Differential Privacy Defenses and Sampling Attacks for Membership Inference

Shadi Rahimian
CISPA Helmholtz Center for Information Security
Saarbrücken, Germany
shadi.rahimian@cispa.saarland

Tribhuvanesh Orekondy
Max Planck Institute for Informatics
Saarbrücken, Germany
orekondy@mpi-inf.mpg.de

Mario Fritz
CISPA Helmholtz Center for Information Security
Saarbrücken, Germany
fritz@cispa.de

ABSTRACT

Machine learning models are commonly trained on sensitive and personal data such as pictures, medical records, financial records, etc. A serious breach of the privacy of this training set occurs when an adversary is able to decide whether or not a specific data point in her possession was used to train a model. While all previous membership inference attacks rely on access to the posterior probabilities, we present the first attack which only relies on the predicted class label - yet shows high success rate.

CCS CONCEPTS

• **Computing methodologies** → **Machine learning**; • **Security and privacy**;

KEYWORDS

membership inference attacks; deep learning; privacy-preserving machine learning

ACM Reference Format:
Shadi Rahimian, Tribhuvanesh Orekondy, and Mario Fritz. 2021. Differential Privacy Defenses and Sampling Attacks for Membership Inference. In *Proceedings of the 14th ACM Workshop on Artificial Intelligence and Security (AISec '21), November 15, 2021, Virtual Event, Republic of Korea.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3474369.3486876

1 INTRODUCTION

Recent machine learning (ML) methods – especially deep learning approaches – largely owe their success to the availability of big data and the computation power to train huge models with millions of parameters on this *training dataset*. Often, people might assume that since these models are designed to learn statistical properties of their training dataset, they protect the privacy of the individuals who contribute to these datasets. Unfortunately, this is not the case and the ML models are proven to violate the privacy of this training set in many unintended ways such as memorizing details about the individuals [37], leaking features of their training set [25] and contributing to re-identification problems e.g. via saving users' geodata [23].

AISec '21, November 15, 2021, Virtual Event, Republic of Korea.

ACM ISBN 978-1-4503-8657-9/21/11…$15.00
https://doi.org/10.1145/3474369.3486876

In this study we are concerned with membership inference (MI) attacks. These are attacks carried out by an adversary who has complete or partial data records and access to a trained victim ML model and wishes to know whether that data record has been used to train the victim model. MI attacks pose serious threats on the privacy of the individuals present in databases and have been successfully applied on models trained on image data, medical data, transaction records, etc. [27, 29, 35, 39].

MI attacks are carried out in a black-box setting, where adversary has no access to the internal parameters of the victim model. The assumption is that posterior vectors from the victim classifier suffice to reveal the membership status of queried points. We go one step further and show that a victim model that only reveals the labels of the queried points is still vulnerable to membership inference attacks. For this purpose, we introduce a novel membership inference attack technique called *sampling attack*. The intuition behind this attack is that the trained victim model returns more consistent labels for small random perturbations of data points that have been used to train the model.

We also employ differential privacy (DP) [9] to defend against our sampling attack. DP is becoming more and more popular as a method to protect the privacy of individuals involved in a machine learning setting. It offers flexible techniques which can be applied locally on the data [17, 31], or during the training on training parameters such as gradients [1] or the objective function [44] or it can be applied on the output of the function [24, 33]. We select DP-Stocahstic Gradient Descent (DP-SGD) [1] which is an extensively-used gradient perturbation technique and compare the protection it offers against MI attacks to randomized response mechanism which is an output perturbation technique.

Our contributions:

- We introduce our novel *sampling attack* model which performs membership inference under the severe restriction of no access to confidence scores of the attacked classifier.
- We compile a comprehensive list of all the prominent datasets in membership attack studies and compare them under unified metrics and alongside each other. This helps us better understand and analyze membership inference attacks.
- Focusing mainly on the practical implications of differential privacy (DP) [9] rather than the theoretical bounds on the privacy budget, we use DP as a method to defend against our sampling attack. We show the interplay between datasets, attack models and defenses.

The structure of this paper is as follows: We first introduce membership inference attacks in details in Section 2. We then summarize

possible defense methods against membership adversaries in Section 3. We formulate the novel sampling attack and explain the details and practical methods to implement it in Section 4 and also suggest two DP methods that could be applied to defend against this attack. We finally presents our experimental results in Section 5.

2 MEMBERSHIP INFERENCE

Membership inference attack decides whether or not a certain data point is a member of a dataset. The privacy risks of membership inference attacks were first brought into attention by Homer et al. [14] when they demonstrated that they could successfully resolve the presence of an individual in a highly-complex DNA mixture. One of their key findings is that publishing only the composite statistics over a collection of genomic data would not protect the privacy of the individuals who are members of that collection. In a follow up paper [40] they use more sophisticated test statistics to achieve better results with less prior knowledge on the victims. Similar attacks on other biomarkers such as microRNA have also been successfully performed [2].

2.1 Membership Inference on Machine Learning Models

The increased popularity of machine learning models translates to an ever-increasing need for data to train these models. This data often contains sensitive information from individuals and protecting the privacy of it is of great importance. Therefore, our focus in this paper is on membership inference attacks on machine learning models.

These models are usually trained on a set of data points x so that the function $f(x;\theta)$, which is characterized by parameters θ, is learned. In this context, the goal of the adversary is to determine: given a trained *victim* model, is a certain data point x_i a member of the training set of this model? i.e:

$$A(x;\phi) : X \rightarrow \{0 \text{ (non-member)}, 1 \text{ (member)}\} \tag{1}$$

where A is the adversary, ϕ are the parameters that the adversary utilizes and X is the space of possible data points.

Now we will explain two adversarial models that cover standard membership attacks. The first adversary requires training with the objective of finding statistical differences between members and non-member data points; whereas the second model has a prior belief on these statistical differences and no learning phase is required.

Learning Based Adversary. First introduced in [34], this adversary relies on training a *shadow* model and a binary *attack* model. Shadow models mimic the behavior of the victim model and are in the possession of the adversary. The adversary can freely study the behavior of these models as a surrogate for the victim models with restricted access. The task of the binary attack classifier is to classify its input as member/non-member of the training set.

The following steps are taken by the adversary:

(1) **Shadow model training:** Train the shadow model with a set of data point from the same distribution as the training set of the vicitm model.
(2) **Binary classifier training:** Query the trained shadow model with its training set as well as a hold-out test set. Collect the posteriors and pass them to the binary attack classifier.
(3) **Attack the victim model:** Query the victim model with the desired data points and use the trained binary classifier to decide the membership status.

The assumption about this attack is that the adversary has a black box access to the victim model and can only study the returned posterior vectors. In this method, only one shadow model and one binary classifier are used. This can be viewed as a relaxed version of multiple shadow models and multiple binary classifiers of Shokri et al. [35].

We will refer to this attack model as `LRN` adversary.

Learning Free Adversary. The dependence of the `LRN` adversary on the shadow model and the binary classifier as well as data to train these models, is an undesirable factor. For this reason, Salem et al. [34] suggest a more versatile model that requires no shadow model or attack binary classifier. A black-box access to the victim model is also assumed for this attack. The posterior vectors from the victim model are inspected and if the maximum element exceeds a calibrated threshold it will be classified as a member, otherwise non-member. This can be summarized in the following function:

$$A(x;T) = \begin{cases} 1 & \text{if} \quad \max_y \Pr(y|x) \geq T \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

which is parameterized by a single threshold T. The adversary queries the victim model by data point x and decides based on the maximum of the returned posterior vector $\Pr(y|x)$ whether or not it is a member of the training set of the vicitm model.

The name "learning free" comes from the fact that no shadow or attack classifiers are trained. We will refer to this attack as `LRN-Free` adversary.

3 DEFENSES FOR MEMBERSHIP INFERENCE ATTACKS

In the previous section we defined the membership inference attacks and described two generic models to carry out these attacks in practice. Now, we will explore methods to defend against them.

To defend against these attacks, we need to understand what factors make these attacks possible and how we can limit and paralyze the adversary. Most of the previous work on defenses against membership inference attacks can be summarized into two categories:

3.1 Generalization-based techniques

[35] was the first to define the membership inference attacks in a machine learning setting. They also identify the overfitting of the victim model as one of the main culprits for vulnerability to membership inference attacks. They hypothesize that the victim model memorizes its training set such that the posteriors show a statistical difference between the seen and hold-out data. A more comprehensive study about the correlation of overfitting to membership inference attacks can be found in [43].

These findings prompt a line of defense that relies on enforcing generalization on the victim model. [35] suggest using L2 regularization of the parameters and restricting the number of training epochs. [34] use dropout and ensemble learning to train the victim model to help it generalize better. In a slightly different approach, [28] utilizes adversarial training of the victim model in the form of a min-max game to help the model generate indistinguishable predictions on its training set and an unseen dataset.

3.2 Noising-based techniques

Adding randomness to different parameters of the victim model at different stages is one of the most natural ways to confuse any adversary. In fact, the first defenses against membership inference attacks on the genome data [40] proposes adding carefully-crafted noise to the published dataset.

Jia et al [16] suggest adding noise to the output of the victim model. They generate specially-composed noise vectors for the victim model's posteriors such that they act as adversarial examples for the attacker.

In a more formal and mathematics-driven line of work *differential privacy* is leveraged to add noise the gradients during the training of the victim model [15, 32].

In this work we mainly focus on differential privacy as a defense since it is a well-defined privacy framework and very flexible with respect to the methods that can be applied to build a differentially-private model. We will next introduce differential privacy.

Differential Privacy. Differential privacy (DP) [7–10] is a mathematical definition bounding the maximum divergence between the probability distributions of the outputs of a mechanism M when it is applied on two *adjacent datsets* d and d'. Two datasets $d, d' \in \mathcal{D}$ are adjacent when they differ in only one entry, e.g. when the data of one user is removed from one of the two identical datasets.

Definition. We formally define a differentially private algorithm $M : \mathcal{D} \rightarrow \mathcal{R}$ when the following condition holds:

$$\Pr[M(d) \in S] \leq e^{\epsilon} \Pr[M(d') \in S] + \delta \tag{3}$$

where M is a randomized algorithm with domain $\mathcal{D}$ of all possible datasets and range $\mathcal{R}$, d and d' are two adjacent datasets and $S \subseteq \mathcal{R}$ is the output of the algorithm M. The privacy parameters (privacy budget) ϵ and δ bound the probability of the output being more likely for one dataset compared to the other.

We say that the randomized algorithm M is (ϵ, δ)-differentially private if Equation. 3 holds for some parameters $\epsilon, \delta \geq 0$. It is usually suitable to set the value of $\delta = 1/|d|$ where $|d|$ is the number of data points in the dataset. So we are mostly concerned with and aim to achieve a smaller value of ϵ since it guarantees more privacy.

In Section. 4 we will explain two DP methods used to defend against our sampling attack, in more detail.

3.3 Argmax Defense

Diverging from differential privacy and the other suggested defense methods, we can take a step back and tackle this problem from a different perspective. To date, all the membership inference adversaries that we are aware of, rely on and utilize the posterior vectors from the victim model. This means that if the victim model returns the most confident 'argmax' label $k = \arg\max_k \Pr(y = k|x_i)$ instead of the full posterior, the adversary is unable to carry out the attack. We refer to this method as `argmax` defense.

Note that the `argmax` defense is not always feasible, mostly as a result of the problem design setting where the scores are required and expected by the benign user.

4 SAMPLING ATTACK AND DEFENSES

In Section 3.3 we argued that the `argmax` defense is effective against all the previously-suggested membership inference adversaries due to their dependence on the posteriors vectors. Now we will introduce our novel attack method which is designed to work under this severe restriction. We will first explain how the attack works and provide the algorithm, then pick two differentially-private mechanisms as defense against sampling attack.

4.1 Sampling Attack

Membership inference adversaries are designed to study the posterior vector, so the idea behind our sampling adversary is to reconstruct these vectors from the returned labels. We achieve this by populating a sphere around each data point with multiple perturbations of it and counting the number of perturbed samples that fall under each label. Our hypothesis is that, accuracy of the model reflects the data point's distance to the decision boundary and members of the training set lie farther from these decision boundaries compared to the non-members and data points that the model is unsure about. So at a specific perturbation level, the returned labels of perturbed member data points are less likely to change, compared to returned labels of the perturbations of the non-member data points.

Algorithm. 1 demonstrates how the sampling adversary works. The perturbation function pert() acts on each data point x to generate N perturbed samples. We assume that these perturbations provide an approximate of the data points distance to the decision boundary (proven for linear models in e.g. [11]). $\mathbf{I}$ is an identity function which builds histograms over the labels of the perturbed points. Here, c is a specific class label. We hypothesize that this histogram can be a suitable replacement for the posterior.

Algorithm 1: Sampling Attack

Input: Data points $\{x_1, \ldots, x_M\}$, neural network $\mathcal{N}(x)$ that outputs label of x, perturbation function $\text{pert}(x_i; p)$.
Parameters: number of perturbations N, perturbation scale p.
for $i \in [M]$ **do**
  $l = []$
  for $n \in [N]$ **do**
    get labels: $l_n = \mathcal{N}(\text{pert}(x_i; p))$
    l.append(l_n)
  end for
  build histograms: $\Pr(y = c|x_i) = \frac{1}{N} \sum \mathbf{I}(l = c)$
end for
Output: Posterior vectors $\Pr(y|x)$

In practice, the following steps are taken by the adversary:

(1) **Shadow model training:** Train a shadow model with data from the same distribution as the training set of the victim model.
(2) **Sampling on the shadow model:** Execute algorithm. 1 for the training set as well as a hold-out test set of the shadow model. Repeat for different perturbation levels p.
(3) **Attack the shadow model:** Using the reconstructed posteriors from the previous step, attack the shadow model with one of the conventional adversaries.
(4) **Attack the victim model:** Choose the optimum value of p according to some performance metric of the adversary. Attack the victim model with the chosen p value and the adversary from step 3.

In Table. 1 we compare the sampling attack with the `LRN` and `LRN-Free` adversaries. The sampling attack requires the training of a shadow model. If we choose `LRN-Free` adversary for steps 3 and 4 of the sampling attack, no binary classifier training is required.

4.2 Defenses for Sampling Attacks

For defense against our sampling attacks, we look at DP inspired techniques. We choose two different DP approaches:

- DP-SGD which is applied during the training of the victim model and in general guarantees the privacy of the model after the training process.
- Randomized Response (RR) mechanism which protects the output of the victim model, here the returned labels.

DP-Stochastic Gradient Descent (DP-SGD) [1]. This method achieves privacy by adding noise to the parameters of the model during the training. First, the gradients are clipped then an additive noise is applied to them:

$$\tilde{\mathbf{g}}_t(x_i) \leftarrow \mathbf{g}_t(x_i)/\max(1, \frac{\|\mathbf{g}_t(x_i)\|_2}{C}) \tag{4}$$

$$\tilde{\mathbf{g}}_t \leftarrow \frac{1}{L}(\sum_i \tilde{\mathbf{g}}_t(x_i) + \mathcal{N}(0, (\sigma C\mathbb{I})^2)) \tag{5}$$

where $\mathbf{g}_t$ is the gradient vector at epoch t, C is the clipping norm, L is the number of samples randomly chosen for calculation of the gradient and $\mathcal{N}(0, \sigma^2)$ is a Gaussian with standard deviation σ. This process can be viewed as Gaussian mechanism on top of the stochastic gradient descent. The reason behind the clipping step is to bound the $l2$-sensitivity of the gradients.

A drawback of this method is that due to the operations during the training, it slows down the training process. It is also designed to guarantee privacy for a white-box access which results in higher ϵ value for a specific accuracy, compared to other DP mechanisms that are applied e.g. after the training of the model.

Randomized Response (RR) mechanism [42]. One of the oldest privacy-ensuring mechanisms, randomized response was first used to protect participants in surveys with a "yes/no" possible answer.

We choose randomized response on the returned labels, assuming that a trained model that follows `argmax` protocol only allows adjusting of the labels. Since we have more than two possible answers (labels), we modify the randomized response with a fair coin, in he following way:

if tails: reveal the returned label

if heads: toss the fair coin again. If *heads* reveal the returned label; otherwise uniformly at random choose one of the remaining classes.

In this case, the privacy budget can be calculated as:

$$e^{\epsilon} = \frac{\Pr(\text{revealed = k}|\text{returned = k})}{\Pr(\text{revealed = k}|\text{returned = k'})} = \frac{0.75}{0.25/(C-1)} = 3(C-1)$$

where C is the total number of classes in the labels. So this mechanism is $(\ln(3C-3), 0)$ differentially private.

This DP method has the advantage of being fast and easy to apply and any model owner, even those with access to only the output labels can use this mechanism to protect the privacy of their model.

5 EXPERIMENTS

In this section we start off with a study of the traditional membership inference attacks (`LRN` and `LRN-Free` of Section. 2) on a collection of 8 different datasets that were used in the most influential membership inference attack studies (e.g. [16, 28, 34, 35]). This has two benefits; first, we are able to unify the results on all these datasets using a single metric. Second, we observe the behavor of the traditional attackers and build a baseline to compare with out novel sampling attack technique.

After these preliminary experiments we move to the main part of our experiments, i.e. the study of sampling attack technique and the DP defenses against it.

5.1 Datasets

The datasets that we begin our experiments with, are:

MNIST. The MNIST* dataset consists of 60,000 training data and 10,000 test data of grayscale images of size 28×28. These images depict handwritten digits $(0-9)$ and are centered with respect to the frame of the image.

FashionMNIST. Created by Zalando†, this dataset consists of 60,000 and 10,000 training and test set data points, respectively. These images are also 28×28 and in grayscale and represent 10 classes of fashion items such as "tops", "trousers", "sneakers", etc

CH-MNIST. This preprocessed dataset obtained from Kaggle‡ contains 5000 greyscale images of different types of tissue in colorectal cancer patients. The size of images is 64×64 and the task is to classify these images into one the 8 possible tissue categories.

CIFAR10, CIFAR100. We used CIFAR-10 and CIFAR-100§ for our experiments. Both consists of color images of size 32×32 and have 50,000 raining data and 10,000 test data. CIFAR-10 has 10 classes such as "air plane", "dogs", "cats", etc.:5000 randomly-selected images per class in its training set and 1000 randomly-selected images per class in its test set. On the other hand, CIFAR-100 has 20 super classes, each containing 5 class (in total 100 classes) of different subjects such as animals, humans and vehicles. Similar to CIFAR-10,

*http://yann.lecun.com/exdb/mnist/
†https://www.kaggle.com/zalando-research/fashionmnist
‡https://www.kaggle.com/kmader/colorectal-histology-mnist
§https://www.cs.toronto.edu/ kriz/cifar.html

Table 1: Comparison of LRN and LRN-Free to the sampling adversary. Full circles mean that the condition is required and the half full circle means flexibility in terms of training.

adversary	shadow model	binary classifier	data distribution access	posterior access	training
LRN	●	●	●	●	●
LRN-Free	-	-	-	●	-
sampling	●	-	●	-	◖

it also has 5000 randomly-selected images per class in its training set and 1000 randomly-selected images per class in its test set.

Purchase100. Purchase[¶] is dataset of shopping history of several thousand customers and the aim is to classify the customers into k different classes of shopping styles so that accurate coupon promotions can be suggested to them. This dataset has no ground truth for the labels. Similar to [34, 35], we use a simplified version of this dataset with $\sim 200,000$ data points and a 600-dimensional vector of purchases per data point where each element can take a value of either 0 or 1 (present or not present in the shopping history). Afterwards, k-means clustering algorithm [21] is used to cluster these vectors into 100 classes. We call this version of the Purchase dataset with 100 classes Purchase100.

Texas100. This includes patients' data published by the Texas Department of State Health Services[‖]. This dataset contains 6, 169 binary features of 67, 330 patients, such as diagnosis of various disease, procedures performed on the patient and other properties of each patient. We use a preprocessed version obtained from [16]. Given the input data of each patient, the task is to choose among the most suitable procedure among the available 100 most frequent ones.

Location. This dataset is the binary representation of 446 locations[**] (either visited or not visited) by users. This comes with 5, 010 data points and the task is to classify the datapoint into one of the 30 possible classes. We obtained a preprocessed version from [16].

5.2 How Successful MI Attacks Really Are

Throughout our studies we encountered many papers on membership inference attacks and/or defenses against them where each used different datasets and different metrics to evaluate the success of the attacks and defenses. This motivated us to first combine all these datasets under a unified metric to be able to compare and better understand the results. We chose LRN and LRN-Free (see Section 2) as our conventional adversaries. Below we explain the details of our experiments:

Evaluation metrics. To evaluate the performance of the adversary we chose Area Under the ROC Curve (AUC) since it is independent of the threshold that the adversary chooses to distinguish members from non-members and gives a better overview of the performance of the attacker. An AUC value of 0.5 means random guessing and implies completely unsuccessful attack, whereas AUC value of 1.0 implies a perfect attack.

Data splits. We divide each dataset into 4 equal parts $D^{\text{victim}}_{\text{train}}$, $D^{\text{victim}}_{\text{test}}$, $D^{\text{shadow}}_{\text{train}}$ and $D^{\text{shadow}}_{\text{test}}$. For this purpose, all of the training and test sets of MNIST, FashionMNIST, CH-MNINST, CIFAR10, CIFAR100 and Location datasets are combined and used. For Purchase100 and Texas100 we use a total of 80,000 and 40,000 randomly selected points, respectively.

LRN *adversary.* We use $D^{\text{shadow}}_{\text{train}}$ to train the shadow model. After training we query the shadow model by its training set and the unseen data points of $D^{\text{shadow}}_{\text{test}}$ and use these posteriors to train the binary classifier. We then test the performance of the binary classifier on the posterior prediction of the victim model when queried by its training set $D^{\text{victim}}_{\text{train}}$ and the unseen set $D^{\text{victim}}_{\text{test}}$. To avoid choosing a decision threshold, at this stage we only take the output of the sigmoid function of the binary classifier. This allows us to calculate the AUC values over decisions of the binary classifier.

LRN-Free *adversary.* For a fair comparison with the LRN adversary we only use $D^{\text{victim}}_{\text{train}}$ and $D^{\text{victim}}_{\text{test}}$ for our learning-free method. We query the trained victim model by its training data $D^{\text{victim}}_{\text{train}}$ and the hold-out $D^{\text{victim}}_{\text{test}}$ and take the maximum element of the returned posterior vectors. Similar to LRN adversary, we refrain from choosing a decision threshold and calculate AUC values over all the possible thresholds.

Victim model. For all the image datasets (MNIST, FashionMNIST, CH-MNIST, CIFAR10 and CIFAR100) we use a VGG-like [36] convolutional neural network (CNN) as shown in Figure. 1. For Location we use a fully connected neural network with layer sizes $[256, 128, 128, 30]$. For Texas 100 and Purchase100 we use a fully connected neural network with layer sized $[512, 256, 128, 100]$. We train these models with `AdamOptimizer` with learning rate of 0.001. The maximum number of epochs is set to 50 but an early stopping criterion is also set.

Shadow model. We use the same structures as the victim model for the shadow model and train the model with $D^{\text{shadow}}_{\text{train}}$ using the same procedure as the victim model.

Attack binary classifier. For the attack classifier we use a neural network with one 64-unit hidden layer and a sigmoid output for the final binary classification.

Results and discussion. Table. 2 demonstrates the evaluated performance of the LRN and LRN-Free adversaries on each dataset. We observe that the adversaries are more successful (higher AUC) on datasets with lower test accuracy. Based on this observation we have divided the table into 3 zones: green zone is where the adversaries do not achieve any meaningful success. Yellow zone is where

[¶] https://www.kaggle.com/c/acquire-valued-shoppers-challenge/data
[‖] https://www.dshs.texas.gov/THCIC/Hospitals/Download.shtm
[**] https://sites.google.com/site/yangdingqi/home/foursquare-dataset

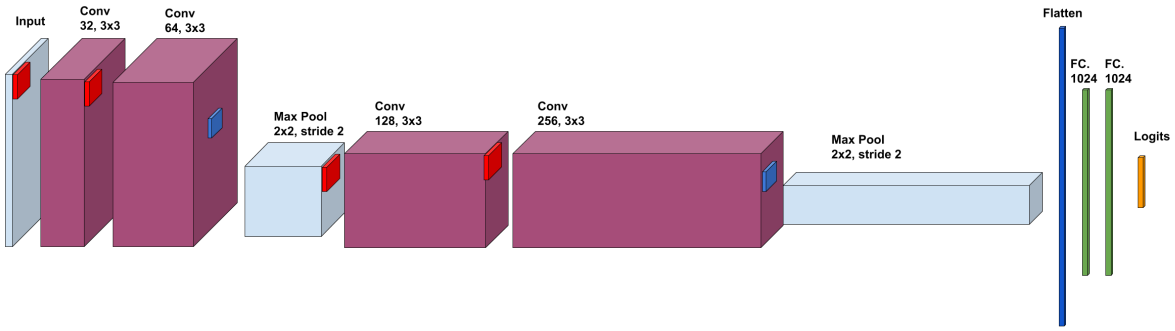

Figure 1: The structure of the CNN used for all the image datasets. All the red squares indicate 3×3 convolution filters and the blue squares show 2×2 pooling with stride of 2. In the end there are two fully-connected (FC) layers.

the adversary starts to pick up the differences between members and non-members and finally the red zone is where the advarsaries have acceptable performance. We also symbolically show that with the argmax defense, the performance of both adversaries for all datasets would drop to chance level.

The existence of the green zone can be contributed to the acceptable generalization of the models. Keep in mind that we have used early stopping for training, unlike many previous work that have a set number of epochs for training. This prevents overfitting on the training set and can be seen as a generalized-based technique (see Section. 3). These models, that generally achieve higher test accuracy, do not overfit on their training data and output posterior vectors that are statistically homogeneous across all the data.

We can also see that the number of classes in each dataset plays a role in the success rate of adversaries. For datasets with comparable number of training points, those that have fewer number of classes are less prone to attacks. This can also be explained with the fact that it is easier for models to generalize and perform for fewer number of classes.

We noticed that some datasets that were well-protected in our experiments, that is MNIST, FashionMNIST and CH-MNIST, were reported with high success rate of the adversary in the previous literature. We relate this to a better network structure and the application of an early stopping mechanism in the training such that overfitting is avoided.

From this point on we proceed with only CIFAR10/100, Texas100, Purchase100 and Location since the AUC value for both adversaries for the remaining datasets is close to chance level and this does provide a good baseline for our further experiments.

5.3 Novel Sampling Attack and Defenses for It

With an understanding of how the conventional membership inference attacks perform in practice, we can start our experiments on the sampling attack and the defenses against it. As mentioned in the previous section, we only choose the datasets that show meaningful weaknesses towards adversaries.

The structure of the victim and shadow models are the same as the previous section.

Sampling adversary. For image datasets (i.e. CIFAR10/100) we choose $\text{pert}(x; p) = x + \mathcal{N}(0, p^2)$ which is an additive Gaussian noise to each pixel of the image in each channel. We chose $p = \{i \times 0.01 | 0 \leq i \leq 20, i \in \mathbb{N}\}$.

For binary datasets (Location, Texas100, Purchase100) $\text{pert}(x; p) = flip(x)$. So we flip the values of each dimension with a probability p to the other value, that is, we flip 1 to 0 and 0 to 1. The steps of perturbation are chosen such that $p = \{i \times 0.005 | 0 \leq i \leq 20, i \in \mathbb{N}\}$

For all of the datasets, we chose to generate $N = 100$ perturbed samples for the attack.

For steps 3 and 4 of the sampling adversary (see Section 4) We choose `LRN-Free` as the conventional adversary. This is based on our findings from the previous section that showed no significant difference between the performance of `LRN` and `LRN-Free` . `LRN-Free` is more versatile and reduces one extra step of the training an attack binary classifier for the sampling adversary. The structure of the `LRN-Free` is the same as the previous section.

DP-SGD. We use DP-SGD as the parameter perturbation mechanism against this adversary. We pick the optimum value of noise level based on the performance of the adversary on a shadow model and run the sampling attack on the model trained with this noise level of DP-SGD.

Randomized response. We apply the RR mechanism as described in Section. 4. For this defense we can also calculate the expected drop of the accuracy due to the application of the RR mechanism:

$$\text{accuracy} = \frac{n_T}{n_T + n_F}$$

$$\rightarrow \mathbb{E}[\text{accuracy}_{DP}] = \frac{0.75 * n_T}{n_T + n_F} + \frac{0.25/(C-1) * n_F}{n_T + n_F}$$

$$= 0.75 * accuracy + \frac{0.25}{C-1}(1 - accuracy)$$

where n_T and n_F indicate the number of points which correctly match the ground truth labels and the number of points which deviate from the ground truth labels, respectively. And $n_T + n_F$ is the total number of data points that the model was tested on.

Results and discussion. In Figure. 2 we show the performance of the sampling adversary for different perturbation levels and

Table 2: Test accuracies versus performance of the LRN and LRN-Free attacker for all of the datasets. The last column symbolically shows that when using the argmax defense, both LRN and LRN-Free adversaries fail to perform.

#classes/dataset	accuracy$_{D^{victim}_{test}}$	AUC$_{\text{LRN}}$	AUC$_{\text{LRN-Free}}$	w/ argmax
10 MNIST	0.98	0.50	0.51	0.5
10 FashionMNIST	0.88	0.505	0.505	0.5
10 CH-MNIST	0.76	0.52	0.505	0.5
100 Purchase100	0.78	0.60	0.58	0.5
10 CIFAR10	0.69	0.60	0.59	0.5
30 Location	0.61	0.81	0.88	0.5
100 Texas100	0.56	0.72	0.67	0.5
100 CIFAR100	0.35	0.76	0.75	0.5

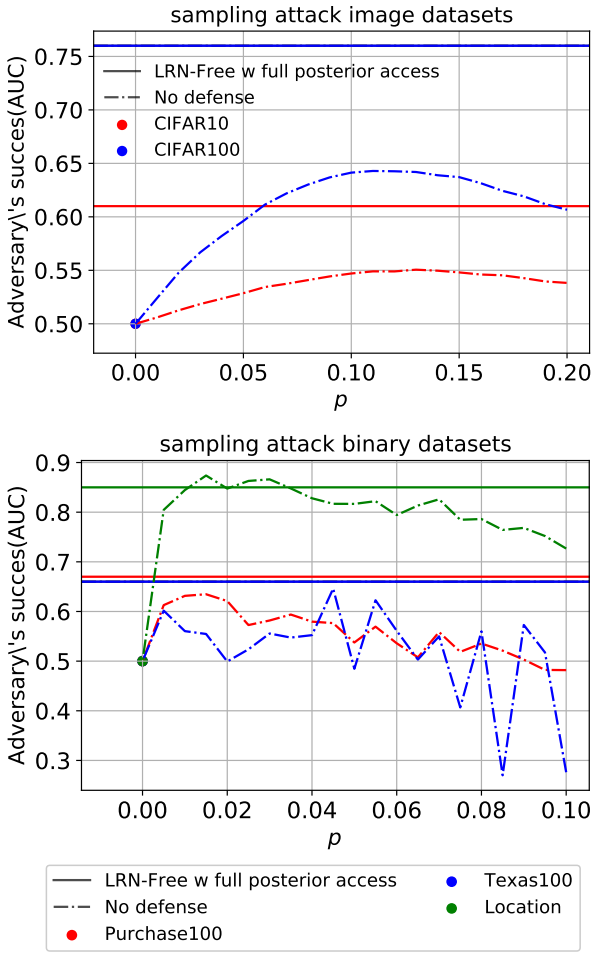

Figure 2: Sampling attack versus the LRN adversary with full posterior access for binary and image datasets. On x-axis the perturbation scale is shown

Table 3: Best p value for sampling attack for each dataset

	CIFAR10	CIFAR100	Purchase100	Texas100	Location
p^*	0.13	0.11	0.015	0.005	0.015

Table 4: Sampling attack with the best p value on the victim model.

Dataset	w/ access	sampling	DP-SGD	RR
CIFAR10	0.59	0.55	0.51	0.53
CIFAR100	0.78	0.66	0.51	0.61
Purchase100	0.69	0.67	0.52	0.57
Texas100	0.64	0.63	0.51	0.59
Location	0.89	0.89	0.61	0.8

compare it to the horizontal baseline of the LRN adversary when the full posterior access is allowed.

We observe that the sampling adversary is able to gain up to 50% of its initial performance for image datasets and 100% of the performance for binary datasets. The best result is achieved for Location dataset and this could correlate with the fact that location has the fewest number of features among all datasets and we achieve more meaningful perturbations towards the boundaries of the other classes and building the histograms over the labels is more representative of the true posteriors.

Table 3 includes the best perturbation scales p with the highest AUC of adversary, found for each dataset in the shadow model setting. We see that this value is very small and particularly in the case of the binary datasets a $p < 0.015$ works best. After this range, further perturbation results in random and noisy behavior of the sampling attack.

Next, we pick the optimum perturbation level p^* of the shadow model from Table. 3 and attack the victim model. Table 4 shows the results for the LRN with access to posteriors, the sampling attack AUC and sampling attack when DP-SGD and randomized response are used. For DP-SGD we chose a noise level such that $\text{acc}_{\text{DP-SGD}}/\text{acc}_{\text{no-defense}} \geq 0.8$ for vitim model performance. Both DP-SGD and randomized response mitigate the risks of such attacks, however DP-SGD seems to protect better and the attacker's performance drops to almost chance level, for most datasets.

We then study the effect of sampling number N on the performance of the adversary on the victim model. We choose $N \in \{10, 100, 1000\}$ and calculate the AUC of the adversary for the optimum p values from Table 3. The results are shown in Table 5. As expected, higher number of queries improve the attack as they provide a better estimate of the true position of the data point to

Table 5: Effect of number of samples on attacks.

Dataset	w/ access	N=10	N=100	N=1000
CIFAR10	0.59	0.52	0.55	0.56
CIFAR100	0.78	0.60	0.66	0.68
Purchase100	0.69	0.57	0.67	0.68
Texas100	0.64	0.54	0.63	0.63
Location	0.89	0.80	0.89	0.89

Table 6: Transferring the best p between datasets

p	0.11	0.13	0.005	0.015
CIFAR10	0.55	0.54		
CIFAR100	0.64	0.63		
Purchase100			0.65	0.67
Texas100			0.63	0.54
Location			0.81	0.89

Table 7: Comparison of the sampling attack's best accuracy with the accuracy of a naive generalization adversary

	sampling attack	naive attack
CIFAR10	0.56	0.51
CIFAR100	0.67	0.76
Purchase100	0.62	0.60
Texas100	0.57	0.65
Location	0.87	0.69

the decision boundaries of the model. It is important to remember that DP guarantees degrade with the number of queries, since the adversary can eventually zero out the noise e.g. in randomized response defense. On the other hand, a victim model can detect and block multiple queries from the adversary. So in general, querying a victim model multiple times is not a desirable action and the adversary should aim to attack with the least number of points possible. We observe that the increase in performance is more noticeable between $N = 10 - 100$ than $N = 100 - 1000$, so for these datasets $N = 100$ is an acceptable and safe value.

We also show the AUC values of the attack when the optimum p of another dataset is used to attack. Table 6 shows the AUC of adversary on the victim model for different p^* of datasets. We transfer the best perturbation scale among image datasets and binary datasets, separately. These results show us that an acceptable attack performance can be achieved even when the adversary trains the shadow model for a different dataset. With this strategy, the attacker is able to train the shadow model once on a similar type of dataset and carry out attacks on other data and save on the training time.

At last, we compare the performance of our sampling attack to the naive baseline of a *generalization error* adversary as proposed in [43]. This naive adversary classifies all the correctly labeled data points as members and the incorrectly classified ones as non-members of the data set. For our same-sized dataset splits, the accuracy of this attacker can be calculated by querying the victim model:

$$\text{attack accuracy} = \frac{1}{2}(1 + \text{accuracy}_{D^{victim}_{train}} - \text{accuracy}_{D^{victim}_{test}})$$

For this purpose, we chose the best attack accuracy of our sampling attacker at the optimal p value of each dataset. The results are shown in Table 7. We observe that the naive attacker performs better than the sampling attack on CIFAR100 and Texas100 datasets. Looking back at Table 2 we see that these two datasets have the lowest test set accuracies and suffer the most from the generalization error. Another factor contributing to the poor performance of the sampling attack on Texas100 might be the feature space which is the largest among the binary datasets, making it hard to generate meaningful perturbations of data points that cross the classifier's boundaries for each label.

In real-world application where the classifiers are trained on massive datasets, we expect the generalization error and consequently the accuracy of the naive base attacker to drop. However, since the sampling attack works based on the assumption that data points that are unseen by the model should be located closer to decision boundaries, we expect that our attack model would still be applicable to real world models.

6 RELATED WORK

Membership inference. Membership inference attacks have been first suggested as a concern for medical data privacy [2, 14, 40]. In 2017, Shokri et al. [35] studied the membership inference attacks on machine learning models. They showed that machine learning models also suffer from memorization of their training data and an adversary can infer the membership status of the data with the help of a few shadow models and attack classifiers. They propose overfitting of the machine learning models on their training set and inefficient generalization as a possible cause of this issue. [43] and [22] carry out extensive studies on the relationship between overfitting and the risk of MI attacks.

Later, Salem et al. [34] suggested relaxing the assumptions made in [35] to construct more versatile adversaries that achieved comparable performance.

So far MI attacks on machine learning models have been studied on a variety of tasks ranging from attacks on audio recordings and natural language processing [19, 26, 38], aggregate location data [30] and smart meters data [3] to generative models [4, 12, 13, 20] as well as in collaborative and decentralized machine learning settings [25, 29].

In all of the previously-mentioned works, the access of the MI adversary to the full posterior is crucial. Machine learning attacks under limited access to the posterior vectors have been studied before, for example, [6] propose evasion attacks when the score of the detector is not accessible to the adversary. To fool the detector, they generate perturbations of the malicious sample and attempt to find the first perturbed sample that traverses the malicious/benign boundary of the detector. However, this evasion attack does not directly depend on the posterior vectors and the goal is to only evade the detector. In an attempt to relax the assumption of posterior access for membership inference attacks, [43] suggested a naive

attacker that performs proportional to the generalization gap of the victim model.

Concurrent to our work, [18] and [5] study methods to attack the models with access to labels only. [18] suggests transfer-based and perturbation-based attacks. Transfer-based attack relies on a shadow model to mimic the behavior of the victim model and return similar posteriors when queried by the attacked data. The perturbation-based method classifies data points as members/non-members based on the amount of perturbation needed to change the label. Unlike our sampling attack, they do not aim to reconstruct the posteriors. [5] investigates different perturbation functions and builds on the assumption that the confidence scores are proportional to distances to the decision boundary.

Defenses against membership inference. Naturally, the attempts to protect against membership inference attacks dates back to when the risks were first exposed. Wang et al. [40] proposed adding carefully-crafted noise to the published dataset that needed to be protected. Later [41] they suggested a method that splits an aggregate of data into what is common among most participants and the sensitive part and only publishing the common part.
Based on the hypothesis that machine learning models leak the membership information due to overfitting on their training set, Shokri et al. [35] suggest using L2 regularization of the parameters and also restricting the number of passes on the training data during the training (epochs). [34] suggest using dropout and ensemble learning. As a method of regularization, adversarial training is used in [28].
Another approach to defending against MI attacks is adding noise to the output, or to the parameters of the model during the training. Adding noise helps by making the distribution over the training data set and the non-training data converge and become indistinguishable. Jia et al. [16] add carefully composed noise vectors to the posteriors of the victim model such that it acts as an adversarial example for the adversary's attack classifier. In a more mathematically-driven line of work differential privacy (DP [7–10]) is leveraged to add noise to the gradients during the training of the model [5, 15, 32]. [15] and [5] demonstrate that trying to preserve the test accuracy results in privacy budgets that are not formally acceptable by DP ($\epsilon \gg 1$). In this paper we use DP only as a practical method to protect against our specific MI attack technique with its restricted assumption and are not concerned with the strict theoretical guarantees of DP.

7 DISCUSSION AND CONCLUSION

By evaluating membership inference attacks over a large scope of different dataset, we highlight issues with the rapidly developing research thread of membership inference. Often attack performance is assessed with different type of models, while such performance cannot be seen in isolation of data and training procedure. We urgently need more transparency in reporting membership attack performance in order to be really in a position to compare and measure progress in this area.

We investigate the *argmax* defense, which previously was thought to prevent membership inference attacks. While for previous attacks this is true, we show a new sampling attack that attracts by repeated querying of the model surrogate information of the model that is able to recover a large fraction of the attack performance. In turn, we also present a modification of the randomized response defense, that is in part capable of mitigating the new attack vector. We also study the effect DP-SGD in protecting of the model against our sampling attacks.

REFERENCES

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. 308–318.

[2] Michael Backes, Pascal Berrang, Mathias Humbert, and Praveen Manoharan. 2016. Membership privacy in MicroRNA-based studies. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. 319–330.

[3] Niklas Buescher, Spyros Boukoros, Stefan Bauregger, and Stefan Katzenbeisser. 2017. Two is not enough: Privacy assessment of aggregation schemes in smart metering. *Proceedings on Privacy Enhancing Technologies* 2017, 4 (2017), 198–214.

[4] Dingfan Chen, Ning Yu, Yang Zhang, and Mario Fritz. 2019. *GAN-Leaks: A Taxonomy of Membership Inference Attacks against GANs.* arXiv:1909.03935. https://arxiv.org/abs/1909.03935https://arxiv.org/pdf/1909.03935.pdf

[5] Christopher A Choquette Choo, Florian Tramer, Nicholas Carlini, and Nicolas Papernot. 2020. Label-Only Membership Inference Attacks. *arXiv preprint arXiv:2007.14321* (2020).

[6] Hung Dang, Yue Huang, and Ee-Chien Chang. 2017. Evading classifiers by morphing in the dark. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. 119–133.

[7] Cynthia Dwork. 2011. A firm foundation for private data analysis. *Commun. ACM* 54, 1 (2011), 86–95.

[8] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*. Springer, 265–284.

[9] Cynthia Dwork, Aaron Roth, et al. 2014. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science* 9, 3–4 (2014), 211–407.

[10] Cynthia Dwork, Adam Smith, Thomas Steinke, Jonathan Ullman, and Salil Vadhan. 2015. Robust traceability from trace amounts. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*. IEEE, 650–669.

[11] Justin Gilmer, Nicolas Ford, Nicholas Carlini, and Ekin Cubuk. 2019. Adversarial examples are a natural consequence of test error in noise. In *International Conference on Machine Learning*. PMLR, 2280–2289.

[12] Jamie Hayes, Luca Melis, George Danezis, and Emiliano De Cristofaro. 2019. LOGAN: Membership inference attacks against generative models. *Proceedings on Privacy Enhancing Technologies* 2019, 1 (2019), 133–152.

[13] Benjamin Hilprecht, Martin Härterich, and Daniel Bernau. 2019. Monte carlo and reconstruction membership inference attacks against generative models. *Proceedings on Privacy Enhancing Technologies* 2019, 4 (2019), 232–249.

[14] Nils Homer, Szabolcs Szelinger, Margot Redman, David Duggan, Waibhav Tembe, Jill Muehling, John V Pearson, Dietrich A Stephan, Stanley F Nelson, and David W Craig. 2008. Resolving individuals contributing trace amounts of DNA to highly complex mixtures using high-density SNP genotyping microarrays. *PLoS genetics* 4, 8 (2008).

[15] Bargav Jayaraman and David Evans. 2019. Evaluating differentially private machine learning in practice. In *28th {USENIX} Security Symposium ({USENIX} Security 19)*. 1895–1912.

[16] Jinyuan Jia, Ahmed Salem, Michael Backes, Yang Zhang, and Neil Zhenqiang Gong. 2019. MemGuard: Defending against Black-Box Membership Inference Attacks via Adversarial Examples. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. 259–274.

[17] Peter Kairouz, Sewoong Oh, and Pramod Viswanath. 2014. Extremal mechanisms for local differential privacy. In *Advances in neural information processing systems*. 2879–2887.

[18] Zheng Li and Yang Zhang. 2020. Label-Leaks: Membership Inference Attack with Label. *arXiv preprint arXiv:2007.15528* (2020).

[19] Gaoyang Liu, Chen Wang, Kai Peng, Haojun Huang, Yutong Li, and Wenqing Cheng. 2019. Socinf: Membership inference attacks on social media health data with machine learning. *IEEE Transactions on Computational Social Systems* 6, 5 (2019), 907–921.

[20] Kin Sum Liu, Bo Li, and Jie Gao. 2018. Generative model: Membership attack, generalization and diversity. *CoRR, abs/1805.09898* (2018).

[21] Stuart Lloyd. 1982. Least squares quantization in PCM. *IEEE transactions on information theory* 28, 2 (1982), 129–137.

[22] Yunhui Long, Vincent Bindschaedler, Lei Wang, Diyue Bu, Xiaofeng Wang, Haixu Tang, Carl A. Gunter, and Kai Chen. 2018. Understanding Membership Inferences on Well-Generalized Learning Models. *CoRR* abs/1802.04889 (2018). arXiv:1802.04889 http://arxiv.org/abs/1802.04889

[23] Mohamed Maouche, Sonia Ben Mokhtar, and Sara Bouchenak. 2017. Ap-attack: a novel user re-identification attack on mobility datasets. In *Proceedings of the 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*. 48–57.

[24] Frank McSherry and Kunal Talwar. 2007. Mechanism design via differential privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*. IEEE, 94–103.

[25] Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. 2019. Exploiting unintended feature leakage in collaborative learning. In *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 691–706.

[26] Yuantian Miao, Ben Zi Hao Zhao, Minhui Xue, Chao Chen, Lei Pan, Jun Zhang, Dali Kaafar, and Yang Xiang. 2019. The audio auditor: Participant-level membership inference in voice-based iot. *arXiv preprint arXiv:1905.07082* (2019).

[27] A Naranyanan and V Shmatikov. 2008. Robust de-anonymization of large datasets. In *Proceedings of the 2008 IEEE Symposium on Security and Privacy*.

[28] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2018. Machine learning with membership privacy using adversarial regularization. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. 634–646.

[29] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2019. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 739–753.

[30] Apostolos Pyrgelis, Carmela Troncoso, and Emiliano De Cristofaro. 2017. Knock knock, who's there? Membership inference on aggregate location data. *arXiv preprint arXiv:1708.06145* (2017).

[31] Zhan Qin, Yin Yang, Ting Yu, Issa Khalil, Xiaokui Xiao, and Kui Ren. 2016. Heavy hitter estimation over set-valued data with local differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. 192–203.

[32] Md Atiqur Rahman, Tanzila Rahman, Robert Laganière, Noman Mohammed, and Yang Wang. 2018. Membership Inference Attack against Differentially Private Deep Learning Model. *Transactions on Data Privacy* 11, 1 (2018), 61–79.

[33] Vibhor Rastogi, Michael Hay, Gerome Miklau, and Dan Suciu. 2009. Relationship privacy: output perturbation for queries with joins. In *Proceedings of the twenty-eighth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*. 107–116.

[34] Ahmed Salem, Yang Zhang, Mathias Humbert, Mario Fritz, and Michael Backes. 2018. ML-Leaks: Model and Data Independent Membership Inference Attacks and Defenses on Machine Learning Models. *Annual Network and Distributed System Security Symposium (NDSS)* (2018). arXiv:1806.01246 http://arxiv.org/abs/1806.01246

[35] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE, 3–18.

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

[37] Congzheng Song, Thomas Ristenpart, and Vitaly Shmatikov. 2017. Machine learning models that remember too much. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. 587–601.

[38] Congzheng Song and Vitaly Shmatikov. 2019. Auditing data provenance in text-generation models. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 196–206.

[39] Liwei Song, Reza Shokri, and Prateek Mittal. 2019. Privacy risks of securing machine learning models against adversarial examples. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. 241–257.

[40] Rui Wang, Yong Fuga Li, XiaoFeng Wang, Haixu Tang, and Xiaoyong Zhou. 2009. Learning your identity and disease from research papers: information leaks in genome wide association study. In *Proceedings of the 16th ACM conference on Computer and communications security*. 534–544.

[41] Rui Wang, XiaoFeng Wang, Zhou Li, Haixu Tang, Michael K Reiter, and Zheng Dong. 2009. Privacy-preserving genomic computation through program specialization. In *Proceedings of the 16th ACM conference on Computer and communications security*. 338–347.

[42] Stanley L Warner. 1965. Randomized response: A survey technique for eliminating evasive answer bias. *J. Amer. Statist. Assoc.* 60, 309 (1965), 63–69.

[43] Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. 2018. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st Computer Security Foundations Symposium (CSF)*. IEEE, 268–282.

[44] Jun Zhang, Zhenjie Zhang, Xiaokui Xiao, Yin Yang, and Marianne Winslett. 2012. Functional mechanism: regression analysis under differential privacy. *arXiv preprint arXiv:1208.0219* (2012).