PRIVACY AUDITING OF MACHINE LEARNING USING MEMBERSHIP INFERENCE ATTACKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Membership inference attacks are used as an auditing tool to quantify the private information that a model leaks about the individual data points in its training set. Membership inference attacks are influenced with different uncertainties that an attacker has to resolve about training data, training algorithm, and the underlying data distribution. Thus attack success rates, of many attacks in the literature, do not precisely capture the information leakage of models about their data, as they also reflect other uncertainties that the attack algorithm has. In this paper, we present a hypothesis testing framework that can explain the implicit assumptions and also the simplifications made in the prior work. We also derive new attack algorithms from our framework that can achieve a high AUC score while also highlighting the different factors that affect their performance. Our algorithms capture a very precise approximation of privacy loss in models, and can be used as a tool to perform an accurate and informed estimation of privacy risk in machine learning models. We provide a thorough empirical evaluation of our attack strategies on various machine learning tasks trained on benchmark datasets.

1 INTRODUCTION

Machine learning systems have come under intense scrutiny of the regulatory authorities in the past few years. Especially, the article 35 of GDPR [1] emphasizes on the importance of performing a data protection impact assessment (DPIA) when building machine learning systems on sensitive data. Veale et al. (2018) even argue that machine learning models could be considered personal data due to their susceptibility to inference attacks that can recover sensitive information about training data just from the models. Membership inference attacks (Homer et al., 2008; Dwork et al., 2015; Shokri et al., 2017) and reconstruction attacks (Dinur & Nissim, 2003; Song et al., 2017; Carlini et al., 2020) are the main inference attacks that highlight, and can quantify, the privacy risk of releasing aggregate information computed on sensitive data (Dwork et al., 2017). The focus of this paper will be on membership inference attacks for measuring privacy risk. Organizations such as the ICO (UK) and NIST (US) have highlighted membership inference as a potential confidentiality confidentiality violation and privacy threat to the training data (Murakonda & Shokri, 2020). This has lead to the development of open-source tools, [2] capabilities in widely-used ML libraries [3] for measuring privacy risk from machine learning models using membership inference attacks. See Appendix B for a comprehensive survey of the related work.

Although the approach of quantifying privacy risk through membership inference attacks is gaining traction, the attack success, as measured by a lot of works, cannot be completely attributed to information leakage from the models and hence their privacy risk. Various factors such as the distribution of training data, difference in distributions of the train and test data may provide an over-estimate or under-estimate of the actual privacy risk from the model (Erlingsson et al., 2019; Humphries et al., 2020). Theoretical analyses that connect the success of membership inference to privacy risk through the framework of differential privacy avoid this issue by slightly modifying how the attack performance is measured (Yeom et al., 2018; Jagielski et al., 2020; Nasr et al., 2021; Malek et al.,

[1] https://gdpr-info.eu/art-35-gdpr/

[2] https://github.com/privacytrustlab/ml_privacy_meter

[3] https://blog.tensorflow.org/2020/06/introducing-new-privacy-testing-library.html

2021). All these works rely on measuring privacy risk through the performance of membership inference attacks but the subtle differences in how the attack is formulated mean that they measure different types of information leakage. When auditing machine learning models, we need to pay attention to what leakage we are measuring and hence how membership inference attacks are formulated. In this work, we provide a framework that can help reason about how different factors, beyond information leakage from the model, affect the performance of membership inference attacks, and how to design attacks that cancel out the effects of other factors. Hence, it can be used for a more accurate privacy analysis of machine learning models through informed choices about the different factors that should be controlled while auditing these models. Below, we first briefly review the related work in this domain, explain how different works measure privacy risk, and then explain the contributions of our work.

In this work, we provide a framework for understanding the relationship between success of membership inference attacks and information leakage from machine learning models. We do this by explaining about the uncertainties that an attacker has to reduce for performing the attack and highlighting the uncertainties the released model helps resolve (via leakage about its training data). The framework formulates membership inference as a game, measures the performance of the attack as the trade-off between type-I and type-II errors, and relies on hypothesis testing for crafting attack strategies. We also derive multiple attack strategies (existing and new) from the same hypothesis test formulation through different approximations. The goal of deriving these attack strategies is to illustrate how different factors such as the attacker dataset, shadow models, distribution of the train and test datasets affect the privacy risk estimated using different attack strategies. We also provide an empirical analysis of these attacks on multiple datasets, showing the effectiveness of these strategies. Privacy auditing has a different computational cost using different attack strategies. The cost could be measured in terms of the number of model queries and the number of extra trained models needed for different attacks. In this paper, we focus on formalizing the attacks and analyzing their performance. A very important follow-up work is to design approximation algorithms for the attacks to optimize their computational cost while preserving their power.

2 Attack Framework

Our objective is to design a framework that enables auditing the privacy loss of machine learning models in the black-box setting (where only model outputs —and not their parameters or internal computations— are observable). Our framework has three elements: (i) the inference game as the evaluation setup; (ii) the indistinguishably metric to measure the privacy risk, and (iii) the construction of membership inference attack as hypothesis testing. The notion of privacy underlying our framework is primarily based on differential privacy, and multiple pieces of this framework are generalizations of existing inference attacks against machine learning algorithms. Instead of designing another attack, we present the important design choices for constructing and evaluating membership inference attacks, for the purpose of having a precise privacy auditing. We identify different sources of uncertainty that influence the error of inference attacks, and can lead to miscalculation of the privacy loss of a model.

We quantify privacy loss of a model in a hypothetical **inference game** between a challenger and an adversary. The challenger first chooses a private training set D, and trains a model θ on it using a training algorithm $\mathcal{T}$. The challenger samples a random data point (x_1, y_1) from the training set (a member), and a random data point (x_0, y_0) from the data *population* outside the training set (a non-member). He then randomly selects one of the two (member or non-member) $b \sim \{0, 1\}$ with probability $\frac{1}{2}$, and shares the selected data point (x_b, y_b) and the model's output $f(x_b; \theta)$ with the adversary. The adversary's task is to determine if the data point is a member or not (i.e., guess b).[4]

[4]We need to emphasize that there is a difference between the privacy loss of a machine learning *algorithm* and that of the specific *models* trained with the algorithm. A model is a given instance of a training algorithm, thus its leakage needs to be computed with respect to the individual data records in its specific training set. This subsequently means that the privacy loss of an algorithm varies depending on the randomness in sampling of its training data, and the randomness of the training algorithm. One can analyze the privacy loss of an algorithm as, for example, its worst case privacy loss (as in differential privacy). Differentially private algorithms enforce an upper bound on the privacy loss of an algorithm over all models with respect to all possible training data. Thus, in the inference game for differential privacy, the privacy loss of the training *algorithms*, would be the worst case privacy loss over all choices of D, (x_0, y_0), and (x_1, y_1) in our inference game for model privacy.

We use an **indistinguishability** measure (which is the basis of differential privacy) to define privacy of individual training data of a model. According to this measure, the *privacy loss* of the model with respect to its training data is the adversary's success in distinguishing between the two possibilities $b = 0$ vs $b = 1$ over multiple repetitions of the inference game. Naturally, the inference attack is a **hypothesis test**, and adversary's error is composed of the false positive (i.e., inferring a non-member as member) and false negative of the test. In practice the error of adversary in each round of the inference game depends of multiple factors:

- The true leakage of the model about the target data (x_b, y_b)
- The uncertainty (belief) of attack algorithm about the population data
- The adversary's uncertainty about the training algorithm $\mathcal{T}$
- The attack dependency on the target data (x_b, y_b), and the model θ
- The uncertainty about all training data except the target data (x_b, y_b)

In the ideal setting, we only want the attack error to be dependent on the true leakage of the model about the target data (i.e., whether the same model trained with and without (x_b, y_b) are distinguishable from each other). To this end, and to cancel out the effect of other uncertainties, an attack algorithm and the evaluation setup for the inference game need to be designed based on the following principle: The population data used for constructing the attack algorithm, and evaluating the inference game, need to be similar, in distribution, to the training data. This is to minimize the impact of prior belief (what could have been sampled for the training set) in the performance of the inference attack. This is not hard to achieve as all the process (of constructing the hypothesis testing attack, and evaluating it) is controlled by the auditor. By violating this principle, we might overestimate the privacy loss (by making the test dependent on a distinct prior knowledge) or underestimate the privacy loss (by evaluating the inference attack on a population data distribution for which it was not constructed).

Another crucial requirement is that the privacy audit needs to output a detailed report, which captures the uncertainty of the attack. Reporting one number as the accuracy of the attack, as it is mostly reported in the literature, is not an informative report. Given the attack being a hypothesis test, the audit report needs to include the analysis of the error versus power of the test: if we can tolerate a certain level of false positive rate in the inference attack, how much would be the true positive rate of the attack, over the random samples from member and non-member data? The area under the curve for such an analysis reflects the chance that the membership of a random data point from the population or the training set can be inferred correctly.

3 Constructing Membership Inference Attacks

The adversary in the membership inference game can observe the output of a target machine learning model θ, trained on unknown dataset D. He also gets a precise target data point z_{tar} as input, and is expected to output 0 or 1 to guess whether the sample z_{tar} is in the dataset D or not. We use likelihood ratio test (LRT) as the most powerful criterion for choosing among membership hypotheses, under the following assumptions.

1. The adversary knows the underlying data distribution $\pi(z)$ over population $z \in D_{pop}$.
2. A randomized training algorithm $\mathcal{T} : D \mapsto \theta$ takes in a training dataset D, which consists of i.i.d. samples from the data population, and produces a model θ that incurs a low loss $\sum_{(x,y)\in D} \ell(\theta, x, y)$ on dataset D.

We denote $P(\theta|D)$ to be the posterior distribution of trained model θ given training dataset D,

$$P(\theta|D) = \frac{e^{-\frac{1}{T}\sum_{(x,y)\in D}\ell(\theta,x,y)}}{\int e^{-\frac{1}{T}\sum_{(x,y)\in D}\ell(\theta,x,y)} d\theta}, \tag{1}$$

where T is a temperature constant specified by the training algorithm $\mathcal{T}$.

Definition 3.1 (Approximated LRT for membership inference) *Let (θ, z) be random samples from the joint distribution of target model and target data point, specified by one of the following*

membership hypotheses.

$$H_0 : D \xleftarrow{n\ i.i.d. samples \sim \pi(z)} D_{pop}, \theta \sim \mathcal{T}(D), z \xleftarrow{sample \sim \pi(z)} D_{pop} \tag{2}$$

$$H_1 : D \xleftarrow{n\ i.i.d. samples \sim \pi(z)} D_{pop}, \theta \sim \mathcal{T}(D), z \xleftarrow{sample} D \tag{3}$$

The likelihoods function of hypothesis H_0 and H_1, given observed target model θ and target data point z, is as follows (detailed derivations are in the Appendix).

$$L(H_0|\theta, z) = P_{H_0}(\theta, z) = \pi(z) \cdot \mathbf{E}_{D \sim \pi^n}[P(\theta|D)] \tag{4}$$

$$L(H_1|\theta, z) = P_{H_1}(\theta, z) = \pi(z) \cdot \mathbf{E}_{D' \sim \pi^{n-1}}[P(\theta|D' \cup z)] \tag{5}$$

Therefore, the LRT statistics can be computed as follows (detailed derivations are in the Appendix).

$$LR(\theta, z) = \frac{L(H_0|\theta, z)}{L(H_1|\theta, z)} = \frac{\mathbf{E}_{D \sim \pi^n}[P(\theta|D)]}{\mathbf{E}_{D' \sim \pi^{n-1}}[P(\theta|D' \cup z)]} \tag{6}$$

$$\approx e^{\frac{1}{T}\ell(\theta, x_z, y_z)}, \tag{7}$$

The LRT hypothesis test rejects H_0 when the LRT statistic is small. By equation 7, the rejection region $\{(\theta, z) : LR(\theta, z) \leq c\}$ can be approximated as follows.

$$\left\{(\theta, z) : \ell(\theta, x_z, y_z) \leq T \cdot \log c\right\} \tag{8}$$

The above approximated LRT strategy compares a constant threshold $T \cdot \log c$ with the loss $\ell(\theta, x_z, y_z)$ of the target model θ on the target data point z. However, by equation 6, a more accurate LRT attack strategy would compare the loss $\ell(\theta, x_z, y_z)$ with a threshold function $T \cdot \log\left(\frac{c \cdot \mathbf{E}_{D' \sim \pi^{n-1}}[P(\theta|D' \cup z)]}{\mathbf{E}_{D \sim \pi^n}[P(\theta|D)]}\right)$ that depends on the target model θ and the target sample z. Therefore, the attack in equation 8 with constant threshold c_α can overly simplify the LRT, thus lowering its attack performance. This motivates the design of our attacks with model-dependent and sample-dependent thresholds, with the objective of improving the attack performance.

Our general template for attack construction. Building on the approximate LRT equation 8, we derive the following variant of sample-dependent and (or) model dependent attack strategy.

$$\text{If } \ell(\theta, x_z, y_z) \leq c_\alpha(\theta, x_z, y_z), \text{ reject } H_0, \tag{9}$$

where $c_\alpha(\theta, x_z, y_z)$ is a threshold function chosen by the attacker under α tolerance of false positive rate (FPR), i.e., $c_\alpha(\theta, x_z, y_z)$ satisfies the following equation which controls false positive rate.

$$\mathbf{E}_{P_{H_0}(\theta, x_z, y_z)}\left[\mathbf{1}_{\ell(\theta, x_z, y_z) \leq c_\alpha}\right] = \alpha \tag{10}$$

In the following attacks, we approximate the joint distribution $P_{H_0}(\theta, z)$ with empirical distribution over its samples. This facilitates solving equation 10, which give valid attack threshold $c_\alpha(\theta, x_z, y_z)$.

3.1 Attack S: MIA via Shadow models

We first formalize a shadow model membership inference attack S based on Shokri et al. (2017), that effectively uses label-dependent attack threshold $c_\alpha(y_z)$, as follows.

$$\text{If } \ell(\theta, x_z, y_z) \leq c_\alpha(y_z), \text{ reject } H_0, \tag{11}$$

where the threshold function $c_\alpha(y_z)$ satisfies equation 10 and ensures false positive rate α. To approximate the joint distribution $P_{H_0}(\theta, x_z, y_z)$ in equation 4, the attacker samples the following set S of shadow models and shadow data points from the joint distribution $P_{H_0}(\theta, z)$.

$$S = \cup_{i=1,2,\cdots} \{(\theta_i, z_1^i), (\theta_i, z_2^i), \cdots\} \tag{12}$$

$$\forall i = 1, 2, \cdots, \quad \theta_i \sim \mathcal{T}(D_i)\ , D_i \xleftarrow{n\ i.i.d. samples \sim \pi(z)} D_{pop} \tag{13}$$

$$\forall i = 1, 2, \cdots, \quad z_1^i, z_2^i, \cdots i.i.d. \sim \pi(z) \tag{14}$$

The shadow models $\theta_s = \{\theta_i\}_{i=1,2,\cdots}$ trained on population datasets approximate the distribution $\mathbf{E}_{D \sim \pi^n}[P(\theta|D)]$ in equation 4, and the shadow points $z_1^i, z_2^i, \cdots$ for the i-th shadow model

approximate the population data distribution $\pi(z)$ in equation 4. Therefore, we approximate the joint distribution $P_{H_0}(\theta, z)$ in equation 10 with the empirical distribution over set S, as follows.

$$\frac{|\{(\theta, z) \in S : \ell(\theta, x_z, y_z) \leq c_\alpha(y_z)\}|}{|S|} = \alpha \tag{15}$$

One sufficient condition for equation 15 to hold is that for any fixed data label $y \in Y$, we have

$$\frac{|\{(\theta, z) \in S : \ell(\theta, x_z, y_z) \leq c_\alpha(y_z) \text{ and } y_z = y\}|}{|\{(\theta, z) \in S : y_z = y\}|} = \alpha \tag{16}$$

By solving equation 16 for every $y \in Y$, we compute that $c_\alpha(y)$ equals the the α-percentile for the histogram of loss values $\ell(\theta, x_z, y_z)$ over samples $\{(\theta, z) \in S : y_z = y\}$. This recovers the class-dependent attack threshold function $c_\alpha(y_z)$ that we use in Attack S.

3.2 Attack P: model-dependent MIA via Population data

We design a new membership inference Attack P that uses attack threshold function $c_\alpha(\theta)$ dependent on the target model θ. The rationale for this design is to construct an inference attack which exploits the similar statistics as in Attack S, in a more accurate way by computing it on the target model, yet with less computations (without the need to train shadow models). The hypothesis test is as follows.

$$\text{If } \ell(\theta, x_z, y_z) \leq c_\alpha(\theta), \text{ reject } H_0, \tag{17}$$

where the threshold function $c_\alpha(\theta)$ satisfies equation 10 and ensures false positive rate α. Using equation 23, we rewrite the false positive rate in equation 10 as follows.

$$\mathbf{E}_{P_{H_0}(\theta)} \left[\mathbf{E}_{\pi(z)} \left[\mathbf{1}_{\ell(\theta, x_z, y_z) \leq c_\alpha(\theta)} \right] \right] = \alpha \tag{18}$$

One sufficient condition for equation 18 to hold is that, for any fixed target model θ, we have

$$\mathbf{E}_{\pi(z)} \left[\mathbf{1}_{\ell(\theta, x_z, y_z) \leq c_\alpha(\theta)} \right] = \alpha \tag{19}$$

To approximate the data distribution $\pi(z)$ in equation 19, the attacker samples the following set P of population data points from distribution $\pi(z)$.

$$P = \{z_i\}_{i=1,2,\cdots}, \text{ where } z_1, z_2, \cdots i.i.d. \sim \pi(z) \tag{20}$$

Therefore, we approximate $\pi(z)$ in equation 19 with the empirical distribution over samples $z \in P$.

$$\frac{|\{z \in P : \ell(\theta, x_z, y_z) \leq c_\alpha(\theta)\}|}{|P|} = \alpha \tag{21}$$

By solving equation 21 for every fixed target model θ, we compute that $c_\alpha(\theta)$ equals the the $\alpha -$ percentile for the histogram of loss values $\ell(\theta, x_z, y_z)$ over population data samples $z = (x_z, y_z) \in P$. This recovers the model-dependent attack threshold $c_\alpha(\theta)$ for Attack P.

3.3 Attack R: sample-dependent MIA via Reference models

The privacy loss of the model with respect to the target data could be directly related to how susceptible the target data is to be memorized (e.g., being an outlier) Feldman (2020). Based on this finding, we design the membership inference Attack R that uses attack threshold function $c_\alpha(x_z, y_z)$ which depends on the target data (both its input features x_z and the label y_z). This attack is very similar to the membership inference attacks designed for summary statistics and graphical models, which use reference models to compute the probability of the null hypothesis Sankararaman et al. (2009); Murakonda et al. (2021). We define the test in our case as follows.

$$\text{If } \ell(\theta, x_z, y_z) \leq c_\alpha(x_z, y_z), \text{ reject } H_0, \tag{22}$$

where the threshold function $c_\alpha(x_z, y_z)$ satisfies equation 10 and ensures false positive rate α. Using the chain rule of joint distribution, and the independence of θ and $z = (x_z, y_z)$ under null hypothesis H_0 in equation 2, we prove that

$$P_{H_0}(\theta, x_z, y_z) = P_{H_0}(x_z, y_z) \cdot P_{H_0}(\theta | x_z, y_z) = \pi(z) \cdot P_{H_0}(\theta), \tag{23}$$

where $P_{H_0}(\theta) = \mathbf{E}_{D\sim\pi^n}[P(\theta|D)]$ is the target model distribution specified by the null hypothesis H_0. Using equation 23, we rewrite the false positive rate constraint in equation 10 as follows.

$$\mathbf{E}_{\pi(z)}\left[\mathbf{E}_{P_{H_0}(\theta)}\left[\mathbf{1}_{\ell(\theta,x_z,y_z)\leq c_\alpha(x_z,y_z)}\right]\right] = \alpha \tag{24}$$

One sufficient condition for equation 24 to hold is that, for any fixed data point $z = (x_z, y_z)$,

$$\mathbf{E}_{P_{H_0}(\theta)}\left[\mathbf{1}_{\ell(\theta,x_z,y_z)\leq c_\alpha(x_z,y_z)}\right] = \alpha \tag{25}$$

To approximate the target model distribution $P_{H_0}(\theta)$ in equation 25, the attacker then samples the following set R of reference models.

$$R = \{\theta_i\}_{i=1,2,\cdots}, \text{ where } \theta_i \sim \mathcal{T}(D_i)\ ,\ D_i \xleftarrow{n\ i.i.d. samples\sim\pi(z)} D_{pop} \tag{26}$$

Because the reference models $\theta_r = \{\theta_i\}_{i=1,2,\cdots}$ are trained on population datasets, they serve as samples from the distribution $P_{H_0}(\theta) = \mathbf{E}_{D\sim\pi^n}[P(\theta|D)]$. Therefore, we replace $P_{H_0}(\theta, z)$ in equation 25 with the empirical distribution over samples $\theta_i \in R$, as follows.

$$\frac{|\{\theta_i \in R : \ell(\theta_i, x_z, y_z) \leq c_\alpha(x_z, y_z)\}|}{|R|} = \alpha \tag{27}$$

By solving equation 27 for fixed target sample $z = (x_z, y_z)$, we compute that $c_\alpha(x_z, y_z)$ equals the the α-percentile for the histogram of loss values $\ell(\theta_i, x_z, y_z)$ over reference models samples $\theta_i \in R$. This computes the sample-dependent attack threshold $c_\alpha(x_z, y_z)$ that in Attack R.

3.4 Attack D: Model-dependent and sample dependent MIA via Distillation

Can we design an attack that takes advantage of all the information available in the target model and the target data that can increase the chance of identifying the right hypothesis? We design a membership inference Attack D whose threshold function $c_\alpha(\theta, x_z, y_z)$ depends on both the target sample z and the target model θ, as follows.

$$\text{If } \ell(\theta, x_z, y_z) \leq c_\alpha(\theta, x_z, y_z), \text{ reject } H_0, \tag{28}$$

where the threshold function $c_\alpha(\theta, x_z, y_z)$ satisfies equation 10 and ensures false positive rate α. However, the degree of freedom in the threshold function $c_\alpha(\theta, x_z, y_z)$ is still too large for us to directly solve equation 10. Therefore, we restrict $c_\alpha(\theta, x_z, y_z)$ to take the following form.

$$c_\alpha(\theta, x_z, y_z) = c_\alpha(D_\theta, x_z, y_z), \tag{29}$$

where $D_\theta = \mathcal{T}^{-1}(\theta)$ is the training dataset for target model θ. For the simplicity of derivation, let us first assume that the randomized training algorithm $\mathcal{T}$ has a deterministic inverse mapping $\mathcal{T}^{-1} : \theta \to D$, i.e. the training dataset for a given model θ is uniquely specified. (Later we also show how to approximate the training datasets D_θ for model θ when the training algorithm $\mathcal{T}$ is not invertible.) Plugging equation 29 into equation 10, we rewrite the FPR constraint as follows.

$$\mathbf{E}_{P_{H_0}(\theta,x_z,y_z)}\left[\mathbf{1}_{\ell(\theta,x_z,y_z)\leq c_\alpha(D_\theta,x_z,y_z)}\right] = \alpha \tag{30}$$

By deterministic mapping from θ to D_θ, the chain rule of joint distribution, and the independence between random variables θ and $z = (x_z, y_z)$ under null hypothesis H_0 in equation 2, we prove that

$$\begin{aligned} P_{H_0}(\theta, x_z, y_z) &= P_{H_0}(\theta, x_z, y_z) \cdot 1 = P_{H_0}(\theta, x_z, y_z) \cdot P(D_\theta|\theta) && (31)\\ &= P_{H_0}(D_\theta, \theta, x_z, y_z) = P_{H_0}(D_\theta, x_z, y_z) \cdot P_{H_0}(\theta|D_\theta, x_z, y_z) && (32)\\ &= P_{H_0}(D_\theta, x_z, y_z) \cdot P(\theta|D_\theta) && (33) \end{aligned}$$

where $P(\theta|D_\theta)$ is distribution of trained model under a fixed training dataset D_θ, as specified in equation 1. Plugging equation 31 into equation 30, we rewrite the false positive rate constraint as

$$\mathbf{E}_{P_{H_0}(D_\theta,x_z,y_z)}\left[\mathbf{E}_{P(\theta|D_\theta)}\left[\mathbf{1}_{\ell(\theta,x_z,y_z)\leq c_\alpha(D_\theta,x_z,y_z)}\right]\right] = \alpha \tag{34}$$

One sufficient condition for equation 34 to hold is that, for any fixed sample z and target model θ,

$$\mathbf{E}_{P(\theta'|D_\theta)}\left[\mathbf{1}_{\ell(\theta',x_z,y_z)\leq c_\alpha(D_\theta,x_z,y_z)}\right] = \alpha, \tag{35}$$

where D_θ is an implicitly fixed training dataset for the target model θ, and the distribution $P(\theta'|D_\theta)$ captures retrained models on the training dataset $D_\theta = \mathcal{T}^{-1}(\theta)$ for given target model θ, as follows.

$$\mathcal{T}\left(\mathcal{T}^{-1}(\theta)\right) = \theta' \text{ with probability } P(\theta'|D_\theta). \tag{36}$$

To approximate this distribution of retrained model $\mathcal{T}\left(\mathcal{T}^{-1}(\theta)\right) \sim P(\theta'|D_\theta)$, the attacker generates the following set of self-distilled models using the target model θ.

$$M = \{\theta_i\}_{i=1,2,\cdots}, \text{ where } \theta_i \sim \mathcal{T}(D_i)\ , D_i \xleftarrow{\text{soft-labeled with } \theta} D_i' \xleftarrow{i.i.d. samples \sim \pi(z)} D_{pop} \tag{37}$$

These distilled models $M = \{\theta_i\}_{i=1,2,\cdots}$ approximate samples from the retrained model distribution $\mathcal{T}(\mathcal{T}^{-1}(\theta)) \sim P(\theta|D_\theta)$. This is because θ_i is trained on distillation dataset D_i consisting of population data points which are soft-labeled with the target model θ. This roughly recovers the target model θ trained on D_θ, however without its potential dependence on z. Therefore, the attacker approximate $P(\theta|D_\theta)$ in the false positive rate constraint equation 35 with the empirical distribution over distilled models samples in M, as follows.

$$\frac{|\{\theta_i \in M : \ell(\theta_i, x_z, y_z) \leq c_\alpha(D_\theta, x_z, y_z)\}|}{|M|} = \alpha \tag{38}$$

By solving equation 38 for fixed target model θ and target point z, we compute that $c_\alpha(D_\theta, x_z, y_z)$ equals the the α-percentile for the histogram of loss values $\ell(\theta_i, x_z, y_z)$ on distilled models $\theta_i \in M$. This recovers the model-dependent and sample-dependent attack threshold $c_\alpha(x_z, y_z)$ in Attack D.

3.5 Attack L: Leave one out attack

An ideal attack, that removes the randomness over the training data (except the target data that could potentially be part of the training set) would be the *leave one out* attack. In this attack, the adversary trains reference models θ' on $D \setminus \{(x_b, y_b)\}$. The attack would be in the same class of attacks as in Attack D, as it will be a model-dependent and data-dependent attack. It also runs a similar hypothesis test, however the attack requires assuming the adversary already knows the $n-1$ data records in $D \setminus \{(x_b, y_b)\}$. This is a totally acceptable assumption in the setting of privacy auditing.

Note that Attack D aims at reproducing the results of the leave-one-out attack without assuming the knowledge of $n-1$ data records in $D \setminus \{(x_b, y_b)\}$.

3.6 Summary and comparison of different attacks

For identifying whether a data point z has been part of the training set of θ, here are the main underlying questions for the attacks we present in this section:

• How likely is the loss $\ell(\theta, z)$ to be a sample from the distribution of loss of random samples from the population on (Attack P: the same model) (Attack S: models trained on the population data)? Depending on the tolerable false positive rate α and the estimated distribution of loss, we reject the null hypothesis.

• How likely is the loss $\ell(\theta, z)$ to be a sample from the distribution of loss of z on (Attack R: models trained on population data) (Attack D: models trained to be as close as possible to the target model, using distillation) (Attack L: models trained on $n-1$ records from D excluding z)? Depending on the tolerable false positive rate α and the estimated distribution of loss, we reject the null hypothesis.

Effectively, these questions cover different types of hypothesis tests that could be designed for performing membership inference attacks. We expect these attacks to have a different error due to the uncertainties that can influence their performance.

Attacks S and P are of the same nature. However, attack S could potentially have a higher error due to its imprecision in using other models to approximate the loss distribution of the target model on population data. Attacks R, D, and L are also of the same nature. However, we expect attacks D to have more confidence in the tests due to reducing the uncertainty of other training data that can influence the model's loss. Thus, we expect attack D to be the closest to the strongest attack which is the leave-one-out attack.

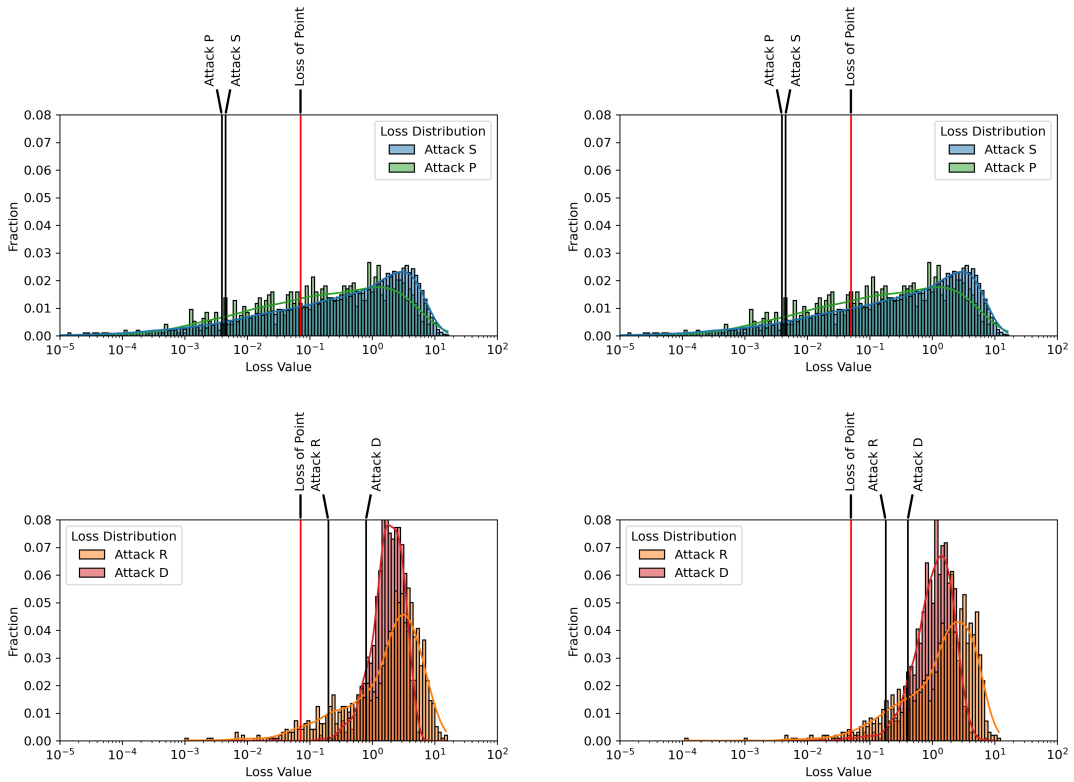


Figure 1: Loss distributions used by Attack S, P, R, D for points z_1, z_2 and target models θ_1, θ_2 in the Purchase100 2a experimental setup, $\alpha = 0.1$. Note that both z_1 and z_2 are datapoints from the same class and are members of both the target models' training datasets. We show the loss distributions for z_1, z_2 on target model θ_1 here, and on θ_2 in the appendix.

4 EXPERIMENTS

In this section, we empirically measure and study the performance of different attack strategies that we derived from our framework. A detailed analysis of the internal mechanics of these attacks and how they differ from each other is provided in the Appendix A.2.

Evaluation of attacks performance Which one of the attacks S, P, R, D has the best performance? How to evaluate the strength of attack besides attack accuracy? We quantify the attacker's performance using two metrics: its true positive rate (TPR), and its false positive rate (FPR), over the random samples from member and non-member data and random target models. The ROC curve captures the tradeoff between the TPR and FPR of an attack, as its threshold c_α is varied across different FPR tolerance α. The AUC (area under the ROC curve) score measures the strength of an attack. Therefore, we plot the ROC curves of all attacks on the Purchase100 dataset, and computes their AUC (area under the ROC curve) score in Figure 3 (in the appendix). The attack with the highest AUC score on Purchase100 is Attack D, which has the least level of uncertainty, as discussed Section 3. Table 1 and Table 3 shows more detailed AUC score results for different attacks under more settings.

Comparison between the Attacks Besides attack strength, how differently are the attacks performing on random input target models and target points? How do the different internal mechanics of attacks changing the attacker's guess qualitatively? Do the attacks succeed on similar or different samples of member data and target model? Do the attacks have different level of confidence on the same input? How far away are the attack performance from the most ideal leave-one-out attacks described in Section 3? Answers to these questions require understanding how and why the attacks perform differently, for which we do detailed comparisons between attacks as follows.

	Train Acc.	Test Acc.	Attack S	Attack P	Attack R	Attack D
1a	96.2 ± 0.031	52.5 ± 0.026	**0.809** ± 0.017	**0.822** ± 0.014	**0.84** ± 0.023	**0.876** ± 0.009
1b	51.3 ± 0.153	35.0 ± 0.095	**0.628** ± 0.035	**0.646** ± 0.026	**0.643** ± 0.045	**0.652** ± 0.045
2a	99.5 ± 0.004	65.4 ± 0.009	**0.752** ± 0.008	**0.755** ± 0.006	**0.799** ± 0.009	**0.821** ± 0.004
2b	64.1 ± 0.039	49.9 ± 0.019	**0.599** ± 0.009	**0.609** ± 0.009	**0.613** ± 0.012	**0.635** ± 0.004
3	100.0 ± 0.0	75.5 ± 0.004	**0.687** ± 0.003	**0.687** ± 0.003	**0.755** ± 0.004	**0.768** ± 0.002

Table 1: AUC Scores of all attacks on Purchase100 Dataset. Configurations 1a, 1b are trained on 2500 datapoints, configurations 2a, 2b are trained on 5000 datapoints, and configuration 3 is trained on 10000 datapoints. Configurations 1b, 2b are trained using L2 regularization with regularization penalty $\lambda = 0.01$.

• **Similarity of attacks with each other.** The scatter plot comparing Attack S and P in Figure 4 is roughly linearly centered around the diagonal, with slightly more points in NorthWest then in SouthEast. This shows that Attack S and Attack P almost always guesses membership of train points similarly, while Attack P performs slightly better than Attack S. Meanwhile, Attack D dominates Attack R for correctly guessing membership of train points because their comparison scatter plot is biased towards the NorthWest direction.

• **Gap between attacks to the ground truth.** From Table 4, among all attacks, Attack D agrees with the ground truth the most on train points, while the least on test points. This matches our observation that Attack D has a larger threshold under given α from Figure 1, which causes it to guess both more points as members. Besides that, we see that the agreement rate between Attack S and Attack P is as high as 0.94 both on train points and test points, this matches their linear comparison scatter plot in Figure 4. This may be because Attack P and Attack R both reduces one degree of uncertainty for the joint distribution under null hypothesis H_0, as discussed in Section 3.

• **Closeness of attacks to ideal leave-one-out attack.** One interesting observation from Table 4 is that, among all the attacks, Attack D agrees with the Attack L the most, with agreement rate 0.98 on train points and 0.82 on test points. We believe this is because Attack D is highly similar in nature with Attack L, by approximating the training dataset of a target model and performing retraining, as discussed in Section 3.

	GT	L	S	P	R	D
GT		0.98	0.66	0.64	0.78	1.00
L	0.56		0.64	0.62	0.80	0.98
S	0.72	0.80		0.94	0.60	0.66
P	0.66	0.82	0.94		0.58	0.64
R	0.72	0.80	0.84	0.82		0.78
D	0.42	0.82	0.70	0.76	0.66	

Table 2: Agreement rate between ground truth (GT) membership values, and Attacks L, S, P, R, D for 50 train and test datapoints. The upper triangle of the table corresponds to the agreement rates of train datapoints, whereas the lower triangle corresponds to the agreement rates of test datapoints. The experimental setup is Purchase100 2a, with $\alpha = 0.3$.

5 Summary

We provide a framework for auditing the privacy risk from machine learning models through membership inference attacks. The framework is used to derive attack strategies and also highlight the factors beyond leakage from the models that affect the attack performance. We also empirically analyze the performance of these attack strategies against models trained on benchmark datasets.

References

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

Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th {USENIX} Security Symposium ({USENIX} Security 19)*, pp. 267–284, 2019.

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. Extracting training data from large language models. *arXiv preprint arXiv:2012.07805*, 2020.

Hongyan Chang and Reza Shokri. On the privacy risks of algorithmic fairness. *arXiv preprint arXiv:2011.03731*, 2020.

Christopher A Choquette-Choo, Florian Tramer, Nicholas Carlini, and Nicolas Papernot. Label-only membership inference attacks. In *International Conference on Machine Learning*, pp. 1964–1974, 2021.

Ganesh Del Grosso, Georg Pichler, Catuscia Palamidessi, and Pablo Piantanida. Bounding information leakage in machine learning. *arXiv preprint arXiv:2105.03875*, 2021.

Irit Dinur and Kobbi Nissim. Revealing information while preserving privacy. In *Proceedings of the twenty-second ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pp. 202–210, 2003.

Jinshuo Dong, Aaron Roth, and Weijie J Su. Gaussian differential privacy. *arXiv preprint arXiv:1905.02383*, 2019.

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

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

Cynthia Dwork, Adam Smith, Thomas Steinke, and Jonathan Ullman. Exposed! a survey of attacks on private data. *Annual Review of Statistics and Its Application*, pp. 61–84, 2017.

Úlfar Erlingsson, Ilya Mironov, Ananth Raghunathan, and Shuang Song. That which we call private. *arXiv preprint arXiv:1908.03566*, 2019.

Farhad Farokhi and Mohamed Ali Kaafar. Modelling and quantifying membership information leakage in machine learning. *arXiv preprint arXiv:2001.10648*, 2020.

Vitaly Feldman. Does learning require memorization? a short tale about a long tail. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 954–959, 2020.

Jamie Hayes, Luca Melis, George Danezis, and Emiliano De Cristofaro. Logan: Membership inference attacks against generative models. *Proceedings on Privacy Enhancing Technologies*, pp. 133–152, 2019.

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. Resolving individuals contributing trace amounts of dna to highly complex mixtures using high-density snp genotyping microarrays. *PLoS genetics*, pp. e1000167, 2008.

Hongsheng Hu, Zoran Salcic, Gillian Dobbie, and Xuyun Zhang. Membership inference attacks on machine learning: A survey. *arXiv preprint arXiv:2103.07853*, 2021.

Thomas Humphries, Matthew Rafuse, Lindsey Tulloch, Simon Oya, Ian Goldberg, Urs Hengartner, and Florian Kerschbaum. Differentially private learning does not bound membership inference. *arXiv preprint arXiv:2010.12112*, 2020.

Matthew Jagielski, Jonathan Ullman, and Alina Oprea. Auditing differentially private machine learning: How private is private sgd? *arXiv preprint arXiv:2006.07709*, 2020.

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

Peter Kairouz, Sewoong Oh, and Pramod Viswanath. The composition theorem for differential privacy. In *International conference on machine learning*, pp. 1376–1385, 2015.

Klas Leino and Matt Fredrikson. Stolen memories: Leveraging model memorization for calibrated white-box membership inference. In *29th {USENIX} Security Symposium ({USENIX} Security 20)*, pp. 1605–1622, 2020.

Zheng Li and Yang Zhang. Membership leakage in label-only exposures. *arXiv preprint arXiv:2007.15528*, 2020.

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

Mani Malek, Ilya Mironov, Karthik Prasad, Igor Shilov, and Florian Tramèr. Antipodes of label differential privacy: Pate and alibi. *arXiv preprint arXiv:2106.03408*, 2021.

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

Sasi Kumar Murakonda and Reza Shokri. Ml privacy meter: Aiding regulatory compliance by quantifying the privacy risks of machine learning. *arXiv preprint arXiv:2007.09339*, 2020.

Sasi Kumar Murakonda, Reza Shokri, and George Theodorakopoulos. Quantifying the privacy risks of learning high-dimensional graphical models. In *International Conference on Artificial Intelligence and Statistics*, pp. 2287–2295, 2021.

M. Nasr, R. Shokri, and A. Houmansadr. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *IEEE Symposium on Security and Privacy (SP)*, pp. 1022–1036, 2019.

Milad Nasr, Shuang Song, Abhradeep Thakurta, Nicolas Papernot, and Nicholas Carlini. Adversary instantiation: Lower bounds for differentially private machine learning. *arXiv preprint arXiv:2101.04535*, 2021.

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

Md Atiqur Rahman, Tanzila Rahman, Robert Laganiere, Noman Mohammed, and Yang Wang. Membership inference attack against differentially private deep learning model. *Transactions on Data Privacy*, pp. 61–79, 2018.

Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, Yann Ollivier, and Hervé Jégou. White-box vs black-box: Bayes optimal strategies for membership inference. In *International Conference on Machine Learning*, pp. 5558–5567, 2019.

Ahmed Salem, Yang Zhang, Mathias Humbert, Pascal Berrang, Mario Fritz, and Michael Backes. Ml-leaks: Model and data independent membership inference attacks and defenses on machine learning models. *arXiv preprint arXiv:1806.01246*, 2018.

Sriram Sankararaman, Guillaume Obozinski, Michael I Jordan, and Eran Halperin. Genomic privacy and limits of individual detection in a pool. pp. 965, 2009.

Muhammad A Shah, Joseph Szurley, Markus Mueller, Athanasios Mouchtaris, and Jasha Droppo. Evaluating the vulnerability of end-to-end automatic speech recognition models to membership inference attacks. *Proc. Interspeech 2021*, pp. 891–895, 2021.

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

Reza Shokri, Martin Strobel, and Yair Zick. On the privacy risks of model explanations. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 231–241, 2021.

Congzheng Song and Ananth Raghunathan. Information leakage in embedding models. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, pp. 377–390, 2020.

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

Congzheng Song, Thomas Ristenpart, and Vitaly Shmatikov. Machine learning models that remember too much. In *Proceedings of the 2017 ACM SIGSAC Conference on computer and communications security*, pp. 587–601, 2017.

Liwei Song and Prateek Mittal. Systematic evaluation of privacy risks of machine learning models. In *30th {USENIX} Security Symposium ({USENIX} Security 21)*, 2021.

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

Michael Veale, Reuben Binns, and Lilian Edwards. Algorithms that remember: model inversion attacks and data protection law. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, pp. 20180083, 2018.

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

Yang Zou, Zhikun Zhang, Michael Backes, and Yang Zhang. Privacy analysis of deep learning in the wild: Membership inference attacks against transfer learning. *arXiv preprint arXiv:2009.04872*, 2020.

A Appendix

A.1 Detailed derivation of approximated LRT for membership inference

We first prove two useful approximation inequalities about the posterior distribution $P(\theta|D)$ as follows.

1. For arbitrary data point $z = (z_x, z_y)$, and arbitrary dataset D, we have

$$P(\theta|D \cup z) \geq e^{-\frac{\ell(\theta, z_x, z_y)}{T}} \cdot P(\theta|D). \tag{39}$$

2. Let $z_1, z_2, \cdots, z_n$ be i.i.d. samples from the data distribution $\pi(z)$. Then when n is large enough, for any model parameter θ, we have

$$\mathbf{E}_{z_1,\cdots,z_n\sim\pi}[P(\theta|z_1,\cdots,z_n)] \approx \mathbf{E}_{z_1,\cdots,z_{n-1}\sim\pi}[P(\theta|z_1,\cdots,z_{n-1})] \tag{40}$$

We now offer details for deriving the approximated likelihood ratio test (LRT) for membership inference.

Definition A.1 (Approximated LRT for membership inference) *Let (θ, z) be random samples from the joint distribution of target model and target data point, specified by one of the following membership hypotheses.*

$$H_0 : D \xleftarrow{n\ i.i.d. samples \sim \pi(z)} D_{pop}, \theta \sim \mathcal{T}(D), z \xleftarrow{sample \sim \pi(z)} D_{pop} \tag{41}$$

$$H_1 : D \xleftarrow{n\ i.i.d. samples \sim \pi(z)} D_{pop}, \theta \sim \mathcal{T}(D), z \xleftarrow{sample} D \tag{42}$$

The likelihoods function of hypothesis H_0 and H_1 given observed target model θ and target data point z is as follows.

$$L(H_0|\theta, z) = P_{H_0}(\theta, z) = \pi(z) \cdot \mathbf{E}_{D\sim\pi^n}[P(\theta|D)] \tag{43}$$

$$L(H_1|\theta, z) = P_{H_1}(\theta, z) = \sum_D P_{H_1}(D, \theta, z) = \sum_D \pi(z) \cdot P_{H_1}(D|z) \cdot P(\theta|D) \tag{44}$$

$$= \pi(z) \cdot \mathbf{E}_{D'\sim\pi^{n-1}}[P(\theta|D' \cup z)] \tag{45}$$

$$\textit{(By equation 39)} \geq \pi(z) \cdot e^{-\frac{1}{T}\ell(\theta, z_x, z_y)} \cdot \mathbf{E}_{D'\sim\pi^{n-1}}[P(\theta|D')] \tag{46}$$

$$\textit{(By equation 40)} \approx \pi(z) \cdot e^{-\frac{1}{T}\ell(\theta, z_x, z_y)} \cdot \mathbf{E}_{D\sim\pi^n}[P(\theta|D)] \tag{47}$$

Therefore the LRT statistics is

$$\lambda(\theta, z) = \frac{L(H_0|\theta, z)}{L(H_1|\theta, z)} \leq e^{\frac{1}{T}\ell(\theta, z_x, z_y)} \frac{\mathbf{E}_{D\sim\pi^n}[P(\theta|D)]}{\mathbf{E}_{D'\sim\pi^{n-1}}[P(\theta|D')]} \approx e^{\frac{1}{T}\ell(\theta, z_x, z_y)} \tag{48}$$

The LRT hypothesis test rejects H_0 when the LRT statistic is small. By equation 6, the rejection region $\{(\theta, z) : \lambda(\theta, z) \leq c\}$ can be approximated as follows.

$$\left\{(\theta, z) : \ell(\theta, z_x, z_y) \leq T \cdot \log c\right\} \tag{49}$$

Datasets

- The **Purchase100** dataset is based on Kaggle's "Acquire Valued Shoppers Challenge" that contains shopping histories for thousands of individuals[5]. We use a simplified preprocessed purchase dataset provided in Shokri et al. (2017). There are $197,324$ data points in this dataset, where each data point has 600 binary features. The data points are clustered into 100 classes to represent different shopping style.
- The **CIFAR100** dataset is a widely used benchmark dataset for image classification. The CIFAR100 dataset consists of $60,000$ data points in 100 classes, where each data point is a 32×32 color image and each class has 600 images. In total, there are $50,000$ training images and $10,000$ test images.
- The **MNIST** dataset is a widely used benchmark dataset for image classification. The MNIST dataset consists of $70,000$ data points in 10 classes, where each data point is a 28×28 handwritten digit image. In total, there are $60,000$ training images and $10,000$ test images.

Experiment setup

1. For each configuration, we train $n = 1000$ models on random IID splits of the data. For the Purchase100 configurations, we use a 4 layer MLP with layer units = [512, 256, 128, 64]. For CIFAR100 and MNIST configurations, we use a 2 layer CNN with filters = [32, 64] and max pooling. For all target models and datasets, we use SGD as the optimization function and categorical crossentropy loss.
2. Target models: We use the first 10 models as target models on each of the Purchase100, CIFAR100 and MNIST datasets. The performance of Attacks S, P, and R are evaluated on all 10 models, whereas Attack D is evaluated on the first 3 models.
3. Shadow models for Attack S and Reference models for Attack R: Given a target model, we use the remaining $(n-1) = 999$ models as shadow models or reference models for Attack S and Attack R respectively.
4. Distilled models for Attack D: Given a target model, we use the same random IID splits used to train $n = 1000$ models to distill $(n-1) = 999$ models using the soft labels from the target model.

[5]https://www.kaggle.com/c/acquire-valued-shoppers-challenge

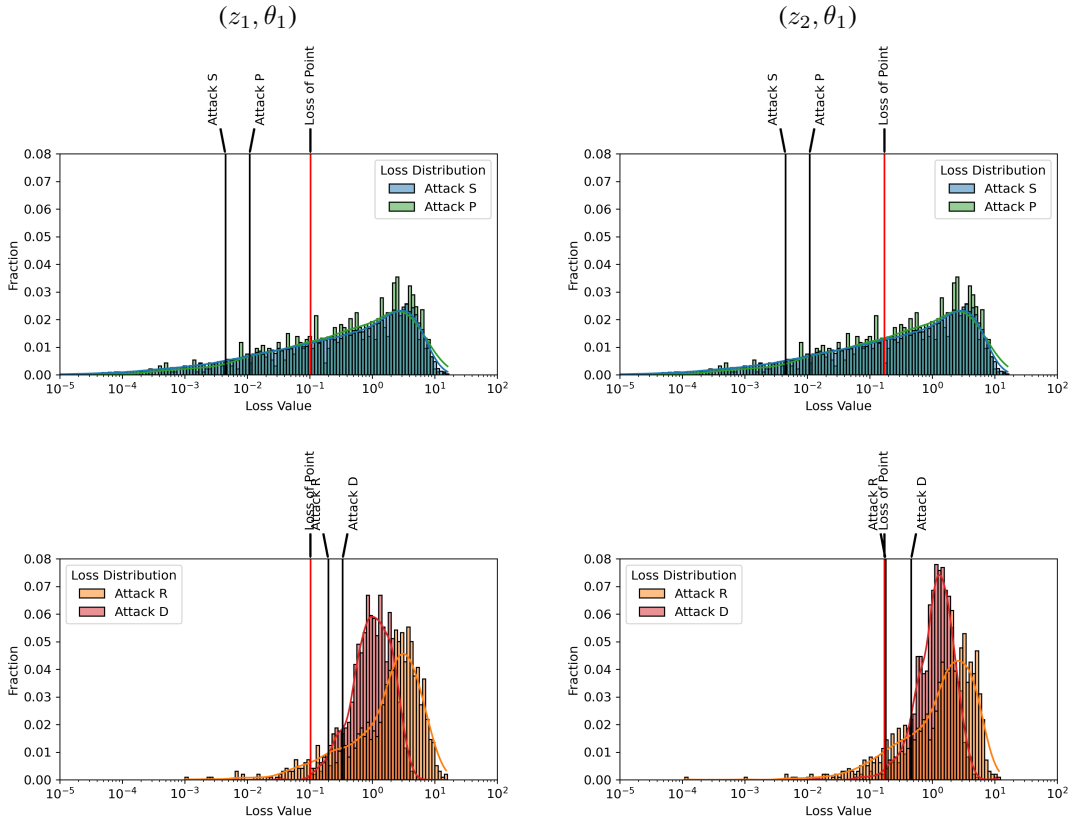


Figure 2: Loss distributions used by Attack S, P, R, D for points z_1, z_2 and target models θ_1, θ_2 in the Purchase100 2a experimental setup, $\alpha = 0.1$. Note that both z_1 and z_2 are datapoints from the same class and are members of both the target models' training datasets. We show the loss distributions for z_1, z_2 on target model θ_1 in the main paper, and on θ_2 here.

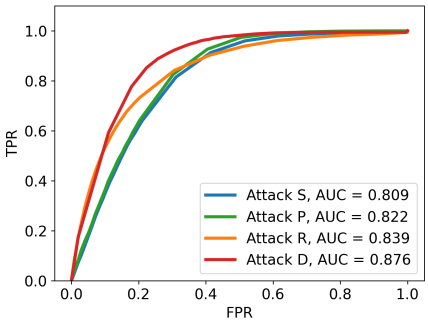
A.2 Additional Empirical Evaluation Results

In this section, we provide a detailed analysis of the internal workings of the attack strategies. We specifically compare how the thresholds on loss values derived from these strategies differ on various target points and target models.

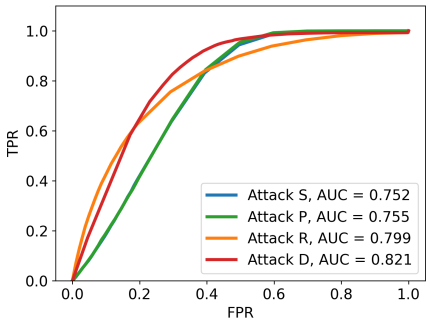
Illustration of Attack threshold How differently does the attack threshold in Attack S, P, R, D depend on the target model θ and the target data point z? How is the different dependence of threshold on θ and z in different attacks affecting their success? To investigate these important questions, we plot the loss histograms that different attacks use to compute the thresholds, on different input θ_1, θ_2 and z_1, z_2 in Figure 1. The loss histogram approximates the LRT statistics distribution under null hypothesis H_0 in equation 2, with different uncertainty for different attacks, as discussed in 3. We see that model-dependence and sample-dependence of attack threshold reduces the uncertainty in the loss histogram. The attacks (Attack D) with more concentrated loss histogram are more likely to succeed.

B Related Works

The current work in this domain can be broadly grouped into three categories: 1) Empirical works for improving the existing attack strategies or adapting them to different settings and models 2) Theoretically/empirically analyzing the privacy risk in various systems using the existing attack strategies 3) Exploring the connections with differential privacy and using them to establish lower



Purchase100 (1a)



Purchase100 (2a)

Figure 3: FPR vs TPR with AUC scores for all attacks on Purchase100 Dataset experimental setups.

	Train Acc.	Test Acc.	Attack S	Attack P	Attack R	Attack D
1a	97.4 ± 0.013	14.9 ± 0.009	$\mathbf{0.959} \pm 0.005$	$\mathbf{0.960} \pm 0.004$	$\mathbf{0.964} \pm 0.006$	$\mathbf{0.957} \pm 0.0003$
1b	89.8 ± 0.035	13.9 ± 0.011	$\mathbf{0.937} \pm 0.007$	$\mathbf{0.941} \pm 0.004$	$\mathbf{0.938} \pm 0.012$	$\mathbf{0.941} \pm 0.004$
2a	97.9 ± 0.006	20.4 ± 0.006	$\mathbf{0.944} \pm 0.004$	$\mathbf{0.945} \pm 0.003$	$\mathbf{0.945} \pm 0.006$	$\mathbf{0.936} \pm 0.0$
2b	87.9 ± 0.020	19.63 ± 0.007	$\mathbf{0.905} \pm 0.007$	$\mathbf{0.906} \pm 0.007$	$\mathbf{0.904} \pm 0.008$	$\mathbf{0.893} \pm 0.002$

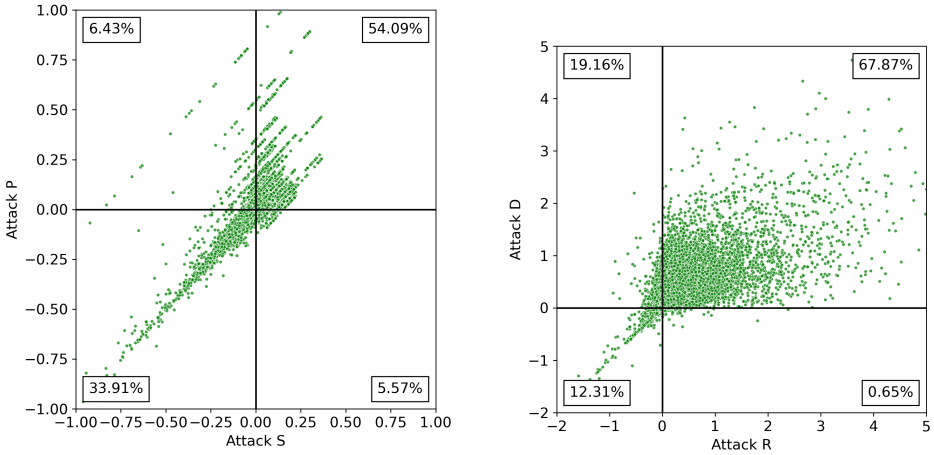
Table 3: AUC Scores of all attacks on CIFAR100 Dataset. Configurations 1a, 1b are trained on 2500 datapoints, and configurations 2a, 2b are trained on 5000 datapoints. Configurations 1b, 2b are trained using L2 regularization with regularization penalty $\lambda = 0.005$.

bounds on leakage and/or select privacy parameters. Below, we provide a brief summary of works in all the three categories.

Empirical attack strategies for membership inference: Shokri et al. (2017) demonstrated the vulnerability of machine learning models to membership inference attacks in the black-box setting, where the adversary has only a query access to the target model. The attack algorithm is based on the concept of shadow models, which are models trained on some attacker dataset that is similar to that of the training data. Membership inference is modeled as a binary classification task for an attack model that is trained on the predictions of shadow models on the attacker dataset. A huge literature followed this work extending the attacks to different setting like white box access (Nasr et al., 2019; Sablayrolles et al., 2019; Leino & Fredrikson, 2020), label-only access (Li & Zhang, 2020; Choquette-Choo et al., 2021), federated learning (Melis et al., 2019), transfer learning (Zou et al., 2020) and different types of data, models such as aggregate location data (Pyrgelis et al., 2017), generative models (Hayes et al., 2019), language models (Carlini et al., 2019; Song & Shmatikov, 2019; Carlini et al., 2020), sentence embeddings (Song & Raghunathan, 2020), and speech recognition models (Shah et al., 2021). Multiple works have looked at improving the attack methodology through a more fine grained analysis or by reducing the background knowledge and the compute power required to execute the attack (Long et al., 2018; Song & Mittal, 2021; Salem et al., 2018). See (Hu et al., 2021) for a more comprehensive list of membership inference attacks against machine learning models. All these works follow the same attack framework for membership inference but they either exploit a slightly different signal that is correlated with membership of a point in the training set or find an efficient way to exploit the already known signals.

	Train Acc.	Test Acc. Attack S	Attack P	Attack R	Attack D	
1a	97.9 ± 0.004	95.8 ± 0.007	$\mathbf{0.50} \pm 0.004$	$\mathbf{0.50} \pm 0.005$	$\mathbf{0.557} \pm 0.009$	$\mathbf{0.549} \pm 0.006$
1b	94.2 ± 0.005	93.4 ± 0.006	$\mathbf{0.497} \pm 0.005$	$\mathbf{0.496} \pm 0.004$	$\mathbf{0.544} \pm 0.014$	$\mathbf{0.522} \pm 0.005$
2a	98.6 ± 0.001	97.1 ± 0.002	$\mathbf{0.496} \pm 0.005$	$\mathbf{0.496} \pm 0.006$	$\mathbf{0.551} \pm 0.011$	$\mathbf{0.544} \pm 0.004$
2b	94.6 ± 0.003	94.5 ± 0.003	$\mathbf{0.491} \pm 0.005$	$\mathbf{0.49} \pm 0.005$	$\mathbf{0.52} \pm 0.011$	$\mathbf{0.497} \pm 0.004$

Table 4: AUC Scores of all attacks on MNIST Dataset. Configurations 1a, 1b are trained on 2500 datapoints, and configurations 2a, 2b are trained on 5000 datapoints. Configurations 1b, 2b are trained using L2 regularization with regularization penalty $\lambda = 0.05$.



(a) (b)

Figure 4: Scatterplots comparing Attacks S and P, and Attacks R and D on Purchase100 2a experimental setup. Each dot on a scatterplot corresponds to a particular train datapoint. The two coordinates of the dot are for the two attacks being compared. Each coordinate of the dot corresponds to the difference between the loss of the point on the target model and the loss threshold used by the particular attack on the point. Plot (a) compares Attacks S and P, whereas plot (b) compares Attacks R and D.

Privacy risk analysis with membership inference attacks: Homer et al. (2008) performed the first membership inference attack on genome data to identify the presence of an individual's genome in a mixture of genomes. Sankararaman et al. (2009) provided a formal analysis of this risk of detecting the presence of an individual from aggregate statistics computed on independent and binary attributes. Murakonda et al. (2021) extended this analysis to the case of releasing Discrete Bayesian networks learned from data with dependent attributes. Backes et al. (2016) perform an analysis similar to that of (Sankararaman et al., 2009) but for MicroRNA data (aggregate statistics computed on independent and continuous attributes). Dwork et al. (2015) provide a more extensive analysis when the released statistics are noisy and the attacker has only one reference sample to perform the attack. The key results of all these works establish the privacy risk of releasing aggregate statistics by quantifying the success of membership inference attacks as a function of the number of statistics released and the number of individuals in the dataset. Similar attempts were made to analyze the privacy risk of machine learning models through membership inference via the lens of mutual information (Farokhi & Kaafar, 2020) and generalization error (Yeom et al., 2018; Del Grosso et al., 2021). Beyond these theoretical analyses, membership inference attacks are also used to empirically study the trade-offs between privacy and other desirable characteristics for machine learning models such as fairness (Chang & Shokri, 2020), robustness to adversarial examples (Song et al., 2019), and providing explanations (Shokri et al., 2021).

Differential privacy and Membership inference: The definitions of differential privacy (Dwork et al., 2006) and membership inference (Homer et al., 2008; Dwork et al., 2015; Shokri et al., 2017) are very closely connected and the hypothesis testing interpretation of differential privacy provides a clear view of the relationship between them. Satisfying differential privacy is equivalent to imposing a bound on the ability to distinguish any two neighboring datasets that differ by the presence of one individual i.e., inferring about the presence/absence of the individual. The bound is stated as a trade-off between the type-I and type-II errors in distinguishing the two neighboring datasets. Hypothesis testing interpretation of differential privacy is very useful in deriving tight compositions (Kairouz et al., 2015) and has even motivated a new relaxed notion of differential privacy called f-DP (Dong et al., 2019). By definition, differentially private algorithms bound the success of membership inference attacks. Multiple works (Yeom et al., 2018; Erlingsson et al., 2019; Humphries et al., 2020), each improving on the previous work, have provided upper bounds on the success of membership inference attacks as a function of the parameters in differential privacy. Jayaraman & Evans (2019) evaluated the performance of membership inference attacks on machine learning models trained

with different values of epsilon under different relaxed notions of differential privacy. Rahman et al. (2018) also use membership inference attacks to measure the privacy loss on models trained with differentially private algorithms. The empirical performance of membership inference attacks has also been used to provide lower bounds on the privacy guarantees achieved by various differentially private algorithms (Jagielski et al., 2020; Nasr et al., 2021; Malek et al., 2021). The key difference between the empirical analysis of membership inference in the previous three works and other works is that they simulate the exact adversary in differential privacy i.e., they train multiple models with and without one particular training point and keep the rest of training set fixed. The performance of the attack (type I and type II errors) is averaged over these models, whereas in the previous works the model is fixed and the performance is computed as an average over points in the training and test sets. This simulation of the DP adversary helps in removing the effects of other points in the dataset when measuring the leakage through the model about a particular point of interest.