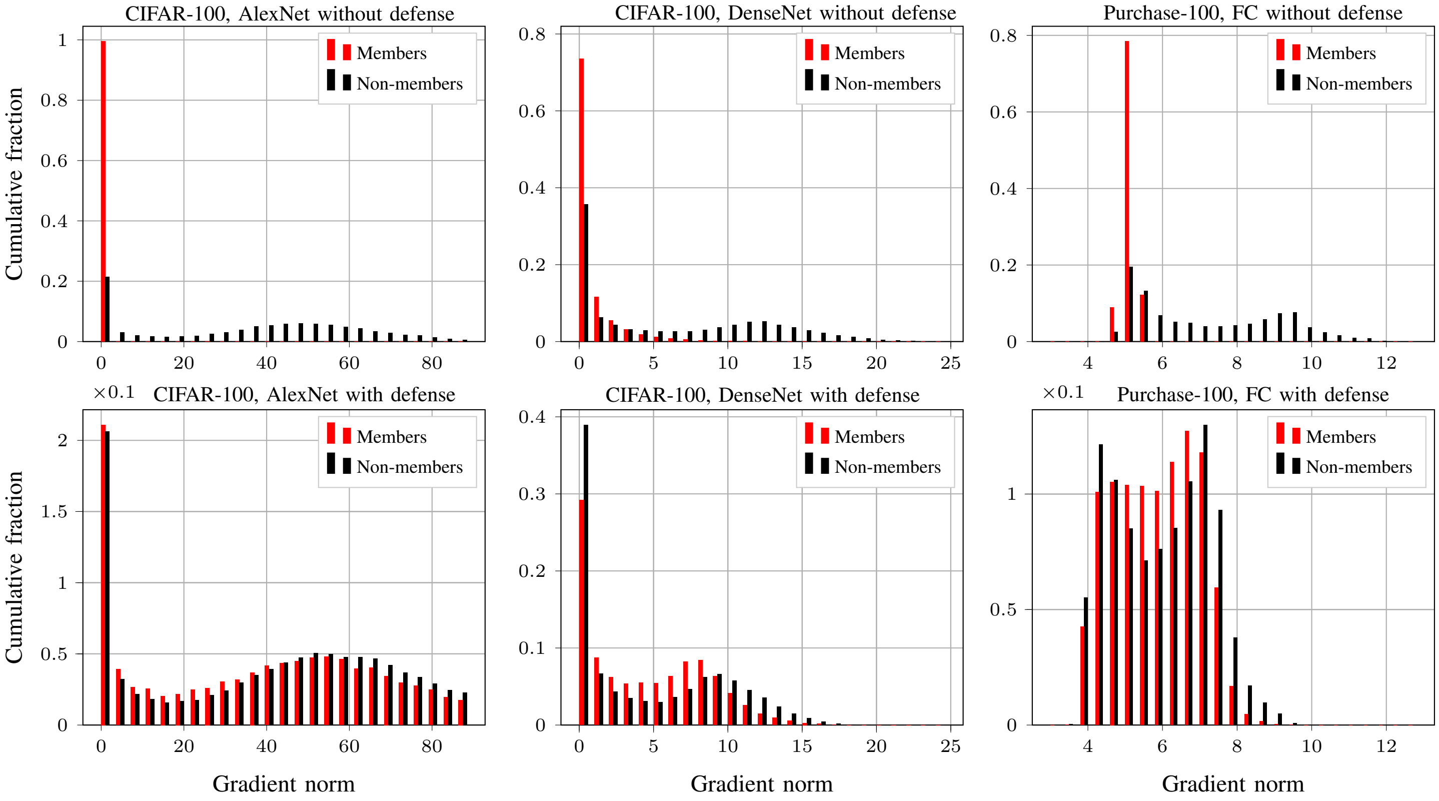


Figure 9: Distribution of gradient norms of members and non-members. (Upper row): Unlike the non-member distributions, the member distributions of the unprotected model, θ_{up}, are skewed towards 0 due to memorization of the members by the networks. (Lower row): The distribution of gradient norms of the protected model, θ_{p}, for members and non-members of the private training data. DMP significantly *increases* the members' gradient norms making them indistinguishable from the non-members' norms.

in Section II-C, the blackbox and whitebox membership inference attacks [36], [50], [33], [19] exploit these statistical differences.

Figure 7 shows the effect of softmax T on the training accuracy on the private training data, D_{tr}, and test accuracy of θ_{p} as the training progresses. In theory, with increase in T of softmax layer of θ_{up}, the generalization error of θ_{p} should decrease due to reduced membership leakage. We observe this in Figure 7: From left to right, the generalization errors of θ_{p} trained with the temperatures of softmax layer of θ_{up} set at 2, 4, and 6 are 4.7% (66.3, 61.6), 3.6% (66.7, 63.1), and 0.8% (55.7, 54.9), respectively. In parentheses are shown the corresponding training and test accuracies, respectively. We keep the temperature of softmax layer in θ_{p} constant at 4.0. This shows that *increasing the temperature T of the softmax layer of θ_{up} reduces the membership leakage and strengthens the membership resistance of θ_{p}, as discussed in Section IV-C.*

The adversarial regularization [35] claims to improve the generalization performance of its resulting models. We show in Figure 8 that, the DMP-trained models have significantly better generalization performance than the adversarially regularized models. In Figure 8, we show the cumulative fraction of classes on y-axis for which the generalization error of the target models is lesser than the corresponding value on the x-axis. Here, closer the line to the line $x = 0$, lower the generalization error. We observe that, with the no defense case as the baseline, *the reduction in the generalization error using DMP is more than twice that using the adversarial regularization. DMP reduces the error by half for Purchase-100 and the reduction is 10-folds for CIFAR-100 dataset*. It is worth mentioning that, the adversarial regularization performs well for large training datasets, but fails to protect small training datasets. We explicitly consider small training datasets to evaluate the efficacy of DMP, as they are harder to prevent from overfitting and therefore from the membership inference attacks.

To assess the efficacy of DMP against the stronger whitebox membership inference attacks [36], [33], we study the gradients of loss of the predictions of θ_{up} and θ_{p} on members and non-members of D_{tr}. Figure 9 shows the fraction of members and non-members given on y-axes that fall in a particular range of gradient norm values given on x-axes. Gradients are computed with respect to the parameters of the given model. We note that the distribution of the norms of θ_{up} (upper figures) is heavily skewed to the left for the members, i.e., towards lower gradient norm values, unlike that for the non-members. This is because θ_{up} tends to memorize D_{tr}, and therefore, the loss and gradient of the loss of the predictions of θ_{up} on the members is very small compared to the non-members. Therefore, the gradients for the non-members are more evenly distributed over a large range of the norm values. However, for the DMP-trained θ_{p}, both members and non-members are evenly distributed across a large range of gradient norm values. In other words, the loss of DMP-trained θ_{p} on members increases significantly. This implies that *DMP significantly reduces the unintended memorization of D_{tr} in the model parameters and makes gradients of the loss on members and non-members indistinguishable*. This is reflected in the significant reduction (27.6%) in the membership inference risk to the large capacity Dense19 model as shown in Table III. This indistinguishability of different statistics of features of θ_{p} on members and non-members mitigates the membership inference risk to D_{tr} with either of blackbox and whitebox access to θ_{p}.