Formalizing Attribute and Membership Inference Attacks on Machine Learning Models

Ganesh Del Grosso
ganeshdg95@gmail.com
INRIA, Ecole Polytechnique
France

Georg Pichler
georg.pichler@ieee.org
TU Wien
Austria

Pablo Piantanida
pablo.piantanida@centralesupelec.fr
CNRS, CentraleSupelec
France

Catuscia Palamidessi
catuscia@lix.polytechnique.fr
INRIA, Ecole Polytechnique
France

1 Introduction

Machine learning (ML) models learn the correlation between input features and output labels through a set of examples, the training set. These models have been shown to memorize information specific to their training set, which causes severe privacy concerns when the training data is sensitive by nature.

In our work, we focus on membership inference attacks as the main method to assess privacy. Membership inference attacks aim at determining if a sample belongs to the training set of a model. For such an attack, the adversary has access to a target sample and the model. If an attacker cannot even determine membership, it is considered infeasible to obtain more detailed information. Therefore, robustness against membership inference attacks prevents other, more severe, privacy violations.

Furthermore, we expose 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. We study several attribute inference strategies against ML models. Our framework allows us to formalise these problems, draw privacy guarantees in a worst-case scenario and find connections between generalization and privacy.

Summary of contributions. We propose a novel and flexible formalism for the study of inference attacks that captures both membership and attribute inference problems by modeling the target attribute as a finite random variable.

1. *Characterization of the optimal attacker.* By considering the success probability of the optimal attacker, we are able to draw strong conclusions about the privacy of a ML model. The optimal attacker has perfect knowledge of the underlying probability distributions and from that it evaluates the conditional probability mass functions of the sensitive attribute, given the observed data. As such, it provides an upper bound to the probability of success of any attack strategy (Theorem 3.1). As a matter of fact, this bound represents a privacy guarantee for any ML model and can be useful to guide the design of privacy defense mechanisms.

2. *A ML model that does not generalize well is susceptible to membership inference attacks.* Theorem 3.2, which generalizes [15, Theorem 1], provides a lower bound for the case of bounded, and tail-bounded loss functions. 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.*

3. *We show how our theoretical results might be used in practical scenarios.* For all our use-cases, we find the connection between the success rate of the optimal membership inference attack and the generalization gap via Theorem 3.2. We consider linear regression of Gaussian data, which allows us to analytically compute the success rate of the optimal attacker. Subsequently, we study Deep Neural Networks (DNNs) for classification of images (CIFAR10) and hand-written digits (PenDigits).In the former we assess the quality of the bound given by Theorem 3.2 using the *likelihood attack* strategy. In the latter we apply several strategies for attribute inference and compare their effectiveness.

Related Work. Yeom *et al.* [15] study the interplay between generalization, differential privacy, 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 their 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. Nonetheless, they prove this equivalence does not hold in general, in the sense that there are learning algorithms which generalize well but still leak a large amount of sensitive information. Similar results were obtained in [3, 9]. In contrast, we consider the optimal 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. In this line of work, Sablayrolles *et at.* [11] derive an optimal attack strategy for membership

inference. 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 optimal attacker can be specialized to their framework and models.

References [12] and [10] utilize membership inference attacks to measure privacy leakage in deep neural networks. These works train a classifier to distinguish members from non-members in both black-box and white-box scenarios. The attack strategies we consider do not require to train an attacker model.

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, 7, 13]. 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. [1, 2, 5, 13, 14]. 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.

2 Preliminaries

We assume a fully Bayesian framework, where $Z = (X, Y) \sim p_{XY} \equiv p_Z$ denotes data X and corresponding labels Y, drawn from sets $\mathcal{X}$ and $\mathcal{Y}$, respectively. The training set consists of n i.i.d. copies $\mathcal{D}_n \triangleq \{z_1, \dots, z_n\}$ drawn according to $\mathbf{Z} \sim p_Z^n$. Let $\mathcal{F} \triangleq \left\{f_\theta \,|\, \theta \in \Theta \subseteq \mathbb{R}^d\right\}$ 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 $\mathcal{Y}$. The symbol $\hat{Y}_\theta(x)$ will be used to denote the random variable on $\mathcal{Y}$ distributed according to $f_\theta(\,\cdot\,; x)$.

A *learning algorithm* is a (possibly randomized) algorithm $\mathcal{A}$ that assigns to every training set $\mathcal{D}_n \in (\mathcal{X} \times \mathcal{Y})^n$ a probability distribution on the parameter space Θ (and, thus, also on the hypothesis space $\mathcal{F}$). We have $\mathcal{A}\colon \mathcal{D}_n \mapsto \mathcal{A}(\,\cdot\,; \mathcal{D}_n)$, where $\mathcal{A}(\,\cdot\,; \mathcal{D}_n)$ is a probability distribution on Θ. The symbol $\widehat{\theta}(\mathcal{D}_n)$ is used to denote a random variable on Θ, distributed according to $\mathcal{A}(\,\cdot\,; \mathcal{D}_n)$.

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 2.1 (Generalization Gap)**.** The *generalization error*[1] and the *empirical risk*[2] of a learning algorithm $\mathcal{A}$ at training set $\mathcal{D}_n$ are respectively defined as:

$$\mathcal{E}(\mathcal{A}, \mathcal{D}_n) \triangleq \mathbb{E}\left[\ell\left(\hat{Y}_{\widehat{\theta}(\mathcal{D}_n)}(X), Y\right)\right], \tag{1}$$

$$\mathcal{E}_{\mathrm{emp}}(\mathcal{A}, \mathcal{D}_n) \triangleq \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n \ell\left(\hat{Y}_{\widehat{\theta}(\mathcal{D}_n)}(x_i), y_i\right)\right]. \tag{2}$$

The difference between generalization error and empirical risk is the *generalization gap*

$$\mathcal{E}_{\mathrm{G}}(\mathcal{A}, \mathcal{D}_n) = \mathcal{E}(\mathcal{A}, \mathcal{D}_n) - \mathcal{E}_{\mathrm{emp}}(\mathcal{A}, \mathcal{D}_n)\ . \tag{3}$$

In order to make privacy guarantees for an algorithm $\mathcal{A}$, we need to specify an attacker model and the capabilities of the attacker. We will not make assumptions about the computation power afforded to an attacker. We will 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 $\mathcal{D}_n$ from $\widehat{\theta}(\mathcal{D}_n)$. 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 membership inference attack) 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 $p_T = p_{T|S}$.

Definition 2.2. The Bayes *success probability* of an attack strategy φ is given by $\mathcal{P}_{\mathrm{Suc}}(\varphi) = \mathrm{P}\{\varphi(\widehat{\theta}(\mathbf{Z}), S) = T\}$.

Definition 2.3 (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.

Definition 2.4 (Membership inference attack)**.** In a membership inference attack, let T be a Bernoulli variable on $\mathcal{T} = \{0, 1\}$ and J is independently, uniformly distributed on $\{1, 2, \dots, n\}$. 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. We also define the expected loss function $\varrho(\theta, (x, y)) \triangleq \mathbb{E}[\ell(\hat{Y}_\theta(x), y)]$ and the corresponding random variable $R \triangleq \varrho(\widehat{\theta}(\mathbf{Z}), S)$.

3 Main Results

We begin by establishing a theorem that provides upper bounds on the success probability on an arbitrary attacker.

[1] The expectation is taken over all quantities: $\widehat{\theta} \sim \mathcal{A}(\,\cdot\,; \mathcal{D}_n)$, $\hat{Y}_{\widehat{\theta}(\mathcal{D}_n)}(X) \sim f_{\widehat{\theta}}(\,\cdot\,; X)$ and, $(X, Y) \sim p_Z$.

[2] Note that the empirical risk is computed using the training data of the algorithm.

The theorem considers the general case in which the target attribute T is not necessarily binary, but discrete. This case includes both membership and feature inference attacks. In this case the Bayes classifier is the best possible attacker.

Theorem 3.1 (Success of the optimal attacker). *Assume that $\mathcal{T}$ is a finite set and φ is an arbitrary attack strategy. The Bayes success probability is upper bounded by,*

$$\mathcal{P}_{\text{Suc}}(\varphi) \leq \mathbb{E}\left[\max_{t\in\mathcal{T}} p_{T|\widehat{\theta}(\mathbf{Z})S}(t|\widehat{\theta}(\mathbf{Z}), S)\right], \tag{4}$$

where the upper bound is achieved by the attack strategy,

$$\varphi^{\star}(\theta, s) = \arg\max_{t\in\mathcal{T}} p_{T|\widehat{\theta}(\mathbf{Z})S}(t|\theta, s)\ . \tag{5}$$

If the arg max *in Eq. (5) is not unique, any $t \in \mathcal{T}$ achieving the maximum can be chosen.*

Given white-box access to the model and its parameters, as well as side information, the attacker in 5 has the highest probability of successfully identifying a record in the training set. Thus, robustness against strategy Eq. (5) provides a strong privacy guarantee.

Now we explore the connection between generalization gap and the success probability of membership inference attacks. Large generalization gap implies poor privacy guarantees against membership inference attacks. Moreover, depending on characteristics of the loss function, the probability of success of the attacker is lower bounded by the generalization gap:

Theorem 3.2 (Lower Bounds on Success Rate of the Optimal Attacker). *Provided the loss $|\ell| \leq \ell_{\max}$, then there is an attack strategy φ for a membership inference attack (Definition 2.4) such that,*

$$\mathcal{P}_{\text{Suc}}(\varphi) \geq \max\left\{P_{\text{m}}, P_{\text{m}}\left(\frac{|\mathbb{E}[\mathcal{E}_{\text{G}}(\mathcal{A}, \mathbf{Z})]|}{2\ell_{\max}} - 1\right) + 1\right\}, \tag{6}$$

where $P_{\text{m}} \triangleq \max_{t\in\{0,1\}} \mathrm{P}\{T = t\}$. Moreover, for a tail-bounded loss function, we obtain the following result. In a membership inference problem (Definition 2.4), assume that $R \triangleq \varrho(\widehat{\theta}(\mathbf{Z}), S)$ 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,

$$\mathcal{P}_{\text{Suc}}(\varphi) \geq \max\left\{P_{\text{m}}, P_{\text{m}}\left(\frac{|\mathbb{E}[\mathcal{E}_{\text{G}}(\mathcal{A}, \mathbf{Z})]|}{2R_{\max}} - \frac{R_{\max} + 2\sigma_R^2}{R_{\max}(1 - P_{\text{m}})} \cdot e^{-\frac{R_{\max}}{2\sigma_R^2}} - 1\right) + 1\right\}. \tag{7}$$

Note that an attacker, knowing the prior probabilities of T, will have a success probability of at least P_{m}. Theorem 3.2 indicates that strong privacy guarantees (i.e., small success probability for any attacker), imply that the generalization gap is also small. Remark that, on the other hand, ensuring that the generalization gap is small does not make a model robust against membership inference attacks.

4 Examples and Numerical Experiments

4.1 Linear Regression on (Synthetic) Gaussian Data

We implement the optimal attacker from Theorem 3.1 and estimate its success probability to monitor the privacy leakage of the model as a function of the number of training samples. Additionally, since the loss is tail-bounded exponentially, we use Theorem 3.2 to derive lower bounds on the success probability of the 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 $\mathcal{D}_n = \{y_1, \ldots, y_n\}$, 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,

$$\mathbb{E}[\mathcal{E}_{\text{G}}(\mathcal{A}, \mathbf{Z})] = \frac{2d}{n}\sigma^2. \tag{8}$$

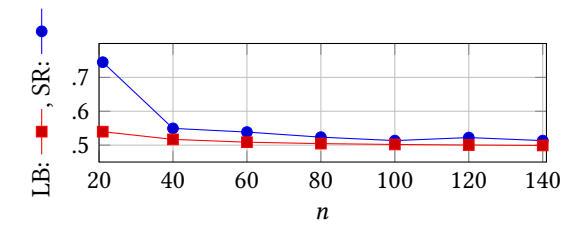
Assuming the noise W to be Gaussian, the scalar response $\mathbf{Y} = \beta^T\mathbf{x} + \mathbf{W}$ then also follows a Gaussian distribution, with $\mathbf{W}$ a row vector of i.i.d components. Similarly, the model parameters $\widehat{\theta}(\mathbf{Y})$ are normally distributed. Now choose a test sample $S = T(Y_J) + (1 - T)(Y'_J)$, where J is a random 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, we derive the success probability of the optimal attacker. In our experiments we perform a Monte Carlo estimation of the expectation in Eq. (4), 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 use Theorem 3.2 to obtain the lower bound

$$\mathcal{P}_{\text{Suc}}(\varphi^{\star}) \geq \frac{1}{2} + \frac{d}{2n}\frac{\sigma^2}{R_{\max}} - \exp\left(-\frac{R_{\max}}{2\sigma^2}\right)\left(1 + \frac{2\sigma^2}{R_{\max}}\right), \tag{9}$$

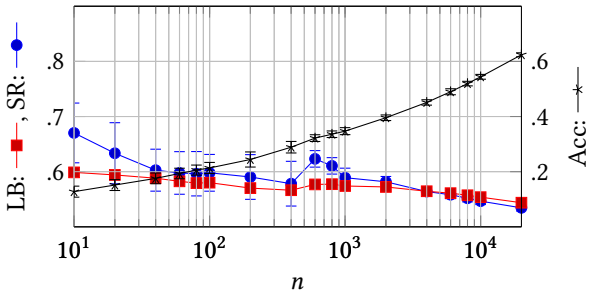
where we used Eq. (8). $R_{\max}$ can be chosen to maximize the lower bound.

In our experiments we vary n to study how the generalization gap and 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 estimate the success rate of the optimal attacker. Additionally, we compute the generalization gap Eq. (8) to obtain the lower bound Eq. (9). Figure 1(a) shows the success rate (SR) of the optimal attacker as a function of n, the number of training samples. Along with it is the lower bound (LB) provided by Theorem 3.2. This example shows that the bounds are not vacuous and they may serve

[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.



(a) Multivariate Gaussian data.



(b) CIFAR10 dataset.

Figure 1. Success Rate (SR), Lower Bound (LB), Accuracy (Acc; legend on the right axis) as function of training samples n.

as a framework for understanding the connection between information leakage and generalization in ML.

4.2 Examples on DNNs

We train DNNs on CIFAR10 [8] to study the interplay between generalization gap and the success rate of an attacker that uses the confidence of the target model as a criterion for the attack. We compare the success rate of the attacker to the lower bound provided by Theorem 3.2, to assess the quality of the bound.

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, which allows us to apply Theorem 3.2 to lower bound the success probability of the optimal attacker. However, in this setup it results impossible to estimate the success probability of the optimal attacker, due to the high number of model parameters. To circumvent this limitation and assess the quality of the bound provided by Theorem 3.2, we implement the likelihood attack and compare its success rate to the bound provided. This 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.

The success rate (SR) of the likelihood attack, the lower bound (LB) provided by Theorem 3.2, and the accuracy on the test set (Acc) are computed as a function of the number of samples in the training set n and reported in Figure 1(b).

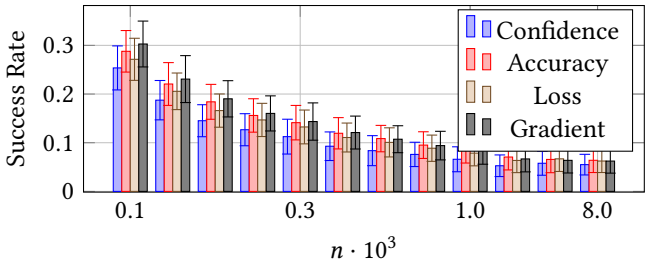


Figure 2. Success Rate of different attribute inference attack strategies.

The lower bound predicts the behaviour of the success rate of the likelihood attack as a function of the generalization gap; both approach 0.5 as the generalization gap vanishes.

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 [4], 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.

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.

Confidence: The intuition behind this attack is that a model is more confident on samples that were part of its training. 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.,

$$\varphi(v,\theta) = \arg\max_{t\in\mathcal{T}} \left[\max_{i\in|\mathcal{Y}|} f_\theta^i((v,t)) \right], \tag{10}$$

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

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

$$\varphi(v,y,\theta) = \arg\max_{t\in\mathcal{T}: x\in\widehat{X}_{y\theta}} \left[\max_{i\in|\mathcal{Y}|} f_\theta^i((v,t)) \right]. \tag{11}$$

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.,

$$\varphi(v, y, \theta) = \arg\min_{t \in \mathcal{T}} \ell\left(f_\theta((v, t)), y\right) \ . \tag{12}$$

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. 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.,

$$\varphi(v, y, \theta) = \arg\min_{t \in \mathcal{T}} \|\nabla_\theta \ell\left(f_\theta((v, t)), y\right)\|_2^2 \ . \tag{13}$$

In our experiments we perform attribute inference attacks using each of these strategies as we vary n. The success rates for each strategy are computed and reported in Figure 2. 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.

5 Acknowledgement

This research was supported by DATAIA
"Programme d'Investissement d'Avenir" (ANR-17-CONV-0003).

References

[1] Samyadeep Basu, Rauf Izmailov, and Chris Mesterharm. 2019. Membership Model Inversion Attacks for Deep Networks. *NeurIPS 2019, Workshop on Privacy in Machine Learning* abs/1910.04257 (2019). arXiv:1910.04257 http://arxiv.org/abs/1910.04257

[2] Thomas Baumhauer, P. Schöttle, and M. Zeppelzauer. 2020. Machine Unlearning: Linear Filtration for Logit-based Classifiers. *ArXiv* abs/2002.02730 (2020).

[3] Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. 2019. The Secret Sharer: Evaluating and Testing Unintended Memorization in Neural Networks. In *28th USENIX Security Symposium*. USENIX Association, Santa Clara, CA, 267–284.

[4] Dheeru Dua and Casey Graff. 2017. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml

[5] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. 2015. Model Inversion Attacks That Exploit Confidence Information and Basic Countermeasures. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security* (Denver, Colorado, USA) *(CCS '15)*. Association for Computing Machinery, New York, NY, USA, 1322–1333. https://doi.org/10.1145/2810103.2813677

[6] Matthew Fredrikson, Eric Lantz, Somesh Jha, Simon Lin, David Page, and Thomas Ristenpart. 2014. Privacy in Pharmacogenetics: An End-to-End Case Study of Personalized Warfarin Dosing. In *23rd USENIX Security Symposium (USENIX Security 14)*. USENIX Association, San Diego, CA, 17–32. https://www.usenix.org/conference/usenixsecurity14/technical-sessions/presentation/fredrikson_matthew

[7] S. Hidano, T. Murakami, S. Katsumata, S. Kiyomoto, and G. Hanaoka. 2017. Model Inversion Attacks for Prediction Systems: Without Knowledge of Non-Sensitive Attributes. In *2017 15th Annual Conference on Privacy, Security and Trust (PST)*. 115–11509. https://doi.org/10.1109/PST.2017.00023

[8] Alex Krizhevsky. 2009. *Learning multiple layers of features from tiny images*. Technical Report.

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

[10] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2019. Comprehensive Privacy Analysis of Deep Learning: Passive and Active White-box Inference Attacks against Centralized and Federated Learning. *2019 IEEE Symposium on Security and Privacy (SP)* (May 2019). https://doi.org/10.1109/sp.2019.00065

[11] Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, Yann Ollivier, and Herve Jegou. 2019. White-box vs Black-box: Bayes Optimal Strategies for Membership Inference. In *ICML (Proc. of Machine Learning Research, Vol. 97)*, Kamalika Chaudhuri and Ruslan Salakhutdinov (Eds.). PMLR, 5558–5567.

[12] R. Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership Inference Attacks Against Machine Learning Models. *2017 IEEE Symposium on Security and Privacy (SP)* (2017), 3–18.

[13] Xi Wu, Matthew Fredrikson, Somesh Jha, and Jeffrey F. Naughton. 2016. A Methodology for Formalizing Model-Inversion Attacks. In *2016 IEEE 29th Computer Security Foundations Symposium (CSF)*. 355–370. https://doi.org/10.1109/CSF.2016.32

[14] Ziqi Yang, Jiyi Zhang, Ee-Chien Chang, and Zhenkai Liang. 2019. Neural Network Inversion in Adversarial Setting via Background Knowledge Alignment. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security* (London, United Kingdom) *(CCS '19)*. Association for Computing Machinery, New York, NY, USA, 225–240. https://doi.org/10.1145/3319535.3354261

[15] S. Yeom, I. Giacomelli, M. Fredrikson, and S. Jha. 2018. 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, 268–282. https://doi.org/10.1109/CSF.2018.00027