Understanding Membership Inferences on Well-Generalized Learning Models

Yunhui Long[1], Vincent Bindschaedler[1], Lei Wang[2], Diyue Bu[2], Xiaofeng Wang[2], Haixu Tang[2], Carl A. Gunter[1], and Kai Chen[3,4]

[1]University of Illinois at Urbana-Champaign
[2]Indiana University Bloomington
[3]State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences
[4]School of Cyber Security, University of Chinese Academy of Sciences

Abstract

Membership Inference Attack (MIA) determines the presence of a record in a machine learning model's training data by querying the model. Prior work has shown that the attack is feasible when the model is overfitted to its training data or when the adversary controls the training algorithm. However, when the model is *not* overfitted and the adversary does *not* control the training algorithm, the threat is not well understood. In this paper, we report a study that discovers overfitting to be a *sufficient* but not a *necessary* condition for an MIA to succeed. More specifically, we demonstrate that even a well-generalized model contains vulnerable instances subject to a new generalized MIA (GMIA). In GMIA, we use novel techniques for selecting vulnerable instances and detecting their subtle influences ignored by overfitting metrics. Specifically, we successfully identify individual records with high precision in real-world datasets by querying black-box machine learning models. Further we show that a vulnerable record can even be *indirectly* attacked by querying other related records and existing generalization techniques are found to be less effective in protecting the vulnerable instances. Our findings sharpen the understanding of the fundamental cause of the problem: the unique influences the training instance may have on the model.

1 Introduction

The recent progress on machine learning has brought in a new wave of technological innovations, ranging from automatic driving, face recognition, natural language processing to intelligent marketing, advertising, healthcare data management, etc. To support the emerging machine learning ecosystem, major cloud providers are pushing *Machine Learning as a Service* (MLaaS), providing computing platforms and learning frameworks to help their customers conveniently train their own models based upon the datasets they upload. Prominent examples include Amazon Machine Learning (ML), Google Prediction API, and Microsoft Azure Machine Learning. The models trained on these platforms can be made available by the data owners to their users for online queries. What is less clear are the privacy implications of these exported models, particularly whether uncontrolled queries on them could lead to exposure of training data that often includes sensitive content such as purchase preferences, patients' health information, and recorded commands and online behavior.

Membership inference attack. Prior research demonstrates that a *membership inference attack* (MIA) can succeed on *overfitted* models with only black-box access to the model [32]. In such an attack, the adversary, who can only query a given *target model* without knowing its internal parameters, can determine whether a specific record is inside the model's training dataset. This type of attacks can have a significant privacy implication such as re-identifying a cancer patient whose data is used to train a classification model. For this purpose, the prior research trains an attack model that utilizes the target model's classification result for a given input to determine whether the input is present in the target model's training set. Such an attack model can be constructed using labeled datasets generated by a set of *shadow models* trained to imitate the behaviors of the target model. This approach is effective when the target models are *overfitted* to training data.

It remains unclear whether it is feasible to perform MIA on *well-generalized* models with *only* black-box access. This problem should be distinguished from prior attacks on non-overfitted models under *a different adversarial model*. In these attacks, the adversary controls the training algorithm and stealthily embeds information in the model [33, 39].

Rethinking ML privacy risk. In our research, we revisited the threat of MIA, in an attempt to answer the following questions: (1) is overfitting a root cause of membership disclosure from a machine learning model? (2) if so, is generalization the right answer to the problem? (3) if not, what indeed causes the information leak? The

findings of our study, though still not fully addressing these issues, make a step closer toward that end, helping us better understand the threat of MIA.

(1) Is overfitting a root cause of membership disclosure from a machine learning model?

We discover that overfitting, as considered in evaluating machine learning models, can be *sufficient* but is by no means *necessary* for exposing membership information from training data. As evidence, we run a new MIA (called generalized MIA or GMIA) that successfully identifies *some* individuals in the training sets from three neural-network models, for predicting salary class, cancer diagnosis and written digits, *even when these models are not overfitted*. Particularly, our attack automatically picks 5 vulnerable patient samples from the Cancer dataset [26], 16 images from the MNIST dataset [24], and 13 individuals from the Adult dataset [26] as the attack object. We identified 73.88% of the models, from which the target images in the MNIST dataset can be inferred with a precision of 93.36%. Similarly, we inferred the presence of target patients in the Cancer dataset with a precision of 88.89% in 3.2% of the models and the presence of target individuals in the Adult dataset with a precision of 73.91% in 5.23% of the models.

Further interesting is the observation that the adversary does not even need to query the models for the target record to determine its presence, as it does in the prior research: instead, the adversary can search for different but related records and use their classifications by models to determine the object's membership in the training data.

(2) Is generalization the right solution for membership disclosure?

We find that existing regularization approaches are *insufficient* to defeat our attack, which can still determine the presence of an image in the MNIST dataset in 34% of all the models with a precision of 100% even when L2 regularization is applied. This finding deviates from what is reported in the prior research, whose MIA can be effectively suppressed by regularization [32].

(3) What is the fundamental cause of membership disclosure?

We observe that such information leaks are caused by the unique influences a specific instance in the training set can have on the learning model. The influences affect the model's outputs (i.e., predictions) with regards to a single or multiple inputs. So once the adversary has asked enough questions, no guarantee will be there that he cannot capture the influences (possibly from multiple queries) to infer the presence of a certain training instance. It is important to note that overfitting is essentially a *special* case of such unique influences but the generic situation is much more complicated. In detection of overfitting, we look for an instance's positive impact on a model's accuracy for the training set and limited/negative impact for the testing set. On the contrary, finding the unique influences in general needs to consider the case when an instance both contributes useful information to the model for predicting other instances and brings in noise uniquely characterizing itself. The model generalization methods that suppress overfitting may reduce the noise introduced by training instances, but cannot completely remove their unique influences, particularly the influences essential for the model's prediction power. On the other hand, noise adding techniques based on the concept of differential privacy [11] can guarantee the low influence of each training instance, while also reduces the prediction accuracy of the model. How to capture the non-noise influences of learning instances through the model's output and how to identify all the *vulnerable* instances with identifiable influences on the model remain open questions.

Generalized MIA. These discoveries are made possible by a novel inference attack we call *Generalized MIA (GMIA)*. In GMIA, we propose a new technique for identifying vulnerable records in a large dataset and new methods for detecting the small influence of these records that are ignored by overfitting metrics and the prior attack.

Unlike overfitted models, whose answers (probabilities) to the queries on the training instances differ significantly from those to other queries, a well-generalized model behaves similarly on the training data and test data. As a result, no longer can we utilize shadow models to generate a meaningful training set for the attack model, since most positive instances here (those inside shadow models' training sets) can be less distinguishable from the negative instances (not in their training sets), in terms of their classification probabilities. To address this challenge, our approach focuses on detecting and analyzing vulnerable target records (outliers) to infer their membership. More specifically, GMIA first estimates whether a given instance is an outlier with regards to the data accessible to the adversary. This estimation is done by extracting high-level feature vector from the intermediate outputs of models trained on these data. We believe that an outlier is more likely to be a vulnerable target record when it is indeed inside the target model's training set. Then we train a set of *reference* models without the target record in the training set, and use these models to build the distribution for the target record's classification probabilities. After that, we run a hypothesis test to decide whether its classification by the target model is in line with this distribution. This approach successfully identifies training records of well-generalized models. For example, on the MNIST dataset, our attack achieves a precision of 93.36% in 73.88% of the models (on the vulnerable objects) when the cutoff p-value is 0.01.

It is even more challenging to attack a target record without directly querying it (*an indirect inference*), which has *never* been done before. To find the right queries for

the object, GMIA trains two sets of reference models: those include the object (*positive reference models*) and those do not (*reference models*). These two sets of models are used to filter out random queries, finding those whose probabilities for receiving the object's class labels are almost always higher with the positive reference models than with the reference models. The selected queries are run against the target model, and their results (classification probabilities) are compared with their individual distributions built from the *reference models* through a set of hypothesis tests. Finally, the test results (p-value) of individual queries are combined using Kost's method [23] to determine the object's presence in the target model's training set. On the Adult dataset, with a cut-off p-value of 0.01, this indirect attack inferred the presence of a record with a precision of 100% in 16% of the models when a direct attack failed to infer any of them.

Contributions. The contributions of the paper are summarized as follows:

• *New understanding about generalization and privacy*. We revisit the membership inference problem and find that overfitting is not a necessary condition for information leaks: even a well-generalized model still cannot prevent MIA, whenever some of its training instances have unique impacts on the learning model. This discovery reveals the fundamental challenges in protecting data privacy for machine learning models and can potentially inspire follow-up research on mitigating this risk.

• *New techniques for membership inference attacks*. We present new techniques for membership inferences on a well-generalized model. Our approach addresses the challenges of finding vulnerable target records, identifying their small influences on the target model, and attacking the target model without directly querying the target records.

• *Implementation and evaluation*. We implement our attacks and evaluate them against real-world datasets. Our studies demonstrate their effectiveness and also highlight the challenges in protecting machine learning models against such threats.

2 Background

2.1 Membership Inference Attacks

In a membership inference attack, the adversary's goal is to infer the membership status of a target individuals data in the input dataset to some computation. For a survey, the adversary wishes to ascertain, from aggregate survey responses, whether the individual participated in the survey. For machine learning, the adversary wishes to ascertain whether the target's record was part of the dataset used to train a specific model.

One of the first prominent examples of membership inference attacks occur in the context of Genome-Wide Association Studies (GWAS). The seminal work of Homer et al. [18] show that p-values, a type of aggregated statistics routinely published when reporting the results of studies, could be used to successfully infer membership status. Although this attack requires that the adversary know the genome of the target individual, it teaches an important lesson: seemly harmless aggregate statistics may contain sufficient information for successful membership inferences. As a consequence of this attack, NIH has removed all aggregate data of GWAS from public websites [40].

More recently, it was shown that membership inference attacks can occur in the context of machine learning. Shokri et al. [32] demonstrated that an adversary with only black-box access to a classifier could successfully infer membership status. However, their attack only works when the classifier is highly overfitted to its training dataset.

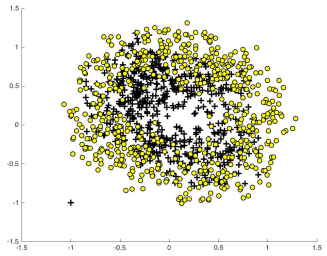
2.2 Model Generalization

A desirable property of any model is having low generalization error. Informally, this means the model should have good performance on unseen examples. More precisely, we adopt the approach of [7] which defines generalization error as the expected model error with the expectation taken over a random example from the population.

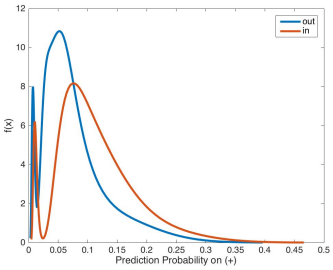
In practice, one does not have access to the population but only to samples from it. Thus, we must estimate the generalization error empirically. To do this, we can simply measure the generalization error with respect to a hold out set instead of the population. Informally, a good indication of low generalization error is if the model's performance is (almost) the same on the training and testing datasets.

2.3 Adversary Model

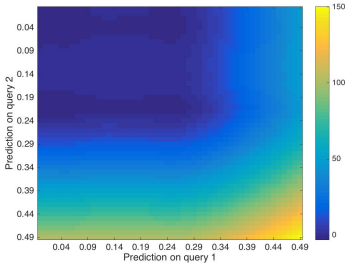
We consider an adversary mounting a membership inference attack against an already trained machine learning model, in which the adversary attempts to infer if a target record r is used as a training record for the target model M. We assume that the adversary has black-box access to the target model, i.e., he can issue arbitrary queries and retrieve the answers (e.g., the probability vector) from the model; the number of queries, however, may be limited. Similar as the previous work [32], we further assume that the adversary either (1) knows the structure of the target model (e.g., the depth and the number of neurons each layer of the neural network) and the training algorithm used to build the model, or (2) has black-box access to the machine learning algorithm used to train the model. We also assume that the adversary has some prior knowledge about the population from which the training records are drawn. Specifically, the adversary can access a set of records that are drawn independently from that population, which may or may not overlap with the actual training data for the target models; but the adversary does



(a) The toy example dataset. The dataset is composed records with real-valued features (x-axis and y-axis) and a binary label ($+$ or $-$) used for classification.



(b) Influence on the probability density function ($f(x)$) of the model's prediction. The `in` and `out` distributions do not fully overlap, which allows the adversary to distinguish them.



(c) Estimation of $\frac{\Pr_{\text{in}}}{\Pr_{\text{out}}}$ based on the model's predictions for two queries (query 1 on x-axis and query 2 on y-axis). When prediction for both queries are above 0.4, it is significantly more likely than not that the record was part of the training data.

Figure 1: Understanding the unique influence of a record through a toy example dataset (a). The adversary performs MIA by fingerprinting the target record's influence on the model's outputs (predicted class probabilities). There are two competing hypotheses: (1) H_{in}: record is part of the training data, and (2) H_{out}: record is not part of the training data. The adversary infers membership status by estimating which hypothesis is more likely based on the model's outputs.

not have any additional information about whether these records are present in the training data. These records can often be obtained from public dataset with similar attributes or from previous data breaches.

3 Understanding Membership Inference Attacks

Previous work demonstrates the vulnerability of machine learning (ML) models to membership inference attacks (MIAs), but little was known about its root cause. We revisit this based on the new results presented in this paper, in an attempt to understand the source of information leaks in machine learning models that can be exploited by MIAs.

3.1 Overfitting and Vulnerability of Machine Learning Models

Shokri et al. [32] show that overfitting is a sufficient condition for MIA. But is overfitting necessary for membership inference? This question is crucial yet not answered by prior work. If overfitting is a prerequisite for successful MIA then attacks can be mitigated using techniques to enforce generalization (e.g., model regularization).

Our research leads to the new observations that MIAs can still succeed even when the target model is well generalized. For example, using the MNIST set, we find 16 records (out of 20,000) whose membership can be inferred successfully with greater than 90% precision in 74% of the models. Moreover, we find that, while model regularization improves the generalization, it does not reliably eliminate the threat. For instance, using an image dataset, after applying L2 regularization with a coefficient of 0.01, the membership status of one image can still be inferred with 100% precision in 34% of the models.

3.2 Influence and Uniqueness

What causes models to leak membership information? Our research uncovers a new way of thinking about this question in terms of the unique influence of vulnerable records on the model. Informally, a record has a unique influence if there exists a set of queries for which the model outputs can reliably reveal the record's presence in the training data. In such a case, the adversary effectively infers the membership of a target record by the *fingerprint* of the record, i.e., the model outputs to queries of the target record and relevant records when the target record is included in the training set of the target model.

To explain why the unique influence of a target record is the key for a successful MIA, we consider an adversary attempting to determine the membership status of a target record r through black-box access to a target model M, using hypothesis testing between two hypotheses:

(H_{in}) r is in the training set of M

(H_{out}) r is *not* in the training set of M

By querying the model M, the adversary gathers evidence in favor of either H_{in} or H_{out}, eventually deciding in the favor of the more likely hypothesis.

To illustrate this approach, we use a toy dataset with 1,181 records (as shown in Figure 1a) to train a neural network model with two fully connected layers for binary classification. Suppose we want to infer the membership of a record r by querying a record q. Let $M(q)$ be the models output to q. Over the record space from which the training records are sampled, we derive two probability distributions of the output of q on the two different sets of models, respectively: 1) the models trained with r, and 2) the models trained without r. Specifically, as shown in Figure 1b, the probability density functions (pdfs) of the model outputs (i.e., the output probability of the positive class) under the hypotheses H_{in} and H_{out}, respectively, do not fully overlap, indicating an adversary can decide in

favor of H_{in} if the output probability of the positive class is above a threshold (e.g., 0.15).

The key to this strategy is that the two distributions are distinguishable. This happens because for a significant number of models, the outputs on q are consistently different when r is or is not included in the training data. In other words, r has a unique influence on models with respect to the record q. In practice, an adversary can only approximate these two distributions; but, as we show in this paper, it is feasible to identify vulnerable records, i.e., those with unique influences on the models. For example, for the MNIST dataset, we can efficiently infer the presence of 16 vulnerable images in the training dataset with precisions greater than 90% in 73.88% of the time.

We stress that what matters here is not the strength of the influence of records, but the influence being unique with respect to other records in the training set and in the record space. The attack fails if there are other records in the record space that would show a similar influence on the model as r if some of them were included in the training set. In such a case, r does not have a *unique* influence to the model, and the two distributions largely overlap.

3.3 Types of Influences

Our work here also demonstrates distinguishing between different kinds of influences that a record may have on the model, which lead to different mechanisms of MIA. Intuitively, we expect that strongest influence of a record is on the model's output for the query of this specific record. In fact, this is precisely consistent with our experimental observationsHowever, inclusion of a target record in the training set may influence the model's output behavior on the queries of other records, which may or may not be strongly correlated with the target record with respect to their features. Surprisingly, in our experiments, we observe that the attacks leveraging these *indirect* influences (i.e., the *indirect inferences*) are sometimes more effective than those based on the direct influences the target records. Specifically, in the Adult dataset, we identify a record whose presence can be inferred by the indirect inference with 100% precision in 14% of the models, whereas the direct inference failed to infer the record's precense in any of the models.

The indirect inferences are powerful because they allow an adversary to accumulate evidence from multiple queries. The more queries the adversary submits, the more likely an adversary can gather the unique influences of the target record, and thus the easier it is to discern the two distributions under different hypotheses as described above. Figure 1c illustrates this concept through a heatmap of the likelihood ratio under the hypotheses of H_{in} and H_{out}.

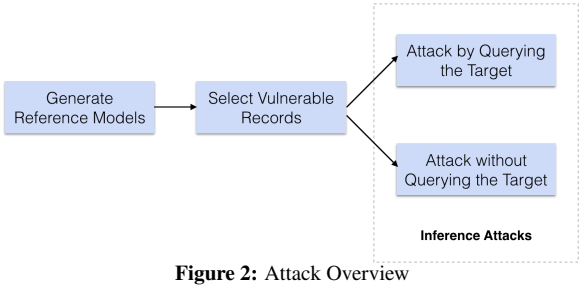


Figure 2: Attack Overview

4 Generalized Membership Inference Attack

In this section, we present the major components of the generalized MIA (GMIA) framework: reference model generation, vulnerable records selection, and the inference attacks. The latter includes the *direct inference* which queries the target record and the *indirect inference* which queries selected non-target records.

4.1 Attack Overview

Figure 2 shows the attack components in their logical sequence. Below, we briefly describe the methods involved in each component and the motivation. We present the details in the follow-up sections.

Building Reference Models. We build reference machine learning (ML) models to imitate the prediction behaviors of the target model, using reference records accessible to the adversary that represent the space where the actual training data are sampled. As the number of available reference records may be limited, we adopt bootstrap sampling [12] to construct training dataset for building multiple reference models. Once constructed, the reference models are exploited in each steps of the GMIA framework, including target record selection, query selection, and hypothesis testing.

Selecting Vulnerable Target Records. In well-generalized models, not all training records are vulnerable to MIA. Therefore, identifying vulnerable target records is the key to an effective attack. We develop a method for selecting vulnerable records by estimating the number of neighbors they have in the the sample space represented by the reference dataset. Records with fewer neighbors are more vulnerable under MIA because they are more likely to impose unique influence on the machine learning models. In order to identify neighbors of a given record, we construct a new feature vector for each record based on the intermediate outputs of reference models on this record, which implies this record's influence on the target machine learning model.

Direct Inference by Querying the Target Record. A training record usually influences the model's predictions on itself. However, in well-generalized models, this influence is usually small and hard to detect. In a direct inference, we attack a machine learning model by sub-

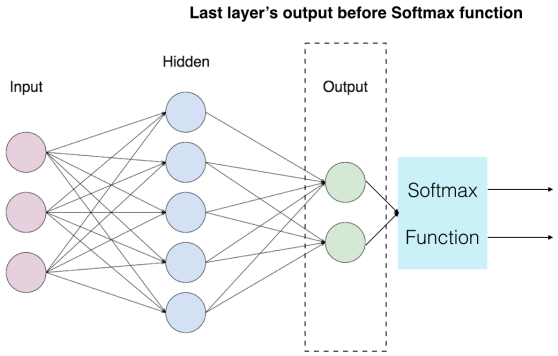


Figure 3: Last layer output of a two-layer neural network

mitting a query of the target record. We use a hypothesis test to determine whether the target model's prediction is deviated from the predictions of reference models for that the target records are not used in the training. The p-value from the hypothesis test indicates the confidence of the attack and thus allows the adversary to efficiently estimate the performance of the attack.

Indirect Inference without Querying the Target Record . We observe that a training record influences a model's predictions not only on itself but also on other seemingly uncorrelated records (called *enhancing records* in GMIA). In GMIA, we use novel techniques that iteratively search for and select enhancing records. Our indirect inferences using the enhancing records can successfully infer the presence of a target record without querying it. Moreover, the indirect inferences sometimes outperform direct inferences by accumulating more information from multiple queries.

4.2 Building Reference Models

GMIA exploits a target record's unique influence on the outputs of a machine learning model to infer the presence of the record in the training set of the target model (called target training set). To identify such influence, we need to estimate the model's behavior when the target record is *not* in the target training set. To achieve this goal, we build *reference models*, which are trained using the same algorithm on *reference datasets* sampled from the same space as the target training set, but not containing the target record. The process of building reference models are illustrated below.

To start with, we need to construct k reference datasets with the same size as the target training set. Since most practical machine learning models are trained on large training datasets, it is difficult for an adversary to get access to an even larger dataset with k times records as the target training set. Consequently, if we build the reference datasets by sampling without replacement from the whole set of reference records, the resulting datasets may share many records, and the reference models built from them would be alike and give similar outputs. To address this issue, we use bootstrap sampling [12] to generate the

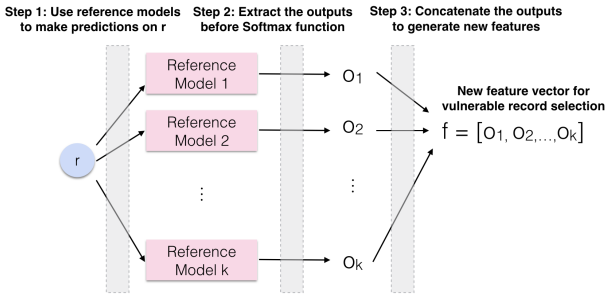


Figure 4: Generate new features for vulnerable record selection

reference datasets, where each dataset is sampled with replacement. Bootstrap sampling reduces overlaps among reference datasets, providing a better approximation of datasets sampled from distribution of the target training set. Each reference dataset is then used to train a reference model using the same training algorithms as used for training the target model.

4.3 Selecting Vulnerable Records

Not all training records are vulnerable to MIA. In an extreme case, if two records are nearly identical, it is difficult to discern which one of them is indeed present in the training dataset because their influence on the model is indistinguishable. In general, we want to measure the potential influence of a target record so as to select vulnerable records with the greatest influences and subject them to MIA in the subsequent steps. It is worth noting that, although the the training records imposing unique influence on the model are often *outlier records* (i.e., with distinct feature vectors) in the training set, the outlier records do not always have unique influence on the model because the training algorithm may decide that some features should be given higher weights than others and some features should be combined in the model. For example, a neural network trained on hand written digit datasets learns the contour of written digits is more important feature than individual pixels [25]. Therefore, instead of using the input features, we extract high level features more relevant to the classification task to detect vulnerable records.

Specifically, when attacking neural networks (e.g., see Figure 3 for a two-layer fully connected neural network), we construct new feature vectors by concatenating the outputs of the last layer before the Softmax function from the reference models (Figure 4), as the deeper layers in the network are more correlated with the classification output [16]. We then measure the unique influences of each record using its new feature vector. Let $\mathbf{f}$ be the the new feature vector of the record r. We call two records r_1 and r_2 *neighbors* if the cosine distance between their feature vectors $\mathbf{f}_1$ and $\mathbf{f}_2$ is smaller than a *neighbor-threshold* δ.

Note that the neighboring records are difficult to be distinguished by MIA because they have similar influence

on the model. When a neighbor of r occurs in the training dataset, the model may behave as if r is used to train the model, leading to the incorrect membership inference result. Our goal is to select the vulnerable records in the entire record space with fewer or no neighbors likely to be present in the training set (assuming the training records are independently drawn from the record space) as putative targets of MIA.

Given a training dataset with N records and a reference dataset with N' records, both sampled from the same record space, and a target record r, we count the number of neighbors of r in the reference dataset, denoted as N'_n. Then, the expected number of neighbors of r in the training dataset, N_n, can be estimated as $\mathbb{E}[N_n] = N'_n \times \frac{N'}{N}$.

A record r is considered to be potentially vulnerable (and as the attack object), only if $\mathbb{E}[N_n] < \beta$, where β is the *probability-threshold* for target record selection. We stress that the approach for vulnerable records selection presented here relies only on the record space (represented by the reference records accessible by an adversary) and the reference models (built using reference records), and is independent of the target model; as a result, the computation can be done off-line even when used to attack a machine learning as a service (MLaaS).

4.4 Direct Inference by Querying the Target Record

In well-generalized models, a single record's influence on the model's prediction is usually small and hard to detect. Moreover, the extent of this influence varies between records, so the approach in the prior MIA [32] no longer works. Instead, we attack each target record separately by computing the deviation between its output given by the target model and those given by the reference models. We expect that each training record has a unique influence on the model, which can be measured by comparing the target model's output with the output of reference models (trained without the target record) on the record. We quantify the difference between the outputs using the log loss function. Given a classifier M and a record r with class label y_r, let p_{y_r} be M's output probability of class label y_r. The log loss function [30] $\mathscr{L}(M,r)$ is defined as: $\mathscr{L}(M,r) = -\log p_{y_r}$. The log loss function is commonly used as a criterion function [30] when training neural network models. $\mathscr{L}(M,r)$ is small when M gives high probabilities on correct labels.

Given a target model M, a target record r, and k reference models, we first obtain the log loss of all the reference models on r as $L_1, L_2, \ldots, L_k$. We view these losses as samples independently drawn from a distribution $\mathscr{D}(L)$, and estimate the empirical cumulative distribution function (CDF) of $\mathscr{D}_L$ as $F(L)$, which takes a real-valued loss L as input. We use the shape-preserving piecewise cubic interpolation [34] to smooth the estimated CDF. Based on the log loss of the target model M on the target record r, $\mathscr{L}(M,r)$, we estimate the confidence of r to be present in the training set by performing a left-tailed hypothesis test: under the null hypothesis H_0, r is not present in the training set (i.e., $\mathscr{L}(M,r)$ is randomly drawn from $\mathscr{D}(L)$), while under the alternative hypothesis r is used to train M (i.e., $\mathscr{L}(M,r)$ is smaller than samples in $\mathscr{D}(L)$ because of the influence of r in the training). Therefore, we calculate the p-value as: $p = F(\mathscr{L}(M,r))$, which gives the confidence that r is used for training M only if p is smaller than a threshold (e.g. 0.01) so that the null hypothesis is rejected.

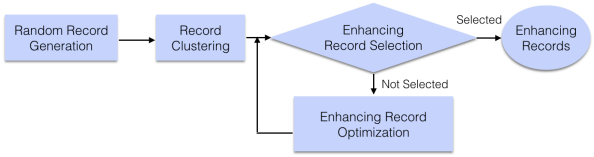


Figure 5: Steps for generating enhancing records.

4.5 Indirect Inference without Querying the Target Record

Besides reducing a model's loss on its own, a training record also influences the model's outputs on other records. This influence is desirable to improve model generalization: in order to give correct predictions on unseen records, a model needs to use the correlation it learns from a training record to make predictions on queries with similar features. On the other hand, however, these influences can be exploited by an adversary to obtain more information about the target record through multiple queries to enhance MIA. Interestingly, we show that MIA can be achieved by queries of records seemingly uncorrelated with the target record, making the attack hard to detect.

The key challenge for inference without querying the target record is to efficiently identify the *enhancing records* whose outputs from the target model are expected to be influenced by the target record. To address this problem, we develop a method consisting of the following steps: random record generation, record clustering, enhancing record selection, and enhancing records optimization (as shown in Figure 5).

Random Record Generation. To start with, we randomly generate records from which the enhancing records are selected. Specifically, we adopt one of the following two methods for random record generation: (1) when the feature space is relatively small, we uniformly sample records from the whole feature space; (2) when the feature space is large, since the chance of getting enhancing records by uniform sampling is slim, we generate random records by adding Gaussian noise to pre-selected vulnerable target records.

Enhancing Record Selection. To identify records whose target model's output may be influenced by the target record r, we approximate the target model's behavior using a group of *positive reference models* that are trained using reference records plus the target record r. To save

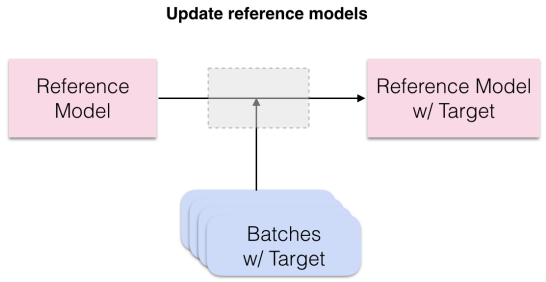


Figure 6: Building positive reference models by updating the model with the training set including the reference records plus the target record.

the effort of retraining the positive reference models, we add the target record into batches sampled from the original reference dataset and update the reference models by training on the batches plus the target record. Figure 6 shows the process of updating reference models.

We select the *enhancing records* by comparing the predictions between the positive reference models (i.e., "in models") and the original reference models (that are trained without the target records, i.e., "out models"). We denote the ith original and the ith positive reference model as M_{ref_i} and $M^r_{\text{ref}_i}$, respectively. Given a record r with class label y_r and another arbitrary record q, let $M(q, y_r)$ be the model M's output probability of y_r on the query q. We calculate r's influence on q as follows:

$$I(r,q) = \frac{1}{k}\sum_{i=1}^{k} \mathrm{t}\left(M^r_{\text{ref}_i}(q,y_r) - M_{\text{ref}_i}(q,y_r)\right) , \quad (1)$$

where k is the total number of original (or positive) reference models, and t is a threshold function defined as follows:

$$\mathrm{t}(x) = \begin{cases} 1 & \text{if} \quad x > 0, \\ 0 & \text{otherwise.} \end{cases}$$

Algorithm 1 Enhancing Records Selection Algorithm

```
procedure select_θ(q)                          ▷ Input a random query
    I(r,q) ← Σ_{i=1}^{k} t( M^r_{ref_i}(q,y_r) − M_{ref_i}(q,y_r) ) / k
    if I > θ then
        Accept q                               ▷ Use q in MIA
    else
        Reject q
```

We identify a randomly generated record q is an enhancing record for the record r if $I(r,q)$ approaches 1, which indicates that adding r to the training dataset *almost always* increase the models' output probability on the class label y_r for the query q. In practice, we use q in the MIA on the target record r only if $I(r,q)$ is greater than a threshold θ (e.g. 0.95). Algorithm 1 summarizes the entire algorithm for query selection.

Enhancing Record Optimization. When the target model has a large record space (e.g., with high-dimension feature vectors), the chance of finding an enhancing record among randomly generated records is slim. To address this issue, we propose an algorithm to search for enhancing records for a target record r by optimizing the following objective function:

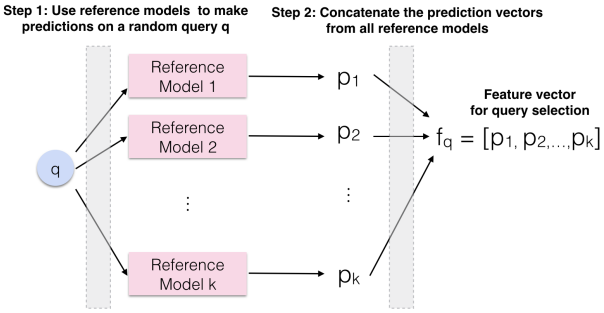


Figure 7: Generate query features for query selection.

$$\max_q I(r,q), \quad (2)$$

where $I(r,q)$ is the influence function defined in Equation 1. Optimizing $I(r,q)$ is time-consuming because $I(r,q)$ consists of a non-differentiable threshold function t. Therefore, instead of solving the optimization function in equation 2, For simplification, we approximate the maximization of $I(r,q)$ with the minimization of the sum of multiple hinge loss functions defined as follows [15]:

$$\min_q \sum_{i=1}^{k} \max\left(0, \gamma - \left(M^r_{\text{ref}_i}(q,y_r) - M_{\text{ref}_i}(q,y_r)\right)\right), \quad (3)$$

where γ is a parameter indicating the margin width. If a randomly generated record are rejected by the query selection algorithm, we minimize the objective function in Equation 3 using gradient descent [9] to check if the resulting record is acceptable as an enhancing record.

Record Clustering (Optional). Note that it is inefficient to repeat the query selection and optimization algorithms on all random records because the predictions of the models on most records are highly correlated: the models giving high output probabilities on some record are also likely to give high output probabilities on correlated records. To improve the efficiency of query selection, we propose an algorithm to identify the *least correlated* enhancing records from a large number of randomly generated records.

First, we estimate the correlation between records based on the model's predictions on them. We construct a feature vector $\mathbf{f}_q$ for a record q by concatenating the reference models' outputs on it (Figure 7). If two queries q_1 and q_2 have highly correlated feature vectors, the models' outputs on q_2 do not add much information to the models' outputs on q_1.

Next, we formulate the problem of selecting a subset of least correlated records as a graph theoretical problem. We build a graph where records are the nodes and pairwise correlation between records is the weight on edges

connecting the corresponding nodes. This allows us to recast our problem as the k-lightest subgraph problem [37], which is NP-hard. We obtain an approximate solution using hierarchical clustering [20]. For this, we cluster the records into k disjoint clusters based on their pairwise cosine distance. Finally, in each cluster, we select the record with least average cosine distance to all other records in the same cluster.

As shown in Figure 5, we use the enhancing record clustering algorithm before the enhancing record selection and enhancing record optimization steps to improve the efficiency of the attack.

Indirect Inference with Multiple Queries. After identifying multiple enhancing records, we repeat the attack in section 4.4 by querying each of these records. Because the outputs on these queries may be correlated, we combine the resulting p-values using Kost's method [23], with the covariance matrix estimated from the query features generated in the query selection step (Figure 7).

5 Evaluation

5.1 Experimental Setup

We evaluated two aspects of the performance of our attack: (1) *How many target records are considered to be vulnerable according to the GMIA selection criterion?* and (2) *How likely are vulnerable records to be inferred by GMIA when they are in the training dataset?*

To answer the first question, we ran the GMIA vulnerable record selection algorithm over all the target records. We compared the number of selected vulnerable records across different datasets, varying neighbor threshold δ, and probability threshold β. We evaluated the performance of GMIA over the selected vulnerable target records instead of the whole dataset since, in real attacks, adversary is likely to choose a few vulnerable targets instead of attacking all individuals.

To answer the second question, we evaluated the performance of the attack over multiple models. We constructed 100 target models, half of which are trained with the target record. To guarantee that each target record occurred in exactly 50 out of 100 target models, we generated training datasets by randomly splitting the target records into two datasets of the same size, each serving as a training set for a target model. We repeated this process for 50 times and generated the training datasets for 100 target models.

For each vulnerable target record, we performed GMIA on all the target models, and calculated for what percentage of models it can be correctly identified. When there are multiple vulnerable target records, we repeated the attack on every vulnerable target record over all the target models. An inference takes place only if the adversary has high confidence in the success of the attack (e.g. $p < 0.01$). The *precision* of the attack is defined as the percentage of successful inferences (i.e., the target record is indeed in the training dataset) among all inferences. The *recall* of the attack is defined as the percentage of successful inferences among all the cases that the target record is in the training set (i.e. $50n$). It indicates the likelihood that the membership of a vulnerable target record can be inferred. We define *true positive (TP)* to be the case that the target record is indeed in the training dataset when the adversary inferred it as in and *false positive (FP)* to be the case that the target record is *not* in the training dataset when the adversary inferred it as in.

5.2 Dataset

UCI Adult. The UCI Adult dataset [26] is a census dataset containing 48,842 records and 14 attributes. The attributes are demographic features and the classification task is to predict whether an individual's salary is above \$50K a year. We normalized the numerical attributes in the dataset and used one hop encoding [38] to construct the binary representation of categorical features. We randomly selected 20,000 records for training target models, and each training dataset contains 10,000 records. The remaining 28,842 records served as the adversary's background knowledge.

UCI Cancer. The UCI cancer dataset [26] contains 699 records and 10 numerical features ranging between 1 to 10. The features are characteristics of the cell in an image of a fine needle aspirate (FNA) of a breast mass. The classification task is to determine whether the cell is malignant or benign. We randomly selected 200 records for training, and each training dataset contains 100 records. The remaining 499 records served as the adversary's background knowledge.

MNIST Dataset. The MNIST dataset [24] is an image dataset of handwritten digits. The classification task is to predict which digit is represented in an image. We randomly selected 20,000 images for training and 40,000 images as the adversary's background knowledge. Each training set for target models and reference models contains 10,000 images.

5.3 Models

Neural Network. For the Adult dataset, we constructed a fully connected neural network with 2 hidden layers with 10 units and 5 units respectively. We use `Tanh` as the activation function and `SoftMax` as the output layer. The model is trained with batchsize of 100 and 20,000 epochs. For the MNIST dataset, we constructed 2 convolutional layers with ReLu as the activation function, followed with max pooling layers. We then added a fully connected layer of 1,024 neurons, and we also used dropout techniques to reduce overfitting. Finally, we added an output layer and a Softmax layer. The model is trained with batchsize of 50 and 10,000 epochs. For the

Table 1: GMIA by Direct Inference

	p-value	precision	recall	TP	FP
Adult (13 records)	0.001	-	0	0	0
	0.002	1	0.31%	2	0
	0.01	73.91%	5.23%	34	12
Cancer (5 records)	0.001	-	0	0	0
	0.008	1	2.4%	6	0
	0.01	88.89%	3.2%	8	1
MNIST (16 records)	0.0001	1	24.37%	195	0
	0.001	96.55%	45.50%	364	13
	0.01	93.36%	73.88%	591	42
Adult(Google) (7 records)	0.001	-	0	0	0
	0.009	1	2.33%	7	0
	0.01	80%	2.67%	8	2
MNIST(Google) (1 record)	0.001	-	0	0	0
	0.01	1	4%	2	0
	0.014	1	8%	4	0

Cancer dataset, we used a vanilla neural network with no hidden layer. The model is trained with batchsize of 10 and 3,000 epochs.

Google ML Engine. Since the Google Predictions API [2] used in the prior attack is deprecated, we used Google ML Engine [1] to train target models on ML cloud. When training the model, we used the samples provided by Google, which has pre-built model structures for training models on Adult dataset and MNIST dataset. Specifically, for Adult dataset, the sample code uses Google estimator [3] which hides low-level model structure from the user; for MNIST dataset, the sample code builds a neural network with 2 fully-connected hidden layers.

5.4 Direct Inference

In our first attack, we inferred the membership of vulnerable target records from the target models' predictions on these records. Based on the vulnerable target record selection algorithm in Section 4, using a probability threshold $\beta = 0.1$ (i.e. the likelihood that a target record's neighbor occurs in the training dataset of the target model is smaller than 0.1) , we selected 13 (out of 20,000) target records in the Adult dataset, 5 (out of 200) target records in the Cancer dataset, and 16 (out of 20,000) target records in the MNIST dataset. The neighborhood threshold δ used for these three datasets are 0.4, 0.1, and 0.2 respectively. We discuss the influence of these parameters in Section 5.5. For models trained on Google ML engine, we selected 7 (out of 20,000) target records for Adult(Google) and 1 target record for MNIST(Google). We performed GMIA on each of the selected target record and on all 100 target models.

Figure 8 shows the precision-recall curve of GMIA by querying the target record. Table 1 reflects the attack performance under different cut-off p-values. The recall reflects the likelihood that the membership of selected target records will be identified. When using 0.01 cut-off threshold for p-values, an adversary can attack with 73.91% precision on the Adult dataset, 88.89% precision on the Cancer dataset, and 93.36% precision on MNIST. All the target models we successfully attacked are well-generalized with difference between training and testing

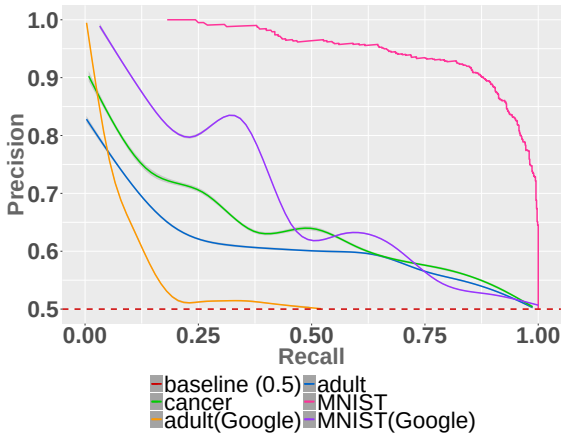


Figure 8: GMIA by Direct Inference

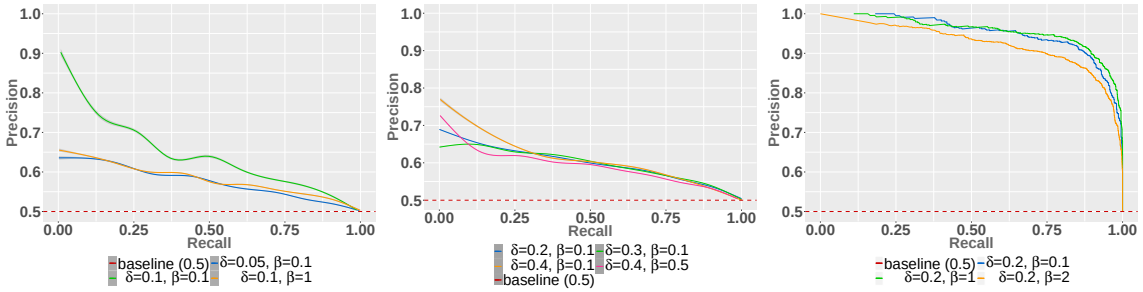
accuracy below 0.01(Table 6 in the Appendix). In comparison, the prior MIA [32] has low precision ($< 70\%$) on the same models and the same target records as shown in Table 7 in the Appendix.

Our attack had better performance on the local MNIST model compared to the Google ML ones because the CNN we constructed locally was more complex. Note that our local CNN improved upon the model on Google ML engine in testing accuracy by 8%, indicating an increase in model utility. However, the privacy risk also increased significantly. When $p < 0.01$, the attack recall increased by more than 70%. This result indicates the high privacy risk of applying complex models even when these models are not overfitted.

Our vulnerable record selection mechanism was less effective on the Adult Google ML model since we did not have access to the exact model structure due to the use of Google estimator. Instead, we used raw features to select target records. This limitation reduced the number of vulnerable target records we identified from 13 to 7.

5.5 Influence of Vulnerable Target Record Selection

Before launching the attack, we selected vulnerable target records by finding out records with unique high level feature vectors. This selection process helps reducing the incorrect inference caused by similar records in the training dataset. The selection criterion depends on two parameters: the neighbor threshold δ, which determines the criterion of neighbors, and the probability threshold β, which indicates how likely a neighbor is to occur in the training dataset. We studied selected vulnerable target records under different thresholds. Table 2 and Figure 9 shows performance of GIA w.r.t. varying target record selection threshold. Smaller neighbor thresholds or higher probability thresholds increased the number of selected vulnerable target records. However, as we tried to attack more records at the same time, there was a higher chance that we would make false positive inferences due to the influence of a record similar to one of the target records, which decreased the attack precision. Moreover, the recall



(a) Cancer Dataset **(b)** Adult Dataset **(c)** MNIST Dataset

Figure 9: Effect of different target selection threshold on attack performances

Table 2: GMIA w.r.t. Target Record Selection ($p = 0.01$)

	δ	β	# of Targets	precision	recall
Adult	0.4	0.1	13	73.91%	5.23%
	0.4	0.5	26	68%	3.92%
	0.3	0.1	53	65.81%	2.91%
	0.2	0.1	127	66.21%	2.19%
Cancer	0.1	0.1	5	88.89%	3.2%
	0.1	1	21	68.75%	3.14%
	0.05	0.1	33	66.67%	2.6%
MNIST	0.2	0.1	16	93.36%	73.88%
	0.2	1	27	95.05%	66.89%
	0.2	2	52	90.84%	68.31%

of the attack also decreased since we included records with weaker influence on the model as vulnerable target records.

5.6 Indirect Inference

For some vulnerable target records, we achieved the same level of attack performance by querying enhancing records. For each dataset, we randomly sampled 5,000 records, selected 50 of them by record clustering, and tested them with the enhancing record selection algorithm [6]. If less than 10 enhancing records were selected, we ran the enhancing record optimization algorithm to improve the records. The initial records for the Cancer dataset and the Adult dataset were randomly sampled from the feature space while the records for the MNIST dataset were generated by adding noise to the target records due to the large feature space.

We selected 1 target record in each dataset. For the Cancer dataset, we selected 47 enhancing records whose euclidean distance to the target record range between 6 and 19.3 with a selection criterion $I(r,q) > 0.95$. Since the Cancer dataset has relatively low dimensional features, enough enhancing records were accepted, and enhancing record optimization was not needed. For the Adult dataset, we relaxed the enhancing record selection criterion to $I(r,q) > 0.9$ and found 15 enhancing records after the optimization step. For the MNIST dataset, we further relaxed the enhancing record criterion to $I(r,q) > 0.8$ due to the high dimensional feature space. We identified 41 enhancing records generated by adding noise to the target record.

Table 3 and Figure 10 show the performance of indirect inferences. For both the Cancer dataset and the Adult dataset, attacking with the enhancing records has compatible performance as querying the target record. Moreover, for the Adult dataset, querying the target record did not

Table 3: Comparison between Direct and Indirect Inferences

Dataset	p-value	prec. (direct)	recall (direct)	prec. (indirect)	recall (indirect)
Adult	0.01	-	0	1	14%
	0.1	70.83%	34%	75%	24%
Cancer	0.01	1	6%	-	0
	0.1	66.67%	52%	88.89%	16%
MNIST	0.01	96.15%	1	1	2%
	0.1	89.29%	1	52.38%	22%

successfully infer any cases with a 0.01 cut-off p-value, but by combining the predictions on enhancing records, we achieved a precision of 1 and a recall of 14%. For the MNIST dataset, we achieved a precision of 1 and a recall of 2% when $p \leq 0.01$. Although this performance is less impressive compared to a direct inference on the same record (whose precision and recall are both close to 1) it's still an indication that membership inference attack can succeed without querying the target record. Moreover, we plotted both the target record and the enhancing records and found that the enhancing records in no means represent the target record, indicating that GMIA is hard to detect (Figure 13 in the Appendix).

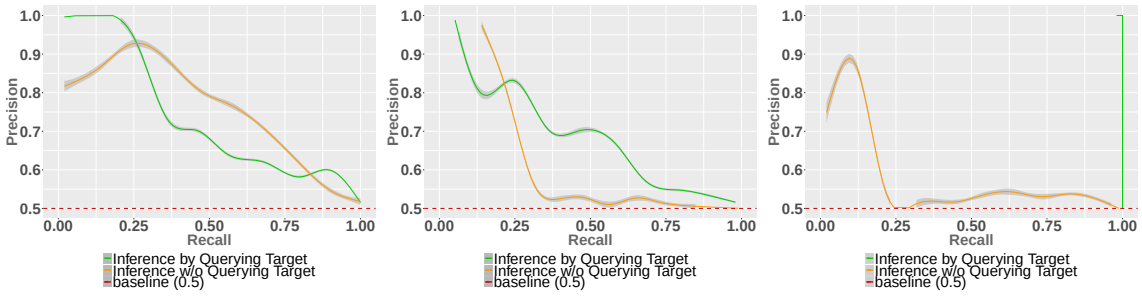
5.7 Influence of Training Epochs

In machine learning, one way of preventing overfitting is to stop training the model as soon as the testing accuracy stops increasing [8]. This method is called "early stop". To study the influence of maximum training epochs on GMIA, we trained neural networks on MNIST dataset with 1k maximum training epochs. Unlike "early stop" method, which stops the training process after testing accuracy stops increasing, we stopped training the models *before* the testing accuracy stopped increasing and performed the attack on potentially underfitted models. Table 4 shows the training and testing accuracy of the models

Table 4: GMIA w.r.t. Training Epochs ($p = 0.01$)

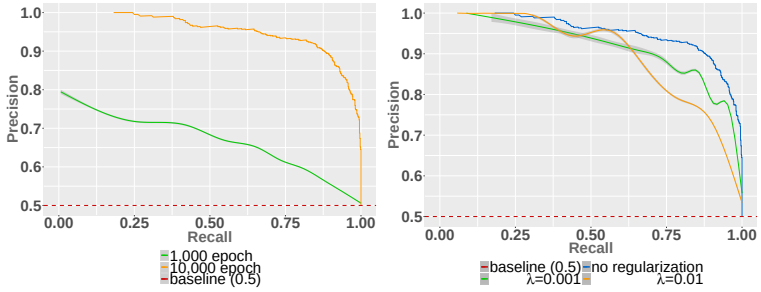
Training Epoch	Training Acc.	Test Acc.	δ	β	# of Targets	prec.	recall
1,000	0.97	0.96	0.2	0.1	28	72.27%	6.14%
			0.3	0.1	4	1	2.5%
10,000	0.99	0.98	0.2	0.1	16	93.36%	73.88%

Figure 11a shows the GMIA performance on models trained with 1k epochs and 10k epochs respectively. Reducing the training epoch did not eliminate membership privacy risk because a few records in the dataset were still identified with high precision. Specifically, when we increased the neighborhood threshold δ from 0.2 to



(a) Cancer Dataset **(b)** Adult Database **(c)** MNIST Dataset

Figure 10: Attack without Querying the Target Record



(a) GMIA Performance w.r.t Training epoch **(b)** GMIA Performance w.r.t. Regularization

Figure 11: Influence of training epochs and regularization

0.3, the 4 vulnerable target records were identified with a precision of 1 and a recall of 2.5%.

Moreover, as we increased the maximum training epoch, the model's testing accuracy increased, indicating an improvement in model generalization and model utility. However, this small improvement in model utility came at a huge cost for privacy—it increased the attack precision from 72.27% to 93.36% and recall from 6.14% to 73.88%.

5.8 Influence of Regularization

Regularization is a common method for improving model generalization. It is shown to be an effective defense against the prior MIA [32]. To study its effectiveness on GMIA, we applied L2 regularization on neural networks trained on MNIST set even though the models were *not overfitted*. In doing so, we limited the model capacity which increased the risk of underfitting. Specifically, when the regularization coefficient λ went from 0.001 to 0.01, testing accuracy decreased by 0.01 indicating that the model might be underfitted due to over regularization.

Table 5 and Figure 11b shows the model accuracy and GMIA performance before and after applying L2 regularization with varying coefficients λ. Applying regularization reduced the number of vulnerable target records in the dataset, but did not completely eliminate the privacy risk. The remaining vulnerable records were attacked with high precision. Specifically, when L2 regularization was applied with coefficient $\lambda = 0.01$, we still identified 1 vulnerable target record, which was inferred with precision close to 1 and a recall of 4%.

Like reducing training epoch, applying regularization mitigated the model's privacy risk but did not eliminate the risk. Moreover, since the most vulnerable record was identified with high precision, regularization may not be a good approach when the data owner wants to provide privacy protection for *all* individuals whose records are in the dataset.

Table 5: GMIA w.r.t. Regularization ($\delta = 0.2$, $\beta = 2$, $p = 0.01$)

Regularization Coefficient λ	Training Acc.	Test Acc.	# of Targets	prec.	recall
0	0.99	0.98	52	90.84%	68.31%
0.001	0.99	0.99	1	1	54.8%
0.01	0.98	0.98	1	93.36%	4%

6 Discussion

6.1 Understanding GMIA

Intuitions. As mentioned in Section 3, MIA can succeed by querying a record q if the target record has a unique influence on the predictions on q. Specifically, when we attack by querying the target record, the target record r is vulnerable to MIA when there is a non-overlapping area between the two distributions: the distribution of predictions on r when r is not used to train the model and the distribution of predictions on r when r is used. To verify our understanding, we plot the distribution of predictions on a vulnerable record r^* (Figure 13 in the Appendix) in the MNIST dataset. Firgure 14 in the Appendix shows this distribution. In section 5, the membership of r^* is inferred with a precision of 1 and a recall of 1 when directly querying r^*. This high vulnerability is explained by the fact that there is almost no overlapping between the distributions of predictions on r when r is included and not included in the training dataset.

Limitations. In the meantime, our current design of GMIA is preliminary. Our techniques for identifying outliers cannot find all vulnerable instances: it is possible that some instances not considered to be outliers

by our current design still exert unique influences on the model, which need to be better understood in the follow-up research. Moreover, the current way to search for the for the enhancing records, through filtering out random queries, is inefficient, and often does not produce any results. More effective solutions could utilize a targeted search based upon a better understanding about the relations between the target record and other records. Also in line with the prior research [32], we assume the adversary to either know the training algorithm or have black-box access to the training algorithm as an oracle. In practice, we may not be able to use the model identical to the target to train our references. Our preliminary study shows that it is still possible to attack some vulnerable instances in an online target model (though at lower success rate) using off-line models. How to make this more effective needs further investigation. Fundamentally, it remains unclear how much information about the training set is leaked out through querying a machine learning model and whether more sensitive techniques can be developed to capture even a small signals for a record's unique impact.

6.2 Mitigation

Generalization and perturbation. As mentioned earlier, generalization has limited effect on mitigating GMIA: as demonstrated in our study, even after applying the L2 regularization (with a coefficient of 0.01), still a vulnerable record in MNIST dataset can be attacked with a precision of 1 (Section 5.8). In the meantime, adding noise to the training set or to the model to achieve differential privacy can suppress the information leak [11]. However, in the presence of high-dimensional data, which is particularly vulnerable to our attack, perturbation significantly undermines the utility of the model before its privacy risk can be effectively controlled [21]. As an example, a recent study reports that a differentially-private stochastic gradient descent (SGD) only has an accuracy of 0.6 with $\varepsilon = 1$ and an accuracy of 0.5 with $\varepsilon = 0.5$ [29] on the MNIST dataset. So we believe that a practical solution should apply generalization and perturbation together with proper training set selection, detecting and removing those vulnerable training instances.

Training record selection. We believe that there is a fundamental contention between selecting useful training instances, which bring in additional information, and suppressing their unique influence to protect their privacy. An important step we could take here is to automatically identify outliers and drop those not contributing much to the utility of the model. To this end, new techniques need to be developed to balance the risk mitigation and the utility reduction for those risky instances. A machine learning model could be built to automatically decide whether an instance should be in the training set or not.

7 Related Work

Attacks on Machine Learning Models. Different attacks against machine learning models have been proposed in recent years. For example, reverse engineering attacks [17, 36] steal model parameters and structures; adversarial learning [13, 22, 27, 35] generates misleading examples that will be misclassified by the model; model inversion attacks [10, 14] infer the features of a record based on the model's predictions on it; membership inference attacks [32] infer the presence of a record in the model's training dataset.

Privacy and Model Generalization. There is a connection between privacy and model generalization. Differential privacy can improve model generalization when data is reused for validation [11]. Moreover, the prior membership inference attack [32] achieves high accuracy on highly overfitted models while barely works on non-overfitted ones. Previous research also points out that privacy leakage can happen on non-overfitted models *when the adversary has control over the training algorithm*. Specifically, the adversary can encode private information of the training dataset into the predictions of well-generalized models [33]. These two attacks [32, 33] can be formalized under a uniform theoretical framework [39]. The risk of membership inferences can be empirically measured based on the influence of each training record [28].

Privacy-Preserving Machine Learning Differential privacy [11] is a prominent way to formalize privacy against membership inference. It has been applied to various machine learning models including decision trees [19], logistic regression [41], and neural networks [4, 31]. However, there are no generic methods to achieve differential privacy for all useful machine learning models. More importantly, even if these methods are developed, their applications to real-world machine learning problems may significantly decrease the accuracy of the models, and thus will reduce their utility [5].

8 Conclusion

In this paper, we take a step forward on understanding the information leaks from machine learning models. Our study demonstrates that overfitting contributes to the information leaks but is *not* the fundamental cause of the problem. This understanding is achieved through a series of membership inference attacks on well-generalized models, discovering vulnerable instances (cancer patients, images, and individual data) even without directly querying the vulnerable target records, and even in the presence of regularization protection. Our study highlights the contention between selecting informative training instances and preventing their identification through their unique influences on the model, and points to the direction of using training data analysis and selection to complement existing approaches.

References

[1] Google cloud machine learning engine. https://cloud.google.com/ml-engine/reference/rest/.

[2] Google prediction api. https://developers.google.com/prediction/.

[3] ABADI, M., AGARWAL, A., BARHAM, P., BREVDO, E., CHEN, Z., CITRO, C., CORRADO, G. S., DAVIS, A., DEAN, J., DEVIN, M., ET AL. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467* (2016).

[4] ABADI, M., CHU, A., GOODFELLOW, I., MCMAHAN, H. B., MIRONOV, I., TALWAR, K., AND ZHANG, L. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security* (2016), ACM, pp. 308–318.

[5] ALVIM, M. S., ANDRÉS, M. E., CHATZIKOKOLAKIS, K., DEGANO, P., AND PALAMIDESSI, C. Differential privacy: On the trade-off between utility and information leakage. *Formal Aspects in Security and Trust 7140* (2011), 39–54.

[6] BERKHIN, P., ET AL. A survey of clustering data mining techniques. *Grouping multidimensional data 25* (2006), 71.

[7] BOUSQUET, O., AND ELISSEEFF, A. Stability and generalization. *Journal of Machine Learning Research 2*, Mar (2002), 499–526.

[8] CARUANA, R., LAWRENCE, S., AND GILES, C. L. Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping. In *Advances in neural information processing systems* (2001), pp. 402–408.

[9] CAUCHY, A. Méthode générale pour la résolution des systemes déquations simultanées. *Comp. Rend. Sci. Paris 25*, 1847 (1847), 536–538.

[10] CORMODE, G. Personal privacy vs population privacy: learning to attack anonymization. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining* (2011), ACM, pp. 1253–1261.

[11] DWORK, C. Differential privacy in the 40th international colloquium on automata. *Languages and Programming* (2006).

[12] EFRON, B. *The jackknife, the bootstrap and other resampling plans*. SIAM, 1982.

[13] EVTIMOV, I., EYKHOLT, K., FERNANDES, E., KOHNO, T., LI, B., PRAKASH, A., RAHMATI, A., AND SONG, D. Robust physical-world attacks on deep learning models.

[14] FREDRIKSON, M., LANTZ, E., JHA, S., LIN, S., PAGE, D., AND RISTENPART, T. Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing. In *23rd USENIX Security Symposium (USENIX Security 14)* (2014), pp. 17–32.

[15] GENTILE, C., AND WARMUTH, M. K. Linear hinge loss and average margin. In *Advances in Neural Information Processing Systems* (1999), pp. 225–231.

[16] HINTON, G. E., OSINDERO, S., AND TEH, Y.-W. A fast learning algorithm for deep belief nets. *Neural computation 18*, 7 (2006), 1527–1554.

[17] HITAJ, B., ATENIESE, G., AND PEREZ-CRUZ, F. Deep models under the gan: Information leakage from collaborative deep learning. *arXiv preprint arXiv:1702.07464* (2017).

[18] HOMER, N., SZELINGER, S., REDMAN, M., DUGGAN, D., TEMBE, W., MUEHLING, J., PEARSON, J. V., STEPHAN, D. A., NELSON, S. F., AND CRAIG, D. W. Resolving individuals contributing trace amounts of dna to highly complex mixtures using high-density snp genotyping microarrays. *PLoS genetics 4*, 8 (2008), e1000167.

[19] JAGANNATHAN, G., PILLAIPAKKAMNATT, K., AND WRIGHT, R. N. A practical differentially private random decision tree classifier. In *Data Mining Workshops, 2009. ICDMW09. IEEE International Conference on* (2009), IEEE, pp. 114–121.

[20] JAIN, A. K., AND DUBES, R. C. *Algorithms for clustering data*. Prentice-Hall, Inc., 1988.

[21] JOHNSON, A., AND SHMATIKOV, V. Privacy-preserving data exploration in genome-wide association studies. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining* (2013), ACM, pp. 1079–1087.

[22] KOS, J., FISCHER, I., AND SONG, D. Adversarial examples for generative models. *arXiv preprint arXiv:1702.06832* (2017).

[23] KOST, J. T., AND MCDERMOTT, M. P. Combining dependent p-values. *Statistics & Probability Letters 60*, 2 (2002), 183–190.

[24] LECUN, Y., CORTES, C., AND BURGES, C. J. Mnist handwritten digit database. *AT&T Labs [Online]. Available: http://yann. lecun. com/exdb/mnist 2* (2010).

[25] LEE, H., GROSSE, R., RANGANATH, R., AND NG, A. Y. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *Proceedings of the 26th annual international conference on machine learning* (2009), ACM, pp. 609–616.

[26] LICHMAN, M. UCI machine learning repository, 2013.

[27] LIU, Y., CHEN, X., LIU, C., AND SONG, D. Delving into transferable adversarial examples and black-box attacks. *arXiv preprint arXiv:1611.02770* (2016).

[28] LONG, Y., BINDSCHAEDLER, V., AND GUNTER, C. A. Towards measuring membership privacy. *arXiv preprint arXiv:1712.09136* (2017).

[29] MCMAHAN, H. B., RAMAGE, D., TALWAR, K., AND ZHANG, L. Learning differentially private language models without losing accuracy. *arXiv preprint arXiv:1710.06963* (2017).

[30] MURATA, N., YOSHIZAWA, S., AND AMARI, S.-I. Network information criterion-determining the number of hidden units for an artificial neural network model. *IEEE Transactions on Neural Networks 5*, 6 (1994), 865–872.

[31] SHOKRI, R., AND SHMATIKOV, V. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security* (2015), ACM, pp. 1310–1321.

[32] SHOKRI, R., STRONATI, M., AND SHMATIKOV, V. Membership inference attacks against machine learning models. *arXiv preprint arXiv:1610.05820* (2016).

[33] SONG, C., RISTENPART, T., AND SHMATIKOV, V. Machine learning models that remember too much. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security* (2017), ACM, pp. 587–601.

[34] SPRAGUE, T. B. Shape preserving piecewise cubic interpolation.

[35] SZEGEDY, C., ZAREMBA, W., SUTSKEVER, I., BRUNA, J., ERHAN, D., GOODFELLOW, I., AND FERGUS, R. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199* (2013).

[36] TRAMÈR, F., ZHANG, F., JUELS, A., REITER, M. K., AND RISTENPART, T. Stealing machine learning models via prediction apis. *arXiv preprint arXiv:1609.02943* (2016).

[37] WATRIGANT, R., BOUGERET, M., AND GIROUDEAU, R. The k-sparsest subgraph problem.

[38] WU, Y., PADHYE, J., CHANDRA, R., PADMANABHAN, V., AND CHOU, P. A. The local mixing problem. In *Proc. Information Theory and Applications Workshop* (2006).

[39] YEOM, S., FREDRIKSON, M., AND JHA, S. The unintended consequences of overfitting: Training data inference attacks. *arXiv preprint arXiv:1709.01604* (2017).

[40] ZERHOUNI, E. A., AND NABEL, E. G. Protecting aggregate genomic data. *Science 322*, 5898 (2008), 44–44.

[41] ZHANG, J., ZHANG, Z., XIAO, X., YANG, Y., AND WINSLETT, M. Functional mechanism: regression analysis under differential privacy. *Proceedings of the VLDB Endowment 5*, 11 (2012), 1364–1375.

Appendix

Training and Testing Accuracy of Target Models. Table 6 shows the training and testing accuracy of the target models in our attacks. All the target models are well-generalized models with difference between training and testing accuracy smaller than 0.1.

Table 6: Training and Testing Accuracy of Target Models

Dataset (Model)	Training Accuracy	Test Accuracy
Adult	0.85 ± 0.01	0.85
Cancer	0.95 ± 0.04	0.94 ± 0.03
MNIST	0.99	0.98
Adult(Google)	0.84 ± 0.03	0.84 ± 0.02
MNIST(Google)	0.90	0.90

Vulnerable Records in MNIST Dataset. To study what kinds of records are vulnerable to GMIA, we plotted the vulnerable target records selected from MNIST dataset with $\delta = 0.2$ and $\beta = 0.1$ (Figure 12). As we expected, some of the vulnerable target records are outliers in the dataset. However, some vulnerable examples actually increase model utility by providing rare but useful features for the classification task. For example, the images of digit 8 written in different directions may help a model on recognizing similar written digits in testing examples. However, since these images are rare in the dataset, they have a unique influence on the target models, making them vulnerable to GMIA, and the fact that this influence is useful in predicting unseen examples does not mitigate the risk.

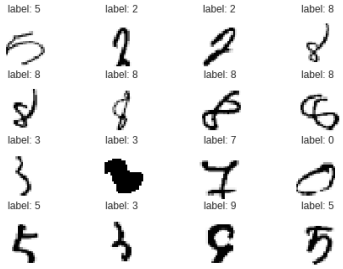


Figure 12: Vulnerable Examples in MNIST Dataset

Indirect Inference on MNIST Dataset. To study the correlation between a target record and its enhancing records in indirect inferences, we plotted the target record with its enhancing records in the MNIST dataset 13. Surprisingly, the enhancing records seem like images of random noise and by no means represent the target record.

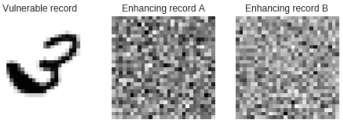


Figure 13: Vulnerable record from MNIST with its two enhancing records.

Intuitions on GMIA. Figure 14 shows a vulnerable record's influence on the machine learning model's predictions on itself. The image of the record is plotted in Figure 13. All the positive reference models (i.e., reference models trained with the target record) predict high

probability for the correct class label while all the reference models (i.e., models trained without the target record) predict low probability for the correct class label. This difference allows us to successfully infer the presence of this record in the training dataset.

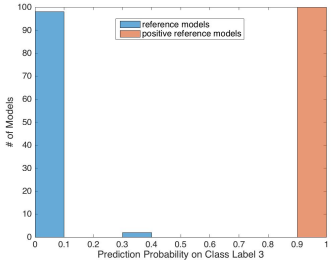


Figure 14: The histogram of predictions on r^* when r^* is in the training dataset (i.e., positive reference models) v.s. not in the training dataset (i.e., reference models)

Comparison with the Prior MIA. To compare with the prior MIA [32], we reproduced the attack in [32] on the same target models and same vulnerable records in GMIA. Specifically, we trained one attack classifier per class for each dataset. The attack classifiers are neural networks with one hidden layer of 64 units. We used `ReLU` as the activation function and `SoftMax` as the output layer. We only performed the attack when the probability given by the attack classifier was higher than a certain threshold (called attack confidence threshold). We evaluated the performance of the attack under various attack threshold as shown in Table 7. The attack precision was relatively low (e.g. $< 70\%$) on all three datasets even when a high attack confidence threshold was used.

Table 7: Performance of the Prior MIA on the Same Target Models

Dataset	Attack Confidence Threshold	Attack Precision	Attack Recall
Cancer (3 records)	0.8	50.25%	40%
	0.9	-	0
Adult (13 records)	0.6	66.67%	4.92%
	0.7	-	0
MNIST (16 records)	0.6	50%	56.25%
	0.7	19.6%	6.25%
	0.8	-	0