Membership Leakage in Label-Only Exposures

Zheng Li and Yang Zhang
CISPA Helmholtz Center for Information Security

ABSTRACT

Machine learning (ML) has been widely adopted in various privacy-critical applications, e.g., face recognition and medical image analysis. However, recent research has shown that ML models are vulnerable to attacks against their training data. Membership inference is one major attack in this domain: Given a data sample and model, an adversary aims to determine whether the sample is part of the model's training set. Existing membership inference attacks leverage the confidence scores returned by the model as their inputs (score-based attacks). However, these attacks can be easily mitigated if the model only exposes the predicted label, i.e., the final model decision.

In this paper, we propose decision-based membership inference attacks and demonstrate that label-only exposures are also vulnerable to membership leakage. In particular, we develop two types of decision-based attacks, namely transfer attack and boundary attack. Empirical evaluation shows that our decision-based attacks can achieve remarkable performance, and even outperform the previous score-based attacks in some cases. We further present new insights on the success of membership inference based on quantitative and qualitative analysis, i.e., member samples of a model are more distant to the model's decision boundary than non-member samples. Finally, we evaluate multiple defense mechanisms against our decision-based attacks and show that our two types of attacks can bypass most of these defenses.[1]

CCS CONCEPTS

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

KEYWORDS

machine learning, membership leakage, label-only

ACM Reference Format:
Zheng Li and Yang Zhang. 2021. Membership Leakage in Label-Only Exposures. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security (CCS '21), November 15–19, 2021, Virtual Event, Republic of Korea.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3460120.3484575

[1]Our code is available at https://github.com/zhenglisec/Decision-based-MIA.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
CCS '21, November 15–19, 2021, Virtual Event, Republic of Korea.
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-8454-4/21/11…$15.00
https://doi.org/10.1145/3460120.3484575

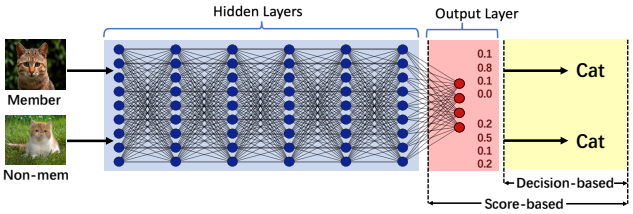


Figure 1: An illustration of accessible components of the target model for each of the two threat models. A score-based threat model assumes access to the output layer; a decision-based threat model assumes access to the predicted label alone.

1 INTRODUCTION

Machine learning (ML) has witnessed tremendous progress over the past decade and has been applied across a wide range of privacy-critical applications, such as face recognition [28, 61] and medical image analysis [9, 29, 51]. Such developments rely on not only novel training algorithms and architectures, but also access to sensitive and private data, such as health data. Various recent research [23, 25, 31, 35, 36, 45, 46, 48, 49, 54, 57, 60] has shown that ML models are vulnerable to privacy attacks. One major attack in this domain is membership inference: An adversary aims to determine whether or not a data sample is used to train a target ML model.

Existing membership inference attacks [25, 31, 35, 46, 48, 49, 57] rely on the confidence scores (e.g. class probabilities or logits) returned by a target ML model as their inputs. The success of membership inference is due to the inherent overfitting property of ML models, i.e., an ML model is more confident facing a data sample it was trained on, and this confidence is reflected in the model's output scores. See Figure 1 for an illustration of accessible components of an ML model for such score-based threat model. A major drawback for these score-based attacks is that they can be trivially mitigated if the model only exposes the predicted label, i.e., the final model decision, instead of confidence scores. The fact that score-based attacks can be easily averted makes it more difficult to evaluate whether a model is truly vulnerable to membership inference or not, which may lead to premature claims of privacy for ML models.

This motivates us to focus on a new category of membership inference attacks that has so far received fairly little attention, namely *Decision-based attacks*. Here, the adversary solely relies on the final decision of the target model, i.e., the top-1 predicted label, as their attack model's input. It is more realistic to evaluate the vulnerability of a machine learning system under the decision-based attacks with sole access to the model's final decision. First, compared to score-based attacks, decision-based attacks are much more relevant in real-world applications where confidence scores

are rarely accessible. Furthermore, decision-based attacks have the potential to be much more robust to the state-of-the-art defenses, such as confidence score perturbation [27, 38, 56]. In label-only exposure, a naive *baseline attack* [57] infers that a candidate sample is a member of a target model if it is predicted correctly by the model. However, this baseline attack cannot distinguish between members and non-members that are both correctly classified as shown in Figure 1.

In this paper, we propose two types of decision-based attacks under different scenarios, namely *transfer attack* and *boundary attack*. In the following, we abstractly introduce each of them.

Transfer Attack. We assume the adversary has an auxiliary dataset (namely shadow dataset) that comes from the same distribution as the target model's training set. The assumption also holds for previous score-based attacks [35, 46, 48, 49]. The adversary first queries the target model in a manner analog to cryptographic oracle, thereby relabeling the shadow dataset by the target model's predicted labels. Then, the adversary can use the relabeled shadow dataset to construct a local shadow model to mimic the behavior of the target model. In this way, the relabeled shadow dataset contains sufficient information from the target model, and membership information can also be transferred to the shadow model. Finally, the adversary can leverage the shadow model to launch a score-based membership inference attack locally.

Boundary Attack. Collecting data, especially sensitive and private data, is a non-trivial task. Thus, we consider a more difficult and realistic scenario in which there is no shadow dataset and shadow model available. To compensate for the lack of information in this scenario, we shift the focus from the target model's output to the input. Here, our key intuition is that it is harder to perturb member data samples to different classes than non-member data samples. The adversary queries the target model on candidate data samples, and perturb them to change the model's predicted labels. Then the adversary can exploit the magnitude of the perturbation to differentiate member and non-member data samples.

Extensive experimental evaluation shows that both of our attacks achieve strong performance. In particular, our boundary attack in some cases even outperforms the previous score-based attacks. This demonstrates the severe membership risks stemming from ML models. In addition, we present a new perspective on the success of current membership inference and show that the distance between a sample and an ML model's decision boundary is strongly correlated with the sample's membership status.

Finally, we evaluate our attacks on multiple defense mechanisms: generalization enhancement[46, 50, 54], privacy enhancement [4] and confidence score perturbation [27, 38, 56]. The results show that our attacks can bypass most of the defenses, unless heavy regularization is applied. However heavy regularization can lead to a significant degradation of the model accuracy.

In general, our contributions can be summarized as the following:

- We perform a systematic investigation on membership leakage in label-only exposures of ML models, and introduce decision-based membership inference attacks, which is highly relevant for real-world applications and important to gauge model privacy.
- We propose two types of decision-based attacks under different scenarios, including transfer attack and boundary attack. Extensive experiments demonstrate that our two types of attacks achieve better performances than the baseline attack, and even outperform the previous score-based attacks in some cases.
- We propose a new perspective on the reasons for the success of membership inference, and perform a quantitative and qualitative analysis to demonstrate that members of an ML model are more distant from the model's decision boundary than non-members.
- We evaluate multiple defenses against our decision-based attacks and show that our novel attacks can still achieve reasonable performance unless heavy regularization is applied.

Paper Organization. The rest of this paper is organized as follows. Section 2 presents the definitions of membership inference for the ML models, threat models, datasets, and model architectures used in this paper. Section 3 and Section 4 introduce our two attack methods and evaluation methods. In Section 5, we provide an in-depth analysis of the success of membership inference. Section 6 provides multiple defenses against decision-based attacks. Section 7 presents related work, and Section 8 concludes the paper.

2 PRELIMINARIES

2.1 Membership Leakage in Machine Learning Models

Membership leakage in ML models emerges when an adversary aims to determine whether a candidate data sample is used to train a certain ML model. More formally, given a candidate data sample x, a trained ML model $\mathcal{M}$, and external knowledge of an adversary, denoted by Ω, a membership inference attack $\mathcal{A}$ can be defined as the following function.

$$\mathcal{A} : x, \mathcal{M}, \Omega \rightarrow \{0, 1\}.$$

Here, 0 means x is not a member of $\mathcal{M}$'s training set and 1 otherwise. The attack model $\mathcal{A}$ is essentially a binary classifier. Depending on the assumptions, it can be constructed in different ways, which will be presented in later sections.

2.2 Threat Model

Here, we outline the threat models considered in this paper. The threat models are summarized in Table 1. There are two existing categories of attacks, i.e., score-based attacks and decision-based attacks. The general idea of score-based attacks is to exploit the detailed output (i.e., confidence score) of the target model to launch an attack. In decision-based attacks, an adversary cannot access to confidence scores, but relies on the final predictions of the target model launch an attack. The baseline attack predicts a data sample as a member of the training set when the model classifies it correctly. However, this naive and simple approach does not work at all in the case shown in Figure 1. In the following, we introduce the adversarial knowledge that an adversary has in our decision-based attacks.

Adversarial Knowledge. For our decision-based attacks, the adversary only has black-box access to the target model, i.e., they are

Table 1: An overview of membership inference threat models. "✓" means the adversary needs the knowledge and "-" indicates the knowledge is not necessary.

Attack Category	Attacks	Target Model's Structure	Training Data Distribution	Shadow Model	Detailed Model Prediction (e.g. probabilities or logits)	Final Model Prediction (e.g. max class label)
Score-based	[25, 31, 35, 46, 48, 49, 57]	✓or -	✓or -	✓or -	✓	✓
Decision-based	Baseline attack [57]	-	✓	-	-	✓
	Transfer attack	-	✓	✓	-	✓
	Boundary attack	-	-	-	-	✓

not able to extract a candidate data sample's membership status from the confidence scores. Concretely, our threat model comprises the following entities. (1) Final decision of the target model $\mathcal{M}$, i.e., predicted label. (2) A shadow dataset $\mathcal{D}_{shadow}$ drawn from the same distribution as target model's dataset $\mathcal{D}_{target}$. (3) A local shadow model $\mathcal{S}$ trained using the shadow dataset $\mathcal{D}_{shadow}$. For boundary attack, the adversary only has the knowledge of (1).

2.3 Datasets and Target Model Architecture

Datasets. We consider four benchmark datasets of different size and complexity, namely CIFAR-10 [1], CIFAR-100 [1], GTSRB [2], and Face [3], to conduct our experiments. Since the images in GTSRB are of different sizes, we resize them to 64×64 pixels. For the Face dataset, we only consider people with more than 40 images, which leaves us with 19 people's data, i.e., 19 classes. We describe them in detail in Appendix Section A.1.

Target Model Architecture. Typically, for image classification tasks, we use neural networks which is adopted across a wide of applications. In this work, we build the target model using 4 convolutional layers and 4 pooling layers with 2 hidden layers containing 256 units each at last. The target models are trained for 200 training epochs, iteratively using Adam algorithm with a batch-size of 128 and a fixed learning rate of 0.001.

3 TRANSFER ATTACK

In this section, we present the first type of decision-based attacks, i.e., transfer attack. We start by introducing our key intuition. Then, we describe the attack methodology. Finally, we present the evaluation results.

3.1 Key Intuition

The intuition of this attack is that the transferability property holds between shadow model $\mathcal{S}$ and target model $\mathcal{M}$. Almost all related works [15, 33, 37, 39] focus on the transferability of adversarial examples, i.e., adversarial examples can transfer between models trained for the same task. Unlike these works, we focus on the transferability of membership information for benign data samples, i.e., the member and non-member data samples behaving differently in $\mathcal{M}$ will also behave differently in $\mathcal{S}$. Then we can leverage the shadow model to launch a score-based membership inference attack.

3.2 Methodology

The transfer attack's methodology can be divided into four stages, namely shadow dataset relabeling, shadow model architecture selection, shadow model training, and membership inference. The algorithm can be found in Appendix algorithm 1.

Shadow Dataset Relabeling. As aforementioned, the adversary has a shadow dataset $\mathcal{D}_{shadow}$ drawn from the same distribution as the target model $\mathcal{M}$'s dataset $\mathcal{D}_{target}$. To train a shadow model, the first step is to relabel these data samples using the target model $\mathcal{M}$ as an oracle. In this way, the adversary can establish a connection between the shadow dataset and the target model, which facilitates the shadow model to be more similar to the target model in the next step.

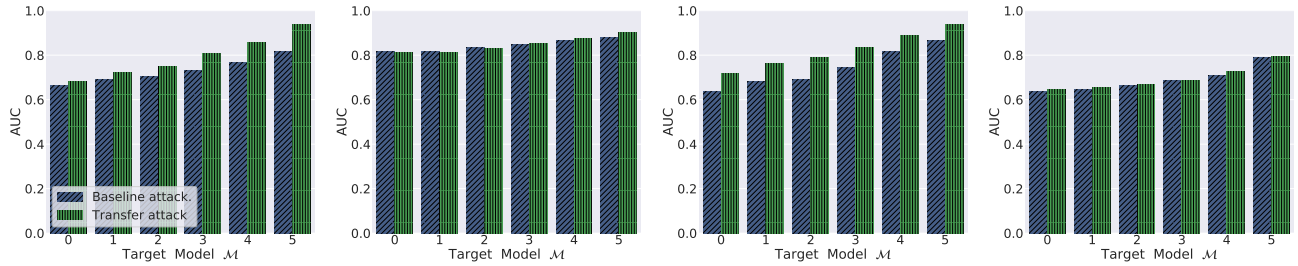
Shadow Model Architecture Selection. As the adversary knows the main task of the target model, it can build the shadow model using high-level knowledge of the classification task (e.g., convolutional networks are appropriate for vision). As in prior score-based attacks, we also use the same architecture of target models to build our shadow models. Note that we emphasize that the adversary does not have the knowledge of the concrete architecture of the target model, and in Section 3.4, we also show that a wide range of architecture choices yield similar attack performance.

Shadow Model Training. The adversary trains the shadow model $\mathcal{S}$ with the relabeled shadow dataset $\mathcal{D}_{shadow}$ in conjunction with classical training techniques.

Membership Inference. Finally, the adversary feeds a candidate data sample into the shadow model $\mathcal{S}$ to calculate its cross-entropy loss with the ground truth label.

$$\text{CELoss} = -\sum_{i=0}^{K} \mathbf{1}_y \log(p_i), \tag{1}$$

where $\mathbf{1}_y$ is the one-hot encoding of the ground truth label y, p_i is the probability that the candidate sample belongs to class i, and K is the number of classes. If the loss value is smaller than a threshold, the adversary then determines the sample being a member and vice versa. The adversary can pick a suitable threshold depending on their requirements, as in many machine learning applications. [7, 19, 22, 27, 43, 46]. In our evaluation, we mainly use area under the ROC curve (AUC) which is threshold independent as our evaluation metric.



(a) CIFAR-10 (b) CIFAR-100 (c) GTSRB (d) Face

Figure 2: Comparison of our transfer attack performance with the baseline attack by Yeom et al. [57]. The x-axis represents the target model being attacked and the y-axis represents the AUC score.

3.3 Experimental Setup

Following the attack strategy, we split each dataset into $\mathcal{D}_{target}$ and $\mathcal{D}_{shadow}$: One is used to train and test the target model, and the other is used to train the shadow model $\mathcal{S}$ after relabeled by the target model. For evaluation, $\mathcal{D}_{target}$ is also split into two: One is used to train the target model $\mathcal{M}$, i.e., $\mathcal{D}_{train}$, and serves as the member samples of the target model, while the other $\mathcal{D}_{test}$ serves as the non-member samples.

It is well known that the inherent overfitting drives ML models to be vulnerable to membership leakage [46, 48]. To show the variation of the attack performance on each dataset, we train 6 target models $\mathcal{M}$-0, $\mathcal{M}$-1, ..., $\mathcal{M}$-5 using different size of the training set $\mathcal{D}_{train}$, exactly as performed in the prior work by Shokri et al. [48] and many subsequent works [35, 46, 49, 54]. The sizes of $\mathcal{D}_{train}$, $\mathcal{D}_{test}$, and $\mathcal{D}_{shadow}$ are summarized in Appendix Table 7.

We execute the evaluation on randomly reshuffled data samples from $\mathcal{D}_{target}$, and select sets of the same size (i.e, equal number of members and non-members) to maximize the uncertainty of inference, thus the baseline performance is equivalent to random guessing. We adopt AUC as our evaluation metric which is threshold independent. In addition, we further discuss methods to pick threshold for our attack later in this section.

3.4 Results

Attack AUC Performance. Figure 2 depicts the performance of our transfer attack and baseline attack. First, we can observe that our transfer attack performs at least on-par with the baseline attack. More encouragingly, on the CIFAR-10 and GTSRB datasets, our transfer attack achieves better performance than the baseline attack. For example, in Figure 2 ($\mathcal{M}$-5, CIFAR-10), the AUC score of the transfer attack is 0.94, while that of the baseline attack is 0.815. The reason why our transfer attack outperforms the baseline attack on CIFAR-10 and GTSRB rather than on CIFAR-100 and Face, is that the size of the shadow dataset for the first two datasets is relatively larger than that of the latter two, compared to the size of each dataset (see Appendix Table 7). In the next experiments, we make the same observation that a larger shadow dataset implies better attack performance.

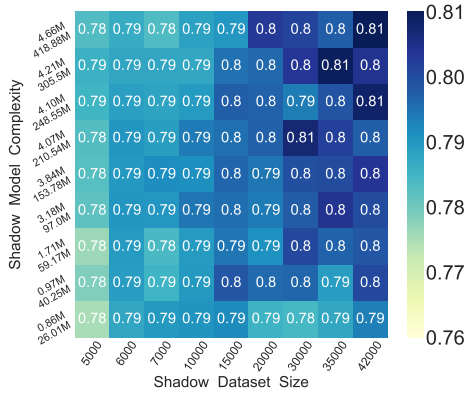
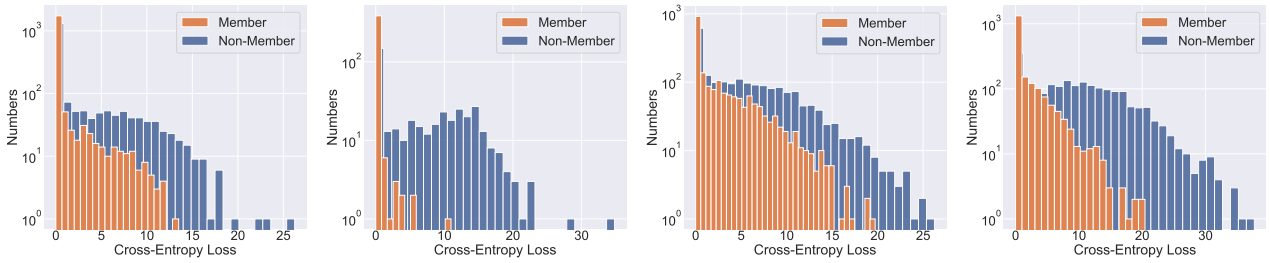


Figure 3: Attack AUC under the effect of changing the dataset size and shadow model complexity (upper is the number of parameters, lower is the computational complexity FLOPs). The target model ($\mathcal{M}$-0, CIFAR-100)'s training set size is 7,000, and complexity is 3.84M parameters and 153.78M FLOPs.

Effects of the Shadow Dataset and Model. We further investigate the effects of shadow dataset size and shadow model complexity (structure and hyper-parameter) on the attack performance. More concretely, for the target model ($\mathcal{M}$-0, CIFAR-100), we vary the size of the shadow dataset $\mathcal{D}_{shadow}$ from 5,000 to 42,000, where the target training set $\mathcal{D}_{train}$ is 7,000. We also vary the complexity of the shadow model from 0.86M (number of parameters) and 26.01M (FLOPs,[2] computational complexity) to 4.86M and 418.88M, where the complexity of the target model is 3.84M and 153.78M, respectively. We conduct extensive experiments to simultaneously tune these two hyper-parameters and report the results in Figure 3. Through investigation, we make the following observations.

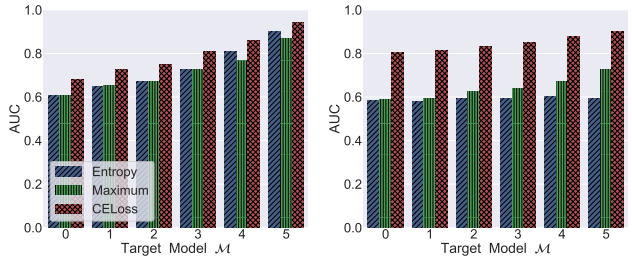
- Larger shadow dataset implies more queries to the target model which leads to better attack performance.
- Even simpler shadow models and fewer shadow datasets (bottom left part) can achieve strong attack performance.

[2]FLOPs represent the theoretical amount of floating-point arithmetic needed when feeding a sample into the model.



(a) CIFAR-10, $\mathcal{M}$-0 (b) CIFAR-10, $\mathcal{M}$-5 (c) CIFAR-100, $\mathcal{M}$-0 (d) CIFAR-100, $\mathcal{M}$-5

Figure 4: The cross entropy loss distribution obtained from the shadow model. The x-axis represents the loss value and the y-axis represents the number of the loss.



(a) CIFAR-10 (b) CIFAR-100

Figure 5: Attack AUC for three different statistical measures. The x-axis represents the target model being attacked and the y-axis represents the AUC score.

- In general, the transfer attack is robust even if the shadow model is much different from the target model.

Effects of Statistical Metrics. As prior works [46, 48] also use other statistical metrics, i.e., maximum confidence scores $Max(p_i)$ and normalized entropy $\frac{-1}{\log(K)} \sum_i p_i \log(p_i)$. Here, we also conduct experiments with these statistical metrics. Figure 5 reports the AUC on the CIFAR-10 and CIFAR-100 datasets. We can observe that the loss metric achieves the highest performance with respect to the different target models. Meanwhile, the AUC score is very close between maximum confidence score and entropy. This indicates that the loss metric contains the strongest signal on differentiating member and non-member samples. We will give an in-depth discussion on this in Section 5.2.

Loss Distribution of Membership. To explain why our transfer attack works, Figure 4 further shows the loss distribution of member and non-member samples from the target model calculated on the shadow model ($\mathcal{M}$-0 and $\mathcal{M}$-5 on CIFAR-10 and CIFAR-100). Though both member and non-member samples are never used to train the shadow model, we still observe a clear difference between their loss distribution. This verifies our key intuition aforementioned: The transferability of membership information holds between shadow model $\mathcal{S}$ and target model $\mathcal{M}$, i.e., the member and non-member samples behaving differently in $\mathcal{M}$ will also behave differently in $\mathcal{S}$.

Threshold Choosing. As mentioned before, in the membership inference stage, the adversary needs to make a manual decision on which threshold to use. For the transfer attack, since we assume that the adversary has a dataset that comes from the same distribution as the target model's dataset, it can rely on the shadow dataset to estimate a threshold by sampling certain part of that dataset as non-member samples.

4 BOUNDARY-ATTACK

After demonstrating our transfer attack, we now present our second attack, i.e., boundary attack. Since curating auxiliary data requires significant time and monetary investment. Thus, we relax this assumption in this attack. The adversary does not have a shadow dataset to train a shadow model. All they could rely on is the predicted label from the target model. To the best of our knowledge, this is by far the most strict setting for membership inference against ML models. In the following section, we start with the key intuition description. Then, we introduce the attack methodology. In the end, we present the evaluation results.

4.1 Key Intuition

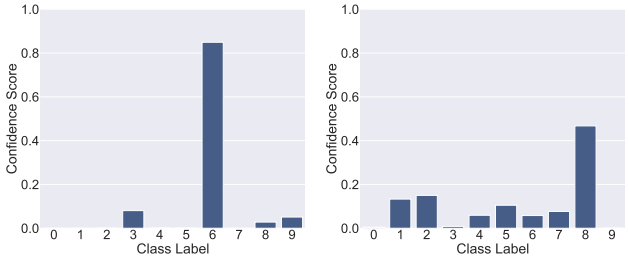
Our intuition behind this attack follows a general observation of the overfitting nature of ML models. Concretely, an ML model is more confident in predicting data samples that it is trained on. In contrast to the prior score-based attacks[25, 31, 35, 46, 48, 49, 57] that directly exploit confidence scores as analysis objects, we place our focus on the antithesis of this observation, i.e., since the ML model is more confident on member data samples, it should be much harder to change its mind.

Intuitively, Figure 6 depicts the confidence scores for two randomly selected member data samples (Figure 6a, Figure 6c) and non-member data samples (Figure 6b, Figure 6d) with respect to $\mathcal{M}$-0 trained on CIFAR-10. We can observe that the maximal score for member samples is indeed much higher than the one of non-member samples. We further use cross entropy (Equation 1) to quantify the difficulty for an ML model to change its predicted label for a data sample to other labels.

Table 2 shows the cross entropy between the confidence scores and other labels for these samples. We can see that member samples' cross entropy is significantly larger than non-member samples. This leads to the following observation on membership information.

Table 2: The cross entropy between the confidence scores and other labels except for the predicted label. ACE represent the Average Cross Entropy.

Status	Truth Label	Predicted Label	Cross Entropy 0	1	2	3	4	5	6	7	8	9	ACE
(a) Member	6	6	7.8156	8.3803	4.1979	1.0942	4.1367	4.3492	-	7.6328	1.5522	1.2923	**4.4946**
(b) Non-member	8	8	2.3274	0.8761	0.8239	2.0793	1.2275	0.9791	1.2373	1.1152	-	5.0451	1.2218
(c) Member	3	3	1.2995	5.2842	5.4212	-	1.5130	4.8059	4.5897	7.1547	3.2411	4.7910	**4.2334**
(d) Non-member	7	9	2.8686	1.8325	3.6480	0.5352	1.8722	1.1689	4.0124	0.6866	3.1071	-	2.1766



(a) Member data sample **(b) Non-member data sample**

(c) Member data sample **(d) Non-member data sample**

Figure 6: The probability distribution of the target model ($\mathcal{M}$-0, CIFAR-10) on member samples and non-member samples.

Observation. Given an ML model and a set of data samples, the cost of changing the target model's predicted labels for member samples is larger than the cost for non-member samples. Furthermore, consider the label-only exposures in a black-box ML model, which means the adversary can only perturb the data samples to change the target model's predicted labels, thus the perturbation needed to change a member sample's predicted label is larger than non-members. Then, the adversary can exploit the magnitude of the perturbation to determine whether the sample is a member or not.

4.2 Methodology

Our attack methodology consists of the following three stages, i.e., decision change, perturbation measurement, and membership inference. The algorithm can be found in Appendix algorithm 2.

Decision Change. The goal of changing the final model decision, i.e., predicted label, is similar to that of adversarial attack [8, 10, 41, 42, 47, 52], For simplicity, we utilize adversarial example techniques to perturb the input to mislead the target model. Specifically, we utilize two state-of-the-art black-box adversarial attacks, namely HopSkipJump [12] and QEBA [30], which only require access to the model's predicted labels.

Perturbation Measurement. Once the final model decision has changed, we measure the magnitude of the perturbations added to the candidate input samples. In general, adversarial attack techniques typically use L_p distance (or Minkowski Distance), e.g., L_0, L_1, L_2, and L_∞, to measure the perceptual similarity between a perturbed sample and its original one. Thus, we use L_p distance to measure the perturbation.

Membership Inference. After obtaining the magnitude of the perturbations, the adversary simply considers a candidate sample with perturbations larger than a threshold as a member sample, and vice versa. Similar to the transfer attack, we mainly use AUC as our evaluation metric. We also provide a general and simple method for choosing a threshold in Section 4.4.

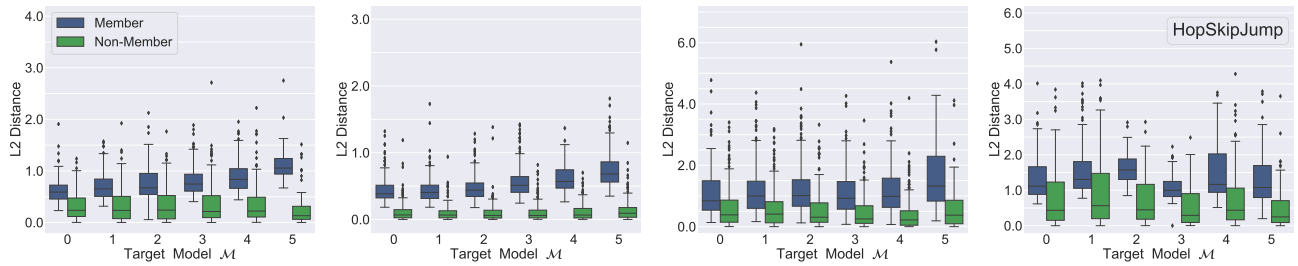
4.3 Experiment Setup

We use the same experimental setup as presented in Section 3.3, such as the dataset splitting strategy and 6 target models trained on different size of training set $\mathcal{D}_{train}$. In the decision change stage, we use the implementation of a popular python library (ART[3]) for HopSkipJump, and the authors' source code[4] for QEBA. Note that we only apply untargeted decision change, i.e., changing the initial decision of the target model to any other random decision. Besides, both HopSkipJump and QEBA require multiple queries to perturb data samples to change their predicted labels. We set 15,000 for HopSkipJump and 7,000 for QEBA. We further study the influence of the number of queries on the attack performance. For space reasons, we report the results of HopSkipJump scheme in the main body of our paper. Results of QEBA scheme can be found in Appendix Figure 14 and Figure 15.

4.4 Results

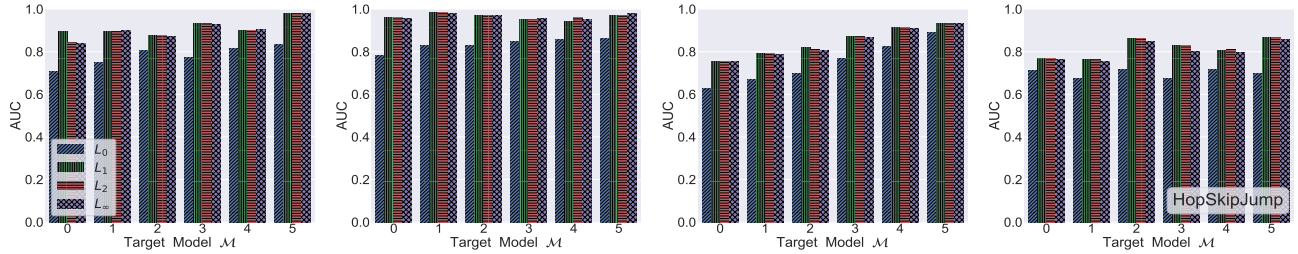
Distribution of Perturbation. First, we show the distribution of perturbation between a perturbed sample and its original one for member and non-member samples in Figure 7. Both decision change schemes, i.e., HopSkipJump and QEBA, apply L_2 distance to limit the magnitude of perturbation, thus we report results of L_2 distance as well. As expected, the magnitude of the perturbation on

[3]https://github.com/Trusted-AI/adversarial-robustness-toolbox
[4]https://github.com/AI-secure/QEBA



(a) CIFAR-10 (b) CIFAR-100 (c) GTSRB (d) Face

Figure 7: L_2 distance between the original sample and its perturbed samples generated by the HopSkipJump attack. The x-axis represents the target model being attacked and the y-axis represents the L_2 distance.



(a) CIFAR-10 (b) CIFAR-100 (c) GTSRB (d) Face

Figure 8: Attack AUC for four different L_p distances between the original samples and its perturbed samples generated by the HopSkipJump attack. The x-axis represents the target model being attacked and the y-axis represents the AUC score.

member samples is indeed larger than that on non-member samples. For instance in Figure 7 ($\mathcal{M}$-5, CIFAR-10), the average L_2 distance of the perturbation for member samples is 1.0755, while that for non-member samples is 0.1102. In addition, models with a larger training set, i.e., lower overfitting level, require less perturbation to change the final prediction. As the overfitting level increases, the adversary needs to modify more on the member sample. The reason is that an ML model with a higher overfitting level has remembered its training samples to a larger extent, thus it is much harder to change their predicted labels, i.e., larger perturbation is required.

Attack AUC Performance. We report the AUC scores over all datasets in Figure 8. In particular, we compare 4 different distance metrics, i.e., L_0, L_1, L_2, and L_∞, for each decision change scheme. From Figure 8, we can observe that L_1, L_2 and L_∞ metrics achieve the best performance across all datasets. For instance in Figure 8 ($\mathcal{M}$-1, CIFAR-10), the AUC scores for L_1, L_2, and L_∞ metrics are 0.8969, 0.8963, and 0.9033, respectively, while the AUC score for L_0 metric is 0.7405. From Figure 15 (in Appendix), we can also observe the same results of QEBA scheme: L_1, L_2 and L_∞ metrics achieve the best performance across all datasets, while L_0 metric performs the worst. Therefore, an adversary can simply choose the same distance metric adopted by adversarial attacks to measure the magnitude of the perturbation.

Effects of Number of Queries. To mount boundary attack in real-world ML applications such as Machine Learning as a Service (MLaaS), the adversary cannot issue as many queries as they want

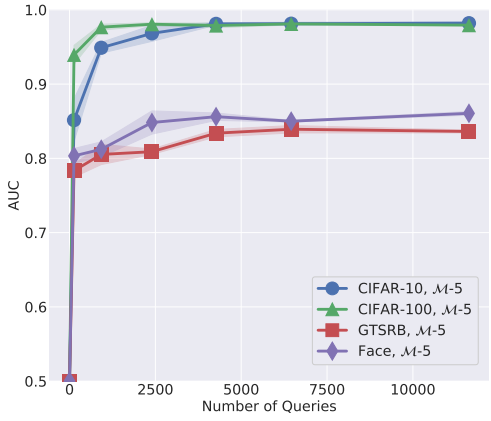
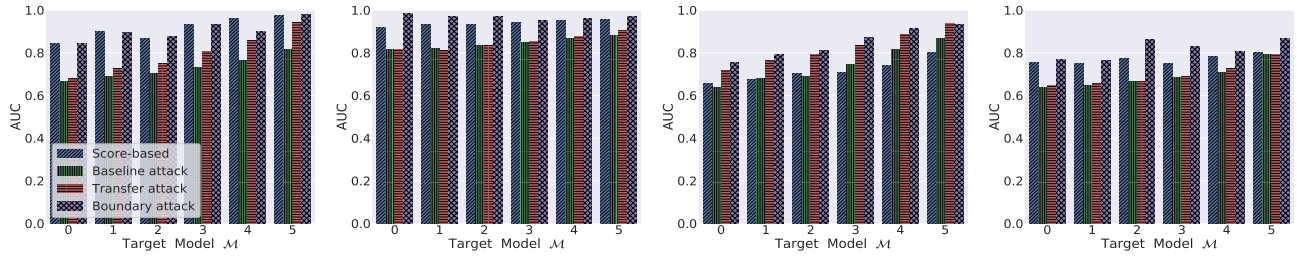


Figure 9: Attack AUC under the effect of number of queries. The x-axis represents the number of queries and the y-axis represents the AUC score for perturbation-based attack.

to the target model, since a large number of queries increases the cost of the attack and may raise the suspicion of the model provider. Now, we evaluate the attack performance with different number of queries. Here, we show the results of the HopSkipJump scheme for $\mathcal{M}$-5 over all datasets. We vary the number of queries from 0 to 15,000 and evaluate the attack performance based on the L_2 metric.



(a) CIFAR-10 (b) CIFAR-100 (c) GTSRB (d) Face

Figure 10: Comparison of our two types of attacks with the baseline attack and score-based attack. The x-axis represents the target model being attacked and the y-axis represents the AUC score.

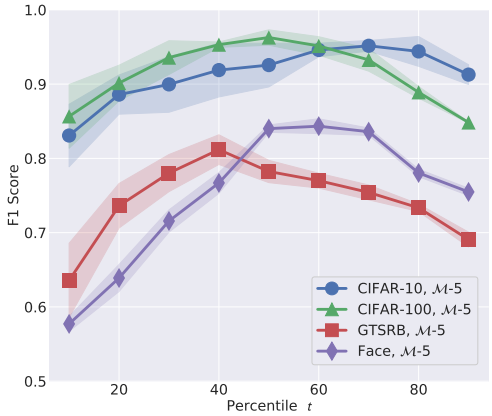


Figure 11: The relation between the top t percentile of the L_2 distance, i.e., threshold, and the attack performance. The x-axis represents the top t percentile and the y-axis represents the F1 score.

As we can see in Figure 9, the AUC increases sharply as the number of queries increases in the beginning. After 2,500 queries, the attack performance becomes stable. From the results, we argue that query limiting would likely not be a suitable defense. For instance, when querying 131 times, the AUC for CIFAR-10 is 0.8228 and CIFAR-100 is 0.9266. At this time, though the perturbed sample is far away from its origin's decision boundary, the magnitude of perturbation for member samples is still relatively larger than that for non-member samples. Thus, the adversary can still differentiate member and non-member samples.

Threshold Choosing. Here, we focus on the threshold choosing for our boundary attack where the adversary is not equipped with a shadow dataset. We provide a simple and general method for choosing a threshold. Concretely, we generate a set of random samples in the feature space as the target model's training set. In the case of image classification, we sample each pixel for an image from a uniform distribution. Next, we treat these randomly generated samples as non-members and query them to the target model. Then, we apply adversarial attack techniques on these random samples to change their initial predicted labels by the target model. Finally, we use these samples' perturbation to estimate a threshold, i.e., finding a suitable top t percentile over these perturbations. The algorithm can be found in Appendix algorithm 3.

We experimentally generate 100 random samples for $\mathcal{M}$-5 trained across all datasets, and adopt HopSkipJump in decision change stage. We again use the L_2 distance to measure the magnitude of perturbation and F1 score as our evaluation metric. From Figure 11, we make the following observations:

- The peak attack performance is bounded between $t = 0\%$ and $t = 100\%$, which means the best threshold can definitely be selected from these random samples' perturbation.
- The powerful and similar attack performance ranges from $t = 30\%$ to $t = 80\%$, reaching half of the total percentile, which means that a suitable threshold can be easily selected.

Therefore, we conclude that our threshold choosing method is effective and can achieve excellent performance.

Comparison of Different Attacks. Now we compare the performance of our two attacks and previous existing attacks. In particular, we also compare our attacks against prior score-based attacks. Following the score-based attack proposed by Salem et al. [46], we train one shadow model using half of $\mathcal{D}_{shadow}$ with its ground truth labels, and one attack model in a supervised manner based on the shadow model's output scores. Here, we do not assume that the attacker knows the exact training set size of the target model, which is actually a strong assumption. Note that this is not a fair comparison, as our decision-based attacks only access to the final model's prediction, rather than the confidence scores.

We report attack performance for our boundary attack using L_2 metric in HopSkipJump scheme. From Figure 10, we can find that our boundary attack achieves similar or even better performance than the score-based attack in some cases. This demonstrates the efficacy of our proposed decision-based attack, thereby the corresponding membership leakage risks stemming from ML models are much more severe than previously shown.

As for cost analysis, the attack logic is different for each method, so it is difficult to evaluate the cost with standard metrics. Besides the adversarial knowledge acquired for each attack, we mainly report training costs and query costs in Table 4. We can find the baseline attack only queries once for a candidate sample. However, in our transfer attack, once a shadow model is built, the adversary will only query the shadow model for candidate samples without

Table 3: Average Certified Radius (ACR) of members and non-members for target models.

Target Model	CIFAR-10		CIFAR-100		GTSRB		Face	
	Member	Non-mem	Member	Non-mem	Member	Non-mem	Member	Non-mem
$\mathcal{M}$-0	0.1392	0.1201	0.0068	0.0033	0.0300	0.0210	0.0571	0.0607
$\mathcal{M}$-1	0.1866	0.1447	0.0133	0.0079	0.0358	0.0215	0.0290	0.0190
$\mathcal{M}$-2	0.1398	0.1170	0.0155	0.0079	0.0692	0.0463	0.0408	0.0313
$\mathcal{M}$-3	0.1808	0.1190	0.0079	0.0074	0.0430	0.0348	0.1334	0.1143
$\mathcal{M}$-4	0.1036	0.1032	0.0141	0.0116	0.0212	0.0176	0.0392	0.0292
$\mathcal{M}$-5	0.1814	0.0909	0.0157	0.0080	0.0464	0.0385	0.1242	0.1110

making any other queries to the target model. Therefore, we cannot prematurely claim that the baseline attack has the lowest cost, but should consider the actual situation.

Table 4: The cost of each attack. Query cost is the number of queries to the target model.

Attack Type	Shadow Model Training Epochs	Query for $\mathcal{D}_{shadow}$	Query for a candidate sample
score-based	200	-	1
baseline attack	-	-	1
transfer attack	200	$\lvert\mathcal{D}_{shadow}\rvert$	-
boundary attack	-	-	Multiple

5 MEMBERSHIP LEAKAGE ANALYSIS

The above results fully demonstrate the effectiveness of our decision-based attacks. Here, we delve more deeply into the reasons for the success of membership inference. Our boundary attack utilizes the magnitude of the perturbation to determine whether the sample is a member or not, and the key to stop searching perturbations is the final decision change of the model. Here, the status of decision change actually contains information about the decision boundary, i.e., the perturbed sample crosses the decision boundary. This suggests a new perspective on the relationship between member samples and non-member samples, and we intend to analyze membership leakage from this perspective. Since previous experiments have verified our key intuition that the perturbation required to change the predicted label of a member sample is larger than that of a non-member, we argue that the distance between the member sample and its decision boundary is typically larger than that of the non-member sample. Next, we will verify it both quantitatively and qualitatively.

5.1 Quantitative Analysis

We introduce the neighboring L_p-radius ball to investigate the membership leakage of ML models. This neighboring L_p-radius ball, also known as *Robustness Radius*, is defined as the L_p robustness of the target model at a data sample, which represents the radius of the largest L_p ball centered at the data sample in which the target model does not change its prediction, as shown in Figure 12d. Concretely, we investigate the L_2 robustness radius of the target model $\mathcal{M}$ at a data sample x. Unfortunately, computing the robustness radius of a ML model is a hard problem. Researchers have proposed many certification methods to derive a tight lower bound of robustness radius $R(\mathcal{M}; x, y)$ for ML models. Here, we also derive a tight lower bound of robustness radius, namely *Certified Radius* [58], which satisfies $0 \leq CR(\mathcal{M}; x, y) \leq R(\mathcal{M}; x, y)$ for any $\mathcal{M}, x$ and its ground truth label $y \in \mathcal{Y} = \{1, 2, \cdots, K\}$. More details about certified radius can be found in Appendix Section A.2.

ACR of Members and Non-members. As we can see from Theorem 1 (see Appendix Section A.2), the value of the certified radius can be estimated by repeatedly sampling Gaussian noises. For the target model $\mathcal{M}$ and a data sample (x, y), we can estimate the certified radius $CR(\mathcal{M}; x, y)$. Here, we use the *average certified radius* (ACR) as a metric to estimate the average certified radius for members and non-members separately, i.e.,

$$ACR_{member} = \frac{1}{|\mathcal{D}_{train}|} \sum_{(x,y)\in\mathcal{D}_{train}} CR(\mathcal{M}; x, y), \tag{2}$$

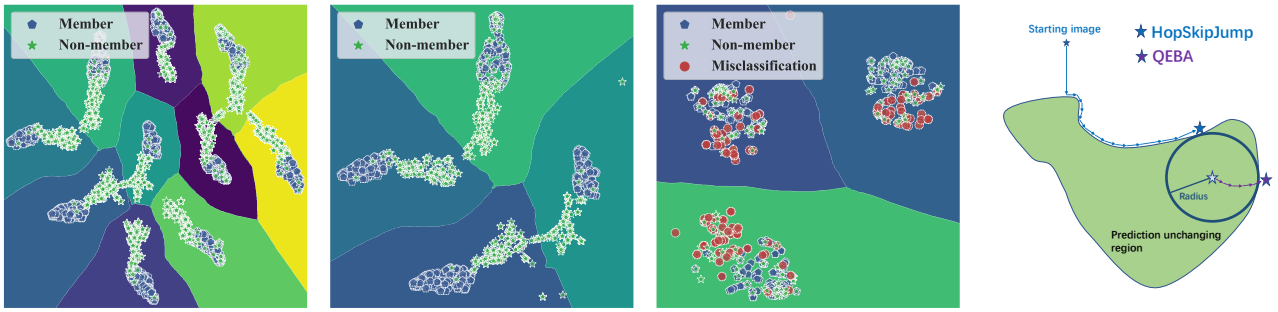
$$ACR_{non-member} = \frac{1}{|\mathcal{D}_{test}|} \sum_{(x,y)\in\mathcal{D}_{test}} CR(\mathcal{M}; x, y). \tag{3}$$

Table 5: Average Certified Radius (ACR) of members and non-members for shadow models.

Shadow Model	CIFAR-10		CIFAR-100	
	Member	Non-mem	Member	Non-mem
$\mathcal{M}$-0	0.1392	0.1301	0.0091	0.0039
$\mathcal{M}$-1	0.1873	0.1516	0.0150	0.0071
$\mathcal{M}$-2	0.1416	0.1463	0.0177	0.0068
$\mathcal{M}$-3	0.1962	0.1452	0.0121	0.0047
$\mathcal{M}$-4	0.1152	0.1046	0.0099	0.0092
$\mathcal{M}$-5	0.1819	0.0846	0.0176	0.0087

We randomly select an equal number of members and non-members for target models and report the results in Table 3. Note that the certified radius is actually an estimated value representing the lower bound of the robustness radius, not the exact radius. Therefore, we analyze the results from a macroscopic perspective and can draw the following observations.

- The ACR of member samples is generally larger than the ACR of non-member samples, which means that in the output space, the ML model maps member samples further away from its decision boundary than non-member samples.



(a) Target Model, Zoom-out (b) Target Model, Zoom-in (c) Shadow Model, Zoom-in (d) Adversarial attack process

Figure 12: The visualization of decision boundary for target model (a, b) and shadow model (c), and the search process of perturbed sample by HopSkipJump and QEBA (d).

- As the level of overfitting increases, the macroscopic trend of the gap between the ACR of members and non-members is also larger, which exactly reflects the increasing attack performance in the aforementioned AUC results.

Furthermore, we also feed the equal member and non-member samples into each corresponding shadow model and obtain the ACR. Note that both member and non-member samples are never used to train the shadow model. We report the results in Table 5, and we can draw the same observations as for the target model. In other words, this again verifies our key intuition for transfer attack: The transferability of membership information holds between shadow model $\mathcal{S}$ and target model $\mathcal{M}$, i.e., the member and non-member samples behaving differently in $\mathcal{M}$ will also behave differently with high probability in $\mathcal{S}$.

5.2 Qualitative Analysis

Next, we investigate the membership leakage of ML models from a visualization approach. We study the decision boundary of the target model (CIFAR-10, $\mathcal{M}$-3) with a given set of data samples, including 1,000 member samples and 1,000 non-member samples. To better visualize the decision boundary, there are two points to note:

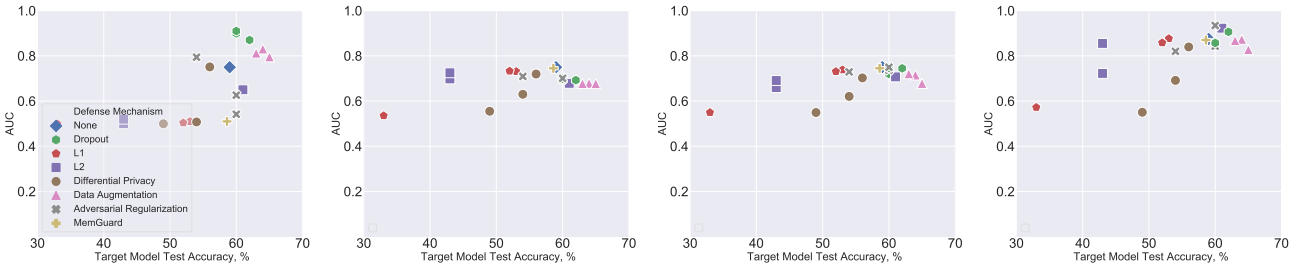
- Both member and non-member samples are mapped from the input space to the output space, which then presents the membership signal. Thus, we visualize the decision boundary in the output space, i.e., the transformed space of the last hidden layer which is fully connected with the final model decision.
- Due to the limitation of the target dataset size, we further sample a large number of random data points in the output space and label them with different colors according to their corresponding classes. This can clearly visualize the decision boundary that distinguishes between different class regions.

To this end, we map the given data samples into the transformed space and embed the output logits or scores into a 2D space using t-Distributed Stochastic Neighbor Embedding (t-SNE) [16]. Figure 12a shows the results for 10 classes of CIFAR-10. We can see that the given data samples have been clearly classified into 10 classes and mapped to 10 different regions. For the sake of analysis, we purposely zoom in four different regions in the left of the whole space. From Figure 12b, we can make the following observations:

- The member samples and non-member samples belonging to the same class are tightly divided into 2 clusters, which explains why the previous score-based attacks can achieve effective performance.
- More interestingly, we can see that the member samples are further away from the decision boundary than the non-member samples, that is, the distance between the members and the decision boundary is larger than that of the non-members. Again, this validates our key intuition.

Recall that in the decision change stage of boundary attack, we apply black-box adversarial attack techniques to change the final model decision. Here, we give an intuitive overview of how HopSkipJump and QEBA schemes work in Figure 12d. As we can see, though these two schemes adopt different strategies to find the perturbed sample, there is one thing in common: The search ends at the tangent samples between the neighboring L_p-radius ball of the original sample and its decision boundary. Only in this way they can mislead the target model and also generate a small perturbation. Combined with Figure 12b, we can find that the magnitude of perturbation is essentially a reflection of the distance from the original sample to its decision boundary.

We again feed the 1,000 member samples and 1,000 non-member samples to the shadow model (CIFAR-10, $\mathcal{M}$-3), and visualize its decision boundary in Figure 12c. In particular, we mark in red the misclassified samples from non-members. First, looking at the correctly classified samples, we can also find that the member samples are relatively far from the decision boundary, i.e., the loss is relatively lower than that of non-member samples. As for the misclassified samples, it is easy to see that their loss is much larger than any other samples. Therefore, we can leverage the loss as metric to differentiate members and non-members. However, we should also note that compared to Figure 12b, the difference between members and non-members towards the decision boundary is much smaller. Thus, if we do not adopt loss metric which considers the ground truth label, then the maximum confidence scores $Max(p_i)$ and normalized entropy $\frac{-1}{\log(K)} \sum_i p_i \log(p_i)$ which are just based on self-information will lead to a much lower difference between



(a) Score-based attack **(b) Baseline attack** **(c) Transfer attack** **(d) Boundary attack**

Figure 13: Attack AUC of transfer attack and boundary attack against multiple defense mechanisms.

members and non-members. This is the reason why the loss metric achieves the highest performance.

Summarizing the above quantitative and qualitative analysis, we verify our argument that the distance between the member sample and its decision boundary is larger than that of the non-member sample, thus revealing the reasons for the success of the membership inference, including score-based and decision-based attacks. In addition, we verify that membership information remains transferable between the target and shadow models. Last but not least, we also show the reason why the loss metric of the transfer attack achieves the best performance.

6 DEFENSES EVALUATION

To mitigate the threat of membership leakage, a large body of defense mechanisms have been proposed in the literature. In this section, we evaluate the performance of current membership inference attacks against the state-of-the-art defenses. We summarize existing defenses in the following three broad categories.

Generalization Enhancement. As overfitting is the major reason for membership inference to be successful, multiple approaches have been proposed with the aim of reducing overfitting, which are first introduced by the machine learning community to encourage generalization. The standard generalization enhancement techniques, such as weight decay (L1/L2 regularization) [46, 54], dropout [50], and data augmentation, have been shown to limit overfitting effectively, but may lead to a significant decrease in model accuracy.

Privacy Enhancement. Differential privacy [11, 17, 26] is a widely adopted for mitigating membership privacy. Many differential privacy based defense techniques add noise to the gradient to ensure the data privacy in the training process of the ML model. A representative approach in this category is DP-Adam [4], and we adopt an open-source version of its implementation in our experiments.[5]

Confidence Score Perturbation. Previous score-based attacks have demonstrated that the confidence score predicted by the target model clearly presents membership signal. Therefore, researchers have proposed several approaches to alter the confidence score. We focus on two representative approaches in this category: MemGuard [27] and adversarial regularization [38], which changes the output probability distribution so that both members and non-members look like similar examples to the inference model built by the adversary. We adopt the original implementation of MemGuard,[6] and an open-source version of the adversarial regularization.[7]

For each mechanism, we train 3 target models (CIFAR-10, $\mathcal{M}-2$) using different hyper-parameters. For example, in L2 regularization, the λ used to constrain the regularization loss is set to 0.01, 0.05, and 0.1, and the λ in L1 regularization is set to 0.0001, 0.001 and 0.005, respectively. In differential privacy, the noise is randomly sampled from a Gaussian distribution $\mathcal{N}(\epsilon, \beta)$, wherein ϵ is fixed to 0 and β is set to 0.1, 0.5 and 1.1, respectively.

Table 6: Attack AUC performance under the defense of MemGuard.

Attack	CIFAR-10, $\mathcal{M}$-2		Face, $\mathcal{M}$-2	
	None	MemGuard	None	MemGuard
score-based	0.8655	0.5151	0.755	0.513
baseline attack	0.705	0.705	0.665	0.665
transfer attack	0.7497	0.7497	0.6664	0.6664
boundary attack	0.8747	0.8747	0.8617	0.8617

We report the attack performance against models trained with a wide variety of different defensive mechanisms in Figure 13, and we make the following observations.

- Our decision-based attacks. i.e., both transfer attack and boundary attack, can bypass most types of defense mechanisms.
- Strong differential privacy (β=1.1), L1 regularization (λ = 0.005) and L2 regularization (λ = 0.1) can reduce membership leakage but, as expected, lead to a significant degradation in the model's accuracy. The reason is that the decision boundary between members and non-members is heavily blurred.
- Data augmentation can definitely reduce overfitting, but it still does not reduce membership leakage. This is because data augmentation drives the model to strongly remember both the original samples and their augmentations.

[5] https://github.com/ebagdasa/pytorch-privacy

[6] https://github.com/jjy1994/MemGuard

[7] https://github.com/SPIN-UMass/ML-Privacy-Regulization

In Table 6, we further compare the performance of all attacks against MemGuard [27], which is the latest powerful defense technique and can be easily deployed. We can find that MemGuard cannot defend against decision-based attacks at all, but is very effective against previous score-based attacks.

7 RELATED WORKS

Various research has shown that machine learning models are vulnerable to security and privacy attacks. In this section, we mainly survey the domains that are most relevant to us.

Membership Inference. Membership inference attack has been successfully performed in various data domains, ranging form biomedical data [6, 22, 24] to mobility traces [43]. Shokri et al. [48] present the first membership inference attack against machine learning models. The general idea behind this attack is to use multiple shadow models to generate data to train multiple attack models (one for each class). These attack models take the target sample's confidence scores as input and output its membership status, i.e., member or non-member. Salem et al. [46] later present another attack by gradually relaxing the assumptions made by Shokri et al. [48] achieving a model and data independent membership inference. In addition, there are several other subsequent score-based membership inference attacks [25, 31, 35, 49, 57]. In the area of decision-based attacks, Yeom et al. [57] quantitatively analyzed the relationship between attack performance and loss for training and testing sets, and proposed the first decision-based attack, i.e., baseline attack aforementioned. We also acknowledge that a concurrent work [13] proposes an approach similar to our boundary attack. Specifically, the concurrent work assumes that an adversary has more knowledge of the target model, including training knowledge (model architecture, training algorithm, and training dataset size), and a shadow dataset from the same distribution as the target dataset to estimate the threshold. In our work, we relax all assumptions and propose a general threshold-choosing method. We further present a new perspective on the reasons for the success of membership inference. In addition, we introduce a novel transfer-attack.

Defenses Against Membership Inference. Researchers have proposed to improve privacy against membership inference via different types of generalization enhancement. For example, Shokri et al. [48] adopted L2 regularization with a polynomial in the model's loss function to penalize large parameters. Salem et al. [46] demonstrated two effective method of defending MI attacks, namely dropout and model stacking. Nasr et al. [38] introduced a defensive confidence score membership classifier in a min-max game mechanism to train models with membership privacy, namely adversarial regularization. There are other existing generalization enhancement method can be used to mitigate membership leakage, such as L1 regularization and data augmentation. Another direction is privacy enhancement. Many differential privacy-based defenses [11, 17, 26] involve clipping and adding noise to instance-level gradients and is designed to train a model to prevent it from memorizing training data or being susceptible to membership leakage. Shokri et al. [48] designed a differential privacy method for collaborative learning of DNNs. As for confidence score alteration, Jia et al. [27] introduce MemGuard, the first defense with formal utility-loss guarantees against membership inference. The basic idea behind this work is to add carefully crafted noise to confidence scores of an ML model to mislead the membership classifier. Yang et al. [56] also propose a similar defense in this direction.

Attacks against Machine Learning. Besides membership inference attacks, there exist numerous other types of attacks against ML models. A major attack type in this space is adversarial examples [12, 30, 40–42, 52]. In this setting, an adversary adds carefully crafted noise to samples aiming at mislead the target classifier. Ganju et al. [20] proposed a property inference attack aiming at inferring general properties of the training data (such as the proportion of each class in the training data). Model inversion attack [18, 19] focuses on inferring the missing attributes of the target ML model. A similar type of attacks is backdoor attack, where the adversary as a model trainer embeds a trigger into the model for her to exploit when the model is deployed [21, 34, 55]. Another line of work is model stealing, Tramèr et al. [53] proposed the first attack on inferring a model's parameters. Other works focus on protecting a model's ownership [5, 32, 44, 59].

8 CONCLUSION

In this paper, we perform a systematic investigation on membership leakage in label-only exposures of ML models, and propose two novel decision-based membership inference attacks, including transfer attack and boundary attack. Extensive experiments demonstrate that our two attacks achieve better performances than baseline attack, and even outperform prior score-based attacks in some cases. Furthermore, we propose a new perspective on the reasons for the success of membership inference and show that members samples are further away from the decision boundary than non-members. Finally, we evaluate multiple defense mechanisms against our decision-based attacks and show that our novel attacks can still achieve reasonable performance unless heavy regularization has been applied. In particular, our evaluation demonstrates that confidence score perturbation is an infeasible defense mechanism in label-only exposures.

ACKNOWLEDGMENTS

This work is partially funded by the Helmholtz Association within the project "Trustworthy Federated Data Analytics" (TFDA) (funding number ZT-I-OO1 4).

REFERENCES

[1] https://www.cs.toronto.edu/~kriz/cifar.html.
[2] http://benchmark.ini.rub.de/?section=gtsrb.
[3] http://vis-www.cs.umass.edu/lfw/.
[4] Martin Abadi, Andy Chu, Ian Goodfellow, Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep Learning with Differential Privacy. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 308–318. ACM, 2016.
[5] Yossi Adi, Carsten Baum, Moustapha Cisse, Benny Pinkas, and Joseph Keshet. Turning Your Weakness Into a Strength: Watermarking Deep Neural Networks by Backdooring. In *USENIX Security Symposium (USENIX Security)*, pages 1615–1631. USENIX, 2018.
[6] Michael Backes, Pascal Berrang, Mathias Humbert, and Praveen Manoharan. Membership Privacy in MicroRNA-based Studies. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 319–330. ACM, 2016.
[7] Michael Backes, Mathias Humbert, Jun Pang, and Yang Zhang. walk2friends: Inferring Social Links from Mobility Profiles. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 1943–1957. ACM, 2017.

[8] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Srndic, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion Attacks against Machine Learning at Test Time. In *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD)*, pages 387–402. Springer, 2013.

[9] Philippe Burlina, David E. Freund, B. Dupas, and Neil M. Bressler. Automatic Screening of Age-related Macular Degeneration and Retinal Abnormalities. In *Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 3962–3966. IEEE, 2011.

[10] Nicholas Carlini and David Wagner. Towards Evaluating the Robustness of Neural Networks. In *IEEE Symposium on Security and Privacy (S&P)*, pages 39–57. IEEE, 2017.

[11] Kamalika Chaudhuri, Claire Monteleoni, and Anand D Sarwate. Differentially Private Empirical Risk Minimization. *Journal of Machine Learning Research*, 2011.

[12] Jianbo Chen, Michael I. Jordan, and Martin J. Wainwright. HopSkipJumpAttack: A Query-Efficient Decision-Based Attack. In *IEEE Symposium on Security and Privacy (S&P)*, pages 1277–1294. IEEE, 2020.

[13] Christopher A. Choquette Choo, Florian Tramèr, Nicholas Carlini, and Nicolas Papernot. Label-Only Membership Inference Attacks. *CoRR abs/2007.14321*, 2020.

[14] Jeremy M. Cohen, Elan Rosenfeld, and J. Zico Kolter. Certified Adversarial Robustness via Randomized Smoothing. In *International Conference on Machine Learning (ICML)*, pages 1310–1320. PMLR, 2019.

[15] Ambra Demontis, Marco Melis, Maura Pintor, Matthew Jagielski, Battista Biggio, Alina Oprea, Cristina Nita-Rotaru, and Fabio Roli. Why Do Adversarial Attacks Transfer? Explaining Transferability of Evasion and Poisoning Attacks. In *USENIX Security Symposium (USENIX Security)*, pages 321–338. USENIX, 2019.

[16] Laurens Van der Maaten and Geoffrey Hinton. Visualizing Data Using t-SNE. *Journal of Machine Learning Research*, 2008.

[17] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating Noise to Sensitivity in Private Data Analysis. In *Theory of Cryptography Conference (TCC)*, pages 265–284. Springer, 2006.

[18] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. Model Inversion Attacks that Exploit Confidence Information and Basic Countermeasures. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 1322–1333. ACM, 2015.

[19] Matt Fredrikson, Eric Lantz, Somesh Jha, Simon Lin, David Page, and Thomas Ristenpart. Privacy in Pharmacogenetics: An End-to-End Case Study of Personalized Warfarin Dosing. In *USENIX Security Symposium (USENIX Security)*, pages 17–32. USENIX, 2014.

[20] Karan Ganju, Qi Wang, Wei Yang, Carl A. Gunter, and Nikita Borisov. Property Inference Attacks on Fully Connected Neural Networks using Permutation Invariant Representations. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 619–633. ACM, 2018.

[21] Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Grag. Badnets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain. *CoRR abs/1708.06733*, 2017.

[22] Inken Hagestedt, Yang Zhang, Mathias Humbert, Pascal Berrang, Haixu Tang, XiaoFeng Wang, and Michael Backes. MBeacon: Privacy-Preserving Beacons for DNA Methylation Data. In *Network and Distributed System Security Symposium (NDSS)*. Internet Society, 2019.

[23] Benjamin Hilprecht, Martin Härterich, and Daniel Bernau. Monte Carlo and Reconstruction Membership Inference Attacks against Generative Models. *Symposium on Privacy Enhancing Technologies Symposium*, 2019.

[24] Nils Homer, Szabolcs Szelinger, Margot Redman, David Duggan, Waibhav Tembe, Jill Muehling, John V. Pearson, Dietrich A. Stephan, Stanley F. Nelson, and David W. Craig. Resolving Individuals Contributing Trace Amounts of DNA to Highly Complex Mixtures Using High-Density SNP Genotyping Microarrays. *PLOS Genetics*, 2008.

[25] Bo Hui, Yuchen Yang, Haolin Yuan, Philippe Burlina, Neil Zhenqiang Gong, and Yinzhi Cao. Practical Blind Membership Inference Attack via Differential Comparisons. In *Network and Distributed System Security Symposium (NDSS)*. Internet Society, 2021.

[26] Roger Iyengar, Joseph P. Near, Dawn Xiaodong Song, Om Dipakbhai Thakkar, Abhradeep Thakurta, and Lun Wang. Towards Practical Differentially Private Convex Optimization. In *IEEE Symposium on Security and Privacy (S&P)*, pages 299–316. IEEE, 2019.

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

[28] Ira Kemelmacher-Shlizerman, Steven M. Seitz, Daniel Miller, and Evan Brossard. The MegaFace Benchmark: 1 Million Faces for Recognition at Scale. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4873–4882. IEEE, 2016.

[29] Konstantina Kourou, Themis P. Exarchos, Konstantinos P. Exarchos, Michalis V. Karamouzis, and Dimitrios I. Fotiadis. Machine Learning Applications in Cancer Prognosis and Prediction. *Computational and Structural Biotechnology Journal*, 2015.

[30] Huichen Li, Xiaojun Xu, Xiaolu Zhang, Shuang Yang, and Bo Li. QEBA: Query-Efficient Boundary-Based Blackbox Attack. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1218–1227. IEEE, 2020.

[31] Jiacheng Li, Ninghui Li, and Bruno Ribeiro. Membership Inference Attacks and Defenses in Supervised Learning via Generalization Gap. In *ACM Conference on Data and Application Security and Privacy (CODASPY)*, pages 5–16. ACM, 2021.

[32] Zheng Li, Chengyu Hu, Yang Zhang, and Shanqing Guo. How to Prove Your Model Belongs to You: A Blind-Watermark based Framework to Protect Intellectual Property of DNN. In *Annual Computer Security Applications Conference (ACSAC)*, pages 126–137. ACM, 2019.

[33] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. Delving into Transferable Adversarial Examples and Black-box Attacks. *CoRR abs/1611.02770*, 2016.

[34] Yingqi Liu, Shiqing Ma, Yousra Aafer, Wen-Chuan Lee, Juan Zhai, Weihang Wang, and Xiangyu Zhang. Trojaning Attack on Neural Networks. In *Network and Distributed System Security Symposium (NDSS)*. Internet Society, 2019.

[35] Yunhui Long, Vincent Bindschaedler, and Carl A. Gunter. Towards Measuring Membership Privacy. *CoRR abs/1712.09136*, 2017.

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

[37] Muzammal Naseer, Salman H. Khan, Muhammad Haris Khan, Fahad Shahbaz Khan, and Fatih Porikli. Cross-Domain Transferability of Adversarial Perturbations. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, pages 12885–12895. NeurIPS, 2019.

[38] Milad Nasr, Reza Shokri, and Amir Houmansadr. Machine Learning with Membership Privacy using Adversarial Regularization. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 634–646. ACM, 2018.

[39] Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. Transferability in Machine Learning: from Phenomena to Black-Box Attacks using Adversarial Samples. *CoRR abs/1605.07277*, 2016.

[40] Nicolas Papernot, Patrick McDaniel, Arunesh Sinha, and Michael Wellman. SoK: Towards the Science of Security and Privacy in Machine Learning. In *IEEE European Symposium on Security and Privacy (Euro S&P)*, pages 399–414. IEEE, 2018.

[41] Nicolas Papernot, Patrick D. McDaniel, Ian Goodfellow, Somesh Jha, Z. Berkay Celik, and Ananthram Swami. Practical Black-Box Attacks Against Machine Learning. In *ACM Asia Conference on Computer and Communications Security (ASIACCS)*, pages 506–519. ACM, 2017.

[42] Nicolas Papernot, Patrick D. McDaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik, and Ananthram Swami. The Limitations of Deep Learning in Adversarial Settings. In *IEEE European Symposium on Security and Privacy (Euro S&P)*, pages 372–387. IEEE, 2016.

[43] Apostolos Pyrgelis, Carmela Troncoso, and Emiliano De Cristofaro. Knock Knock, Who's There? Membership Inference on Aggregate Location Data. In *Network and Distributed System Security Symposium (NDSS)*. Internet Society, 2018.

[44] Bita Darvish Rouhani, Huili Chen, and Farinaz Koushanfar. DeepSigns: A Generic Watermarking Framework for IP Protection of Deep Learning Models. *CoRR abs/1804.00750*, 2018.

[45] Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, Yann Ollivier, and Hervé Jégou. White-box vs Black-box: Bayes Optimal Strategies for Membership Inference. In *International Conference on Machine Learning (ICML)*, pages 5558–5567. PMLR, 2019.

[46] Ahmed Salem, Yang Zhang, Mathias Humbert, Pascal Berrang, Mario Fritz, and Michael Backes. ML-Leaks: Model and Data Independent Membership Inference Attacks and Defenses on Machine Learning Models. In *Network and Distributed System Security Symposium (NDSS)*. Internet Society, 2019.

[47] Ali Shafahi, W Ronny Huang, Mahyar Najibi, Octavian Suciu, Christoph Studer, Tudor Dumitras, and Tom Goldstein. Poison Frogs! Targeted Clean-Label Poisoning Attacks on Neural Networks. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, pages 6103–6113. NeurIPS, 2018.

[48] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership Inference Attacks Against Machine Learning Models. In *IEEE Symposium on Security and Privacy (S&P)*, pages 3–18. IEEE, 2017.

[49] Liwei Song, Reza Shokri, and Prateek Mittal. Privacy Risks of Securing Machine Learning Models against Adversarial Examples. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 241–257. ACM, 2019.

[50] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 2014.

[51] Mary H. Stanfill, Margaret Williams, Susan H. Fenton, Robert A. Jenders, and William R. Hersh. A Systematic Literature Review of Automated Clinical Coding and Classification Systems. *J. Am. Medical Informatics Assoc.*, 2010.

[52] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble Adversarial Training: Attacks and Defenses. In *International Conference on Learning Representations (ICLR)*, 2017.

[53] Florian Tramèr, Fan Zhang, Ari Juels, Michael K. Reiter, and Thomas Ristenpart. Stealing Machine Learning Models via Prediction APIs. In *USENIX Security Symposium (USENIX Security)*, pages 601–618. USENIX, 2016.

[54] Stacey Truex, Ling Liu, Mehmet Emre Gursoy, Lei Yu, and Wenqi Wei. Towards Demystifying Membership Inference Attacks. *CoRR abs/1807.09173*, 2018.
[55] Bolun Wang, Yuanshun Yao, Shawn Shan, Huiying Li, Bimal Viswanath, Haitao Zheng, and Ben Y. Zhao. Neural Cleanse: Identifying and Mitigating Backdoor Attacks in Neural Networks. In *IEEE Symposium on Security and Privacy (S&P)*, pages 707–723. IEEE, 2019.
[56] Ziqi Yang, Bin Shao, Bohan Xuan, Ee-Chien Chang, and Fan Zhang. Defending Model Inversion and Membership Inference Attacks via Prediction Purification. *CoRR abs/2005.03915*, 2020.
[57] Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy Risk in Machine Learning: Analyzing the Connection to Overfitting. In *IEEE Computer Security Foundations Symposium (CSF)*, pages 268–282. IEEE, 2018.
[58] Runtian Zhai, Chen Dan, Di He, Huan Zhang, Boqing Gong, Pradeep Ravikumar, Cho-Jui Hsieh, and Liwei Wang. MACER: Attack-free and Scalable Robust Training via Maximizing Certified Radius. In *International Conference on Learning Representations (ICLR)*, 2020.
[59] Jialong Zhang, Zhongshu Gu, Jiyong Jang, Hui Wu, Marc Ph. Stoecklin, Heqing Huang, and Ian Molloy. Protecting Intellectual Property of Deep Neural Networks with Watermarking. In *ACM Asia Conference on Computer and Communications Security (ASIACCS)*, pages 159–172. ACM, 2018.
[60] Yuheng Zhang, Ruoxi Jia, Hengzhi Pei, Wenxiao Wang, Bo Li, and Dawn Song. The Secret Revealer: Generative Model-Inversion Attacks Against Deep Neural Networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 250–258. IEEE, 2020.
[61] Tianyue Zheng, Weihong Deng, and Jiani Hu. Cross-Age LFW: A Database for Studying Cross-Age Face Recognition in Unconstrained Environments. *CoRR abs/1708.08197*, 2017.

A APPENDIX

A.1 Datasets Description

CIFAR-10/CIFAR-100. CIFAR-10 [1] and CIFAR-100 [1] are benchmark datasets used to evaluate image recognition algorithms. CIFAR-10 is composed of 32×32 color images in 10 classes, with 6000 images per class. In total, there are 50000 training images and 10000 test images. CIFAR-100 has the same format as CIFAR-10, but it has 100 classes containing 600 images each. There are 500 training images and 100 testing images per class.

GTSRB. The GTSRB [2] dataset is an image collection consisting of 43 traffic signs. Images vary in size and are RGB-encoded. It consists of over 51,839 color images, whose dimensions range from 15×15 to 250×250 pixels (not necessarily square). Of these 51,839 images, 39,209 are used for training, and 12,630 are used for testing. Due to the varying sizes of the images, the images are resized to 64×64 before being passed to the model for classification.

Face. The Face [3] dataset consists of about 13,000 images of human faces crawled from the web. It is collected from 1,680 participants with each participant having at least two distinct images in the dataset. In our evaluation, we only consider people with more than 40 images, which leaves us with 19 people's data, i.e., 19 classes. The Face dataset is challenging for facial recognition, as the images are taken from the web and not under a controlled environment, such as a lab. It is also worth noting that this dataset is unbalanced.

A.2 Certified Radius

Randomized Smoothing. In this work, we apply a recent technique, called randomized smoothing [14], which can be extended to any architecture to obtain the certified radius of smoothed deep neural networks. The core of randomized smoothing is to use the smoothed version of $\mathcal{M}$, which is denoted by $\mathcal{G}$, to make predictions. The formulation of $\mathcal{G}$ is defined as follows.

Definition 1. For an arbitrary classifier $\mathcal{M}$ and $\sigma > 0$, the smoothed classifier $\mathcal{G}$ of $\mathcal{M}$ is defined as

$$\mathcal{G}(x) = \arg\max_{c \in \mathcal{Y}} P_{\varepsilon \sim \mathcal{N}(0,\sigma^2 \boldsymbol{I})}(\mathcal{M}(x+\varepsilon) = c). \tag{4}$$

In short, the smoothed classifier $\mathcal{G}$ returns the label most likely to be returned by $\mathcal{M}$ when its input is sampled from a Gaussian distribution $\mathcal{N}(x, \sigma^2 \boldsymbol{I})$ centered at x. Cohen et al. [14] prove the following theorem, which provides an analytic form of certified radius:

Theorem 1. [14] Let $\mathcal{M} : x \to y$, and $\varepsilon \sim \mathcal{N}(0, \sigma^2 \boldsymbol{I})$. Let the smoothed classifier $\mathcal{G}$ be defined as in (4). Let the ground truth of an input x be y. If $\mathcal{G}$ classifies x correctly, i.e.,

$$P_\varepsilon(\mathcal{M}(x+\varepsilon) = y) \geq \max_{y' \neq y} P_\varepsilon(\mathcal{M}(x+\varepsilon) = y'). \tag{5}$$

Then, $\mathcal{G}$ is provably robust at x, with the certified radius given by

$$\begin{aligned} CR(\mathcal{G}; x, y) =& \frac{\sigma}{2}[\Phi^{-1}(P_\varepsilon(\mathcal{M}(x+\varepsilon) = y)) \\ & - \Phi^{-1}(\max_{y' \neq y} P_\varepsilon(\mathcal{M}(x+\varepsilon) = y'))] \\ =& \frac{\sigma}{2}[\Phi^{-1}(\mathbb{E}_\varepsilon \mathbf{1}_{\{\mathcal{M}(x+\varepsilon)=y\}}) \\ & - \Phi^{-1}(\max_{y' \neq y} \mathbb{E}_\varepsilon \mathbf{1}_{\{\mathcal{M}(x+\varepsilon)=y'\}})], \end{aligned} \tag{6}$$

where Φ is the $c.d.f.$ of the standard Gaussian distribution.

Table 7: Dataset splitting strategy. $\mathcal{D}_{train}$ is used to train the target model and serves as the members, while the other $\mathcal{D}_{test}$ serves as the non-members. $\mathcal{D}_{shadow}$ is used to train the shadow model after relabelled by the target model.

Target Model	CIFAR10		CIFAR100		GTSRB		Face	
	$\mathcal{D}_{train}$	$\mathcal{D}_{test}$	$\mathcal{D}_{train}$	$\mathcal{D}_{test}$	$\mathcal{D}_{train}$	$\mathcal{D}_{test}$	$\mathcal{D}_{train}$	$\mathcal{D}_{test}$
$\mathcal{M}$-0	3000	1000	7000	1000	600	500	350	100
$\mathcal{M}$-1	2000	1000	6000	1000	500	500	300	100
$\mathcal{M}$-2	1500	1000	5000	1000	400	500	250	100
$\mathcal{M}$-3	1000	1000	4000	1000	300	500	200	100
$\mathcal{M}$-4	500	1000	3000	1000	200	500	150	100
$\mathcal{M}$-5	100	1000	2000	1000	100	500	100	100
Shadow Model	$\mathcal{D}_{shadow}$							
	46000		42000		38109		1417	

Algorithm 1: Transfer attack algorithm.

Input: shadow dataset $\mathcal{D}_{shadow}$, shadow model $\mathcal{S}$, target model $\mathcal{M}$, a candidate sample (x, y), threshold τ, minibatch m, membership indicator T;
Output: Trained shadow model $\mathcal{S}$, x is member or not;

Initialize the parameters of *shadow*;
Relabel $\mathcal{D}_{shadow}$ by querying to $\mathcal{M}$;
for *number of training epochs* **do**
  for $i = 1; i \leq \frac{|\mathcal{D}_{shadow}|}{m}; i++$ **do**
    sample minibatch of m samples from $\mathcal{D}_{shadow}$;
    update $\mathcal{S}$ by descending its adam gradient
  end
end
Feed x into $\mathcal{S}$ to obtain p_i;
calculate loss: $l = -\sum_{i=0}^{K} \mathbf{1}_y \log(p_i)$;
if $l \leq \tau$ **then**
  $T = 1$; ; /* x is a member */
else
  $T = 0$; ; /* x is a non-member */
end
return $\mathcal{S}$, T;

Algorithm 2: Boundary attack algorithm.

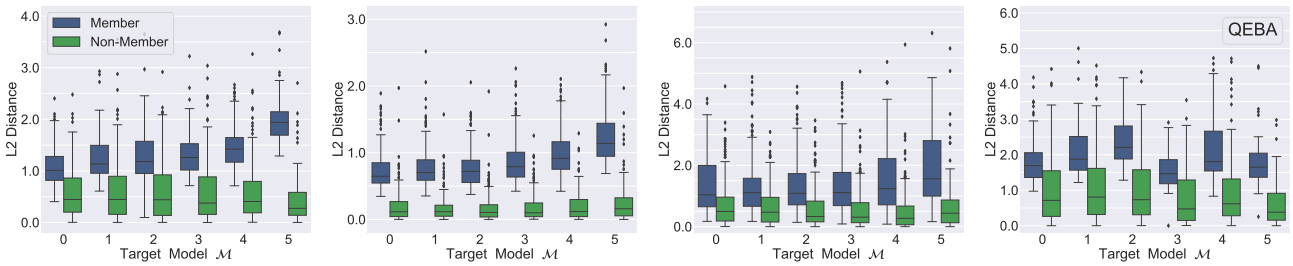
Input: adversarial attack technique *HopSkipJump*, target model $\mathcal{M}$, a candidate sample (x, y), threshold τ, membership indicator T;
Output: x is member or not;

for *number of query* **do**
  Feed x into $\mathcal{M}$ to obtain predicted label y';
  if $y' \neq y$ **then**
    x' = x; ; /* perturbed sample x' */
  else
    Apply *HopSkipJump* to perturb x ;
  end
end
calculate perturbation $P = |x - x'|_2$;
if $P \leq \tau$ **then**
  $T = 0$; ; /* x is a non-member */
else
  $T = 1$; ; /* x is a member */
end
return T;

Algorithm 3: Threshold choosing for boundary attack.

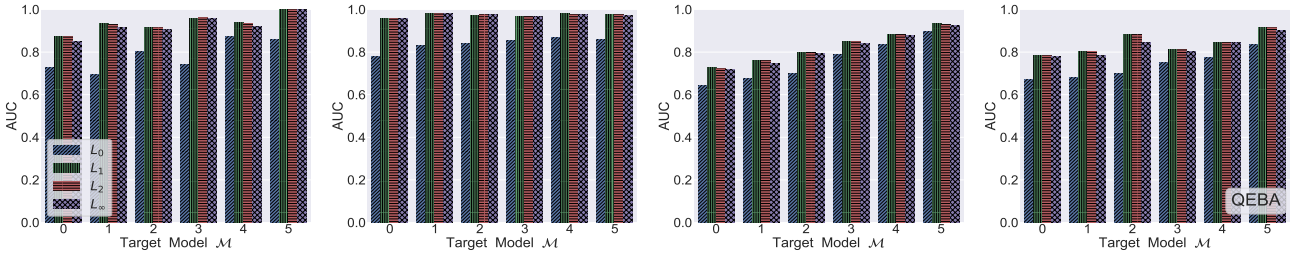
Input: adversarial attack technique *HopSkipJump*, target model $\mathcal{M}$; Gaussian distribution $\mathcal{N}(\epsilon, \beta)$, queue q, top t percentile;
Output: threshold τ;

Initialize q;
Sample multiple random samples $\mathcal{X}$ from $\mathcal{N}(\epsilon, \beta)$ **for** *number of random samples* **do**
  Select one sample $x \in \mathcal{X}$;
  Feed x into $\mathcal{M}$ to obtain predicted label y;
  for *number of query* **do**
    Apply *HopSkipJump* to perturb x to obtain x' ;
    Feed x' into $\mathcal{M}$ to obtain predicted label y';
    if $y' \neq y$ **then**
      push $|x - x'|_2$ into q; break;
    else
      $x = x'$;
    end
  end
end
sort q in descending order;
$\tau = q(t)$;
return τ;



(a) CIFAR-10 (b) CIFAR-100 (c) GTSRB (d) Face

Figure 14: L_2 distance between the original sample and its perturbed samples generated by the QEBA attack. The x-axis represents the target model being attacked and the y-axis represents the L_2 distance.



(a) CIFAR-10 (b) CIFAR-100 (c) GTSRB (d) Face

Figure 15: Attack AUC for four different L_p distances between the original sample and its perturbed samples generated by the QEBA attack. The x-axis represents the target model being attacked and the y-axis represents the AUC score.