Bounding Information Leakage in Machine Learning

Ganesh Del Grosso, George Pichler, Catuscia Palamidessi, Pablo Piantanida

Bounding Information Leakage in Machine Learning

Ganesh Del Grosso[a,b], Georg Pichler[c], Catuscia Palamidessi[a,b], Pablo Piantanida[d]

[a]*Inria Saclay, team COMETE, 1 Rue Honore d'Estienne d'Orves, Palaiseau, 91120, Ile-de-France, France*
[b]*Ecole Polytechnique, LIX, 1 Rue Honore d'Estienne d'Orves, Palaiseau, 91120, Ile-de-France, France*
[c]*TU Wien, Institute of Telecommunications, Gusshausstrasse 25/E389, 1040, Vienna, Austria*
[d]*International Laboratory on Learning Systems (ILLS) Universite McGill - ETS - MILA - CNRS - Universite Paris-Saclay - CentraleSupelec, 1100 Rue Notre Dame O, Montreal, H3C 1K3, Quebec, Canada*

Abstract

Recently, it has been shown that Machine Learning models can leak sensitive information about their training data. This information leakage is exposed through membership and attribute inference attacks. Although many attack strategies have been proposed, little effort has been made to formalize these problems. We present a novel formalism, generalizing membership and attribute inference attack setups previously studied in the literature and connecting them to memorization and generalization. First, we derive a universal bound on the success rate of inference attacks and connect it to the generalization gap of the target model. Second, we study the question of how much sensitive information is stored by the algorithm about its training set and we derive bounds on the mutual information between the sensitive attributes and model parameters. Experimentally, we illustrate the potential of our approach by applying it to both synthetic data and classification tasks on natural images. Finally, we apply our formalism to different attribute inference strategies, with which an adversary is able to recover the identity of writers in the PenDigits dataset.

Keywords: Membership Inference, Privacy, Attacks in Machine Learning.

1. Introduction

Machine Learning (ML) models have been known to leak information about their training records. This raises severe privacy concerns in cases where the training data contains sensitive information, for instance, when using real patients' data in medical applications, e.g., [1, 2]. In order for an ML algorithm to be private, the General Data Protection Regulation (GDPR) and similar laws require for it to be impossible to *single out* any individual from the training set. Recently, [3] pointed out the need for a clear definition of what this means. Nonetheless, it is widely considered in the literature that Membership Inference Attacks (MIAs) measure the privacy of ML algorithms, i.e., if a MIA is effective against a model, it is possible to single out an individual from its training set.

Membership Inference has been studied in the literature, but the efforts have been concentrated on model or data dependent strategies to perform the attacks, rather than developing a general framework to understand these problems. This drives us to propose a novel formalism, providing a general framework to study inference attacks and their connection to generalization and memorization. Compared to previous works (e.g. [4, 5]), we consider a more general framework, where we study the performance of the Bayesian attacker without making any assumptions on the distribution of model parameters given the training set.

Furthermore, the present paper is distinguished from previous work by studying the connection between the information stored by the model and the leakage of sensitive information. Intuitively, the amount of leaked information should be proportional to the amount of training data stored by the model. In our study, we find that the mutual information between training set and model parameters upper bounds the gain of the Bayesian attacker over an attacker that only uses the prior distributions of the sensitive attributes.

Lastly, we consider the risk of sensitive information leakage from ML models in the form of attribute inference attacks. In these attacks, having partial knowledge of a sample in the training set, an adversary tries to extract sensitive information about the sample from the target model. In the case of medical data, the sensitive information could be the genetic profile of a patient [6]. We study several attribute inference strategies against ML models. Our framework allows us to formalize these problems, draw universal bounds on the performance of MIAs and find connections between generalization and privacy.

1.1. Summary of Contributions

Our work investigates fundamental bounds on information leakage and advances the state-of-the-art in multiple ways.

1. *A simple framework for modeling membership and/or attribute inference attacks.* We introduce a probabilistic framework for the analysis of membership and/or attribute inference attacks on machine learning systems. The necessary Definitions 2 to 4 are simple and concise, yet flexible enough to be applied to different problem setups.

2. *Universal bounds on the success rate of inference attacks.* By considering the success rate obtained by the Bayes decision rule, we are able to draw strong conclusions about the privacy of a ML model. The attacker we consider is given with perfect knowledge of the underlying probability distribution. As such, it provides an upper bound for the probability of success of any attack strategy (Theorem 1). As a matter of fact, this bound represents a privacy guarantee for any ML model and may be useful to guide the design of privacy defense mechanisms.

3. *Relation between generalization gap and membership inference.* A model that does not generalize well is susceptible to MIAs. Theorem 2, which generalizes [4, Theorem 1], provides a lower bound for the case of bounded loss functions, while Theorems 3 and 4 cover the case of sub-Gaussian and tail-bounded loss functions, respectively. These results provide formal evidence that bad generalization leads to privacy leakage. However, the converse does not hold in general, i.e., *good generalization does not automatically prevent privacy leakage.* Intuitively, one would think that a model that generalizes well would be agnostic to any particular sample being in its training set. Nevertheless, we show, by providing a suitable counterexample, that this intuition is wrong.

4. *Missing information in inference attacks and its connection to generalization.* Using mutual information, we study the amount of information stored by a trained model about its training set, and the role this information plays when the model is susceptible to privacy attacks (Theorem 5). We find that the mutual information between the sensitive attribute and the model parameters upper bounds the gain of the Bayesian attacker over an attacker that uses the prior distribution of the sensitive attribute. This mutual information is in turn upper bounded by the mutual information between the training set and the trained model parameters.

5. *Numerical experiments.* As proof of concept we consider linear regression with Gaussian data (Section 4.1). The simplicity of this setup allows us to estimate the success rate of the Bayesian white-box attacker and, since the loss is exponentially tail-bounded, we can also apply Theorem 4 to monitor the interplay between success rate and generalization. Then we apply our theoretical results in a more practical setting; namely, Deep Neural Networks (DNNs) for classification (Section 4.2). Considering bounded loss functions, we apply Theorem 2 to lower bound the success probability of the Bayesian attacker. We perform MIAs using state of the art strategies to assess the quality of this bound. Lastly, to illustrate that a model susceptible to membership inference might be susceptible to other, more severe, privacy violations, we consider a model for hand-written digit classification and use this example to apply several attribute inference strategies, comparing their effectiveness (Section 4.3).

1.2. Related Work

Connection between Privacy Leakage and Generalization: [4] study the interplay between generalization, Differential Privacy (DP), attribute and membership inference attacks. Our work investigates related questions, but offers a different and complementary perspective. While their analysis considers only bounded loss functions, we extend the results to the more general case of tail-bounded loss functions. They consider a membership inference strategy that uses the loss of the target model, yielding an equivalence between generalization gap and success rate of this attacker. In contrast, we consider a Bayesian attacker with white-box access, yielding an upper bound on the probability of success of all possible adversaries and also on the generalization gap.

Consequently, a large generalization gap implies a high success probability for the attacker. The converse statement, i.e., "*generalization implies privacy*" has been proven false in previous works, such as [7, 8, 4]. Our work also provides a counter proof, giving an example where the generalization gap tends to 0 while the attacker achieves perfect accuracy.

In this line of work, [5] derived an attack strategy for membership inference that is optimal to their setup. However, their results rely on randomness during training and assume a specific form in the distribution of network parameters given the training set. In this sense, our Bayesian attacker can be specialized to their framework and models.

The authors of concurrent work [9] studied the trade-off between the size of the target model (number of model parameters) and the success rate of an optimal attacker within their framework. That setup differs from ours mainly in terms of the capabilities of the attacker; while our attacker has access to the model parameters and full information on the target sample, their attacker only has access to the target sample data and corresponding model output. The work [9] presents a formal relation between the over-parametrization of the model and the success rate the Bayesian attacker against a linear regression model trained on Gaussian data. Differences in the definition of the sample-space, target model and attacker capabilities lead to orthogonal results, but similar conclusions.

Membership Inference: [10] utilize MIAs to measure privacy leakage in deep neural networks. Their attacks consist in training a classifier that distinguishes members from non-members. While their first work covers the case of black-box attacks, subsequent work by [11] considers white-box attacks, where the adversary has access to the model parameters. Later, [12] studied the influence of model choice on the privacy leakage of ML models via membership inference.

Recent works [13, 14, 15, 16] revise new and old membership inference strategies under the light of new evaluation metrics. In particular, the work in [16] takes inspiration from [5], developing an attack strategy based on estimating the distribution of the loss. Further work [17] proposes to use learned differences in distribution between outputs of intermediate layers to predict membership. In [18], a new MIA strategy is proposed, which is based on the magnitude of the perturbation necessary to successfully make the target model change its prediction. It is compared to state-of-the-art methods [14, 11].

The use of shadow models is prevalent in the MIA literature. These models mimic the behavior of the target model, while allowing an attacker access to the training set and model parameters. Many of the aforementioned MIAs require the training of an attacker model (e.g. [14]), while others require the training of shadow models [10, 11, 17, 19] in addition to training an attacker. The attacks in [15] require only black box access to the model and no additional information, while the attacks in [18] require white box access.

Recent work [20], applies the Modified Entropy strategy proposed by [15] to launch MIAs against poisoned target models. This setup differs from previous works in the sense that the attacker plays an active role in the training by poisoning part of the training data. [20] shows that the effectiveness of

MIAs is highly increased against poisoned models.

Typically, when studying the privacy leakage of ML models, classifiers are considered as the target to privacy attacks. In contrast, [21] were the first to consider MIAs against generative models. A comprehensive study of MIAs against GANs and other generative models is provided by [22].

Attribute Inference/Model Inversion: A more severe violation of privacy is represented by attribute inference attacks. Mainly two forms of these attacks have been considered in the literature. The first consists in inferring a sensitive attribute from a partially known record plus knowledge of a model that was trained using this record, e.g. [6, 23, 24, 25, 26, 27]. The second consists in generating a representative sample of one of the members of the training set, or one of the classes in a classification problem, by exploiting knowledge of the target model, e.g. [28, 29, 30, 31, 32, 33]. Our framework is applicable to both forms, but in this work we focus on the former, i.e., inferring sensitive information from a partially known record. [34] propose a framework that generalizes to both types of attribute inference attacks and connects them to several cryptographic notions. The notion of attribute inference is also formalized by [4]. While their work defines the advantage of an adversary as the difference between the information leaked by the model and the information present in the underlying probability distribution of the data, our formalism only allows the adversary to gain advantage from the target model. Furthermore, we consider and compare different attack strategies, while their work only focuses on the attack introduced by [6], and an attacker with oracle access to a membership inference algorithm.

Model Extraction: A third class of privacy violation consists in stealing the functionality of a model, when the model and its parameters are considered sensitive information, e.g., [35], but this setup is out of the scope of our work.

Unintended Memorization: Leakage of sensitive information might be caused by unintended memorization by the model. [7] studies unintended memorization by generative sequence models. They prove that unintended memorization is persistent and hard to avoid; moreover, they find that a model can present exposure even before overfitting. This is an instance in which a model can leak sensitive information even while generalizing well.

Differential Privacy in Machine Learning: DP, [36, 37] is a widely used definition of privacy, which guarantees the safety of individuals in a database while releasing general information about the group. There have been several works in ML that use DP as a measure for privacy or use DP

mechanisms for defense against inference attacks. [38] proposes a Differentially Private Stochastic Gradient Descent method for training neural networks. Their analysis allows them to estimate the privacy budget when successively applying noise to the model parameters during training. Later, [39] presented a comprehensive analysis of DP in ML by considering the different stages in which noise can be added to make an ML model differentially private. [40] evaluates the effectiveness and cost of DP methods for ML in the light of inference attacks. [41] propose *Bayesian DP*, which takes into account the data distribution to provide more practical privacy guarantees, achieving the same accuracy as DP while providing better privacy guarantees on several models and datasets. Recent work [42] proposes an algorithm to "audit" the privacy of ML models, accurately computing the privacy budget necessary to prevent attacks with minimal impact on the utility of the target model. We do not consider the connection between DP and MIAs, as this is thoroughly analyzed in [4].

Federated Learning: Inference attacks that target federated systems have been investigated by [32, 43]. Privacy preserving methods specific for federated learning have been proposed by [44, 45, 46, 47]. [48] provides a comprehensive study of MIAs against Federated Learning models. In these setups the attacker can influence other entities during training. In our framework the attacker directly obtains the trained model; thus, our framework does not cover such cases.

Adversarial Examples and Privacy: There have been several works that combine the topics of privacy and adversarial examples. [49] studies the impact that securing a Machine Learning model against Adversarial Attacks has on the privacy of the model. [50] makes use of Adversarial Examples as part of a defense mechanism against MIAs. [51] were the first to simultaneously address the issues of robustness and privacy, providing a complete analysis of both aspects of DNNs.

2. Preliminaries

A random variable is indicated by upper case (e.g., X). Lower case letter indicate realizations, while calligraphic case denotes the alphabet (e.g., $X \sim \mathcal{X}$ and $x \in \mathcal{X}$). A probability density function (pdf) is denoted by p (e.g., the pdf of X is denoted by p_X). Expectations $\mathbb{E}[\cdot]$ are taken over all random variables inside the square brackets.

We assume a fully Bayesian framework, where $Z = (X, Y) \sim p_{XY} \equiv p_Z$ denotes data X and according labels Y, drawn from sets $\mathcal{X}$ and $\mathcal{Y}$, respectively. The training set consists of n i.i.d. copies $\mathbf{z} \triangleq \{z_1, \ldots, z_n\}$ drawn according to $\mathbf{Z} \sim p_Z^n$.

2.1. Learning and Inference

Let $\mathcal{F} \triangleq \{f_\theta \,|\, \theta \in \Theta\}$ be a *hypothesis class* of (possibly randomized) decision functions parameterized with θ, i.e., for every $\theta \in \Theta$, $f_\theta(\cdot; x)$ is a probability distribution on $\mathcal{Y}$. We will abuse notation and let $f_\theta(y; x)$ be a probability mass function (pmf) or a pdf in y for every $x \in \mathcal{X}$, depending on the context. The symbol $\hat{Y}_\theta(x)$ will be used to denote the random variable on $\mathcal{Y}$ distributed according to $f_\theta(\,\cdot\,; x)$. In case the decision functions are deterministic, i.e., $f_\theta(y; x) \in \{0, 1\}$ is a one-hot pmf for every $\theta \in \Theta$, $x \in \mathcal{X}$, we write $\hat{y}_\theta(x) \in \mathcal{Y}$ to denote this deterministic decision, i.e., $\hat{y}_\theta(x) = \arg\max_{y \in \mathcal{Y}} f_\theta(y; x)$.

A *learning algorithm* is a (possibly randomized) algorithm $\mathcal{A}$ that assigns to every training set $\mathbf{z} \in (\mathcal{X} \times \mathcal{Y})^n$ a probability distribution on the parameter space Θ (and, thus, also on the hypothesis space $\mathcal{F}$). We have $\mathcal{A}\colon \mathbf{z} \mapsto \mathcal{A}(\,\cdot\,; \mathbf{z})$, where $\mathcal{A}(\,\cdot\,; \mathbf{z})$ is a probability distribution on Θ. The symbol $\widehat{\theta}(\mathbf{z})$ is used to denote a random variable on Θ, distributed according to $\mathcal{A}(\,\cdot\,; \mathbf{z})$. In case of a deterministic learning algorithm, we have a pmf $\mathcal{A}(\theta; \mathbf{z}) \in \{0, 1\}$ for every training set $\mathbf{z}$ and can thus define the function $\widehat{\theta}(\mathbf{z}) = \arg\max_{\theta \in \Theta} \mathcal{A}(\theta; \mathbf{z})$, yielding the (possibly random) decision function $f_{\widehat{\theta}(\mathbf{z})}$.

To judge the quality of a decision function $f \in \mathcal{F}$ we require a loss function $\ell\colon \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$. We naturally extend this definition to vectors by an average over component-wise application, i.e., $\ell(\mathbf{y}, \mathbf{y}') = \frac{1}{n} \sum_{i=1}^n \ell(y_i, y_i')$.

Definition 1 (Expected risks)**.** *We define $\varrho(\theta, (x, y)) \triangleq \mathbb{E}[\ell(\hat{Y}_\theta(x), y)]$ as the expected loss between $f_\theta(x)$ and y. This notation is naturally extended to vectors as*

$$\varrho(\theta, \mathbf{z}) \triangleq \frac{1}{n} \sum_{i=1}^n \mathbb{E}[\ell(\hat{Y}_\theta(x_i), y_i)]. \tag{1}$$

The expected risk *and* empirical risk *of a learning algorithm $\mathcal{A}$ at training*

set $\mathbf{Z}$ *are defined respectively as*[1]

$$\mathcal{R}_{\mathrm{exp}}(\mathcal{A}) \triangleq \mathbb{E}\big[\varrho\big(\widehat{\theta}(\mathbf{Z}),(X,Y)\big)\big]\ ,\ \mathcal{R}_{\mathrm{emp}}(\mathcal{A}) \triangleq \varrho\big(\widehat{\theta}(\mathbf{Z}),\mathbf{Z}\big)\ , \tag{2}$$

where the training set $\mathbf{Z}$ *and* (X,Y) *are independent. The difference between expected and empirical risk is the* generalization gap $\mathcal{G}_{\mathrm{G}}(\mathcal{A})$*, and its expectation* $\mathcal{E}_{\mathrm{G}}(\mathcal{A})$*, which are respectively defined as*

$$\mathcal{G}_{\mathrm{G}}(\mathcal{A}) \triangleq \mathcal{R}_{\mathrm{exp}}(\mathcal{A}) - \mathcal{R}_{\mathrm{emp}}(\mathcal{A})\ ,\ \mathcal{E}_{\mathrm{G}}(\mathcal{A}) \triangleq \mathbb{E}[\mathcal{G}_{\mathrm{G}}(\mathcal{A})]\,. \tag{3}$$

2.2. Attack Model and Assumptions

In order to make privacy guarantees for an algorithm $\mathcal{A}$, we need to specify an attacker model and the capabilities of an attacker. We will adopt a point of view of information-theoretic privacy and will not make assumptions about the computation power afforded to an attacker. We will also assume that the attacker has perfect knowledge of the underlying data distribution p_Z, as well as the algorithm $\mathcal{A}$.

In general, the goal of the attacker is to infer some property of $\mathbf{z}$ from $\widehat{\theta}(\mathbf{z})$. However, in general the attacker may have access to certain side information. This may include the specific potential member of the training set that is queried (in case of a MIA) or any additional knowledge gained by the attacker. This side information is modeled by a random variable $S \in \mathcal{S}$, dependent on $\mathbf{Z}$, the value of which is known to the attacker. The attacker is interested in a target (or concept) property denoted by a random variable $T \in \mathcal{T}$, which is also dependent on $(\mathbf{Z}, S)$. A (white box) *attack strategy* is a (measurable) function $\varphi\colon \Theta \times \mathcal{S} \to \mathcal{T}$.

We shall assume that S and T are independent, but not necessarily conditionally independent given $\mathbf{Z}$. This natural assumption ensures that knowledge of the side-information S does not change the prior $p_T = p_{T|S}$ of the attacker.

Definition 2. *The Bayes* success probability *of a (randomized) attack strategy* φ *is*

$$\mathcal{P}_{\mathrm{Suc}}(\varphi) = \mathrm{P}\{\varphi(\widehat{\theta}(\mathbf{Z}), S) = T\}. \tag{4}$$

[1]Note that the expectation is taken over all random quantities, i.e., $\mathbf{Z} \sim p_Z^n$, $\widehat{\theta}(\mathbf{Z}) \sim \mathcal{A}(\,\cdot\,;\mathbf{Z})$, and $(X,Y) \sim p_Z$.

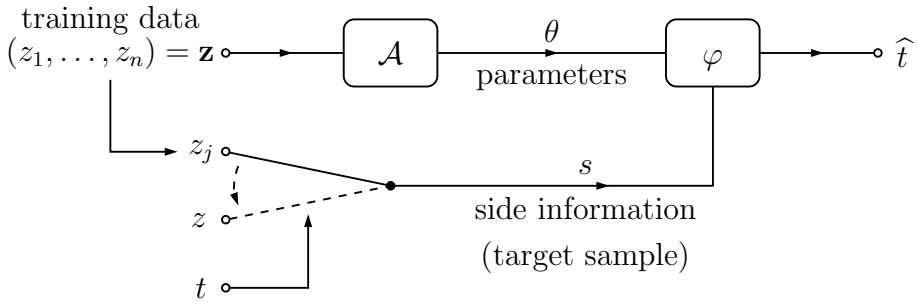

Figure 1: Scheme of a MIA. If $t = 1$, the target sample is drawn from the training set $\mathbf{z} = (z_1, \ldots, z_n)$ used by $\mathcal{A}$ to train the target model. If $t = 0$, the target sample is independently drawn from the data distribution. The attacker φ then uses the parameters θ at the output of $\mathcal{A}$ and the side information s to provide an estimate $\hat{t}$ of t.

We may additionally define the success probability conditioned on side information $S = s$ *as*

$$\mathcal{P}_{\mathrm{Suc}}(\varphi|s) = \mathrm{P}\{\varphi(\widehat{\theta}(\mathbf{Z}), s) = T|S = s\}. \tag{5}$$

Definition 3 (Attribute inference attack)**.** *We model the non-sensitive attribute by a random variable $V \in \mathcal{V}$. In this context, the input to the model is formed by the sensitive and non-sensitive attributes $X \equiv (V, T)$. Thus $\mathcal{X} \subseteq \mathcal{V} \times \mathcal{T}$. The side information given to the attacker can consist of $S = V$ or $S = (V, Y)$, depending on the attack strategy considered. In this work we only consider attribute inference attacks were $\mathcal{T}$ is finite.*

Definition 4 (MIA)**.** *In a MIA, let T be a Bernoulli variable on $\mathcal{T} = \{0, 1\}$ and J is independently, uniformly distributed on $\{1, 2, \ldots, n\}$. Then set $S = TZ_J + (1-T)Z$, where Z_J is a random element of the training set and $Z \sim p_Z$ is independently drawn. Thus, an attacker needs to determine if $T = 1$, i.e., whether S is part of the training set or not.*

For later use we define the random variable $R \triangleq \varrho(\widehat{\theta}(\mathbf{Z}), S)$, i.e., the (random) loss function evaluated at S (cf. Definition 1).

A MIA using an arbitrary strategy is illustrated in Fig. 1.

Although, in practice, the prior distribution of the target attribute T is usually unknown, we define the optimal rejection region of an idealized attacker, having access to all other involved distributions.

Definition 5 (Most powerful test according to Neyman-Pearson lemma)**.** *In a membership inference setup (Definition 4), define, for a threshold* $0 < \gamma < \infty$*, the decision region*

$$\widehat{\mathcal{T}}(\gamma) \triangleq \left\{(\theta, s) \in \Theta \times \mathcal{S} : p_{\widehat{\theta}(\mathbf{Z})S|T}\big(\theta, s|1\big) > \gamma \cdot p_{\widehat{\theta}(\mathbf{Z})S|T}\big(\theta, s|0\big)\right\}. \tag{6}$$

By the Neyman-Pearson lemma [52], the most powerful test at threshold γ *is then given by detecting* $T = 1$ *for all pairs* $(\theta, s) \in \widehat{\mathcal{T}}(\gamma)$*, i.e.,* $\varphi(\theta, s) = 1$ *if and only if* $(\theta, s) \in \widehat{\mathcal{T}}(\gamma)$*.*

In Proposition 1 we will provide lower bounds on the error achieved by this decision region and make the connection to the fully Bayesian case.

3. Main Results

3.1. Performance of the Bayesian Attacker

In this section, we establish two theorems that provide upper bounds on the success probability of an arbitrary attacker. First, consider the general case in which the target attribute T is not necessarily binary, but finite. This case includes both membership and feature inference attacks. In this case the Bayes classifier is the best possible attacker, which arises naturally from a maximum a posteriori optimization of the target attribute.

Theorem 1 (Success of the Bayesian attacker)**.** *Assume that* $\mathcal{T}$ *is a finite set and* φ *is an arbitrary attack strategy.*[2] *The Bayes success probability is upper bounded by,*

$$\mathcal{P}_{\text{Suc}}(\varphi) \leq \mathbb{E}\left[\max_{t \in \mathcal{T}} p_{T|\widehat{\theta}(\mathbf{Z})S}(t|\widehat{\theta}(\mathbf{Z}), S)\right], \tag{7}$$

where the upper bound is achieved by the attack strategy,

$$\varphi^{\star}(\theta, s) = \arg\max_{t \in \mathcal{T}} p_{T|\widehat{\theta}(\mathbf{Z})S}(t|\theta, s)\,. \tag{8}$$

If the $\arg\max$ *in (8) is not unique, any* $t \in \mathcal{T}$ *achieving the maximum can be chosen.*

[2] As this result provides an upper bound on the success probability, no restrictions are placed on the capabilities of the attacker.

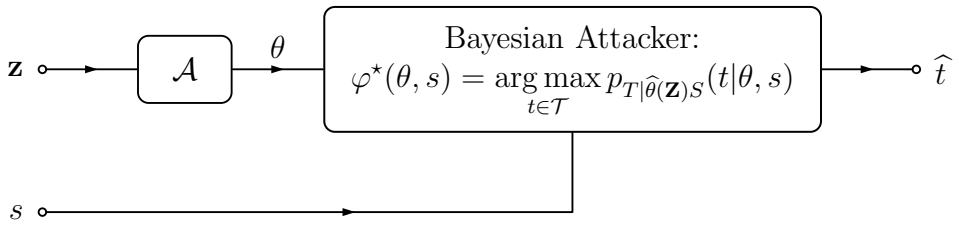

Figure 2: Scheme of the Bayesian attacker. The Bayesian attacker achieves the upper bound shown in Theorem 1, but needs to be able to evaluate the conditional distribution $p_{T|\widehat{\theta}(\mathbf{Z})S}$. The observations required for the attack are the side-information s and model parameters θ.

Proof. Let $\widehat{T}$ denote the random variable defined by $\widehat{T} \triangleq \varphi(\widehat{\theta}(\mathbf{Z}), S)$. Note that $\widehat{T}$ is independent from T given $(\widehat{\theta}(\mathbf{Z}), S)$. First, the upper bound in (7) is shown, then it is shown that this upper bound is achieved by (8). Let φ be an arbitrary attack strategy defining pdf $p_{\widehat{T}|\widehat{\theta}(\mathbf{Z})S}(\widehat{t}|\theta, s)$ for each $(\theta, s) \in \Theta \times \mathcal{S}$,

$$\begin{aligned}\mathcal{P}_{\text{Suc}}(\varphi) &= \mathbb{E}\left[\sum_{\widehat{t}\in\mathcal{T}} p_{\widehat{T}|\widehat{\theta}(\mathbf{Z})S}(\widehat{t}|\widehat{\theta}(\mathbf{Z}), S) p_{T|\widehat{\theta}(\mathbf{Z})S}(\widehat{t}|\widehat{\theta}(\mathbf{Z}), S)\right] \\ &\leq \mathbb{E}\left[\max_{t'\in\mathcal{T}} p_{T|\widehat{\theta}(\mathbf{Z})S}(t'|\widehat{\theta}(\mathbf{Z}), S)\right]. \end{aligned} \tag{9}$$

Now, consider an attack strategy $\varphi^\star$, such that $\varphi^\star(\theta, s)$ is in

$$\left\{t \in \mathcal{T} : p_{T|\widehat{\theta}(\mathbf{Z})S}(t|\theta, s) = \max_{t'\in\mathcal{T}} p_{T|\widehat{\theta}(\mathbf{Z})S}(t'|\theta, s)\right\}, \tag{10}$$

for given $\theta \in \Theta$ and $s \in \mathcal{S}$. Hence,

$$\mathcal{P}_{\text{Suc}}(\varphi^\star) = \mathbb{E}\left[\max_{t'\in\mathcal{T}} p_{T|\widehat{\theta}(\mathbf{Z})S}(t'|\widehat{\theta}(\mathbf{Z}), S)\right]. \tag{11}$$

Note that the bound is achieved as long as (10) is satisfied. □

A schema of the Bayesian attack is shown in Fig. 2. Given white-box access to the model and its parameters, as well as side information, the attacker (8) has the highest probability of successfully identifying a record in the training set. Thus, resilience against strategy (8) provides a strong privacy guarantee. Note that, even though S plays a very specific role in a

MIA, it may contain additional samples, or any other kind of information, making Theorem 1 applicable to other setups.

Theorem 1 can also be applied to the black-box case. A black-box attack is not granted access to the parameters $\theta \in \Theta$, but only to the input-output relation $\left\{\left(x, f_\theta(x)\right) \mid x \in \mathcal{X}\right\}$ where $f_\theta \in \mathcal{F}$ is the model associated to the parameters θ. Thus, any black-box attack strategy $\varphi' : \mathcal{F} \times \mathcal{S} \to \mathcal{T}$ can be seen as a particular case of a white-box strategy defined as $\varphi(\theta, s) = \varphi'(f_\theta, s)$, and therefore the upper bound expressed by Theorem 1 still applies, since it is an upper bound for *all* strategies.

Similarly, when the attacker has access to only a subset of the parameters, it can be seen as a particular case of the attacker considered in Theorem 1, and therefore the result still applies.

The following proposition provides similar results for the membership inference problem.

Proposition 1 (Decision tradeoff)**.** *In a membership inference setup (Definition 4), let $\widehat{\mathcal{T}} \subseteq \Theta \times \mathcal{S}$ be any decision set, and define*

$$\epsilon_1(\widehat{\mathcal{T}}) \triangleq \int_{\widehat{\mathcal{T}}} p_{\widehat{\theta}(\mathbf{Z})S|T}(\theta, s|0)\, d\theta\, ds \ , \ \epsilon_0(\widehat{\mathcal{T}}^c) \triangleq \int_{\widehat{\mathcal{T}}^c} p_{\widehat{\theta}(\mathbf{Z})S|T}(\theta, s|1)\, d\theta\, ds, \quad (12)$$

the average Type-I (false positive) and Type-II (false negative) error probabilities, respectively. Then,

$$\epsilon_0(\widehat{\mathcal{T}}) + \epsilon_1(\widehat{\mathcal{T}}^c) \geq 1 - \Delta, \quad (13)$$

where $\Delta \triangleq \left\| p_{\widehat{\theta}(\mathbf{Z})S|T=1} - p_{\widehat{\theta}(\mathbf{Z})S|T=0} \right\|_{\mathrm{TV}}$ and $\| \cdot \|_{\mathrm{TV}}$ is the total variation distance [53]. Equality is achieved by choosing $\widehat{\mathcal{T}}^\star \equiv \widehat{\mathcal{T}}(1)$ according to Definition 5. If the hypotheses are equality distributed, then the minimum average Bayesian error satisfies

$$\inf_{\varphi} \mathrm{P}\left\{\varphi(\widehat{\theta}(\mathbf{Z}), S) \neq T\right\} = \frac{1}{2}\left(1 - \Delta\right) \ . \quad (14)$$

The proof of this proposition is rather lengthy and so is relegated to Appendix B. Equation (13), similar to (7), provides a lower bound for the total error of an arbitrary attacker. Equation (14) provides the error of the Bayesian attacker from Theorem 1 in the case where the hypotheses are equally distributed.

3.2. Generalization Gap and Success of the Attacker

In this section, we explore the connection between the generalization gap and the success probability of MIAs. Large generalization gap implies poor privacy guarantees against MIAs. Moreover, depending on characteristics of the loss function, the probability of success of the attacker is lower bounded by the generalization gap:

Theorem 2 (Bounded loss function)**.** *If the loss is bounded by* $|\ell| \leq \ell_{\max}$*, then there is an attack strategy* φ *for a MIA (Definition 4) such that,*

$$\mathcal{P}_{\mathrm{Suc}}(\varphi) \geq \max\left\{P_{\mathrm{m}}, P_{\mathrm{m}}\left(\frac{|\mathcal{E}_{\mathrm{G}}(\mathcal{A})|}{2\ell_{\max}} - 1\right) + 1\right\} ,$$

where $P_{\mathrm{m}} \triangleq \max_{t\in\{0,1\}} \mathrm{P}\{T = t\}$.

Proof. Recalling Definitions 1 and 4, and in particular $\varrho(\theta, (x, y)) = \mathbb{E}[\ell(\hat{Y}_\theta(x), y)]$, as well as the random variable $R = \varrho(\widehat{\theta}(\mathbf{Z}), S)$, we obtain

$$\begin{aligned}
|\mathcal{E}_{\mathrm{G}}(\mathcal{A})| &= \left|\int r(p_{R|T}(r|0) - p_{R|T}(r|1))\, dr\right| \\
&\leq \int |r||p_{R|T}(r|0) - p_{R|T}(r|1)|\, dr \\
&\leq \ell_{\max}\|p_{R|T}(\cdot|0) - p_{R|T}(\cdot|1)\|_1. \qquad (15)
\end{aligned}$$

Assume w.l.o.g. that the attacker φ satisfies the condition,

$$p_{RT}(\varrho(\theta, (x, y)), \varphi(\theta, x, y)) \geq p_{RT}(\varrho(\theta, (x, y)), 1 - \varphi(\theta, x, y)). \qquad (16)$$

Thus, we obtain,

$$\begin{aligned}
\mathcal{P}_{\mathrm{Suc}}(\varphi) &= \frac{1}{2}\left(1 + \int |p_{RT}(r, 0) - p_{RT}(r, 1)|\, dr\right) \\
&\geq \frac{1}{2}P_{\mathrm{m}}\|p_{R|T}(\cdot|0) - p_{R|T}(\cdot|1)\|_1 + 1 - P_{\mathrm{m}} \\
&\geq P_{\mathrm{m}}\left(\frac{|\mathcal{E}_{\mathrm{G}}(\mathcal{A})|}{2\ell_{\max}} - 1\right) + 1. \qquad (17)
\end{aligned}$$

Note that the lower bound, $P_{\mathrm{m}}\left(\frac{|\mathcal{E}_{\mathrm{G}}(\mathcal{A})|}{2\ell_{\max}} - 1\right) + 1$, varies from $1 - P_{\mathrm{m}}$ to 1, as the generalization gap increases. However, an attacker with knowledge of the

prior on T can always have a success probability of at least P_{m} by guessing $\widehat{t} = \arg\max_{t \in \mathcal{T}} \mathrm{P}\{T = t\}$; therefore,

$$\mathcal{P}_{\mathrm{Suc}}(\varphi) \geq \max\left\{P_{\mathrm{m}}, P_{\mathrm{m}}\left(\frac{|\mathcal{E}_{\mathrm{G}}(\mathcal{A})|}{2\ell_{\max}} - 1\right) + 1\right\} \qquad \square$$

Theorem 2 indicates that strong privacy guarantees (i.e., small success probability for any attacker), imply that the generalization gap is also small. We remark that, on the other hand, ensuring that the generalization gap is small does not make a model robust against MIAs. We shall return to this important point in Section 3.3.

In the following, we extend the result of Theorem 2 to sub-Gaussian and exponentially tail-bounded loss functions.

Theorem 3 (Sub-Gaussian loss)**.** *In a membership inference problem (Definition 4), assume that $R = \varrho(\widehat{\theta}(\mathbf{Z}), S)$ is a sub-Gaussian random variable with variance proxy σ_R^2. For all $R_{\max} \geq r_0 \triangleq \sqrt{2\sigma_R^2 \log 2}$, there exists an attack strategy φ, such that,*

$$\mathcal{P}_{\mathrm{Suc}}(\varphi) \geq \max\left\{P_{\mathrm{m}}, P_{\mathrm{m}}\left(\frac{|\mathcal{E}_{\mathrm{G}}(\mathcal{A})|}{2R_{\max}} - \frac{C(R_{\max}, \sigma_R)}{1 - P_{\mathrm{m}}} - 1\right) + 1\right\}. \tag{18}$$

where $C(R_{\max}, \sigma_R) \triangleq \exp\left(-\frac{R_{\max}^2}{2\sigma_R^2}\right)\left(1 + \frac{\sigma_R^2}{R_{\max}^2}\right)$.

Proof. Given that R is a sub-Gaussian random variable with variance proxy σ_R^2, we have $\mathrm{P}\{|R| \geq r\} \leq 2e^{-\frac{r^2}{2\sigma_R^2}}$ for all $r \geq 0$ [54]. Define the random variable R_0 to have the distribution function $Q_0(r) \triangleq \mathrm{P}\{R_0 \leq r\} \triangleq 1 - 2e^{-\frac{r^2}{2\sigma_R^2}}$ on its support $[r_0, \infty)$, where $r_0 = \sqrt{2\sigma_R^2 \log 2}$, i.e., the pdf of R_0 is $p_{R_0}(r) = \frac{2r}{\sigma_R^2} e^{-\frac{r^2}{2\sigma_R^2}}$. Let Q be the distribution function of $|R|$. Then, using the construction in the proof of [55, Theorem 1.104], we can write $|R| = Q^{-1} \circ Q_0(R_0)$, where Q^{-1} is the left continuous inverse of Q, noting that Q_0 is continuous. The sub-Gaussian property then implies $Q(r) = 1 - \mathrm{P}\{|R| \geq r\} \geq Q_0(r)$, which immediately yields $Q^{-1} \circ Q_0(r) \leq r$.

We thus have, for $R_{\max} \geq r_0$,

$$\begin{aligned}\int_{|r|\geq R_{\max}} |r| p_R(r) dr &= \int_{Q_0(r)\geq Q(R_{\max})} Q^{-1}(Q_0(r)) p_{R_0}(r) dr \\ &\leq \int_{Q_0(r)\geq Q(R_{\max})} r p_{R_0}(r) dr \\ &\leq \int_{r\geq R_{\max}} r p_{R_0}(r) dr \\ &\leq 2R_{\max} C(R_{\max}, \sigma_R) \end{aligned} \tag{19}$$

Following steps similar to those in (15),

$$\begin{aligned}|\mathcal{E}_{\mathrm{G}}(\mathcal{A})| &\leq \int_{|r|\leq R_{\max}} |r| |p_{R|T}(r|0) - p_{R|T}(r|1)| \, dr \\ &\quad + \int_{|r|> R_{\max}} |r| |p_{R|T}(r|0) - p_{R|T}(r|1)| \, dr \\ &\leq R_{\max} \| p_{R|T}(r|0) - p_{R|T}(r|1) \|_1 + \frac{2R_{\max} C(R_{\max}, \sigma_R)}{1 - P_{\mathrm{m}}} \, , \end{aligned} \tag{20}$$

where the last inequality follows from (19). Consequently,

$$\| p_{R|T}(r|0) - p_{R|T}(r|1) \|_1 \geq \frac{|\mathcal{E}_{\mathrm{G}}(\mathcal{A})|}{R_{\max}} - \frac{2C(R_{\max}, \sigma_R)}{1 - P_{\mathrm{m}}}. \tag{21}$$

The rest of the proof follows identically to that of Theorem 2. □

Theorem 4 (Tail-bounded loss)**.** *In a membership inference problem (Definition 4), assume that* $R = \varrho(\widehat{\theta}(\mathbf{Z}), S)$ *is such that* $\mathrm{P}\{|R| \geq r\} \leq 2\exp(-r/2\sigma_R^2)$ *for all* $r \geq 0$ *with some variance proxy* $\sigma_R^2 > 0$*. Then, for all* $R_{\max} \geq r_0 \triangleq 2\sigma_R^2 \log 2$*, there is an attack strategy* φ *such that, (18) holds with*

$$C(R_{\max}, \sigma_R) \triangleq \exp\left(-\frac{R_{\max}}{2\sigma_R^2}\right)\left(1 + \frac{2\sigma_R^2}{R_{\max}}\right). \tag{22}$$

The proof of this theorem is analogous to that of Theorem 3 and will be omitted here.

Note that in principle both Theorem 3 and Theorem 4 are applicable when the loss is bounded, since all bounded random variables are sub-Gaussian and exponentially tail-bounded; nonetheless, we expect Theorem 2 to provide a tighter bound in this case, as it certainly does for $\ell_{\max} = R_{\max}$.

In practice the distribution of the loss for a particular model is often unknown; however, it can be estimated and fitted to one of the cases presented in this section. Then, these results can be applied to measure the potential impact of generalization on the privacy leakage of the model.

3.3. Good Generalization is not Enough to Prevent Successful Attacks

Generalization does not imply privacy. The purpose of this section is to prove that in general the success rate of the attacker may not be directly proportional to the generalization gap. We show this by constructing a synthetic example of a membership inference problem, where the generalization gap can be made arbitrarily small, while T can be determined with certainty by an attacker. To construct the counterexample we need to define the random variables X, Y and a loss function ℓ for fixed parameters $0 < \delta < D$. Let p_X be an arbitrary continuous pdf on $\mathbb{R}$, e.g., $X \sim \mathcal{N}(0, \sigma^2)$, and define $Y = X + U$, where U is independent of X and uniformly distributed on $[-\frac{\epsilon}{2}, \frac{\epsilon}{2}]$. Given the training set $\mathbf{z}$ and an input x, the learned decision function $f(\,\cdot\,; x)$ either outputs the correct label y, if $(x, y) \in \mathbf{z}$, and otherwise $f(\cdot; x) = x + D + U'$, where U' is an i.i.d. copy of U, i.e., uniformly distributed on $[-\frac{\epsilon}{2}, \frac{\epsilon}{2}]$. With Euclidean distance loss $\ell(y, y') = |y - y'|$, these definitions immediately yield $\mathrm{P}\{R = 0|T = 1\} = 1$ and the conditional pdf

$$p_{R|T}(r|0) = \frac{1}{\epsilon}\Lambda((r - D)/\epsilon). \tag{23}$$

where $\Lambda(r) \triangleq \max(1 - |r|, 0)$ is the triangle distribution. The parameters $0 < \epsilon < D$ can be chosen arbitrarily. Clearly then an attacker can simply check whether $R = 0$ to determine T with probability one. On the other hand, from (23), it is easily verified that,

$$|\mathcal{E}_{\mathrm{G}}(\mathcal{A})| = |\mathbb{E}[R|T = 0] - \mathbb{E}[R|T = 1]| = D. \tag{24}$$

Thus, by varying the parameter D, we can make the generalization gap arbitrarily small, while the attacker maintains perfect success. Therefore, good generalization does not prevent the attacker from easily determining which samples were part of the training set. Remark that as NNs are universal approximators, any (reasonable) function, including the decision rule in this example, can be approximated to arbitrary degree by a NN; therefore, this behavior could be seen in practice.

3.4. On the Amount of Missing Information in Inference Attacks and Generalization

We aim at investigating the following simple but fundamental questions, from the perspective of information theory:

- *How much information do the model parameters $\widehat{\theta}(\mathbf{z})$ store about the training set $\mathbf{z}$? How is this information related to the generalization gap?*
- *How much information about the unknown (sensitive) attribute T is contained in the model parameters $\widehat{\theta}(\mathbf{z})$ and the side information S? And how much information is needed for the inference of T?*
- *How do the above information quantities relate or bound to each other?*

From the point of view of information theory these questions make sense only if we consider $\widehat{\theta}(\mathbf{z})$ and T as random variables, that is, attribute probabilities to the target attribute and model parameters, which is perfectly consistent with the investigated framework in this paper.

To state the following theorem, we need the *Fenchel-Legendre dual function* [56] $g^\star\colon \mathbb{R} \to \mathbb{R}$ of a function $g\colon \mathbb{R} \to \mathbb{R}$, which is defined as $g^\star(t) \triangleq \sup\{\lambda \cdot t - g(\lambda) : \lambda \in \mathbb{R}\}$. We will also use the log-moment-generating function $\psi_W\colon \mathbb{R} \to \mathbb{R}$ of a random variable W, defined as $\psi_W(\lambda) \triangleq \log \mathbb{E}[e^{\lambda W}]$. More information on these quantities and their properties are given in the discussion of the Cramér-Chernoff Method in Appendix D.2.

Theorem 5 (Mutual information)**.** *Let $\widehat{T} \triangleq \varphi(\widehat{\theta}(\mathbf{Z}), S)$ be the (random) prediction of any attacker φ (Definition 2). Then,*

$$I\big(T; \widehat{\theta}(\mathbf{Z}) \big| S\big) \geq d_{KL}\Big(\mathcal{P}_{\mathrm{Suc}}(\varphi) \,\Big\|\, \max_{t \in \mathcal{T}} p_T(t)\Big), \tag{25}$$

where $d_{KL}(p\|q)$ denotes the KL divergence between Bernoulli random variables with probabilities (p, q). Moreover, for $\epsilon \geq 0$, the generalization gap $\mathcal{E}_{\mathrm{G}}$ at $\mathbf{Z}$ satisfies

$$\mathrm{P}\big(\mathcal{G}_{\mathrm{G}}(\mathcal{A}) \geq \epsilon\big) \leq \frac{I(\mathbf{Z}; \widehat{\theta}(\mathbf{Z})) + 1}{nK(\epsilon)}, \tag{26}$$

where

$$K(\epsilon) \triangleq \operatorname*{ess\,inf}_{\theta \sim P_{\widehat{\theta}(\mathbf{Z})}} \psi^*_{\mathbb{E}[\varrho(\widehat{\theta},(X,Y))] - \varrho(\theta,(X,Y))}(\epsilon) \tag{27}$$

is an essential infimum w.r.t. $\theta \sim P_{\widehat{\theta}(\mathbf{Z})}$ *of the Fenchel-Legendre dual function* $\psi^\star$ *of the log-moment-generating function of* $\mathbb{E}[\varrho(\theta,(X,Y))] - \varrho(\theta,(X,Y))$. *Furthermore,*

$$I(T;\widehat{\theta}(\mathbf{Z})|S) = I(S;\widehat{\theta}(\mathbf{Z})|T) - I(S;\widehat{\theta}(\mathbf{Z})) \leq I(\mathbf{Z};\widehat{\theta}(\mathbf{Z})) - I(S;\widehat{\theta}(\mathbf{Z})). \quad (28)$$

Theorem 5 is proved in Appendix D.

The mutual information expressions in (25) and (26) are related by the inequality (28), where $I(\mathbf{Z};\widehat{\theta}(\mathbf{Z}))$ represents the average amount of information about the random training set $\mathbf{Z}$ retained in the model parameters $\widehat{\theta}(\mathbf{Z})$; and $I(S;\widehat{\theta}(\mathbf{Z}))$ indicates the amount of information already contained in the side information S before observing the parameters $\widehat{\theta}(\mathbf{Z})$.

From (28) it is clear that by controlling the average number of bits of information about the training set $\mathbf{Z}$ that the model parameters $\widehat{\theta}(\mathbf{z})$ store, i.e., $I(\mathbf{Z};\widehat{\theta}(\mathbf{Z})) \leq r$, it is possible to control both the generalization gap in (26) and the accuracy of any possible attacker in (25). Nevertheless, a more effective defense strategy may aim directly at reducing the mutual information $I(T;\widehat{\theta}(\mathbf{Z})|S)$, which is expected to have less severe impact on the performance of the trained model, i.e., the expected risk $\mathbb{E}\big[\ell(\hat{Y}_{\widehat{\theta}(\mathbf{Z})}(X),Y)\big]$. As (25) indicates, the performance of any attacker must be close to a random guess if the mutual information $I(T;\widehat{\theta}(\mathbf{Z})|S)$ is suitably small. This equation can be numerically computed to obtain an upper bound on $\mathcal{P}_{\mathrm{Suc}}(\varphi)$.

The generalization gap bound in (26) is subtly different from most PAC-Bayes scenarios of learning. In the present case, we are bounding the joint probability over both the training data $\mathbf{Z}$ and the randomness involved in the learning algorithm, which is within the spirit of the work by [57]. But due to the term $K(\epsilon)$, the bound presented in (26) is tighter.

Assuming that the loss is sub-Gaussian or bounded, it is not difficult to provide a lower bound for $K(\epsilon)$ that is independent of the underlying data distribution.

4. Examples and Numerical Experiments

4.1. Linear Regression on (Synthetic) Gaussian Data

The following example allows us to illustrate how the theoretical results from the previous section might be used to assess the privacy guarantees of a specific model. We implement the Bayesian attacker from Theorem 1 and

estimate its success probability to monitor the privacy leakage of the model as a function of the number of training samples. Second, since the loss is tail-bounded exponentially, we use Theorem 4 to derive lower bounds on the success probability of the attacker. Lastly, we utilize (25) from Theorem 5 to upper bound the success probability of the Bayesian attacker.

For $i \in [n]$, let x_i be a fixed vector on $\mathbb{R}^d$ and for a fixed vector $\beta \in \mathbb{R}^d$, let $Y_i = \beta^T x_i + W_i$ with $\mathbb{E}[W_i] = 0$ and $\mathbb{E}[W_i^2] = \sigma^2 < \infty$ for $i \in [n]$. The training set is $\mathbf{z} = \{y_1, \ldots, y_n\}$, a realization of Y_i for each $i \in [n]$. The function space $\mathcal{F}$ consists of linear regression functions $f_\theta(x_i) = \theta^T x_i$ for $\theta \in \mathbb{R}^d$ and the deterministic algorithm $\mathcal{A}$ minimizes squared error on the training set and thus yields[3] $\widehat{\theta}(\mathbf{y}) = (\mathbf{x}\mathbf{x}^T)^{-1}\mathbf{x}\mathbf{y}^T$ and the associated decision function $f_{\widehat{\theta}(\mathbf{y})}(x_i) = \mathbf{y}\mathbf{x}^T(\mathbf{x}\mathbf{x}^T)^{-1}x_i$. Using squared error loss, $\ell(y, y') = (y - y')^2$, we obtain the generalization gap,

$$\mathcal{E}_{\mathrm{G}}(\mathcal{A}) = \frac{2d}{n}\sigma^2 \,, \tag{29}$$

A derivation of this formula is presented in Appendix E. Assuming the noise W to be Gaussian, the scalar response $\mathbf{Y} = \beta^T\mathbf{x} + \mathbf{W}$ then also follows a Gaussian distribution, with $\mathbf{W}$ a row vector of i.i.d. components. Similarly, the model parameters $\widehat{\theta}(\mathbf{Y})$ are normally distributed. Now choose a test sample $S_J = T(Y_J) + (1-T)(Y'_J)$, where J is an index in $[n]$, Y_J is the $J-$th component of the (random) training set and Y'_J is drawn independently of the training set. Assuming a Bernoulli $1/2$ prior on the hypothesis T, the success probability of the Bayesian attacker $\varphi^\star$ is given by

$$\mathcal{P}_{\mathrm{Suc}}(\varphi^\star) = 1 - \frac{1}{2}\left[\epsilon_0\big(\widehat{\mathcal{T}}(1)\big) + \epsilon_1\big(\widehat{\mathcal{T}}(1)^c\big)\right] \,, \tag{30}$$

with the Type-I and Type-II errors defined by (12), and the optimal decision region $\widehat{\mathcal{T}}(1)$ defined by (6). With posteriors defined by,

$$p_{S_J J\widehat{\theta}|T}(s, j, \theta|0) = \frac{1}{n}Q(\theta)p_{Y_j}(s) \,, \tag{31}$$

$$p_{S_J J\widehat{\theta}|T}(s, j, \theta|1) = \frac{1}{n}Q_j(\theta|s)p_{Y_j}(s) \,. \tag{32}$$

[3]Let $\mathbf{x}$ be the $[d \times n]$ matrix $\mathbf{x} = (x_1, x_2, \ldots, x_n)$. Similarly, $\mathbf{y} = (y_1, y_2, \ldots, y_n)$ and $\mathbf{W} = (W_1, W_2, \ldots, W_n)$ are $[1 \times n]$ vectors.

The index j indicates the feature vector x_j from which the test sample s is generated. $Q(\theta)$ is the distribution of the model parameters conditioned to $T = 0$. It is independent of the test sample s and of the index j. $Q_j(\theta|s)$ is the distribution of the model parameters conditioned to $T = 1$. Since, under this hypothesis, the attacker assumes s is one of the samples in the training set, this conditional distribution depends on the test sample s and its corresponding index j. The distribution of the test sample p_{Y_j} is defined by $p_{Y_j}(\cdot) \triangleq \mathcal{N}(\cdot; \beta^T x_j, \sigma^2)$. $Q(\cdot)$ and $Q_j(\cdot|s)$ are defined by $Q(\cdot) \triangleq \mathcal{N}(\cdot; \beta, \sigma^2 \overline{x}^{-1})$ and $Q_j(\cdot|s) \triangleq \mathcal{N}\big(\cdot; \beta + \overline{x}^{-1} x_j (s - x_j^T \beta), \sigma^2 \overline{x}^{-1} (\mathbb{I}^{d\times d} - x_j x_j^T \overline{x}^{-1})\big)$, respectively, where $\overline{x} \triangleq \mathbf{x}\mathbf{x}^T$. These distributions are derived in Appendix E.

The success probability of the Bayesian attack strategy in Theorem 1 is given by (30). In our experiments we perform a Monte Carlo estimation of the integrals in (12), by randomly drawing T, s and θ. The posterior distributions can be computed in closed form with the above definitions. Since the loss is exponentially tail-bounded, we can apply Theorem 4 to obtain the lower bound

$$\mathcal{P}_{\text{Suc}}(\varphi^\star) \geq \frac{1}{2} + \frac{d}{2n} \frac{\sigma^2}{R_{\max}} - C(R_{\max}, \sigma), \tag{33}$$

where we used (29) and $C(R_{\max}, \sigma)$ is defined in expression (22). $R_{\max}$ can be chosen to maximize the upper bound in this expression. In our experiments, we choose the optimal $R_{\max}$ using the golden section search algorithm. Furthermore, from (25) we have,

$$I\big(S_J; \widehat{\theta}(\mathbf{Y}) \big| T\big) \geq d_{\text{KL}}\Big(\mathcal{P}_{\text{Suc}}(\varphi) \,\Big\|\, \max_{t\in\mathcal{T}} p_T(t)\Big) \,. \tag{34}$$

Note that $I\big(S_J; \widehat{\theta}(\mathbf{Y}) \big| T\big) \geq I\big(T; \widehat{\theta}(\mathbf{Y}) \big| S_J\big)$. The mutual information between the testing sample and the model parameters given the sensitive attribute, $I\big(S_J; \widehat{\theta}(\mathbf{Y}) \big| T\big)$, can be explicitly computed in this setup; the details of this computation are relegated to Appendix E. Fixing the prior on the hypothesis T to a Bernoulli 1/2, we can utilize (34) to find an upper bound on the success probability of the Bayesian attacker. This is done by searching for the success rate $\mathcal{P}_{\text{Suc}}(\varphi)$ that makes the l.h.s. of (34) equal to its r.h.s. Namely, the golden section search algorithm is used to minimize the square distance between the mutual information and the KL-divergence with respect to $\mathcal{P}_{\text{Suc}}(\varphi)$.

Algorithm 1 details our simulations to estimate the success rate of the Bayesian attacker. It returns '1' when the attacker successfully predicts whether the test sample s was part of the training set or not, and '0' otherwise. In our experiments we vary n to study how the generalization gap and

Algorithm 1 Experiment 1

Input: feature vectors $\mathbf{x}$, training set size n
Draw t uniform in $\{0, 1\}$
Draw j uniform in $[n]$
$\mathbf{y} \longleftarrow \beta^T \mathbf{x} + \mathbf{W}$
if t **then**
    $s \longleftarrow y_j$
else
    $s \longleftarrow \beta^T x_j + W$
end if
$\theta \longleftarrow (\mathbf{x}\mathbf{x}^T)^{-1}\mathbf{x}\mathbf{y}^T$
return $p_{S_J J\widehat{\theta}|T}(s, j, \theta|1) > p_{S_J J\widehat{\theta}|T}(s, j, \theta|0)$ **XNOR** t

success rate of the attacker evolve as a function of the number of training samples. The dimension of the feature space is fixed to $d = 20$. For each value of n, we fix $\mathbf{x}$ and we repeat (10k times) Algorithm 1 to estimate the success rate of the attacker. The feature vectors $\mathbf{x}$ are generated i.i.d. and then fixed for each value of n. Additionally, for n, we compute the generalization gap (29), which is used to compute the lower bound (33). We also compute the Mutual Information in the l.h.s. of (34), which is used to compute the upper bound on the success probability of the attacker.

Figure 3 (**Top**) shows the success rate (SR) of the Bayesian attacker as a function of the number of samples in the training set n. Along with it is the lower bound (LB) provided by Theorem 4 and the upper bound (UP) provided by equation (34). The lower bound predicts the behavior of the SR as a function of the generalization gap. For large n (small generalization gap), the success rate and its lower bound approach 0.5, the success rate of an attacker that only uses knowledge on the prior of T. While the lower bound seems loose in this setting, it is worth noting that we compare with the best possible strategy. Nonetheless, this example shows that the bounds are not vacuous and they may serve as a framework for understanding the connection between information leakage and generalization in ML. On the other hand, the upper bound provides a strong privacy guarantee. In cases where the success rate of the Bayesian attacker cannot be explicitly computed, its upper bound is the best privacy guarantee that can be provided. Additionally, Fig. 3 (**Bottom**) shows the mutual information (l.h.s. of (34)) that is used to compute the upper bound.

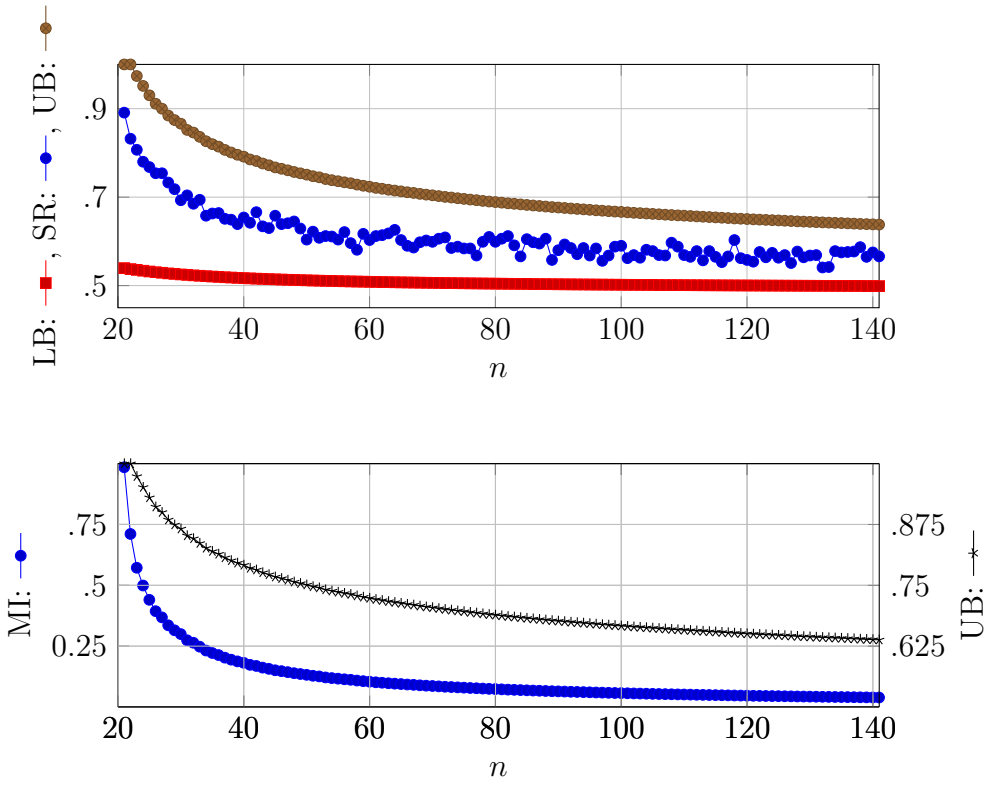

Figure 3: Dependence of success rate of the Bayesian attacker, generalization gap, and mutual information on the number of training samples n, using Gaussian data. **Top:** Success Rate (SR), Lower Bound (LB), and Upper Bound (UB). **Bottom:** Mutual Information (MI), Upper Bound (UB; axis labels on the right).

	Likelihood	Loss	Modified Entropy [15]	[4]	[10]
Attack complexity	One query	One query	One query	One query	Thousands of queries and train shadow models
Required Knowledge	Soft Probabilities	Loss value	Soft Probabilities	Training Loss	Additional Samples
PPV MNIST	0.444 ± 0.000	0.446 ± 0.000	0.444 ± 0.000	0.505	0.517
PPV CIFAR-10	0.446 ± 0.001	0.451 ± 0.001	0.449 ± 0.001	0.694	0.72
PPV Fashion-MNIST	0.445 ± 0.000	0.447 ± 0.001	0.446 ± 0.001	–	–
Recall	> 0.99	> 0.99	> 0.99	> 0.99	> 0.99

Table 1: Comparison of the likelihood attack to previous black-box MIAs from the literature. Precision (PPV; Positive Predictive Value) and recall are reported for CIFAR10, MNIST and Fashion-MNIST.

4.2. Examples on DNNs

We train DNNs on various datasets to study the interplay between generalization gap and the success rate of three different black-box MIA strategies. We compare the success rate of the different attack strategies to the lower bound provided by Theorem 2, to assess the quality of the bound. Our datasets for these experiments are: CIFAR10 [58], MNIST [59], MNIST fashion [60]. Details about datasets, the target model and the experiments are given in Appendix A.1.

The loss function used for training and for computing the generalization gap is the Mean Squared Error (MSE) loss between the one-hot encoded labels and the soft probabilities output by the network. Note that this loss function is bounded by 2. While cross-Entropy is a more common choice for loss function, it is not bounded. On the other hand MSE has a negligible effect on performance and allows us to apply Theorem 2 to lower bound the success probability of the Bayesian attacker. However, in this setup it results impossible to estimate the success probability of the Bayesian attacker, due to the high number of model parameters. To circumvent this limitation and assess the quality of the bound provided by Theorem 2, we implement the likelihood attack, detailed in Algorithm 2, the loss value attack, and the modified entropy attack from [15], and compare their success rate to the bound.

The **likelihood** attack exploits the level of confidence of a trained model in its prediction, based on the assumption that the model will make more confident predictions on samples that were part of its training set. This attack returns $\widehat{t} = 1$ if the score function $\max_{i \in |\mathcal{Y}|} f_\theta^i(x)$ is higher than some given threshold.

The **loss** value attack works by assuming that the loss value will be lower for samples present in the training set, since the training algorithm minimizes

Algorithm 2 Likelihood Attack

Input: Target model NN, threshold h
Draw t uniform in $\{0, 1\}$
if t **then**
    Draw s uniform from the training set.
else
    Draw s uniform from the test set.
end if
likelihood $\longleftarrow$ **max**(NN(s))
return likelihood $> h$ **XNOR** t

the loss on these samples. This attack compares the score $\ell\big(f_\theta(x), y\big)$ to some threshold.

On the other hand, the **modified entropy** attack simultaneously considers the correctness and the entropy of the prediction. We refer the reader to [15] for a more detailed description. The score used by this attack is given by

$$-\left(1 - f_\theta^y(x)\right) \log\left(f_\theta^y(x)\right) - \sum_{i \neq y} f_\theta^i(x) \log\left(1 - f_\theta^i(x)\right). \tag{35}$$

The training set is chosen uniformly at random from a pool of possible training samples. We vary the size n of the training set and observe how this affects the success rate of attacks, the generalization gap and consequently the lower bound derived from Theorem 2. For n, the number of samples in the training set, the success rate of the likelihood attack (LK), the loss value attack (LS), the modified entropy attack (ME), the lower bound (LB) provided by Theorem 2, as well as the accuracy on the test set (Acc) are obtained empirically in 100 runs. The results for CIFAR10, MNIST and FashionMNIST are reported in Figure 4. The lower bound predicts the behavior of the success rate of the likelihood attack as a function of the generalization gap; both approach 0.5 as the generalization gap vanishes. Note that it is possible for the success rate of the likelihood attack to go below the lower bound of the Bayesian attack. For some large n values of MNIST the average success rate of the attacker goes below 0.5. In this region the attacker cannot do better than a random guess and sometimes its success rate goes below 0.5, which implies the model can be more confident in samples outside

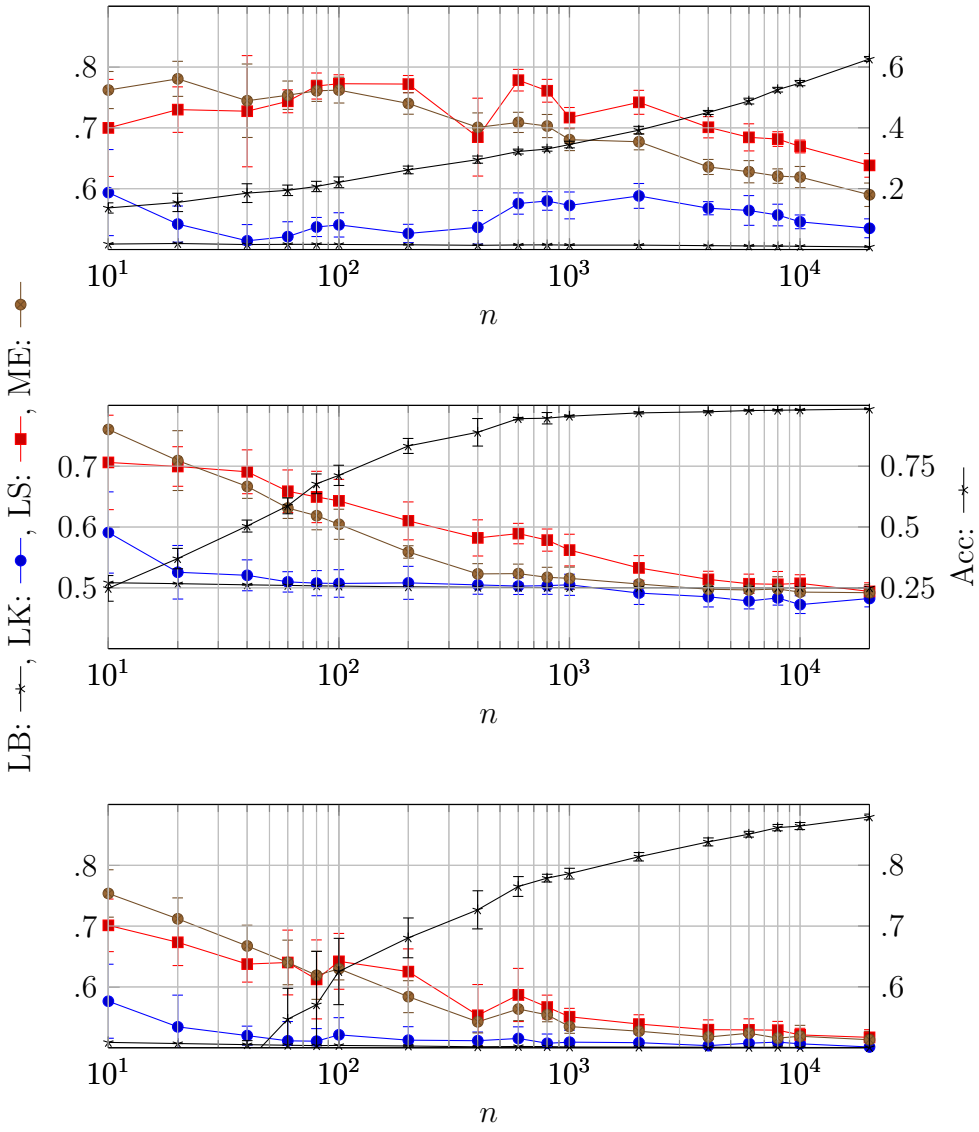

Figure 4: Success rate of the likelihood attack (LK), loss value attack (LS), modified entropy attack (ME), lower bound (LB) and accuracy (Acc; axis labels on the right) depend on the number of training samples n. **Top:** CIFAR10; **Middle:** MNIST; **Bottom:** FashionMNIST.

the training set. This is an artifact of the random sampling of the training set and the training of the model.

Table 1 compares the strategies considered in our experiments to other previous MIA strategies found in the literature. The three strategies here considered do not require access to the model parameters or additional samples, and they only need to query the model once, while the other strategies [4, 10] require extra information or significantly more computing power. The attack is performed against target models with a training set of 8000 samples, to match the setup used in [4, 10]; however, the architectures of the target models, as well as the (random) selection of training samples differ in all three setups.

4.3. Attribute Inference on PenDigits

To demonstrate the risk of information leakage from ML models, we consider attribute inference attacks against a model that classifies hand-written digits. We consider the PenDigits dataset [61], as it contains identity information about the writers, which we use as the sensitive attribute. The target model is a fully-connected network trained to classify hand-written digits. Details about the model and its training are provided, along with information about the dataset and its pre-processing, in Appendix A.2. When performing MIAs, we utilize MSE, which is bounded, as the loss for training. This allows us to apply Theorem 2.

Next, we discuss the attack strategies considered against the model. Since $\mathcal{T}$ is finite, our attack strategy consists on testing every possible value of T and choosing the most likely value according to some criteria. The *Gradient* and *Loss* strategies are inspired by similar strategies from the membership inference literature [11, 10].

Likelihood: The intuition behind this attack is that a model is more confident on samples that were part of its training. Therefore, by choosing the correct value t, the model will maximize its output for the predicted label. Note that this criteria does not care about the model making the right prediction. The side information given to the attacker are the non-sensitive attributes, $s = v$. This strategy chooses the sensitive attribute that outputs the highest score, i.e.,

$$\varphi(v, \theta) = \arg\max_{t \in \mathcal{T}} \left[\max_{i \in |\mathcal{Y}|} f_\theta^i\big((v, t)\big) \right] , \tag{36}$$

where f_θ^i is the i-th component of the output of the model parameterized by θ.

Accuracy: In contrast to the previous one, this strategy chooses the sensitive attribute that produces the *right* prediction with the highest score. This is the closest to the strategy proposed by [6]. The side information given to the attacker are the non-sensitive attributes and the label, $s = (v, y)$. Define set $\widehat{X}_{y\theta} \triangleq \{x \in \mathcal{X} : \arg\max(f_\theta(x)) = y\}$, then,

$$\varphi(v, y, \theta) = \underset{t \in \mathcal{T}:\, x \in \widehat{X}_{y\theta}}{\arg\max} \left[\max_{i \in |\mathcal{Y}|} f_\theta^i\big((v, t)\big) \right] . \tag{37}$$

Loss: This attack, based on the value of the loss, tries to minimize the loss function over samples present in the model's training set; while the next attack uses the norm of its gradient with respect to the model parameters. The side information given to the attacker is the non-sensitive attributes and the label: $s = (v, y)$. This strategy chooses the sensitive attribute that minimizes the loss, i.e.,

$$\varphi(v, y, \theta) = \underset{t \in \mathcal{T}}{\arg\min}\, \ell\left(f_\theta\big((v, t)\big), y\right) . \tag{38}$$

Gradient: Near a minimum, the norm of the gradient of the loss function with respect to its model parameters should approach 0; the attacker exploits this knowledge for the present attack strategy. While the previous attacks only make use of the output of the model or the value of its loss, the present attack makes explicit use of its parameters. The side information given to the attacker are the non-sensitive attributes and the label, $s = (v, y)$. This strategy chooses the sensitive attribute that minimizes the gradient norm, i.e.,

$$\varphi(v, y, \theta) = \underset{t \in \mathcal{T}}{\arg\min} \left\| \nabla_\theta \ell\left(f_\theta((v, t)), y\right) \right\|_2^2 . \tag{39}$$

In our experiments we perform attribute inference attacks using each of these strategies as we vary n. A detailed description of the experimental procedure is given in Appendix A.2. The success rates for each strategy are computed and reported in Figure 5 (**Top**). In this setup, a random guess would amount to a success rate of approximately 2.3%. For a small training set (100 samples), the attacker has a gain of 25% over a random guess. This decreases significantly with the size of the training set; however, even for a large training set, the attacker still has twice as much accuracy as a random guess.

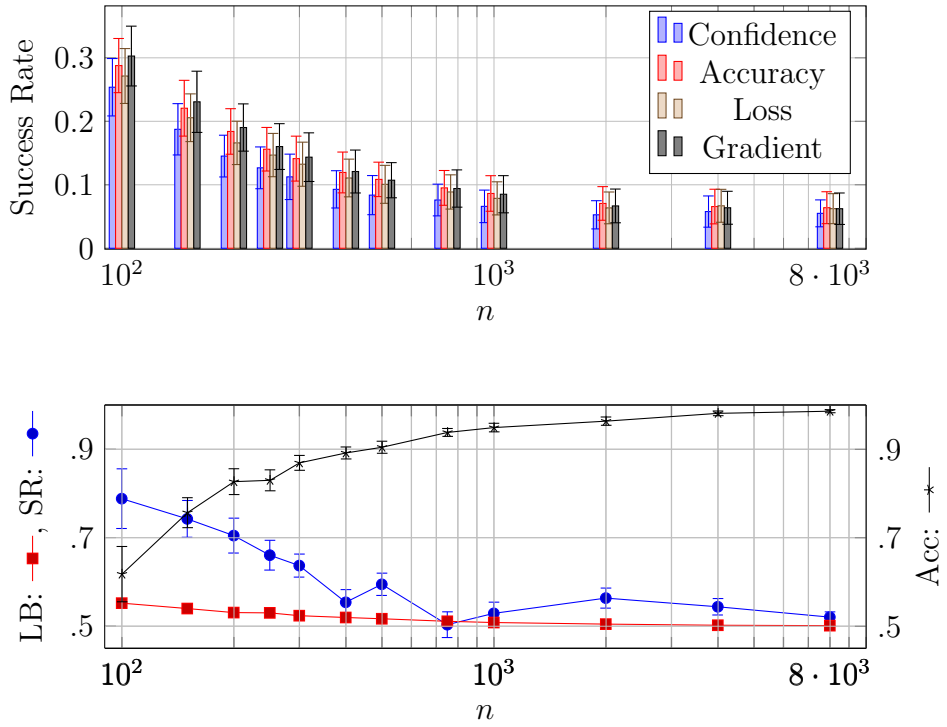

Figure 5: Attribute and Membership Inference Attacks on PenDigits for different sizes n of training sets. **Top:** Success Rate of different attribute inference attack strategies; **Bottom:** Success Rate of the likelihood attack (SR), Lower Bound (LB) and Accuracy (Acc; axis labels on the right).

Additionally, we perform MIAs against the same models. The attack strategy utilized is the Likelihood attack, given in Algorithm 2. The success rate of the attacker, lower bound on the Bayesian attacker and accuracy of the model are presented in Fig. 5 (**Bottom**). We can observe that there is a significant leakage of membership information for low values of n, while this drops almost to the value of a random guess for large values ($n = 8000$).

5. Summary and Concluding Remarks

We proposed a theoretical framework to analyze and bound information leakage from machine learning models. Our framework allows us to draw strong privacy by drawing upper bounds on the success rate of MIAs. Furthermore, we studied how much information is stored in a trained model about its training data and the implications in terms of leakage of sensitive attributes from the model parameters.

Our numerical experiments illustrate how the bounds from Section 3.2 can be used to assess the information leakage of ML models. The success rate of the attacker follows the same behavior as its lower bound, which is a function of the generalization gap. As a lower bound, it cannot guarantee that there is no attack that can perform better under the same conditions. Nevertheless, if the lower bound is above the performance of a random guess, the target model is guaranteed to leak sensitive information about its training data; thus, the generalization gap can be used to alert that a model is leaking sensitive information.

We exposed the leakage of sensitive information via attribute inference attacks, proving that models that are susceptible to membership inference can also be susceptible to other, more severe, privacy violations. We collected several attribute inference strategies and compared their effectiveness, finding that there is gain to be had by fully exploiting white-box access to the model.

The success rate of the Bayesian attacker provides a strong guarantee on the privacy of a model. However, computing the associated decision region seems computationally infeasible. In this paper we provided a synthetic example, using linear regression and Gaussian data, in which it is possible to analytically compute the involved distributions. In future work, we will explore novel tools to extend our illustrative examples to a systematic analysis of complex models. We hope that the present work serves as a common framework to compare different inference attack strategies. Furthermore, we hope that the definition of the Bayesian attacker and its connection to the

generalization gap and the information stored by the model serve as inspiration to devise novel attack strategies and defense mechanisms, when applied to specific models.

Appendix A. Experimental Details

Most of the experiments were run on a Latitude-7400 computer, with an Intel Core i7-8665U CPU @ 1.90GHz x 8 Processor. Part of the experiments were run on a server with two NVIDIA Quadro RTX 6000 GPUs and an AMD EPYC 7302 16-Core processor.

The code and instructions necessary to reproduce our experiments can be found at `https://github.com/anonymus369/Formalizing-Attribute-and-Membership-Infere`

Appendix A.1. Examples on DNNs

The number of samples in the training set, n, varies in our experiments. For fixed n, that many samples are uniformly randomly picked from a pool of training samples. A test set is also fixed to measure the accuracy of the trained model and to empirically compute the generalization gap. In the case of MNIST and MNIST fashion, the training set is picked from a pool of 60k samples. A separate pool of 10k samples is fixed as the test set. For CIFAR10, the pool of training samples is of size 50k, and the pool of test samples is of size 10k.

The target model in this setup is a Deep Neural Network with 4 convolutional layers and 3 fully connected layers. For CIFAR10 the model has a total of 439722 parameters, while for MNIST and MNIST fashion it has only 376714. The model is trained for up to 150 epochs using the Adam optimizer [62] with learning rate $5 \cdot 10^{-3}$. The batch size used for training the models is 200 (this represent the whole training set when $n \leq 200$). An early stop criteria compares the current loss over the training set to the total loss after the previous epoch, and stops training if the difference in below 10^{-3}. The number of epochs of training can change drastically depending on the size of the training set.

Regarding the likelihood attack, note that Algorithm 2 outputs 1 if the attacker infers membership correctly and 0 otherwise. The success rate of the attacker is computed by simply counting the number of times it succeeds, over the total number of trials. Experimentally, we found that a threshold of $h = 0.8$ works best across different values of n.

For each n, we repeat the following process 100 times: Draw uniformly the training set, train the model, compute the generalization gap, compute the lower bound on the success rate of the Bayesian attacker and perform the likelihood attack 10000 times. The results over different realizations of the model are averaged to produce a single value for each n.

Appendix A.2. Attribute Inference on PenDigits

The PenDigits dataset [61], was taken by asking participants to write digits from 0 to 9 on a tablet. The original data contains variable-length time series that correspond to the position of the pen on the tablet over time. We pre-process the data to make the length of the time series uniform (length 32). Since the capture rate of the tablet is fixed, we can infer the time that it took to write a digit by the length of the original series. We keep this information, along with the number of strokes that were used to write the digit.

The target model is a DNN trained to classify hand-written digits. The input to the network consist of two time series (one for each coordinate) indicating the position of the pen over time, an integer indicating the number of strokes, a float between 0 and 1 indicating the length of the original sequences and a one-hot-encoding of the identity of the writer. The latter is considered as the sensitive attribute, while the other inputs are considered non-sensitive.

The target model possesses 4 fully-connected layers and a total of 4650 parameters. The loss for training is the MSE between the soft probabilities and the one-hot-encoded labels; this is a bounded loss function, allowing us to use Theorem 2 to lower bound the success rate of the Bayesian attacker. The model is trained with Adam optimizer (learning rate $5 \cdot 10^{-3}$) for up to 2500 epochs. An early stop criteria compares the current loss over the training set to the total loss after the previous epoch, and stops training if the difference is below 10^{-4}.

In our experiments, we compute the success rate for each of the proposed attribute inference strategies as a function of the number of samples in the training set of the target model. For each value of n, we randomly uniformly select 100 different training sets drawn from a pool containing a total of 11990 samples. For each training set we train a model. Subsequently, we apply each attack criteria to 100 training samples of each trained model. The success rate of the attacker is computed by counting the amount of times the attack is successful. The reported success rate is an average over different target models. Since there are 44 different writers in the data set, a random guess would amount to a success rate of approximately 2.3%.

Regarding membership inference, we compute the success rate of the likelihood attacker (Algorithm 2), the lower bound from Theorem 2, and the accuracy of the model on the test set as a function of n. For each value of n

we train 100 different models and perform the attack 100 times. Results are averaged over different models for each value of n.

Appendix B. Proof of Proposition 1

We recall the definition of the total variation distance when applied to distributions P, Q on a set $\mathcal{X} \subseteq \mathbb{R}^d$ and Scheffé's identity [63, Lemma 2.1]

$$\|P - Q\|_{\mathrm{TV}} \triangleq \sup_{\mathcal{A} \in \mathcal{B}^d} |P(\mathcal{A}) - Q(\mathcal{A})| = \frac{1}{2} \int |p_X(\mathbf{x}) - q_X(\mathbf{x})| d\mu(\mathbf{x}) \,, \quad \text{(B.1)}$$

with respect to a base measure μ, where $\mathcal{B}^d$ denotes the class of all Borel sets on $\mathbb{R}^d$.

Proof. First of all, we prove equality for $\gamma = 1$. Let us denote the optimal decision regions with $\mathcal{T}^\star \equiv \mathcal{T}(1)$ and $\mathcal{T}^{\star c} \equiv \mathcal{T}^c(1)$ (cf. Definition 5). Let $\epsilon_0(\mathcal{T}^{\star c})$ and $\epsilon_1(\mathcal{T}^\star)$ the Type-I and Type-II errors. Then,

$$\begin{aligned}
\epsilon_1(\mathcal{T}^\star) + \epsilon_0(\mathcal{T}^{\star c}) &= \int_{\mathcal{T}^\star} p_{\widehat{\theta}(\mathbf{Z})S|T}(\theta, s|0)\, d\theta ds + \int_{\mathcal{T}^{\star c}} p_{\widehat{\theta}(\mathbf{Z})S|T}(\theta, s|1)\, d\theta ds \\
&= \int_{\mathcal{T}^\star} \min_{t \in \{0,1\}} p_{\widehat{\theta}(\mathbf{Z})S|T}(\theta, s|t)\, d\theta ds + \int_{\mathcal{T}^{\star c}} \min_{t \in \{0,1\}} p_{\widehat{\theta}(\mathbf{Z})S|T}(\theta, s|t)\, d\theta ds \\
&= \int_{\Theta \times \mathcal{S}} \min_{t \in \{0,1\}} p_{\widehat{\theta}(\mathbf{Z})S|T}(\theta, s|t)\, d\theta ds \\
&= 1 - \left\| p_{\widehat{\theta}(\mathbf{Z})S|T}(\cdot|1) - p_{\widehat{\theta}(\mathbf{Z})S|T}(\cdot|0) \right\|_{\mathrm{TV}} = 1 - \Delta, \qquad \text{(B.2)}
\end{aligned}$$

where the last identity follows by applying Scheffé's identity (B.1). From (B.2), we have for any decision region $\widehat{\mathcal{T}} \subseteq \Theta \times \mathcal{S}$,

$$\begin{aligned}
1 - \Delta &= \int_{\Theta \times \mathcal{S}} \min_{t \in \{0,1\}} p_{\widehat{\theta}(\mathbf{Z})S|T}(\theta, s|t)\, d\theta ds \\
&= \int_{\widehat{\mathcal{T}}} \min_{t \in \{0,1\}} p_{\widehat{\theta}(\mathbf{Z})S|T}(\theta, s|t)\, d\theta ds + \int_{\widehat{\mathcal{T}}^c} \min_{t \in \{0,1\}} p_{\widehat{\theta}(\mathbf{Z})S|T}(\theta, s|t)\, d\theta ds \\
&\leq \int_{\widehat{\mathcal{T}}} p_{\widehat{\theta}(\mathbf{Z})S|T}(\theta, s|0)\, d\theta ds + \int_{\widehat{\mathcal{T}}^c} p_{\widehat{\theta}(\mathbf{Z})S|T}(\theta, s|1)\, d\theta ds \\
&= \epsilon_1(\widehat{\mathcal{T}}) + \epsilon_0(\widehat{\mathcal{T}}^c). \qquad \text{(B.3)}
\end{aligned}$$

It remains to show (14), assuming that $\mathrm{P}\{T=1\} = \mathrm{P}\{T=0\} = 1/2$. Using (B.2), we have

$$\begin{aligned}\frac{1}{2}\big[1-\Delta\big] &= \frac{1}{2}\big[\epsilon_1(\mathcal{T}^\star) + \epsilon_0(\mathcal{T}^{\star c})\big] \\ &= \inf_{\varphi} \mathrm{P}\left\{\varphi(\widehat{\theta}(\mathbf{Z}), S) \neq T\right\}, \end{aligned} \tag{B.4}$$

where the last identity follows by the definition of the decision regions. □

Appendix C. Proof of Theorem 4

Proof. Using the definitions from the previous proofs, let $R \triangleq \varrho(\widehat{\theta}(\mathbf{Z}), \overline{X}, \overline{Y})$ be the square of a sub-Gaussian random variable $R_{\mathrm{SG}} \triangleq \sqrt{|R|}$ with variance proxy σ_R^2. Then, we have $\mathrm{P}\{R \geq r^2\} = \mathrm{P}\{|R_{\mathrm{SG}}| \geq r\} \leq 2e^{-\frac{r^2}{2\sigma_R^2}}$ for all $r \geq 0$, which in turn yields $\mathrm{P}\{R \geq r\} \leq 2e^{-\frac{r}{2\sigma_R^2}}$ for all $r \geq 0$. Define the random variable R_0 to have the distribution function $Q_0(r) \triangleq P\{R_0 \leq r\} \triangleq 1 - 2e^{-\frac{r}{2\sigma_R^2}}$ on its support $[r_0, \infty)$, where $r_0 = 2\sigma_R^2 \log 2$, i.e., the pdf of R_0 is $p_{R_0}(r) = \frac{1}{\sigma_R^2} e^{-\frac{r}{2\sigma_R^2}}$.

Let Q be the distribution function of R. Then, using the construction in the proof of in [55, Theorem 1.104], we can write $R = Q^{-1} \circ Q_0(R_0)$, where Q^{-1} is the left continuous inverse of Q, noting that Q_0 is continuous. The tail bound on R then implies $Q(r) = 1 - P\{|R| \geq r\} \geq Q_0(r)$, which immediately yields $Q^{-1} \circ Q_0(r) \leq r$.

Following similar steps as for Theorem 3, for $R_{\max} \geq r_0$, we get,

$$\begin{aligned}
\int_{|r|\geq R_{\max}} |r| p_R(r) dr &= \int_{Q^{-1}\circ Q_0(r)\geq R_{\max}} Q^{-1}\circ Q_0(r) p_{R_0}(r) dr \\
&= \int_{Q_0(r)\geq Q(R_{\max})} Q^{-1}\circ Q_0(r) p_{R_0}(r) dr \\
&\leq \int_{Q_0(r)\geq Q(R_{\max})} r p_{R_0}(r) dr \\
&\leq \int_{r\geq R_{\max}} r p_{R_0}(r) dr \\
&= \int_{R_{\max}}^{\infty} \frac{r}{\sigma_R^2} e^{-\frac{r}{2\sigma_R^2}} dr \\
&= -2\int_{R_{\max}}^{\infty} \frac{\partial\, r e^{-\frac{r}{2\sigma_R^2}}}{\partial r} dr + 2\int_{R_{\max}}^{\infty} e^{-\frac{r}{2\sigma_R^2}} dr \\
&= 2R_{\max} \exp\left(-\frac{R_{\max}}{2\sigma_R^2}\right)\left(1+\frac{2\sigma_R^2}{R_{\max}}\right). \qquad \text{(C.1)}
\end{aligned}$$

The rest of the proof follows identically to that of Theorem 3 and yields,

$$\begin{aligned}
\mathcal{P}_{\text{Suc}}(\varphi) \geq \max\Bigg\{ P_{\text{m}}, P_{\text{m}}\Bigg(&\frac{|\mathbb{E}[\mathcal{E}_{\text{G}}(\mathcal{A},\mathbf{Z})]|}{2R_{\max}} \\
&- \frac{1}{1-P_{\text{m}}} \exp\left(-\frac{R_{\max}}{2\sigma_R^2}\right)\left(1+\frac{2\sigma_R^2}{R_{\max}}\right) - 1\Bigg) + 1\Bigg\}. \qquad \text{(C.2)}
\end{aligned}$$

□

Appendix D. Proof of Theorem 5

Before we proceed with the proof of Theorem 5, we provide a series of definitions and preliminary results.

Appendix D.1. Basic Definitions and Change of Measure

Let us consider two probability measures P and Q on a common measurable space $(\Omega, \mathcal{F})$. Let X denote a random variable $X : \Omega \to \mathcal{X}$ and P_X,

Q_X correspond to the induced distributions. Assuming absolute continuity $P_X \ll Q_X$, the KL-divergence of Q_X with respect to P_X is defined by

$$D_{\mathrm{KL}}(P_X \| Q_X) \triangleq \mathbb{E}_{Q_X}\left[-\log\left(\frac{dP_X}{dQ_X}\right)\right]. \tag{D.1}$$

Consider a kernel (or channel) according to the law $P_{Y|X}$ that produces the random variable Y given X. Let P_Y be the induced distribution of Y when X is generated according to P_X while Q_Y is the distribution of Y when X is generated according to Q_X. Then, by the data-processing inequality for KL-divergence [64, Theorem 2.2 6.], we have

$$D_{\mathrm{KL}}(P_X \| Q_X) \geq D_{\mathrm{KL}}(P_Y \| Q_Y). \tag{D.2}$$

Equality holds if and only if $P_{X|Y} = Q_{X|Y}$, where $P_{X|Y}P_Y = P_{Y|X}P_X$ and $Q_{X|Y}Q_Y = P_{Y|X}P_X$. A simple application of this inequality leads to the following result.

Lemma 1 (Data-processing reduces KL-divergence [65])**.** *For any measurable set $\mathcal{B} \in \mathcal{F}(\mathcal{X})$, inequality (D.2) applied to the degenerate channel based on the indicator function $Y = \mathbb{1}[X \in \mathcal{B}]$ implies:*

$$D_{KL}(P_X \| Q_X) \geq d_{KL}(p_{\mathcal{B}} \| q_{\mathcal{B}}) = d_{KL}(1 - p_{\mathcal{B}} \| 1 - q_{\mathcal{B}}), \tag{D.3}$$

where $d_{KL}(\cdot\|\cdot)$ denotes the binary KL-divergence with parameters $p_{\mathcal{B}} = P_X(\mathcal{B})$ and $q_{\mathcal{B}} = Q_X(\mathcal{B})$. Note that if $t \in [0,1]$ and $M > 1$, then

$$\log_2(M) - d_{KL}(t \| 1 - 1/M) = t \log_2(M-1) + H_2(t)\ , \tag{D.4}$$

where $H_2(t) \triangleq -t\log_2 t - (1-t)\log_2(1-t)$ is the binary entropy function. Equality holds in (D.3) if and only if $P_{X|X\in\mathcal{B}} = Q_{X|X\in\mathcal{B}}$ and $P_{X|X\notin\mathcal{B}} = Q_{X|X\notin\mathcal{B}}$.

The proof of this lemma is rather straightforward from basic properties and will be omitted. We next revisit a well-known result to obtain bounds for the probability of an arbitrary event $\mathcal{B} \in \mathcal{F}(\mathcal{X})$.

Lemma 2 (Change of measure [66])**.** *Let the distributions P_X and Q_X be induced by the random variable X as described. Then,*

$$\sup_{\mathcal{B}\in\mathcal{F}(\mathcal{X})} P_X(\mathcal{B}) \log_2(1/Q_X(\mathcal{B})) \leq D_{KL}(P_X \| Q_X) + 1, \tag{D.5}$$

where the supremum is taken over all measurable sets $\mathcal{B} \in \mathcal{F}(\mathcal{X})$.

Proof. For any set $\mathcal{B} \in \mathcal{F}(\mathcal{X})$, we have by Lemma 1, that

$$\begin{aligned} D_{\mathrm{KL}}(P_X \| Q_X) &\geq d_{\mathrm{KL}}(p_{\mathcal{B}} \| q_{\mathcal{B}}) \\ &= P_X(\mathcal{B}) \log \frac{P_X(\mathcal{B})}{Q_X(\mathcal{B})} + P_X(\mathcal{B}^c) \log \frac{P_X(\mathcal{B}^c)}{Q_X(\mathcal{B}^c)} \\ &= P_X(\mathcal{B}) \log_2 \frac{1}{Q_X(\mathcal{B})} + P_X(\mathcal{B}^c) \log_2 \frac{1}{Q_X(\mathcal{B}^c)} - H_2(p_{\mathcal{B}}) \\ &\geq P_X(\mathcal{B}) \log_2 \left(\frac{1}{Q_X(\mathcal{B})} \right) - 1. \end{aligned} \tag{D.6}$$

The final inequality (D.5) follows by taking the supremum over all measurable sets $\mathcal{B} \in \mathcal{F}(\mathcal{X})$ in (D.6). □

Appendix D.2. Cramér-Chernoff Method

We recall a distribution-dependent deviation bound based on the optimization of the Markov inequality which is known as Cramér-Chernoff method.

Let Z be a real-valued random variable and define its log-moment-generating function as

$$\psi_Z(\lambda) = \log \mathbb{E}\left[\exp \lambda Z\right], \quad \lambda \geq 0. \tag{D.7}$$

For $\lambda \geq 0$, the Markov inequality implies:

$$\begin{aligned} \mathrm{P}(Z \geq t) &\leq \mathrm{P}(e^{\lambda Z} \geq e^{\lambda t}) \\ &\leq e^{-\lambda t} \mathbb{E}[e^{\lambda Z}] \\ &= \exp\big[-\lambda t + \psi_Z(\lambda) \big]. \end{aligned} \tag{D.8}$$

As (D.8) holds for any $\lambda \geq 0$, we immediately obtain $\mathrm{P}(Z \geq t) \leq \exp\left[-\psi_Z^*(t)\right]$ for $t \geq \mathbb{E}[Z]$, where

$$\psi_Z^*(t) = \sup_{\lambda \in \mathbb{R}} \left\{ \lambda t - \log \mathbb{E}\left[e^{\lambda Z}\right] \right\}. \tag{D.9}$$

This expression is known as the Fenchel-Legendre dual function of $\psi_Z(\lambda)$ and it equals $\psi_Z^*(t) = \sup\left\{\lambda t - \psi_Z(\lambda) \,:\, \lambda \geq 0\right\}$ whenever $t \geq \mathbb{E}[Z]$.

And therefore, for $t \geq \mathbb{E}[Z]$,

$$\mathrm{P}(Z \geq t) \leq \exp\left[-\psi_Z^*(t)\right]. \tag{D.10}$$

We will need the following properties:

- If $Z = X_1 + \cdots + X_n$ with $\{X_i\}_{i=1}^n$ being i.i.d. copies of X, then

$$\psi_Z^*(t) = n\psi_X^*\left(\frac{t}{n}\right); \tag{D.11}$$

- For any random variable Z,

$$\psi_{Z/n}^*(t) = \psi_Z^*(nt). \tag{D.12}$$

An immediate consequence of these properties is that the random variable

$$Z = \mathbb{E}[X] - \frac{1}{n}\sum_{i=1}^n X_i,$$

with $\{X_i\}_{i=1}^n$ i.i.d. copies of X, satisfies

$$\mathrm{P}(Z \geq t) \leq \exp\left[-n\psi_{\mathbb{E}[X]-X}^*(t)\right], \ \forall t \geq 0. \tag{D.13}$$

Appendix D.3. Proof

Using these preliminary results, we are now ready to prove Theorem 5. The proof requires three steps which are described below.

Information loss. First, we observe that $(T, S) \leftrightarrow \mathbf{Z} \leftrightarrow \widehat{\theta}(\mathbf{Z})$ and thus $T \leftrightarrow (\mathbf{Z}, S) \leftrightarrow \widehat{\theta}(\mathbf{Z})$ form Markov chains since $\widehat{\theta}$ is a stochastic function of $\mathbf{Z}$. As a consequence of the data-processing inequality [65, Theorem 2.8.1], we obtain (28) from

$$I(T; \widehat{\theta}(\mathbf{Z})|S) \leq I(\mathbf{Z}; \widehat{\theta}(\mathbf{Z})|S) = I(\mathbf{Z}; \widehat{\theta}(\mathbf{Z})) - I(S; \widehat{\theta}(\mathbf{Z})). \tag{D.14}$$

Interestingly, we will show that $I(T; \widehat{\theta}(\mathbf{Z})|S)$ bounds the accuracy of the membership (sensitive attribute) inference while $I(\mathbf{Z}; \widehat{\theta}(\mathbf{Z}))$ bounds the generalization gap of the hypothesis associated with $\widehat{\theta}$.

Generalization gap. The proof of the bound on the generalization gap (26) easily follows from application of well-known results. For $\epsilon \geq 0$ let us define the region

$$\mathcal{B} \triangleq \left\{(\theta, \mathbf{z}) \in \Theta \times \mathcal{Z}^n : \mathbb{E}[\varrho(\theta, Z)] - \frac{1}{n}\sum_{i=1}^n \varrho(\theta, z_i) \geq \epsilon\right\}. \tag{D.15}$$

By the definition of the generalization gap (Definition 1), we have $\mathcal{G}_{\mathrm{G}}(\mathcal{A}, \mathbf{Z}) \geq \epsilon$ if and only if $\big(\widehat{\theta}(\mathbf{Z}), \mathbf{Z}\big) \in \mathcal{B}$. We define the associated fibers $\mathcal{B}_{(\theta)} \triangleq \{\mathbf{z} \in \mathcal{Z}^n : (\theta, \mathbf{z}) \in \mathcal{B}\}$ for $\theta \in \Theta$. First, we apply the Cramér-Chernoff method (Appendix D.2) to the random variable

$$R_\theta \triangleq \mathbb{E}[\varrho(\theta, (X, Y))] - \varrho(\theta, (X, Y)) \tag{D.16}$$

$\mathcal{B}_{(\theta)}$ with respect to the data probability measure $P_{\mathbf{Z}}$, where (D.10)–(D.12) then yield

$$\mathrm{P}\left(\mathbf{Z} \in \mathcal{B}_{(\theta)}\right) \leq \exp\big[-n\psi^*_{R_\theta}(\epsilon)\big], \tag{D.17}$$

where $\psi^*_{R_\theta}$ is the Fenchel-Legendre dual function of the real-valued random variable: . Then, it follows that,

$$\operatorname*{ess\,sup}_{\theta \sim P_{\widehat{\theta}(\mathbf{Z})}} \mathrm{P}\left(\mathbf{Z} \in \mathcal{B}_{(\theta)}\right) \leq \exp\Big[-n \operatorname*{ess\,inf}_{\theta \sim P_{\widehat{\theta}(\mathbf{Z})}} \psi^*_{R_\theta}(\epsilon)\Big]. \tag{D.18}$$

We can now use Lemma 2, rearranging terms and taking the expectation w.r.t. $P_{\widehat{\theta}(\mathbf{Z})}$, we have that

$$\begin{aligned}
\mathrm{P}\left(\mathcal{G}_{\mathrm{G}}(\mathcal{A}, \mathbf{Z}) \geq \epsilon\right) &= \mathrm{P}\left(\big(\widehat{\theta}(\mathbf{Z}), \mathbf{Z}\big) \in \mathcal{B}\right) \\
&\leq \frac{I(\mathbf{Z}; \widehat{\theta}(\mathbf{Z})) + 1}{-\log(P_{\widehat{\theta}(\mathbf{Z})} \times P_{\mathbf{Z}}(\mathcal{B}))} \qquad \text{(D.19)} \\
&\leq \frac{I(\mathbf{Z}; \widehat{\theta}(\mathbf{Z})) + 1}{-\log\big(\int P_{\mathbf{Z}}(\mathcal{B}_{(\theta)})\, dP_{\widehat{\theta}(\mathbf{Z})}(\theta)\big)} \\
&\leq \frac{I(\mathbf{Z}; \widehat{\theta}(\mathbf{Z})) + 1}{-\log\Big[\operatorname*{ess\,sup}_{\theta \sim P_{\widehat{\theta}(\mathbf{Z})}} \mathbf{P}\big(\mathbf{Z} \in \mathcal{B}_{(\theta)}\big)\Big]} \\
&\leq \frac{I(\mathbf{Z}; \widehat{\theta}(\mathbf{Z})) + 1}{n\Big[\operatorname*{ess\,inf}_{\theta \sim P_{\widehat{\theta}(\mathbf{Z})}} \psi^*_{R_\theta}(\epsilon)\Big]}, \qquad \text{(D.20)}
\end{aligned}$$

where inequality (D.19) follows from (D.5) and (D.20) follows from (D.18).

Attribute inference. Let $\varphi^\star$ be the attack strategy given in (8) with $\mathcal{P}_{\mathrm{Suc}}(\varphi^\star) = \mathrm{P}\{\widehat{T} = T\} \geq 1/2$, where $\widehat{T}$ denotes the random variable $\widehat{T} \triangleq$

$\varphi^\star(\widehat{\theta}(\mathbf{Z}), S)$. Note that $\widehat{T}$ is independent of T given $(\widehat{\theta}(\mathbf{Z}), S)$. We will show that,

$$I(T;\widehat{\theta}(\mathbf{Z})|S) \geq d_{\mathrm{KL}}\left(\mathcal{P}_{\mathrm{Suc}}(\varphi^\star)\,\middle\|\,\mathbb{E}\Big[\min_{t\in\mathcal{T}} P_{T|S}(t|S)\Big]\right), \tag{D.21}$$

where,

$$d_{\mathrm{KL}}(p\|q) \triangleq p\log_2\frac{p}{q} + (1-p)\log_2\frac{(1-p)}{(1-q)}. \tag{D.22}$$

To this end, denote by $D_{\mathrm{KL}}(\cdot\|\cdot)$ the KL-divergence between two distributions and observe that, by Lemma 1,

$$D_{\mathrm{KL}}\left(P_{T|\widehat{\theta}(\mathbf{Z})S}(\cdot|\theta,s)\|P_{T|S}(\cdot|s)\right) \geq d_{\mathrm{KL}}\left(P_{T|\widehat{\theta}(\mathbf{Z})S}\big(t|\theta,s\big)\,\big\|\,P_{T|S}\big(t|s\big)\right), \tag{D.23}$$

where $t = t(s,\theta)$ may be any function of $(s,\theta) \in \mathcal{S}\times\Theta$. By taking the expectation over $\theta, s \sim p_{S,\widehat{\theta}(\mathbf{Z})}$, we obtain

$$I(T;\widehat{\theta}(\mathbf{Z})|S) \geq \mathbb{E}\left[d_{\mathrm{KL}}\left(P_{T|\widehat{\theta}(\mathbf{Z})S}\big(t|\widehat{\theta}(\mathbf{Z}),S\big)\,\big\|\,P_{T|S}\big(t|S\big)\right)\right]. \tag{D.24}$$

We choose a mapping $t^\star_{(s,\theta)}$ that satisfies,

$$\mathbb{E}\Big[P_{T|\widehat{\theta}(\mathbf{Z})S}(t^\star|\widehat{\theta}(\mathbf{Z}),S)\Big] = \mathbb{E}\Big[\max_{t\in\mathcal{T}} P_{T|\widehat{\theta}(\mathbf{Z})S}(t|\widehat{\theta}(\mathbf{Z}),S)\Big] = \mathcal{P}_{\mathrm{Suc}}(\varphi^\star). \tag{D.25}$$

It is straightforward to verify that,

$$\mathcal{P}_{\mathrm{Suc}}(\varphi^\star) \geq \mathbb{E}\Big[\max_{t\in\mathcal{T}} P_{T|S}(t|S)\Big]. \tag{D.26}$$

Then, by convexity of the function $(p,q)\mapsto d_{\mathrm{KL}}(p\|q)$, we can continue from (D.24) to show,

$$\begin{aligned} I(T;\widehat{\theta}(\mathbf{Z})|S) &\geq \mathbb{E}\left[d_{\mathrm{KL}}\Big(P_{T|\widehat{\theta}(\mathbf{Z})S}(t^\star|\widehat{\theta}(\mathbf{Z}),S)\Big\|\,P_{T|S}(t^\star|S)\Big)\right] \\ &\geq d_{\mathrm{KL}}\left(\mathcal{P}_{\mathrm{Suc}}(\varphi^\star)\,\big\|\,\mathbb{E}\big[P_{T|S}\big(t^\star|S\big)\big]\right) \\ &\geq d_{\mathrm{KL}}\left(\mathcal{P}_{\mathrm{Suc}}(\varphi^\star)\,\big\|\,\mathbb{E}\big[\max_{t\in\mathcal{T}} P_{T|S}(t|S)\big]\right), \end{aligned} \tag{D.27}$$

where the last inequality (D.27) follows by using (D.26) and noticing that the function $q\mapsto d_{\mathrm{KL}}(p\|q)$ is non-increasing for $q\in[0,p]$.

Finally, notice that we can apply the bound,

$$d_{\mathrm{KL}}(p\|q) \geq \max\left\{2(p-q)^2, -p\log_2(q) - 1\right\}, \tag{D.28}$$

with $p \geq q$. □

Appendix E. Gaussian Data and Linear Regression

Recall the following notation: $\mathbf{x}$ is the $[d \times n]$ matrix given by $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, while $\mathbf{y} = (y_1, y_2, \ldots, y_n)$ and $\mathbf{W} = (W_1, W_2, \ldots, W_n)$ are $[1 \times n]$ vectors. Let each copy of noise W be normal i.i.d.; $W \sim \mathcal{N}(\cdot; 0, \sigma^2)$. Since Y_i is linear in W_i, Y_i is also normal distributed, $Y_i \sim \mathcal{N}(\,\cdot\,; \beta^T x_i, \sigma^2)$. Since model parameters are linear in the training set $\mathbf{Y}$, their pdf is a multivariate Gaussian, $\widehat{\theta}(\mathbf{Y}) \sim Q(\,\cdot\,) \triangleq \mathcal{N}(\,\cdot\,; \beta, \sigma^2 \overline{x}^{-1})$, where $\overline{x} \triangleq \mathbf{x}\mathbf{x}^T$. Furthermore, fixing the $j-$th sample in the training set to s, we have $\widehat{\theta}(\mathbf{Y})$ distributed as $Q_j(\,\cdot\,|s) \triangleq \mathcal{N}\big(\,\cdot\,; \beta + \overline{x}^{-1} x_j (s - x_j^T \beta), \sigma^2 \overline{x}^{-1} (\mathbb{I}^{d\times d} - x_j x_j^T \overline{x}^{-1})\big)$.

Consider a MIA against this model. The attacker possesses side information (S_J, J), that is, a test sample and its corresponding index. Recall our definition $S_J = T(Y_J) + (1-T)(Y'_J)$, where J is a random index in $[n]$. When $T = 0$, $S = Y'_J$, independent of the training set; hence,

$$\begin{aligned} p_{S_J J \widehat{\theta}|T}(s, j, \theta|0) &= \frac{1}{n} p_{S_J \widehat{\theta}(\mathbf{Y})|T,J}(s, \theta|0, j) \\ &= \frac{1}{n} p_{\widehat{\theta}(\mathbf{Y})|T}(\theta|0) p_{S_J|T,J}(s|0, j) \\ &= \frac{1}{n} Q(\theta) p_{Y_j}(s)\ , \end{aligned} \tag{E.1}$$

On the other hand, when $T = 1$, $S = Y_J$ is the $J-$th component of the training set $\mathbf{Y}$; therefore,

$$\begin{aligned} p_{S_J J \widehat{\theta}|T}(s, j, \theta|1) &= \frac{1}{n} p_{S_J \widehat{\theta}(\mathbf{Y})|T,J}(s, \theta|1, j) \\ &= \frac{1}{n} p_{\widehat{\theta}(\mathbf{Y})|T J S_J}(\theta|1, j, s) p_{S_J|T,J}(s|1, j) \\ &= \frac{1}{n} Q_j(\theta|s) p_{Y_j}(s)\ . \end{aligned} \tag{E.2}$$

Note that $Q(\,\cdot\,)$ and $Q_i(\,\cdot\,|s)$ differ only by their mean and variance. The second pdf has shifted mean and reduced variance. The reduced variance is to be expected, since fixing one of the samples in the training set should reduce randomness. Note that if the dimension of the space of features is equal to the amount of samples (i.e., $d \geq n$) an attacker having access to the feature vectors in the training set $\mathbf{x}$ can solve a system of equations to obtain $\mathbf{y}$.

In the following, we derive a theoretical lower bound for (30). Define $R \triangleq x_J^T (\mathbf{x}\mathbf{x}^T)^{-1} \mathbf{x}\mathbf{Y}^T - S_J$. Fixing J and T, R is a linear combination of

Gaussian r.v.s, and thus R is a Gaussian random variable. Regardless of T and J, $\mathbb{E}[R] = 0$. If $T = 0$; then $S_J = Y'_J$, independent of $\mathbf{Y}$,

$$\mathrm{Var}[R|T=0] = \sigma^2 + \frac{d\sigma^2}{n} \ . \tag{E.3}$$

If $T = 1$, then $S_J = Y_J$ is the $J-$th component of $\mathbf{Y}$; consequently,

$$\mathrm{Var}[R|T=1] = \sigma^2 - \frac{d\sigma^2}{n} \ . \tag{E.4}$$

In total,

$$\sigma_R^2 \triangleq \mathrm{Var}[R] = \sigma^2 \ . \tag{E.5}$$

Since R is a Gaussian random variable, the squared error, defined by $R^2 \triangleq \left(x^T(\mathbf{x}\mathbf{x}^T)^{-1}\mathbf{x}\mathbf{Y} - S_J\right)^2$, is exponentially tail-bounded[4]; hence, we can apply Theorem 4 to get a theoretical lower bound on the success probability of the Bayesian MIA. Assume that T is Bernoulli 1/2 distributed; thus,

$$\begin{aligned}\mathcal{P}_{\mathrm{Suc}}(\varphi^\star) &\geq \frac{1}{2} + \frac{|\mathbb{E}[\mathcal{E}_{\mathrm{G}}(\mathcal{A}, \mathbf{Z})]|}{4R_{\max}} \\ &\quad - \exp\left(-\frac{R_{\max}}{2\sigma_R^2}\right)\left(1 + \frac{2\sigma_R^2}{R_{\max}}\right) \ . \\ &= \frac{1}{2} + \frac{d}{2n}\frac{\sigma^2}{R_{\max}} - \exp\left(-\frac{R_{\max}}{2\sigma^2}\right)\left(1 + \frac{2\sigma^2}{R_{\max}}\right) \ , \end{aligned} \tag{E.6}$$

where we use (29).

The Mutual information between a test sample S_J and the model parameters $\widehat{\theta}(\mathbf{Y})$ given the sensitive attribute T is,

[4]See proof of Theorem 4 in Appendix C.

$$
\begin{aligned}
I(S_J;\widehat{\theta}(\mathbf{Y})|T) &= \sum_{t\in\{0,1\}} I(S_J;\widehat{\theta}(\mathbf{Y})|T=t) \\
&= \mathrm{P}\{T=1\} I(S_J;\widehat{\theta}(\mathbf{Y})|T=1) \\
&= \mathrm{P}\{T=1\}\frac{1}{n}\sum_{j=1}^{n}\int Q_j(\theta|s)P_{Y_j}(s)\log\left[\frac{Q_j(\theta|s)P_{Y_j}(s)}{Q(\theta)P_{Y_j}(s)}\right]d\theta ds \\
&= \mathrm{P}\{T=1\}\frac{1}{n}\sum_{j=1}^{n}\int Q_j(\theta|s)P_{Y_j}(s)\log\left[\frac{Q_j(\theta|s)}{Q(\theta)}\right]d\theta ds \\
&= \mathrm{P}\{T=1\}\frac{1}{n}\sum_{j=1}^{n}\mathbb{E}_{S\sim P_{Y_j}}\left[D_{\mathrm{KL}(Q_j(\theta|s)|Q(\theta))}\right] \\
&= \mathrm{P}\{T=1\}\frac{1}{2n}\sum_{j=1}^{n}\mathbb{E}_{S\sim P_{Y_j}}\Big[\mathrm{Tr}\big(\Sigma^{-1}\Sigma_j\big)-\log\left(\frac{|\Sigma_j|}{|\Sigma|}\right)-d \\
&\qquad +(\mu_j(S)-\beta)^T\Sigma^{-1}(\mu_j(S)-\beta)\Big] \\
&= \mathrm{P}\{T=1\}\frac{1}{2n}\sum_{j=1}^{n}\left(\mathrm{Tr}\big(\Sigma^{-1}\Sigma_j\big)-\log\left(\frac{|\Sigma_j|}{|\Sigma|}\right)-d+x_j^T\overline{x}^{-1}x_j\right) \\
&= \mathrm{P}\{T=1\}\frac{1}{2n}\sum_{j=1}^{n}\log\left(\frac{|\Sigma|}{|\Sigma_j|}\right)\ , \qquad \text{(E.7)}
\end{aligned}
$$

with $\mu_j(s) \triangleq \beta + \overline{x}^{-1}x_j(s-x_j^T\beta)$, $\Sigma_j \triangleq \sigma^2\overline{x}^{-1}(\mathbb{I}^{d\times d}-x_jx_j^T\overline{x}^{-1})\big)$ and, $\Sigma=\overline{x}^{-1}\sigma^2$. Using the upper bound $I(S_J;\widehat{\theta}(\mathbf{Y})|T)\geq I(T;\widehat{\theta}(\mathbf{Y})|S_J)$ in combination with (25), we can estimate an upper bound on the probability of success of the Bayesian attacker.

Proof of (29). Recall the definition of the generalization gap, substituting the MSE and the model into the definition, we obtain,

$$
\begin{aligned}
\mathbb{E}\left[\mathcal{E}_{\mathrm{G}}(\mathcal{A},\mathbf{Y})\right] &= \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\ell(f_{\widehat{\theta}(\mathbf{Y})}(x_i),Y_i')-\frac{1}{n}\sum_{i=1}^{n}\ell(f_{\widehat{\theta}(\mathbf{Y})}(x_i),Y_i)\right] \\
&= \frac{1}{n}\mathbb{E}\left[\left\|\widehat{\theta}(\mathbf{Y})^T\mathbf{x}-\mathbf{Y}'\right\|^2-\left\|\widehat{\theta}(\mathbf{Y})^T\mathbf{x}-\mathbf{Y}\right\|^2\right]\ , \qquad \text{(E.8)}
\end{aligned}
$$

Let $\overline{x} \triangleq \mathbf{x}\mathbf{x}^T$, then,

$$\begin{aligned}\mathbb{E}\left[\left\|\widehat{\theta}(\mathbf{Y})^T\mathbf{x} - \mathbf{Y}'\right\|^2\right] &= \mathbb{E}\left[\left(\mathbf{Y}\mathbf{x}^T\overline{x}^{-1}\overline{xx}^{-1}\mathbf{x}\mathbf{Y}^T - 2\mathbf{Y}'\mathbf{x}^T\overline{x}^{-1}\mathbf{x}\mathbf{Y}^T + \|\mathbf{Y}'\|^2\right)\right] \\ &= \mathbb{E}\left[\mathbf{W}\mathbf{x}^T\overline{x}^{-1}\mathbf{x}\mathbf{W}^T - \beta^T\overline{x}\beta - 2\mathbf{W}'x^T\overline{x}^{-1}\mathbf{x}\mathbf{W}^T + \|\mathbf{Y}'\|^2\right] \\ &= \mathbb{E}\left[\mathbf{W}\mathbf{x}^T\overline{x}^{-1}\mathbf{x}\mathbf{W}^T - \beta^T\overline{x}\beta + \|\mathbf{Y}'\|^2\right] , \qquad \text{(E.9)}\end{aligned}$$

Note that $\mathbb{E}\left[2\mathbf{W}'x^T\overline{x}^{-1}\mathbf{x}\mathbf{W}^T\right] = 0$; since $\mathbb{E}[W] = 0$ and $\mathbf{W}'$ is independent from $\mathbf{W}$. On the other hand,

$$\begin{aligned}\mathbb{E}\left[\left\|\widehat{\theta}(\mathbf{Y})^T\mathbf{x} - \mathbf{Y}\right\|^2\right] &= \mathbb{E}\left[\left(\|\mathbf{Y}\|^2 - \mathbf{Y}\mathbf{x}^T\overline{x}^{-1}\mathbf{x}\mathbf{Y}^T\right)\right] \\ &= \mathbb{E}\left[\left(\|\mathbf{Y}\|^2 - \beta^T\overline{x}\beta - \mathbf{W}\mathbf{x}^T\overline{x}^{-1}\mathbf{x}\mathbf{W}^T\right)\right] \qquad \text{(E.10)}\end{aligned}$$

Note that $\mathbb{E}[\|\mathbf{Y}\|^2] = \mathbb{E}[\mathbf{Y}'^2]$, since $\mathbf{Y}$ and $\mathbf{Y}'$ are i.i.d. copies of the same random vector. Hence,

$$\mathbb{E}\left|\mathcal{E}_{\mathrm{G}}(\mathcal{A}, \mathbf{Y})\right| = \frac{2}{n}\mathbb{E}\left[\mathbf{W}\mathbf{x}^T\overline{x}^{-1}\mathbf{x}\mathbf{W}^T\right] , \qquad \text{(E.11)}$$

Taking the trace of the remaining term in the expectation,

$$\begin{aligned}\frac{2}{n}\mathbb{E}\left[\mathbf{W}\mathbf{x}^T\overline{x}^{-1}\mathbf{x}\mathbf{W}^T\right] &= \frac{2}{n}\mathbb{E}\left[\mathrm{Tr}(\mathbf{W}\mathbf{x}^T\overline{x}^{-1}\mathbf{x}\mathbf{W}^T)\right] \\ &= \frac{2}{n}\mathbb{E}\left[\mathrm{Tr}(\mathbf{x}^T\overline{x}^{-1}\mathbf{x}\mathbf{W}^T\mathbf{W})\right] \\ &= \frac{2}{n}\mathrm{Tr}(\mathbf{x}^T\overline{x}^{-1}\mathbf{x}\,\mathbb{E}\left[\mathbf{W}^T\mathbf{W}\right]) \\ &= \frac{2}{n}\mathrm{Tr}(\sigma^2\mathbf{x}^T\overline{x}^{-1}\mathbf{x}) \\ &= \frac{2}{n}\mathrm{Tr}(\sigma^2\mathbb{I}^{d\times d}) = \frac{2d\sigma^2}{n} . \qquad \text{(E.12)}\end{aligned}$$

which gives the desired result. □

Acknowledgment

This research was supported by DATAIA "Programme d'Investissement d'Avenir" (ANR-17-CONV-0003) and by the ERC project Hypatia under the European Union's Horizon 2020 research and innovation program, grant agreement No. 835294.

References

[1] A. S. Lundervold, A. Lundervold, An overview of deep learning in medical imaging focusing on
Zeitschrift für Medizinische Physik 29 (2) (2019) 102 – 127, special Issue: Deep Learning in Medical Physics. doi:https://doi.org/10.1016/j.zemedi.2018.11.002.
URL http://www.sciencedirect.com/science/article/pii/S0939388918301181

[2] J. L. Katzman, U. Shaham, A. Cloninger, J. Bates, T. Jiang, Y. Kluger,
Deepsurv: personalized treatment recommender system using a cox proportional hazards dee
BMC Medical Research Methodology 18 (1) (Feb 2018). doi:10.1186/s12874-018-0482-1.
URL http://dx.doi.org/10.1186/s12874-018-0482-1

[3] E. D. Cristofaro, An overview of privacy in machine learning, CoRR abs/2005.08679 (2020). arXiv:2005.08679.
URL https://arxiv.org/abs/2005.08679

[4] S. Yeom, I. Giacomelli, M. Fredrikson, S. Jha, Privacy risk in machine learning: Analyzing the connection to overfitting, in: 2018 IEEE 31st Computer Security Foundations Symposium (CSF), IEEE Computer Society, Los Alamitos, CA, USA, 2018, pp. 268–282. doi:10.1109/CSF.2018.00027.
URL https://doi.ieeecomputersociety.org/10.1109/CSF.2018.00027

[5] A. Sablayrolles, M. Douze, C. Schmid, Y. Ollivier, H. Jegou, White-box vs black-box: Bayes optimal strategies for membership inference, in: K. Chaudhuri, R. Salakhutdinov (Eds.), ICML, Vol. 97 of Proc. of Machine Learning Research, PMLR, 2019, pp. 5558–5567.

[6] M. Fredrikson, E. Lantz, S. Jha, S. Lin, D. Page, T. Ristenpart, Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing, in: 23rd USENIX Security Symposium (USENIX Security 14), USENIX Association, San Diego, CA, 2014, pp. 17–32.

[7] N. Carlini, C. Liu, Ú. Erlingsson, J. Kos, D. Song, The secret sharer: Evaluating and testing unintended memorization in neural networks, in: 28th USENIX Security Symposium, USENIX Association, Santa Clara, CA, 2019, pp. 267–284.

[8] Y. Long, V. Bindschaedler, L. Wang, D. Bu, X. Wang, H. Tang, C. A. Gunter, K. Chen, Understanding membership inferences on well-generalized learning models, CoRR abs/1802.04889 (2018). arXiv:1802.04889.
URL http://arxiv.org/abs/1802.04889

[9] J. Tan, B. Mason, H. Javadi, R. Baraniuk, Parameters or privacy: A provable tradeoff between overparameterization and membership in
in: A. H. Oh, A. Agarwal, D. Belgrave, K. Cho (Eds.), Advances in Neural Information Processing Systems, 2022.
URL https://openreview.net/forum?id=7nypt7cjNL

[10] R. Shokri, M. Stronati, C. Song, V. Shmatikov, Membership inference attacks against machine learning models, 2017 IEEE Symposium on Security and Privacy (SP) (2017) 3–18.

[11] M. Nasr, R. Shokri, A. Houmansadr, Comprehensive privacy analysis of deep learning: Passive and active white-box inference atta
2019 IEEE Symposium on Security and Privacy (SP) (May 2019). doi:10.1109/sp.2019.00065.
URL http://dx.doi.org/10.1109/SP.2019.00065

[12] S. Truex, L. Liu, M. E. Gursoy, L. Yu, W. Wei, Demystifying membership inference attacks in machine learning as a service, IEEE Transactions on Services Computing (2019) 1–1doi:10.1109/TSC.2019.2897554.

[13] B. Jayaraman, L. Wang, D. E. Evans, Q. Gu, Revisiting membership inference under realistic assumptions, Proceedings on Privacy Enhancing Technologies 2021 (2021) 348 – 368.

[14] S. Rezaei, X. Liu, On the difficulty of membership inference attacks, in: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society, Los Alamitos, CA, USA, 2021, pp. 7888–7896. doi:10.1109/CVPR46437.2021.00780.
URL https://doi.ieeecomputersociety.org/10.1109/CVPR46437.2021.00780

[15] L. Song, P. Mittal, Systematic evaluation of privacy risks of machine learning models, in: USENIX Security Symposium, 2021, pp. 2615–2632.

[16] N. Carlini, S. Chien, M. Nasr, S. Song, A. Terzis, F. Tramer, Membership inference attacks from first principles, in: 2022 2022 IEEE Symposium on Security and Privacy (SP) (SP), IEEE Computer Society, Los Alamitos, CA, USA, 2022, pp. 1519–1519. doi:10.1109/SP46214.2022.00090.
URL https://doi.ieeecomputersociety.org/10.1109/SP46214.2022.00090

[17] K. Leino, M. Fredrikson, Stolen memories: Leveraging model memorization for calibrated white-box membership inference, in: Proceedings of the 29th USENIX Conference on Security Symposium, SEC'20, USENIX Association, USA, 2020.

[18] G. Del Grosso, H. Jalalzai, G. Pichler, C. Palamidessi, P. Piantanida, Leveraging adversarial examples to quantify membership information leakage, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 10399–10409.

[19] A. Salem, Y. Zhang, M. Humbert, P. Berrang, M. Fritz, M. Backes, Ml-leaks: Model and data independent membership inference attacks and defenses on machine learning models, in: Proceedings of the 26th Annual Network and Distributed System Security Symposium (NDSS), 2019.

[20] Y. Chen, C. Shen, Y. Shen, C. Wang, Y. Zhang, Amplifying membership exposure via data poisoning, in: NeurIPS 2022, 2022.
URL https://publications.cispa.saarland/3876/

[21] J. Hayes, L. Melis, G. Danezis, E. D. Cristofaro, Logan: Membership inference attacks against generative models, Proceedings on Privacy Enhancing Technologies 2019 (1) (2019) 133 – 152. doi:https://doi.org/10.2478/popets-2019-0008.
URL https://content.sciendo.com/view/journals/popets/2019/1/article-p133.xml

[22] D. Chen, N. Yu, Y. Zhang, M. Fritz, Gan-leaks: A taxonomy of membership inference attacks against generative models, in: Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security, CCS '20, Association for Computing Machinery, New York, NY, USA, 2020, p. 343–362.

doi:10.1145/3372297.3417238.
URL https://doi.org/10.1145/3372297.3417238

[23] S. Hidano, T. Murakami, S. Katsumata, S. Kiyomoto, G. Hanaoka, Model inversion attacks for prediction systems: Without knowledge of non-sensitive attributes, in: 2017 15th Annual Conference on Privacy, Security and Trust (PST), 2017, pp. 115–11509. doi:10.1109/PST.2017.00023.

[24] C. Song, T. Ristenpart, V. Shmatikov, Machine learning models that remember too much, in: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS '17, Association for Computing Machinery, New York, NY, USA, 2017, p. 587–601. doi:10.1145/3133956.3134077.
URL https://doi.org/10.1145/3133956.3134077

[25] B. Zhao, A. Agrawal, C. Coburn, H. Asghar, R. Bhaskar, M. Kaafar, D. Webb, P. Dickinson, On the (in)feasibility of attribute inference attacks on machine learning models, in: 2021 IEEE European Symposium on Security and Privacy (EuroSamp;P), IEEE Computer Society, Los Alamitos, CA, USA, 2021, pp. 232–251. doi:10.1109/EuroSP51992.2021.00025.
URL https://doi.ieeecomputersociety.org/10.1109/EuroSP51992.2021.00025

[26] L. Melis, C. Song, E. De Cristofaro, V. Shmatikov, Exploiting unintended feature leakage in collaborative learning, in: 2019 IEEE Symposium on Security and Privacy (SP), 2019, pp. 691–706. doi:10.1109/SP.2019.00029.

[27] L. Pengcheng, J. Yi, L. Zhang, Query-efficient black-box attack by active learning, in: 2018 IEEE International Conference on Data Mining (ICDM), 2018, pp. 1200–1205. doi:10.1109/ICDM.2018.00159.

[28] M. Fredrikson, S. Jha, T. Ristenpart, Model inversion attacks that exploit confidence information and basic countermeasures, in: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, CCS '15, Association for Computing Machinery, New York, NY, USA, 2015, p. 1322–1333. doi:10.1145/2810103.2813677.
URL https://doi.org/10.1145/2810103.2813677

[29] S. Basu, R. Izmailov, C. Mesterharm, Membership model inversion attacks for deep networks, NeurIPS 2019, Workshop on Privacy in Machine Learning abs/1910.04257 (2019). arXiv:1910.04257.
URL http://arxiv.org/abs/1910.04257

[30] T. Baumhauer, P. Schöttle, M. Zeppelzauer, Machine unlearning: Linear filtration for logit-based classifiers, ArXiv abs/2002.02730 (2020).

[31] Z. Yang, J. Zhang, E.-C. Chang, Z. Liang, Neural network inversion in adversarial setting via background knowledge alignment, in: Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, CCS '19, Association for Computing Machinery, New York, NY, USA, 2019, p. 225–240. doi:10.1145/3319535.3354261.
URL https://doi.org/10.1145/3319535.3354261

[32] B. Hitaj, G. Ateniese, F. Perez-Cruz, Deep models under the gan: Information leakage from collaborative deep learning, in: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS '17, Association for Computing Machinery, New York, NY, USA, 2017, p. 603–618. doi:10.1145/3133956.3134012.
URL https://doi.org/10.1145/3133956.3134012

[33] A. Salem, A. Bhattacharya, M. Backes, M. Fritz, Y. Zhang, Updates-leak: Data set inference and reconstruction attacks in online learning, in: Proceedings of the 29th USENIX Conference on Security Symposium, SEC'20, USENIX Association, USA, 2020.

[34] X. Wu, M. Fredrikson, S. Jha, J. F. Naughton, A methodology for formalizing model-inversion attacks, in: 2016 IEEE 29th Computer Security Foundations Symposium (CSF), 2016, pp. 355–370. doi:10.1109/CSF.2016.32.

[35] F. Tramèr, F. Zhang, A. Juels, M. K. Reiter, T. Ristenpart, Stealing machine learning models via prediction apis, in: Proceedings of the 25th USENIX Conference on Security Symposium, SEC'16, USENIX Association, USA, 2016, p. 601–618.

[36] C. Dwork, Differential privacy, in: Proceedings of the 33rd International Conference on Automata, Languages and Programming - Volume Part II, ICALP'06, Springer-Verlag, Berlin, Heidelberg, 2006, p. 1–12. doi:10.1007/11787006_1.

[37] C. Dwork, A. Roth, The algorithmic foundations of differential privacy, Found. Trends Theor. Comput. Sci. 9 (3–4) (2014) 211–407. doi:10.1561/0400000042.
URL https://doi.org/10.1561/0400000042

[38] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, L. Zhang, Deep learning with differential privacy, in: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, CCS '16, Association for Computing Machinery, New York, NY, USA, 2016, p. 308–318. doi:10.1145/2976749.2978318.
URL https://doi.org/10.1145/2976749.2978318

[39] B. Z. H. Zhao, M. A. Kaafar, N. Kourtellis, Not one but many tradeoffs, Proceedings of the 2020 ACM SIGSAC Conference on Cloud Computing Security Workshop (Nov 2020). doi:10.1145/3411495.3421352.
URL http://dx.doi.org/10.1145/3411495.3421352

[40] B. Jayaraman, D. Evans, Evaluating differentially private machine learning in practice, in: 28th USENIX Security Symposium (USENIX Security 19), USENIX Association, Santa Clara, CA, 2019, pp. 1895–1912.
URL https://www.usenix.org/conference/usenixsecurity19/presentation/jayaraman

[41] A. Triastcyn, B. Faltings, Improved accounting for differentially private learning, CoRR abs/1901.09697 (2019). arXiv:1901.09697.
URL http://arxiv.org/abs/1901.09697

[42] F. Lu, J. Munoz, M. Fuchs, T. LeBlond, E. V. Zaresky-Williams, E. Raff, F. Ferraro, B. Testa, A general framework for auditing differentially private machine learning, in: A. H. Oh, A. Agarwal, D. Belgrave, K. Cho (Eds.), Advances in Neural Information Processing Systems, 2022.
URL https://openreview.net/forum?id=AKM3C3tsSx3

[43] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, V. Shmatikov, How to backdoor federated learning (2019). arXiv:1807.00459.

[44] J. So, B. Guler, A. S. Avestimehr, A scalable approach for privacy-preserving collaborative machine learning (2020). arXiv:2011.01963.

[45] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, K. Seth, Practical secure aggregation for privacy-preserving machine learning, in: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS '17, Association for Computing Machinery, New York, NY, USA, 2017, p. 1175–1191. doi:10.1145/3133956.3133982.
URL https://doi.org/10.1145/3133956.3133982

[46] R. Shokri, V. Shmatikov, Privacy-preserving deep learning, in: 2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton), 2015, pp. 909–910. doi:10.1109/ALLERTON.2015.7447103.

[47] J. Konecny, H. B. McMahan, D. Ramage, P. Richtarik, Federated optimization: Distributed machine learning for on-device intelligence (2016). arXiv:1610.02527.

[48] V. Mothukuri, R. M. Parizi, S. Pouriyeh, Y. Huang, A. Dehghantanha, G. Srivastava, A survey on security and privacy of federated learning, Future Generation Computer Systems 115 (2021) 619–640. doi:https://doi.org/10.1016/j.future.2020.10.007.
URL https://www.sciencedirect.com/science/article/pii/S0167739X20329848

[49] L. Song, R. Shokri, P. Mittal, Privacy risks of securing machine learning models against advers Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security (Nov 2019). doi:10.1145/3319535.3354211.
URL http://dx.doi.org/10.1145/3319535.3354211

[50] J. Jia, A. Salem, M. Backes, Y. Zhang, N. Z. Gong, Memguard: Defending against black-box membership inference attacks via adversarial examp in: Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, CCS '19, Association for Computing Machinery, New York, NY, USA, 2019, p. 259–274. doi:10.1145/3319535.3363201.
URL https://doi.org/10.1145/3319535.3363201

[51] N. Phan, M. T. Thai, H. Hu, R. Jin, T. Sun, D. Dou, Scalable differential privacy with certified robustness in adversarial learning (2020). arXiv:1903.09822.

[52] J. Neyman, E. S. Pearson, IX. on the problem of the most efficient tests of statistical hypotheses, Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character 231 (694-706) (1933) 289–337.

[53] A. L. Gibbs, F. E. Su, On choosing and bounding probability metrics, International statistical review 70 (3) (2002) 419–435.

[54] V. V. Buldygin, Y. V. Kozachenko, Sub-gaussian random variables, Ukrainian Mathematical Journal 32 (6) (1980) 483–489.

[55] A. Klenke, Probability Theory, Springer, 2013. doi:10.1007/978-1-84800-048-3.

[56] H. H. Bauschke, P. L. Combettes, Convex Analysis and Monotone Operator Theory in Hilbert Spaces, 1st Edition, Springer Publishing Company, Incorporated, 2011.

[57] R. Bassily, S. Moran, I. Nachum, J. Shafer, A. Yehudayoff, Learners that use little information, in: F. Janoos, M. Mohri, K. Sridharan (Eds.), Proceedings of Algorithmic Learning Theory, Vol. 83 of Proceedings of Machine Learning Research, PMLR, 2018, pp. 25–55.
URL http://proceedings.mlr.press/v83/bassily18a.html

[58] A. Krizhevsky, G. Hinton, Learning multiple layers of features from tiny images, Tech. Rep. 0, University of Toronto, Toronto, Ontario (2009).

[59] Y. LeCun, C. Cortes, C. Burges, Mnist handwritten digit database, ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist 2 (2010).

[60] H. Xiao, K. Rasul, R. Vollgraf, Fashion-mnist: a novel image dataset for benchmarking machir
CoRR abs/1708.07747 (2017). arXiv:1708.07747.
URL http://arxiv.org/abs/1708.07747

[61] D. Dua, C. Graff, UCI machine learning repository (2017).
URL http://archive.ics.uci.edu/ml

[62] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, cite arxiv:1412.6980Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015 (2014).
URL http://arxiv.org/abs/1412.6980

[63] A. B. Tsybakov, Introduction to Nonparametric Estimation, 1st Edition, Springer Publishing Company, Incorporated, 2008.

[64] Y. Polyanskiy, Y. Wu, Lecture notes on information theory, Lecture Notes for ECE563 (UIUC) (2019).

[65] T. M. Cover, J. A. Thomas, Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing), Wiley-Interscience, USA, 2006.

[66] R. Bassily, S. Moran, I. Nachum, J. Shafer, A. Yehudayoff, Learners that use little information, in: Algorithmic Learning Theory, PMLR, 2018, pp. 25–55.