Low-Cost High-Power Membership Inference Attacks

Sajjad Zarifzadeh [1] **Philippe Liu** [1] **Reza Shokri** [1]

Abstract

Membership inference attacks aim to detect if a particular data point was used in training a model. We design a novel statistical test to perform robust membership inference attacks (RMIA) with low computational overhead. We achieve this by a fine-grained modeling of the null hypothesis in our likelihood ratio tests, and effectively leveraging both reference models and reference population data samples. RMIA has superior test power compared with prior methods, *throughout the TPR-FPR curve* (even at extremely low FPR, as low as 0). Under computational constraints, where only a limited number of pre-trained reference models (as few as 1) are available, and also when we vary other elements of the attack (e.g., data distribution), our method performs exceptionally well, unlike prior attacks that approach random guessing. RMIA lays the groundwork for practical yet accurate data privacy risk assessment in machine learning.

1. Introduction

Membership inference attacks (MIA) are used to quantify the information leakage of machine learning algorithms about their training data (Shokri et al., 2017). Membership inference attacks originated within the realm of summary statistics on high-dimensional data (Homer et al., 2008). In this context, different hypothesis testing methods were designed to optimize the trade-off between test power and its error (Sankararaman et al., 2009; Visscher & Hill, 2009; Dwork et al., 2015; Murakonda et al., 2021). For deep learning algorithms, these tests evolved from using ML itself to perform MIA (Shokri et al., 2017) to using various approximations of the original statistical tests (Sablayrolles et al., 2019; Ye et al., 2022; Carlini et al., 2022; Watson et al., 2022a; Bertran et al., 2023). Attacks also vary based on the threat models and the computation needed to tailor the attacks to specific data points and models (e.g., global attacks (Shokri et al., 2017; Yeom et al., 2018) versus per-sample tailored attacks (Ye et al., 2022; Carlini et al., 2022; Sablayrolles et al., 2019; Watson et al., 2022a)) which all necessitate training a *large* number of reference models.

Although there have been improvements in the effectiveness of attacks, their **computation cost** renders them useless for practical privacy auditing. Also, as it is shown in the prior work (Carlini et al., 2022; Ye et al., 2022) different strong attacks exhibit mutual dominance *depending on the test scenarios*! Under a practical computation budget, (Carlini et al., 2022) verges on **random guessing**, and in the abundance of computation budget, (Ye et al., 2022) shows low power at low FPR. Through extensive empirical analysis, we observe further **performance instabilities** in the prior attacks across different settings, where we investigate the impact of varying the number of reference models, the number of required inference queries, the similarity of reference models to the target model, the distribution shift in target data versus population data, and the performance on out-of-distribution data. The limitations of existing MIA tests in these scenarios calls for *robust and efficient* membership inference attacks.

Membership inference attack is a hypothesis testing problem, and attacks are evaluated based on their TPR-FPR trade-off curve. We design a **novel statistical test** for MIA by enumerating the fine-grained plausible worlds associated with the null hypothesis, in which the *target data point could have been replaced with any random sample from the population*. We perform the attack by composing the likelihood ratio (LR) tests of these cases. Our test enhances the differentiation between member and non-member data points, enabling a more precise estimation of test statistics. The computation we propose to compute the LR test statistics is also extremely efficient (as it requires very few reference models) and remains powerful under uncertainties about the data distribution. **Our robust attack method RMIA dominates prior work in all test scenarios, and consistently achieves a high TPR *across all* FPR (even as low as 0), given *any* computation budget.** Another significant aspect of our framework is that many *prior attacks can be framed as simplifications of ours*, shedding light on the causes of their instability and low performance.

[1] National University of Singapore (NUS), CS Department. Correspondence to: Sajjad Zarifzadeh <s.zarif@nus.edu.sg>, Reza Shokri <reza@comp.nus.edu.sg>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

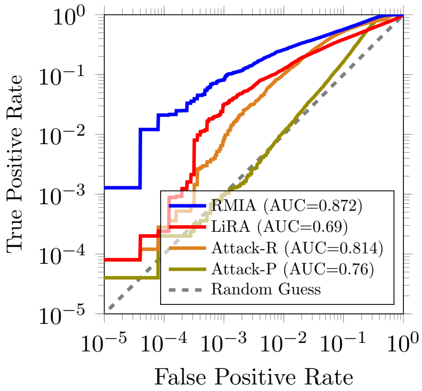


Figure 1: RMIA versus the prior attacks, Attack-P and Attack-R (Ye et al., 2022) and also LiRA (Carlini et al., 2022), on CIFAR-100 models, with the restriction of using only 1 **reference model** (in an offline setting). RMIA outperforms other attacks throughout the TPR-FPR trade-off curve (e.g. by at least 25% higher AUC and an order of magnitude better TPR at zero FPR, compared with LiRA).

We show that RMIA outperforms prior attacks[1] across benchmark datasets[2], by achieving a significantly higher AUC, i.e., TPR throughout all FPR values, and 2× to 4× higher TPR at low FPRs, when using only 1 or 2 reference models. See Figure 1. RMIA's gain is particularly obvious where the adversary exclusively uses pre-trained reference models (i.e., trained independently from the target data).

We also test how shifting the distribution of training data and population data (by using noisy and OOD data), and modifying model architectures, can impact the attack performance. When used as an oracle in reconstruction attacks (Carlini et al., 2021), MIA needs to perform accurately under low FPR regime to filter out the astronomically large number of non-members for discovering members in a high-dimensional space. The vast majority of tested non-members in this application are OOD data. Thus, the advantage of having high TPR at a low FPR primarily comes into play when the attack is evaluated using a large number of non-member (potentially OOD) data for reconstruction attacks. Also, in this setting, *online*[3] MIA methods are useless in practice, as they require training a large number of models per MIA query. Thus, MIA methods need to be both efficient and robust to OOD samples.

We perform extensive tests to analyze the *robustness* of MIAs, and even considering worst-case scenarios, **RMIA consistently outperforms other attacks in all settings**.

2. Performing Membership Inference Attacks

Membership inference attacks (MIA) determine whether a specific data point x was used in the training of a given machine learning model θ. MIA is defined by an indistinguishability game between a challenger and adversary (i.e., privacy auditor). See (Ye et al., 2022) for a comprehensive presentation of MIA games. We use the widely-used game and attack template (Homer et al., 2008; Sankararaman et al., 2009; Shokri et al., 2017; Ye et al., 2022; Carlini et al., 2022; Bertran et al., 2023). The game models random experiments related to two worlds/hypotheses. H_{in}: the model θ was trained on x, and H_{out}: x was not in θ's training set (the null hypothesis). The adversary is randomly placed in one of these two worlds and tasked with inferring which world he is in, using only data point x, the trained model θ, and his background knowledge about the training algorithm and population data distribution.

Definition 2.1 (**Membership Inference Game**). Let π be the data distribution, and let $\mathcal{A}$ be the training algorithm.

i – The challenger samples a training dataset $S \sim \pi$, and trains a model $\theta \sim \mathcal{A}(S)$.

ii – The challenger flips a fair coin b. If $b = 1$, it randomly samples a data point x from S. Otherwise, it samples $x \sim \pi$, such that $x \notin S$. The challenger sends the target model θ and the target data point x to the adversary.

iii – The adversary, having access to the distribution over the population data π, computes $\mathrm{Score}_{\mathrm{MIA}}(x;\theta)$ and uses it to output a membership prediction bit $\hat{b} \leftarrow \mathrm{MIA}(x;\theta)$.

A membership inference attack assigns a membership score $\mathrm{Score}_{\mathrm{MIA}}(x;\theta)$ to every pair of (x,θ), and performs the hypothesis testing by outputting a membership bit through comparing the score with a threshold β:

$$\mathrm{MIA}(x;\theta) = \mathbb{1}_{\mathrm{Score}_{\mathrm{MIA}}(x;\theta) \geq \beta} \tag{1}$$

The adversary's **power** (true positive rate) and **error** (false positive rate) are quantified over numerous repetitions of the MIA game experiment. The threshold β controls the false-positive error the adversary is willing to tolerate (Sankararaman et al., 2009; Murakonda et al., 2021; Ye et al., 2022; Bertran et al., 2023).

The $\mathrm{Score}_{\mathrm{MIA}}(x;\theta)$ and the test equation 1 are designed to maximize the **MIA test performance** as its power (TPR) for any FPR. The (lower-bound for the) *leakage* of the ML algorithm is defined as the power-error trade-off curve (the ROC curve), which is derived from the outcome of the game experiments across all values of β. We primarily compare attacks based on their TPR-FPR curves, but also analyze their computational **efficiency** and their **stability** (i.e., how much their power changes when we vary the data distribution and attacker's computational budget).

[1] We focus on Ye et al. (2022); Carlini et al. (2022); Bertran et al. (2023) that represent prior strong MIA methods.

[2] CIFAR10/100, CINIC10, ImageNet, and Purchase100

[3] We do analyze online attacks in this paper, but we can consider them exclusively as proof-of-concept attacks.

3. Designing RMIA

We propose a novel statistical test for membership inference attacks. We model the null hypothesis (where x is not a member of the training set of θ) as the *composition* of worlds in which the target data point x is replaced by a random data point z sampled from the population. We then compose many **pairwise likelihood ratio tests** each testing the membership of a data point x *relative* to another data point z. To reject the null hypothesis, we need to collect substantial evidence (i.e., a large fraction of population data z) that the probability of observing θ under the hypothesis that x is in its training set is larger than the probability of observing θ when, instead of x, a random z is in the training set. This approach provides a much more fine-grained analysis of leakage, and differentiates between the worlds in which x is not a member (as opposed to relying on the average likelihood of the null hypothesis). We define the likelihood ratio corresponding to the pair of x and z as:

$$\mathrm{LR}_\theta(x, z) = \frac{\Pr(\theta|x)}{\Pr(\theta|z)}, \tag{2}$$

where $\Pr(\theta|.)$ is computed over the randomness of the training algorithm (e.g., SGD). The term $\Pr(\theta|x)$ is the probability that the algorithm produces the model θ given that x was in the training set, while the rest of the training set is randomly sampled from the population distribution π.

Computing the Pairwise Likelihood Ratio. To efficiently compute the pair-wise LR values in the black-box setting (where the adversary can observe the model output), we apply the Bayes rule to compute equation 2:[4]

$$\mathrm{LR}_\theta(x, z) = \left(\frac{\Pr(x|\theta)}{\Pr(x)}\right) \cdot \left(\frac{\Pr(z|\theta)}{\Pr(z)}\right)^{-1} \tag{3}$$

Here, $\Pr(x|\theta)$ is the likelihood function of model θ evaluated on data point x. In the case of classification models, and black-box MIA setting, $\Pr(x|\theta)$ is the prediction score (SoftMax) of output of the model $f_\theta(x_{\text{features}})$ for class x_{label} (MacKay, 2003; Blundell et al., 2015).[5]

It is important to note that $\Pr(x)$ is not the same as $\pi(x)$, which is rather the prior distribution over x. The term $\Pr(x)$ is the normalizing constant in the Bayes rule, and is computed by integrating over all models θ' with the same structure and training data distribution as θ.

$$\begin{aligned}\Pr(x) &= \sum_{\theta'} \Pr(x|\theta') \Pr(\theta') \\ &= \sum_{D,\theta'} \Pr(x|\theta') \Pr(\theta'|D) \Pr(D)\end{aligned} \tag{4}$$

In practice, we compute $\Pr(x)$ as the empirical mean of $\Pr(x|\theta')$ by sampling *reference models* θ', each trained on random datasets D drawn from the population distribution π. As we use only a small number of D, θ' pairs, we need to make sure the reference models are sampled in an *unbiased* way, in particular, with respect to whether x is part of their training data D. Thus, x should be included in the training set of half the reference models (IN models) and be excluded from the training set of the other half (OUT models). This (online attack) is computationally expensive, as customized reference models need to be trained for each MIA query. To avoid this cost, our offline algorithm only computes $\Pr_{OUT}(x)$ by averaging $\Pr(x|\theta')$ over OUT models where $x \notin D$. To approximate the other half, $\Pr_{IN}(x)$, we scale up $\Pr_{OUT}(x)$, as the inclusion of a data point typically increases its probability. See Appendix B.2.2, for the details of computing unbiased $\Pr(x)$ from reference models in both online and offline attack settings. The same computation process applies to $\Pr(z)$.

Constructing RMIA by Composing Pair-Wise LRs. Given $\mathrm{LR}_\theta(x, z)$, we formulate the hypothesis test for our novel membership inference attack RMIA, as follows:

$$\mathrm{Score}_{\mathrm{MIA}}(x; \theta) = \Pr_{z \sim \pi}\big(\mathrm{LR}_\theta(x, z) \geq \gamma\big) \tag{5}$$

We measure the probability that x can γ-*dominate* a random sample z from the population, for threshold $\gamma \geq 1$. The threshold $\gamma \geq 1$ enables us to adjust how much larger the probability of learning θ with x as a training data should be *relative* to a random alternative point z to pass the test.[6] For the simplest setting of $\gamma = 1$, the MIA score reflects the quantile corresponding to $\Pr(x|\theta)/\Pr(x)$ in the distribution of $\Pr(z|\theta)/\Pr(z)$ over random z samples.

We reject the null hypothesis if we find enough fraction of z samples for which the probability of x on target model versus its probability over reference models has a larger gap than that of reference population z (which are not in the training set of θ). The following presents our attack procedure (we provide a detailed pseudo-code in Appendix B.1).

Definition 3.1 (**Robust Membership Inference Attack**). Let θ be the target model, and let x be the target data point. Let γ and β be the MIA test parameters. RMIA determines if x was in the training set of θ, by following these steps:

i – Sample many $z \sim \pi$, and compute $\mathrm{Score}_{\mathrm{MIA}}(x; \theta)$ as the fraction of z samples that pass the pair-wise membership inference likelihood ratio test $\mathrm{LR}_\theta(x, z) \geq \gamma$. See equation 5.

ii – Return MEMBER if $\mathrm{Score}_{\mathrm{MIA}}(x; \theta) \geq \beta$, and NON-MEMBER otherwise. See equation 1.

[4] $\Pr(\theta)$ is canceled from the numerator and denominator.

[5] See Appendix B.2.1 for alternatives for computing $\Pr(x|\theta)$.

[6] Figure 7 shows that the MIA test is not very sensitive to small variations of γ.

By performing the test over all possible values of $\beta \in [0, 1]$, we can compute the ROC power-error trade-off curve. As Figure 8 shows, our test is calibrated in a sense that when γ is set to 1, the expected FPR of the attack is $1 - \beta$. The ability to adjust the attack to achieve a specific FPR is a significant advantage when conducting practical audits to assess the privacy risk of models.

4. Why is RMIA a More Powerful Test Compared with Prior Attacks?

Membership inference attacks, framed as hypothesis tests, essentially compute the *relative* likelihood of observing θ given x's membership in the training set of θ versus observing θ under x's non-membership (null hypothesis). The key to a powerful test is accounting for *all information sources* that distinguish these possible worlds. Membership inference attacks use *references* from the hypothesis worlds, comparing the pair (x, θ) against them. Effectively designing the test involves leveraging all possible informative references, which could be either population data or models trained on them. MIA methods predominantly focus on using reference models. The *way* that such reference models are used matters a lot. As we show in our empirical evaluation, prior attacks (Carlini et al., 2022; Ye et al., 2022) exhibit different behavior depending on the reference models (i.e., in different scenarios, they *dominate each other in opposing ways*). Also, even though they outperform attacks that are based on population data by a large margin, they do not strictly dominate them on all membership inference queries (Ye et al., 2022). They, thus, fall short due to overlooking some type of distinguishing signals.

Table 1 summarizes the MIA scores of various attacks. Our method offers a novel perspective on the problem. This approach leverages both population data and reference models, enhancing attack power and robustness against changes in adversary's background knowledge (e.g. about the distribution of training and population data as well as the structure of the models). Our likelihood ratio test, as defined in equation 5 and equation 3, effectively measures the distinguishability between x and any z based on the shifts in their probabilities when conditioned on θ, through contrasting $\Pr(x|\theta)/\Pr(x)$ versus $\Pr(z|\theta)/\Pr(z)$. The prior work could be seen as *average-case* and *uncalibrated* versions of our test. Attack-P (Ye et al., 2022) and related methods (Shokri et al., 2017; Chang & Shokri, 2021; Bertran et al., 2023) rely primarily on how the likelihood of the target model on x and z change, and neglect or fail at accurately capturing the $\Pr(x)/\Pr(z)$ term of our test (i.e., they implicitly assume $\Pr(x) = \Pr(z)$). The strength of our test lies in its ability to detect subtle differences in $\Pr(x|\theta)/\Pr(z|\theta)$ and $\Pr(z)/\Pr(x)$ LRs, which reflect the noticeable change in LR due to the inclusion of x in the target training set. In our test, a reference data point z would vote for the membership of x only if the ratio $\Pr(x)/\Pr(z)$ is enlarged when point probabilities are computed on the target model θ (which indicates that θ fits x better).

Stronger prior attacks utilizing reference models, especially as seen in (Ye et al., 2022) and similar attacks (Watson et al., 2022b), neglect the $\Pr(z|\theta)/\Pr(z)$ component of our test. Calibration by z would tell us if the magnitude of $\Pr(x|\theta)/\Pr(x)$ is significant (compared to non-members), without which the attacks would under-perform throughout the TPR-FPR curve. LiRA (Carlini et al., 2022) falters in the same way, missing the essential calibration of their test with population data.[7] But, the unreliability of LiRA is not only because of this. To better explain our advantage, we present an alternative method for computing our LR equation 2, which shows LiRA is an average-case of RMIA.

In the black-box setting, the divergence between the output distributions of a model trained on x and its corresponding leave-one-out model (which is not trained on x), when the distribution is computed over the randomness of the training algorithm, is maximum *when* the models are queried on the differing point x (Ye et al., 2024). So, a good approximation for the likelihood ratio in the black-box setting is to evaluate the probability of $f_\theta(x_{\text{features}})$ and $f_\theta(z_{\text{features}})$, where $f_\theta(.)$ is a classification model with parameters θ. A **direct** way to compute LR in equation 2 is the following:

$$\text{LR}_\theta(x, z) = \frac{\Pr(\theta|x)}{\Pr(\theta|z)} \approx \frac{\Pr(f_\theta(x), f_\theta(z)|x)}{\Pr(f_\theta(x), f_\theta(z)|z)}, \tag{6}$$

where the terms can be computed as in Appendix B.3 and (Carlini et al., 2022). LiRA (online) computes the average of this LR over all z, which reduces its test power. In addition, the direct computation of LR requires a large number of reference models for constructing a stable test. We provide an empirical comparison between attack performance of our main computation (equation 3 using Bayes rule) and direct computations of the likelihood ratio in Appendix B.3. The results show that our construction of RMIA using the Bayes rule equation 3 is very robust and dominates the direct computation of LR equation 6 when *a few* reference models are used (Figure 20), and they match when we use a large number of reference models (Figure 19). Thus, our attack strictly dominates LiRA throughout the power-error (TPR-FPR) curve, and the gap increases significantly when we limit the budget for reference models (See Figure 25). The combination of a pairwise LR and its computation using the Bayesian approach results in our robust, high-power, and low-cost attack, which only requires training of OUT models in offline setting.

[7] By applying the Bayes rule on the numerator of LiRA LR and considering that the denominator is the same as $\Pr(\theta)$ when the OUT reference models are sampled from the population (and not a small finite set), the LiRA $\text{Score}_{\text{MIA}}(x; \theta)$ is $\Pr(x|\theta)/\Pr(x)$.

Method	RMIA (this paper)	LiRA (Carlini et al., 2022)	Attack-R (Ye et al., 2022)	Attack-P (Ye et al., 2022)	Global (Yeom et al., 2018)
MIA Score	$\Pr_z\left(\frac{\Pr(\theta\|x)}{\Pr(\theta\|z)} \geq \gamma\right)$	$\frac{\Pr(\theta\|x)}{\Pr(\theta\|\bar{x})}$	$\Pr_{\theta'}\left(\frac{\Pr(x\|\theta)}{\Pr(x\|\theta')} \geq 1\right)$	$\Pr_z\left(\frac{\Pr(x\|\theta)}{\Pr(z\|\theta)} \geq 1\right)$	$\Pr(x\|\theta)$

Table 1: Computation of $\mathrm{Score}_{\mathrm{MIA}}(x;\theta)$ in different membership inference attacks, where the notation $\bar{x}$ (for LiRA) represents the case where x is not in the training set. The attack is $\mathrm{MIA}(x;\theta) = \mathbb{1}_{\mathrm{Score}_{\mathrm{MIA}}(x;\theta)\geq\beta}$ based on Definition 2.1.

5. Empirical Evaluation

In this section, we present a comprehensive empirical evaluation of RMIA and compare its performance with the prior state-of-the-art attacks. Our goal is to analyze:

1. Performance of attacks under *limited computation resources* for training reference models. This includes limiting the attack to the offline mode where reference models are pre-trained.
2. *Ultimate power of attacks* when unlimited number of reference models could be trained (in online mode).
3. The strength of the attacks in distinguishing members from non-members when target data points are *out-of-distribution*. This reflects the *robustness and usefulness* of attacks for acting as an oracle for partitioning the entire data space into members and non-members.
4. Impact of *adversary's knowledge* on the performance of attacks (in particular data distribution shift about the population data, and mismatch of network architecture between target and reference models).

Setup. We perform our experiments on CIFAR-10, CIFAR-100, CINIC-10, ImageNet and Purchase-100, which are benchmark datasets commonly used for MIA evaluations. We use the standard metrics, notably the FPR versus TPR curve, and the area under the ROC curve (AUC), for analyzing attacks. The setup of our experiments, including the description of datasets, models, hyperparameters, and metrics, is presented in Appendix A.

Attack Modes (offline vs. online). Reference models are trained on population samples (Homer et al., 2008; Sankararaman et al., 2009; Shokri et al., 2017; Ye et al., 2022; Watson et al., 2022b). Adversaries can also simulate the leave-one-out scenario (Ye et al., 2024), and perform online attacks by training some reference models on the target data (MIA query) (Carlini et al., 2022). We consider online attacks as proof-of-concept attacks, due to their high cost in practical scenarios (e.g., reconstruction attacks). We refer to models trained on the target data as IN models, and to those that are not trained on it as OUT models. In the offline mode, all reference models are OUT models.

Baseline Attacks. We mainly compare the performance of RMIA with the state-of-the-art attacks (Ye et al., 2022; Carlini et al., 2022) which had been shown to outperform their prior methods (Watson et al., 2022b; Shokri et al., 2017; Song & Mittal, 2021; Sablayrolles et al., 2019; Long et al., 2020; Yeom et al., 2018; Jayaraman et al., 2020). In summary, we use Attack-P (Ye et al., 2022) as a baseline attack that does not use any reference models. Note that Attack-P is equivalent to the LOSS attack (Yeom et al., 2018) which operates by setting threshold on the loss signal from the target model. The both attacks test the same statistic (loss or probability), and the threshold for obtaining a test for a given FPR needs to be set using the population data. We also compare our results with Attack-R (Ye et al., 2022) which uses reference models in an offline mode, and LiRA (Carlini et al., 2022) which is an inference attack in online and offline modes. We also compare the results with Quantile Regression (Bertran et al., 2023) (which improves Attack-P attack in a similar way as (Chang & Shokri, 2021) does) outperforming (Carlini et al., 2022) in some scenarios on ImageNet, due to the high uncertainty of the test signal used by (Carlini et al., 2022) on large models. However, (Bertran et al., 2023)[Table 4] shows relative weakness of the attack on other benchmark datasets, even when baseline (Carlini et al., 2022) uses a few reference models.

Reproducing the Attacks. To reproduce the results for the prior work, we exclusively use the attacks' implementation as provided by the authors.[8] [9] [10] We would like to highlight the discrepancy in the empirical results we have obtained for the offline version of LiRA (using the authors' code) which is also shown in the prior work (Ye et al., 2022; Wen et al., 2023). The obtained attack power is much lower than what is presented in (Carlini et al., 2022).

Our attack RMIA operates in both offline and online modes (See Appendix B.2.2). The reader can reproduce our results using our source code.[11]

[8] https://github.com/privacytrustlab/ml_privacy_meter/tree/master/research/2022_enhanced_mia (Ye et al., 2022)

[9] https://github.com/tensorflow/privacy/tree/master/research/mi_lira_2021 (Carlini et al., 2022)

[10] https://github.com/amazon-science/quantile-mia (Bertran et al., 2023)

[11] https://github.com/privacytrustlab/ml_privacy_meter/tree/master/research/2024_rmia (RMIA)

# Ref	Attack	CIFAR-10			CIFAR-100			CINIC-10			ImageNet		
		AUC	TPR@FPR		AUC	TPR@FPR		AUC	TPR@FPR		AUC	TPR@FPR	
			0.01%	0.0%		0.01%	0.0%		0.01%	0.0%		0.01%	0.0%
0	Attack-P	58.19	0.01	0.0	75.91	0.01	0.0	66.91	0.01	0.0	64.48	0.01	0.0
*	Quantile-Reg.	61.45	0.08	0.03	83.32	0.26	0.04	73.48	0.35	0.12	70.8	0.06	0.0
1	Attack-R	63.65	0.07	0.02	81.61	0.06	0.02	72.04	0.07	0.02	70.91	0.02	0.0
	LiRA	53.2	0.48	0.25	68.95	0.54	0.27	59.93	0.32	0.07	58.66	0.01	0.0
	RMIA	**68.64**	**1.19**	**0.51**	**87.18**	**2.06**	**0.77**	**79**	**0.86**	**0.32**	**72.21**	**0.06**	**0.01**
2	Attack-R	63.35	0.32	0.08	81.52	0.31	0.06	72.02	0.21	0.07	71.03	0.04	0.01
	LiRA	54.42	0.67	0.27	72.21	1.52	0.76	62.18	0.57	0.26	61.05	0.01	0.0
	LiRA (Online)	63.97	0.76	0.43	84.55	1.15	0.55	73.17	0.53	0.12	67.55	0.01	0.01
	RMIA	**70.13**	**1.71**	**0.91**	**88.92**	**4.9**	**1.73**	**80.56**	**2.14**	**0.98**	**73.95**	**0.16**	**0.02**
4	Attack-R	63.52	0.65	0.21	81.78	0.63	0.19	72.18	0.4	0.14	71.11	0.07	0.02
	LiRA	54.6	0.97	0.57	73.57	2.26	1.14	63.07	1.03	0.45	62.69	0.01	0.0
	LiRA (Online)	67	1.38	0.51	87.82	3.64	2.19	77.06	1.34	0.51	71.44	0.01	0.0
	RMIA	**71.02**	**2.91**	**2.13**	**89.81**	**7.05**	**3.5**	**81.46**	**3.2**	**1.39**	**74.98**	**0.45**	**0.06**

Table 2: Performance of attacks where a **few reference models** are used. All attacks, except LiRA, are *offline* (do not need new reference models to be trained per query). The Quantile Regression attack trains regression-based attack models (instead of reference models) to compute the attack threshold for a certain FPR value. For tuning of hyper-parameters in this attack, Bertran et al. (2023) suggested training several models which has a considerable overhead compared to training a reference model. Results are averaged over 10 random target models (Results with standard deviations are shown in Table 9 in Appendix C.5). The result of Purchase-100 is illustrated in Table 10 in Appendix C.8.

5.1. Inference Attack under Low Computation Budget

Using a Few Reference Models. Figure 1 shows the TPR versus FPR tradoff curves for all attacks when the number of reference models is limited to 1. RMIA outperforms Attack-R (Ye et al., 2022) and LiRA (Carlini et al., 2022) across all FPR values. Table 2 compares the result of attacks in a low-cost scenario where the adversary has access to only a limited number of reference models. Our focus is on attacks in offline mode, where a fixed number of reference models are pre-trained and used to perform the inference on all queries. We also include LiRA (Online) as a benchmark, despite its high computation cost (half reference models need to be trained on the target data).

The table also incorporates results from the Quantile-Regression attack by Bertran et al. (2023). Although this method does not rely on a reference model, it instead involves training several regression-based models to determine the attack threshold for each sample (to reach a certain FPR value) and optimizing its hyper-parameters through fine-tuning. The attack can be considered a computationally expensive method as it requires many models to be trained to fine-tune its hyper-parameters, and in the end one attack model needs to be constructed for each FPR.

Our proposed RMIA demonstrates its strict dominance over the prior work across all datasets. For instance, with only 2 CIFAR-10 reference models, it achieves around 10% higher AUC than Attack-R and LiRA and still gains a better TPR at zero FPR. It is important to highlight that, RMIA (in offline mode) also dominates LiRA (Online) . For example, with 4 CIFAR-10 models, it has at least 6% higher AUC and a significant 3× improvement for TPR at zero FPR, compared to LiRA (Online) . In the extreme case of using one single reference model, RMIA shows at least 24% higher AUC and also a better TPR at low FPRs than LiRA over all datasets. In this case, our attack outperforms the Quantile-Regression attack with respect to both AUC and TPR at low FPR. Moreover, it is important to highlight that we can further improve the performance of RMIA by adding more reference models, but there is no room for such an improvement in the Quantile-Regression attack. Attack-R outperforms LiRA (Offline) and results in a relatively high AUC, but it does not perform as well at lower FPR regions. Also, Attack-P does not use any reference models, yet it outperforms LiRA in the offline mode.

Using Population Data. When the number of reference models is low, all attacks must use information from population data to operate effectively. Specifically, LiRA necessitates computing the mean and variance of rescaled-logits across reference models. In scenarios with a limited number of reference models (i.e., below 64 models as indicated by Carlini et al. (2022)[Figure 9]), employing a global variance over the population/test data proves advantageous. Similarly, Attack-R employs smoothing techniques to approximate the CDF for the loss distribution across population/test data to compute percentiles (Ye et al., 2022). The Quantile-Regression attack (Bertran et al., 2023) also uses a large set of population data to train the attack model. Consequently, there is no significant difference between attacks regarding their demand for additional population data.

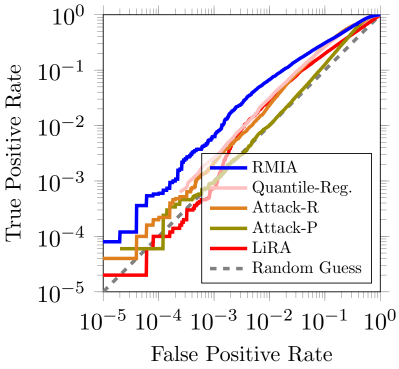


Figure 2: ROC of attacks against **ImageNet** models. The result is obtained on one random target model. We use **1 reference model** (OUT). Table 2 reports AUC of attacks.

Attacking Larger Models with Large Training Sets. As denoted by Bertran et al. (2023), membership inference attacks might experience performance deterioration against models with larger training set. To study this, Table 2 also presents the outcomes of attacks against models trained on the ImageNet dataset, which is 20 times larger than CIFAR datasets. The Quantile-Regression attack outperforms LiRA, even under online LiRA conditions with a couple of reference models. However, the efficacy of offline RMIA with only one reference model surpasses all other attacks in terms of both AUC and TPR at low FPRs, e.g. showing approximately 2% higher AUC compared to the Quantile-Regression attack. Furthermore, the performance gap between our attack and the Quantile-Regression attack widens considerably with the inclusion of additional reference models. Figure 2 shows the ROC of attacks on ImageNet models when we use 1 reference model. RMIA consistently obtains superior TPR across all FPR values.

Using More Reference Models in the Offline Mode. Table 3 compares the performance of offline attacks when using a larger number of OUT models (127 models). The RMIA consistently outperforms other offline attacks across all datasets. RMIA has a much larger power than the Attack-R (which is the strongest offline attack in the prior work). Attack-R may wrongly reject a typical member as a non-member solely because it has a higher probability in reference models. As a result, it yields 5%-10% lower AUC than RMIA across all datasets. RMIA is designed to overcome these limitations by considering both the characteristics of the target sample within reference models and its relative probability among other population records.

Comparing Table 2 and Table 3 shows that the AUC of RMIA when using a few reference models is almost the same as that of using a large number (127) of models. This shows that the overall power of the attack is not significantly dependent on having many reference models. However, when we increase the number of reference models the attack TPR in low FPR regions increases.

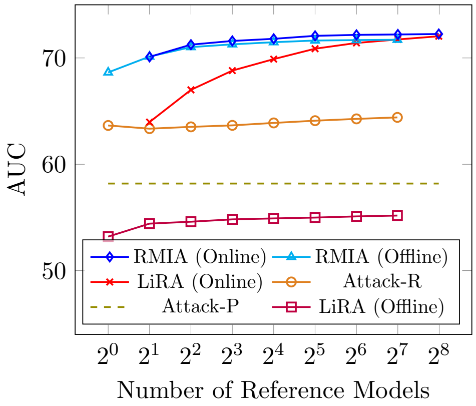


Figure 3: Number of reference models versus the AUC of the attacks on CIFAR-10. In online attacks, half of reference models need to be trained per each MIA query.

To better analyze the difference between the attacks, we compare the variation in MIA scores of member and non-member samples across all attacks in Appendix C.10.

5.2. Ultimate Power of (Online) Inference Attacks

Table 3 presents the performance of all attacks on models trained with different datasets where we have enough resources to train many (254 IN and OUT) reference models. In this setting, RMIA (Online) performs better than LiRA (Online) (which gains significantly from a large number of reference models) with respect to AUC. In this case, even minor AUC improvements are particularly significant when approaching the maximum leakage of the training algorithm in the corresponding MIA game. What is, however, more important to note is that *RMIA (Online) 's TPR at zero FPR is significantly better than that of LiRA (Online) (by up to 50%)*. It is important to note that our offline attack achieves performance comparable to online attacks, which is quite remarkable, considering the large gap between the cost associated with offline and online attacks.

We can further improve attacks by querying the target model with multiple *augmentations* of input query, obtained via simple mirror and shift operations on image data. In the multi-query setting, we use majority voting on our hypothesis test in equation 2: query x is considered to dominate population record z if more than half of all augmentations of x dominate z. See Appendix B.5 for the details. RMIA can leverage this technique to a significantly greater extent, and can achieve *a* $4\times$ *improvement in TPR at zero FPR and about 4.6% higher AUC* compared to LiRA. Our results are based on applying 18 augmented queries.

Attack	CIFAR-10			CIFAR-100			CINIC-10		
	AUC	TPR@FPR		AUC	TPR@FPR		AUC	TPR@FPR	
		0.01%	0.0%		0.01%	0.0%		0.01%	0.0%
Attack-P	58.19 ± 0.33	0.01 ± 0.01	0.00 ± 0.01	75.91 ± 0.36	0.01 ± 0.01	0.0 ± 0.0	66.91 ± 0.3	0.01 ± 0.01	0.0 ± 0.0
Attack-R	64.41 ± 0.41	1.52 ± 0.33	0.80 ± 0.43	83.37 ± 0.24	4.8 ± 0.75	2.59 ± 1.35	73.64 ± 0.34	2.17 ± 0.76	1.12 ± 0.85
LiRA	55.18 ± 0.37	1.37 ± 0.32	0.72 ± 0.31	75.78 ± 0.33	2.53 ± 1.23	1.13 ± 1.23	64.51 ± 0.51	1.33 ± 0.33	0.6 ± 0.38
RMIA	$\mathbf{71.71 \pm 0.43}$	$\mathbf{4.18 \pm 0.61}$	$\mathbf{3.14 \pm 0.87}$	$\mathbf{90.57 \pm 0.15}$	$\mathbf{11.45 \pm 2.6}$	$\mathbf{6.16 \pm 2.8}$	$\mathbf{82.33 \pm 0.32}$	$\mathbf{5.07 \pm 1.77}$	$\mathbf{3.33 \pm 1.47}$
LiRA (online)	72.04 ± 0.47	3.39 ± 0.86	2.01 ± 0.78	$\mathbf{91.48 \pm 0.16}$	10.85 ± 1.69	7.13 ± 2.19	82.44 ± 0.3	4.43 ± 1.54	2.92 ± 1.68
RMIA (online)	$\mathbf{72.25 \pm 0.46}$	$\mathbf{4.31 \pm 0.47}$	$\mathbf{3.15 \pm 0.61}$	91.01 ± 0.14	$\mathbf{11.35 \pm 2.21}$	$\mathbf{7.78 \pm 3.03}$	$\mathbf{82.7 \pm 0.35}$	$\mathbf{6.77 \pm 1.03}$	$\mathbf{4.43 \pm 1.38}$

Table 3: Performance of attacks using a *large number of reference models*, averaged over 10 random target models. The top part (the first four rows) represents offline attacks, using 127 OUT reference models (except Attack-P that uses no reference models). The second part (the bottom two rows) represents online attacks, using 127 OUT and 127 IN models.

5.3. Dependency on Availability of Reference Models

Inference attacks must be robust to changes in their assumptions about the prior knowledge of adversary, for example population data available to train reference models. Low-cost MIA methods should also be less dependent on availability of a large number of reference models.

Figure 3 presents the number of reference models needed to achieve an overall performance for attacks (computed using their AUC). Note that online attacks need at least 2 reference models (1 IN, 1 OUT). As opposed to other strong attacks, RMIA obtains stable results that do not change significantly when reducing the number of reference models. RMIA does gain from increasing the number of reference models, but even with a small number of models, it is very close to its maximal overall performance. However, LiRA (Online) displays a great sensitivity to the changes in number of reference models (with a huge AUC gap of about 8% in CIFAR-10 models when comparing 2 models versus 254 models), underscoring the necessity of a large number of models for this attack to function effectively. Also it is remarkable to note that the RMIA (Offline) is stronger than LiRA (Online) attacks unless when use hundreds of reference models. Appendix C.7 presents additional results obtained with various number of models trained on other datasets, showing a comparable superiority among attacks similar to what was observed with CIFAR-10.

5.4. Robustness of Attacks against OOD Non-members

We challenge membership inference attacks by testing them with non-member out-of-distribution (OOD) data. A strong MIA should be able to rule out *all non-members* regardless of whether they are from the same distribution as its training data or not. This is of a great importance also in scenarios where we might use MIA oracles in applications such as data extraction attacks (Carlini et al., 2021). In such cases, it is essential for the attack to remain accurate (high TPR for all FPR) on out-of-distribution (OOD) non-member data, while detecting members. Note that naive filtering techniques cannot be used to solve this problem. While OOD samples generally exhibit lower confidence levels compared to in-distribution samples, filtering them just based on confidence leads to a low TPR by rejecting hard member samples.

To examine the robustness and effectiveness of attacks in presence of OOD samples, we train our models with CIFAR-10 and use samples from a different dataset (CINIC-10) to generate OOD non-member test queries. The setting of the MIA game described here diverges from the game outlined in Definition 2.1, due to $x \not\sim \pi$. Consequently, the outcomes cannot be directly compared with those of the original game. We focus on offline attacks, as it is practically infeasible to train IN models for each and every individual OOD test sample, especially when dealing with a large volume of queries.

Figure 4 illustrates the ROC curves. RMIA has a substantial performance advantage (at least 21% higher AUC) over Attack-R, with LiRA not performing much better than random guessing. Attack-P demonstrates strong AUC performance by capitalizing on the disparity between confidence/loss of OOD and in-distribution samples. However, as explained above, it struggles at achieving good TPR at low FPRs. It is also not a powerful method to detect in-distribution non-members, as shown in the results in previous figures and tables. Conducting our hypothesis test by assessing the pair-wise LR between the target sample x and several *in-distribution* z samples enables us to capture significant differences between $\frac{\Pr(x|\theta)}{\Pr(x)}$ and $\frac{\Pr(z|\theta)}{\Pr(z)}$ in equation 3 to effectively distinguish OOD queries from members. We also present the results of using pure noise as non-member test queries in Figure 9 (Appendix C.1), revealing a wider disparity between RMIA and other methods compared to OOD samples

5.5. Robustness to Model and Data Distribution Shift

When assessing membership inference attacks, it is crucial to examine how the attack's performance is influenced when the adversary lacks precise knowledge about the distribution of training data and also the structure of the tar-

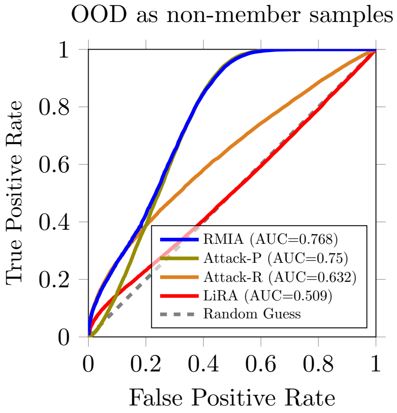


Figure 4: ROC of offline attacks using models trained on CIFAR-10, while non-member test queries are OOD samples from CINIC-10. We use 127 reference models.

get model. Therefore, we compare the result of attacks when the reference models are trained on different datasets than the target models. More specifically, the target models are trained on CIFAR-10, while the reference models are trained on CINIC-10. The performance of all attacks is affected when there is a data distribution shift between the training set of the target model and the reference models. However, compared with other attacks, RMIA always obtains a higher AUC (e.g. by up to 25% in comparison with LiRA, using 2 reference models) and a better TPR at low FPRs. This confirms that our attack remains effective when pre-trained models on different datasets are used as reference models, requiring zero training cost. Table 8 in Appendix C.2 presents the detailed result of this experiment.

We also study the impact of network architecture change between the target model and the reference models. While the optimal performance of all attacks is noted when both the target and reference models share a similar architecture, the superiority of our attack becomes more pronounced in the presence of architecture shifts (we observe up to 3% increase in the AUC gap between our attack and others). See Appendix C.3 for the details of the empirical results.

5.6. Analyzing RMIA Parameters

Number of z samples. Our attack evaluates the likelihood ratio of the target model on a target data x versus other population samples z. Computing the LR versus the population samples enables us to compute reliable test statistics for our attack. Table 4 shows how the result of RMIA changes, as we use different number of reference samples. The AUC increases when we increase the number of z samples, but it is noteworthy that using 2500 population samples, equivalent to 10% of the size of the models' training set, yields results comparable to those obtained with a 10 fold larger population set. Moreover, the TPR at low FPRs is still high even when we consider just 250 reference samples. The trend of results remains consistent when using fewer reference models (See Table 11 in Appendix C.9). In the default setting, when performing an attack on a target model with sample x, we use all non-members of the target model as the set of z samples. We exclude the query x itself from being used as z.

# population samples z	AUC	TPR@FPR	
		0.01%	0.0%
25 samples	58.71 ± 0.26	1.32 ± 0.15	1.15 ± 0.34
250 samples	64.88 ± 0.24	2.23 ± 0.38	1.65 ± 0.46
1250 samples	67.57 ± 0.30	2.22 ± 0.41	1.73 ± 0.49
2500 samples	68.25 ± 0.27	2.26 ± 0.45	1.72 ± 0.49
6250 samples	68.78 ± 0.31	2.28 ± 0.44	1.75 ± 0.51
12500 samples	69.03 ± 0.33	2.28 ± 0.43	1.77 ± 0.52
25000 samples	69.15 ± 0.35	2.26 ± 0.43	1.80 ± 0.55

Table 4: Performance of RMIA (Online) using different number of random z samples. The number of reference models, trained on CIFAR-10, is 254 (127 IN, 127 OUT). Here, we do not use augmented queries. Results are averaged over 10 random target models.

Sensitivity to Pair-Wise LR Test Threshold γ. The result of our experiments, presented in Appendix B.4, shows that our attack's performance is consistent against changes in the value of γ. Both AUC and FPR-TPR curve remain relatively stable with small changes to γ (except for a considerably high value of γ). In fact, by adjusting the value of the threshold β, we achieve roughly the same result across different γ values.

5.7. Applying RMIA to Other ML Algorithms

To assess how attacks perform against alternative ML algorithms, we investigate the privacy risks of Gradient Boosting Decision Tree (GBDT) algorithms. In this case, the TPR obtained by our attack consistently outperforms all other attacks, particularly by an order of magnitude at zero FPR, as demonstrated in Figure 15 (Appendix C.6).

6. Conclusions

We argue that MIA tests, as privacy auditing tools, must be stress-tested with low computation budget, few available reference models, and changes to data distribution and models. A strong test is the one that can outperform others in these scenarios, and not only in typical scenarios. We present a novel statistical test for MIA, and a series of evaluation scenarios to compare MIAs based on their efficiency and robustness. **RMIA can be reliably used to audit privacy risks in ML under realistic practical assumptions.**

Impact Statement

The goal of this paper is to advance the field of Machine Learning and data privacy. There are many potential societal benefits of our work, including advancement of privacy auditing methods required in privacy regulations.

Acknowledgements

The authors would like to thank Vincent Bindschaedler, Jiayuan Ye, and Hongyan Chang for their comments on the earlier versions of this work. The authors would also like to thank the reviewers for their valuable comments and for a very engaging discussion during the rebuttal process. The work of Reza Shokri was supported by the Asian Young Scientist Fellowship 2023, and the Ministry of Education, Singapore, Academic Research Fund (AcRF) Tier 1.

References

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 23rd ACM SIGSAC Conference on Computer and Communications Security (CCS'16)*, pp. 308–318, 2016.

Backes, M., Berrang, P., Humbert, M., and Manoharan, P. Membership privacy in microrna-based studies. In *Proceedings of the 23rd ACM SIGSAC Conference on Computer and Communications Security (CCS'16)*, pp. 319–330, 2016.

Banerjee, K., Prasad, V. C., Raj Gupta, R., Vyas, K., H, A., and Mishra, B. Exploring alternatives to softmax function. In *Proceedings of the 2nd International Conference on Deep Learning Theory and Applications (DeLTA'21)*, pp. 81–86, 2021.

Bertran, M., Tang, S., Kearns, M., Morgenstern, J., Roth, A., and Wu, Z. S. Scalable membership inference attacks via quantile regression. In *Proceedings of the 37th Conference on Neural Information Processing Systems (NeurIPS'23)*, 2023.

Blundell, C., Cornebise, J., Kavukcuoglu, K., and Wierstra, D. Weight uncertainty in neural network. In *International conference on machine learning*, pp. 1613–1622. PMLR, 2015.

Carlini, N., Liu, C., Erlingsson, , Kos, J., and Song, D. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th USENIX Security Symposium (USENIX Security'19)*, pp. 267–284, 2019.

Carlini, N., Tramer, F., Wallace, E., Jagielski, M., Herbert-Voss, A., Lee, K., Roberts, A., Brown, T., Song, D., Erlingsson, U., and et al. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security'21)*, 2021.

Carlini, N., Chien, S., Nasr, M., Song, S., Terzis, A., and Tramer, F. Membership inference attacks from first principles. In *IEEE Symposium on Security and Privacy (S&P'22)*, pp. 1897–1914, 2022.

Chang, H. and Shokri, R. On the privacy risks of algorithmic fairness. In *2021 IEEE European Symposium on Security and Privacy (EuroS&P)*, pp. 292–303. IEEE, 2021.

Chen, D., Yu, N., and Fritz, M. Relaxloss: Defending membership inference attacks without losing utility. In *Proceedings of the 10th International Conference on Learning Representations (ICLR'22)*, 2022.

Chen, M., Zhang, Z., Wang, T., Backes, M., Humbert, M., and Zhang, Y. When machine unlearning jeopardizes privacy. In *Proceedings of the 28th ACM SIGSAC Conference on Computer and Communications Security (CCS '21)*, pp. 896–911, 2021.

Choquette-Choo, C. A., Tramer, F., Carlini, N., and Papernot, N. Label-only membership inference attacks. In *Proceedings of the 38th International Conference on Machine Learning (ICML'21)*, pp. 1964–1974, 2021.

De Brebisson, A. and Vincent, P. An exploration of softmax alternatives belonging to the spherical loss family. In *Proceedings of the 4th International Conference on Learning Representations (ICLR'16)*, 2016.

Dwork, C. Differential privacy. In *Proceedings of 33rd International Colloquium in Automata, Languages and Programming (ICALP'06)*, pp. 1–12, 2006.

Dwork, C., Smith, A., Steinke, T., Ullman, J., and Vadhan, S. Robust traceability from trace amounts. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, pp. 650–669. IEEE, 2015.

Fredrikson, M., Jha, S., and Ristenpart, T. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (CCS'15)*, pp. 1322–1333, 2015.

Ganju, K., Wang, Q., Yang, W., Gunter, C. A., and Borisov, N. Property inference attacks on fully connected neural networks using permutation invariant representations. In *Proceedings of the 25th ACM SIGSAC Conference on Computer and Communications Security (CCS'18)*, pp. 619–633, 2018.

He, K., Zhang, X., Ren, S., , and Sun, J. Deep residual learning for image recognition. In *Proceedings of* *the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16)*, 2016.

Hisamoto, S., Post, M., and Duh, K. Membership inference attacks on sequence-to-sequence models: Is my data in your machine translation system? *Transactions of the Association for Computational Linguistics*, 8:49–63, 2020.

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.

Jayaraman, B., Wang, L., Knipmeyer, K., Gu, Q., and Evans, D. Revisiting membership inference under realistic assumptions. *arXiv preprint arXiv:2005.10881*, 2020.

Jia, J., Salem, A., Backes, M., Zhang, Y., and Gong, N. Z. Memguard: Defending against black-box membership inference attacks via adversarial examples. In *Proceedings of the 26th ACM SIGSAC Conference on Computer and Communications Security (CCS'19)*, pp. 259–274, 2019.

Leemann, T., Pawelczyk, M., and Kasneci, G. Gaussian membership inference privacy. In *Proceedings of the 37th Conference on Neural Information Processing Systems (NeurIPS'23)*, 2023.

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

Li, J., Li, N., and Ribeiro, B. Membership inference attacks and defenses in classification models. In *Proceedings of the 11th ACM Conference on Data and Application Security and Privacy (CODASPY'21)*, pp. 5–16, 2021.

Li, Z. and Zhang, Y. Membership leakage in label-only exposures. In *Proceedings of the 28th ACM SIGSAC Conference on Computer and Communications Security (CCS'21)*, pp. 880–895, 2021.

Li, Z., Liu, Y., He, X., Yu, N., Backes, M., and Zhang, Y. Auditing membership leakages of multi-exit networks. In *Proceedings of the 29th ACM SIGSAC Conference on Computer and Communications Security (CCS'22)*, pp. 1917–1931, 2022.

Liang, X., Wang, X., Lei, Z., Liao, S., and Z., L. S. Soft-margin softmax for deep classification. In *Proceedings of the 24th International Conference on Neural Information Processing (ICONIP'17)*, pp. 413–421, 2017.

Liu, W., Wen, Y., Yu, Z., and Yang, M. Large-margin softmax loss for convolutional neural networks. In *Proceedings of the 33rd International Conference on Machine Learning (ICML'16)*, 2016.

Liu, Y., Wen, R., He, X., Salem, A., Zhang, Z., Backes, M., Cristofaro, E. D., Fritz, M., and Zhang, Y. Ml-doctor: Holistic risk assessment of inference attacks against machine learning models. In *31st USENIX Security Symposium (USENIX Security'22)*, pp. 4525–4542, 2022a.

Liu, Y., Zhao, Z., Backes, M., and Yang, Z. Membership inference attacks by exploiting loss trajectory. In *Proceedings of the 29th ACM SIGSAC Conference on Computer and Communications Security (CCS'22)*, pp. 2085–2098, 2022b.

Long, Y., Wang, L., Bu, D., Bindschaedler, V., Wang, X., Tang, H., Gunter, C. A., and Chen, K. A pragmatic approach to membership inferences on machine learning models. In *IEEE European Symposium on Security and Privacy (EuroS&P'20)*, pp. 521–534, 2020.

MacKay, D. J. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

Melis, L., Song, C., De Cristofaro, E., and Shmatikov, V. Exploiting unintended feature leakage in collaborative learning. In *IEEE Symposium on Security and Privacy (S&P'19)*, pp. 691–706, 2019.

Murakonda, S. K., Shokri, R., and Theodorakopoulos, G. Quantifying the privacy risks of learning high-dimensional graphical models. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pp. 2287–2295, 2021.

Nasr, M., Shokri, R., and Houmansadr, A. Machine learning with membership privacy using adversarial regularization. In *Proceedings of the 25th ACM SIGSAC Conference on Computer and Communications Security (CCS'18)*, pp. 634–646, 2018.

Nasr, M., Shokri, R., and Houmansadr, A. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *IEEE Symposium on Security and Privacy (S&P'19)*, pp. 1022–1036, 2019.

Nasr, M., Song, S., Thakurta, A., Papernot, N., and Carlini, N. Adversary instantiation: Lower bounds for differentially private machine learning. In *IEEE Symposium on Security and Privacy (S&P'21)*, pp. 866–882, 2021.

Rahman, M. A., Rahman, T., Laganiere, R., Mohammed, N., and Wang, Y. Membership inference attack against differentially private deep learning model. *Transactions on Data Privacy*, 11(1):61–79, 2018.

Sablayrolles, A., Douze, M., Schmid, C., Ollivier, Y., and Jégou, H. White-box vs black-box: Bayes optimal strategies for membership inference. In *Proceedings of the 36th International Conference on Machine Learning (ICML'19)*, pp. 5558–5567, 2019.

Salem, A., Zhang, Y., Humbert, M., Fritz, M., and Backes, M. Ml-leaks: Model and data independent membership inference attacks and defenses on machine learning models. In *Network and Distributed Systems Security Symposium (NDSS'19)*, 2019.

Sankararaman, S., Obozinski, G., Jordan, M. I., and Halperin, E. Genomic privacy and limits of individual detection in a pool. *Nature genetics*, 41(9):965–967, 2009.

Shokri, R., Stronati, M., Song, C., and Shmatikov, V. Membership inference attacks against machine learning models. In *IEEE Symposium on Security and Privacy (S&P'17)*, pp. 3–18, 2017.

Song, L. and Mittal, P. Systematic evaluation of privacy risks of machine learning models. In *30th USENIX Security Symposium (USENIX Security'21)*, pp. 2615–2632, 2021.

Steinke, T., Nasr, M., and Jagielski, M. Privacy auditing with one (1) training run. *Proceedings of the 37th Conference on Neural Information Processing Systems (NeurIPS'23)*, 2023.

Tang, X., Mahloujifar, S., Song, L., Shejwalkar, V., Nasr, M., Houmansadr, A., and Mittal, P. Mitigating membership inference attacks by self-distillation through a novel ensemble architecture. In *31st USENIX Security Symposium (USENIX Security'22)*, pp. 1433–1450, 2022.

Thudi, A., Shumailov, I., Boenisch, F., and Papernot, N. Bounding membership inference. In *arXiv preprint*, pp. arXiv:2202.12232, 2022.

Truex, S., Liu, L., Gursoy, M. E., Yu, L., and Wei, W. Demystifying membership inference attacks in machine learning as a service. *IEEE Transactions on Services Computing*, 14(6):2073–2089, 2019.

Visscher, P. M. and Hill, W. G. The limits of individual identification from sample allele frequencies: theory and statistical analysis. *PLoS genetics*, 5(10):e1000628, 2009.

Watson, L., Guo, C., Cormode, G., and Sablayrolles, A. On the importance of difficulty calibration in membership inference attacks. In *Proceedings of International Conference on Learning Representations (ICLR'22)*, 2022a.

Watson, L., Guo, C., Cormode, G., and Sablayrolles, A. On the importance of difficulty calibration in membership inference attacks. In *Proceedings of the 10th International Conference on Learning Representations (ICLR'22)*, 2022b.

Wen, Y., Bansal, A., Kazemi, H., Borgnia, E., Goldblum, M., Geiping, J., and Goldstein, T. Canary in a coalmine: Better membership inference with ensembled adversarial queries. In *Proceedings of the 11th International Conference on Learning Representations (ICLR'23)*, 2023.

Xu, Z., Shi, S., Liu, A. X., Zhao, J., and Chen, L. An adaptive and fast convergent approach to differentially private deep learning. In *Proceedings of 39th IEEE International Conference on Computer Communications (Infocom'20)*, pp. 1867–1876, 2020.

Ye, J., Maddi, A., Murakonda, S. K., Bindschaedler, V., and Shokri, R. Enhanced membership inference attacks against machine learning models. In *Proceedings of the 29th ACM SIGSAC Conference on Computer and Communications Security (CCS'22)*, pp. 3093–3106, 2022.

Ye, J., Borovykh, A., Hayou, S., and Shokri, R. Leave-one-out distinguishability in machine learning. *12th International Conference on Learning Representations (ICLR'24)*, 2024.

Yeom, S., Giacomelli, I., Fredrikson, M., and Jha, S. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st Computer Security Foundations Symposium (CSF'18)*, pp. 268–282. IEEE, 2018.

Yu, L., Liu, L., Pu, C., Gursoy, M. E., and Truex, S. Differentially private model publishing for deep learning. In *IEEE Symposium on Security and Privacy (S&P'19)*, pp. 332–349, 2019.

Zhang, C., Ippolito, D., Lee, K., Jagielski, M., Tramèr, F., and Carlini, N. Counterfactual memorization in neural language models. *arXiv preprint arXiv:2112.12938*, 2021.

Contents

A. Experimental Setup

To conduct attacks, we must first train models. We adopt the same training setup as in (Carlini et al., 2022), wherein, for a given dataset, we train our models on randomly selected training sets, with each set containing half of the dataset. Moreover, each sample of the dataset is included in exactly half of the reference models' training set. We pick our target models from the set of trained models at random. It is worth noting that in this setting, the training set of a target model can overlap by 50% with each reference model.

Datasets. We report the attack results on models trained on five different datasets, traditionally used for membership inference attack evaluations. For CIFAR-10 (He et al., 2016) (a traditional image classification dataset), we train a Wide ResNets (with depth 28 and width 2) for 100 epochs on half of the dataset chosen at random. For CIFAR-100 and CINIC-10 (as other image datasets), we follow the same process as for CIFAR-10 and train a wide ResNet on half of the dataset[12] We set the batch size to 256. We assess the impact of attacks on larger datasets by examining the ImageNet dataset, comprising approximately 1.2 million images with 1000 class labels. We train the ResNet-50 on half of the dataset for 100 epochs, with a batch size of 256, a learning rate of 0.1, and a weight decay of 1e-4. We also include the result of attacks on Purchase-100 dataset (a tabular dataset of shopping records) (Shokri et al., 2017), where models are 4-layer MLP with layer units=[512, 256, 128, 64], trained on 25k samples for 50 epochs. Table 5 displays the accuracy of models trained on various datasets.

Dataset	Train Accuracy	Test Accuracy
CIFAR-10	99.9%	92.4%
CIFAR-100	99.9%	67.5%
CINIC-10	99.5%	77.2%
ImageNet	90.2%	58.6%
Purchase-100	100%	83.4%

Table 5: Average accuracy of models trained on different datasets.

Evaluation Metrics. We measure the performance of each attack using two underlying metrics: its true positive rate (TPR), and its false positive rate (FPR), over all member and non-member records of random target models. Then, we use the ROC curve to reflect the trade-off between the TPR and FPR of an attack, as we sweep over all possible values of threshold β to build different FPR tolerance. The AUC (area under the ROC curve) score gives us the average success across all target samples and measures the overall strength of an attack. Inspired from previous discussions in (Carlini et al., 2022), we also consider TPR at very low FPRs. More precisely, we focus on TPR at 0% FPR, a metric that has seen limited usage in the literature. When attacking a target model, all samples in the population data are used as input queries. Hence, for each target model, half of queries are members and the other half are non-members.

B. Details of RMIA and its Evaluation

B.1. Pseudo-code of RMIA

Membership inference attacks require training reference (or shadow) models in order to distinguish members from non-members of a given target model. We train k reference models on a set of samples randomly drawn from the population data. Concerning the training of models, we have two versions of RMIA. The RMIA (Online) (similar to Carlini et al. (2022)) trains reference models separately for each target data (MIA test query). Specifically, upon receiving a MIA test query x, we train k IN models that include x in their training set. However, such a training is costly and impractical in many real-world scenarios due to the significant resource and time requirements. On the other hand, the offline version uses only k pre-trained reference models on randomly sampled datasets, avoiding any training on test queries (as it exclusively uses OUT models). The two versions differ in the way they compute the normalizing term $\Pr(x)$ in equation 3. The online algorithm computes the terms by averaging prediction probabilities over all IN and OUT models in an unbiased manner (equation 9), while the offline algorithm computes the terms solely with OUT models based on equation 10. In the offline mode, we approximate what $\Pr(x)$ would have been if we had access to IN models (see Appendix B.2.2 for more explanations).

[12] For CINIC-10, we randomly choose 50k samples (out of 270k samples) for training models. In Appendix C.4, we show the result of attacks when training models with larger subsets of CINIC-10.

Algorithm 1 MIA Score Computation with RMIA. The input to this algorithm is k reference models Θ, the target model θ, target (test) sample x, parameter γ, and a scaling factor a as described in Appendix B.2.2. We assume the reference models Θ are pre-trained on random samples from a population dataset available to adversary; each sample from the population dataset is included in training of half of reference models. The algorithm also takes an *online* flag which indicates whether we intend to run MIA in the online mode.

```
1: Randomly choose a subset $Z$ from the population dataset
2: $C \leftarrow 0$
3: if online then
4:     $\Theta_{in} \leftarrow \emptyset$
5:     for $k$ times do
6:         $D_i \leftarrow$ randomly sample a dataset from population data $\pi$
7:         $\theta_x \leftarrow \mathcal{T}(D_i \cup x)$
8:         $\Theta_{in} \leftarrow \Theta_{in} \cup \{\theta_x\}$
9:     end for
10:    $\Pr(x) \leftarrow \frac{1}{2k}\big(\sum_{\theta' \in \Theta} \Pr(x|\theta') + \sum_{\theta' \in \Theta_{in}} \Pr(x|\theta')\big)$    (See equation 9)
11: else
12:    $\Pr(x)_{OUT} \leftarrow \frac{1}{k} \sum_{\theta' \in \Theta} \Pr(x|\theta')$
13:    $\Pr(x) \leftarrow \frac{1}{2}\big((1+a).\Pr(x)_{OUT} + (1-a)\big)$    (See Appendix B.2.2)
14: end if
15: $Ratio_x \leftarrow \frac{\Pr(x|\theta)}{\Pr(x)}$
16: for each sample $z$ in $Z$ do
17:    $\Pr(z) \leftarrow \frac{1}{k} \sum_{\theta' \in \Theta} \Pr(z|\theta')$
18:    $Ratio_z \leftarrow \frac{\Pr(z|\theta)}{\Pr(z)}$
19:    if $(Ratio_x / Ratio_z) > \gamma$ then
20:        $C \leftarrow C + 1$
21:    end if
22: end for
23: $\text{Score}_{\text{MIA}}(x;\theta) \leftarrow C/|Z|$    (See equation 5)
```

Algorithm 1 outlines the pseudo-code for RMIA. The input to this algorithm includes the target model θ, the target sample x, the threshold γ, and the set of reference models denoted by Θ, which are trained prior to computing the MIA score of x. Additionally, it takes an *online* flag indicating if we run the algorithm in online mode. If it we are in offline mode, a scaling factor a is also obtained as described in Appendix B.2.2. The output of the algorithm is $\text{Score}_{\text{MIA}}(x;\theta)$, which is then used according to equation 1 to infer the membership of x. As described in Algorithm 1, we first select our reference population samples (i.e., the set Z) from the population dataset. If we are operating in the online mode, we train k IN models using target data x and a randomly selected dataset form the population data (line 5-9). Depending on the attack mode, we query the reference models to obtain the average prediction probabilities over reference models as $\Pr(x)$ (line 10 and line 12). In the case of offline RMIA, line 13 approximates $\Pr(x)$ using equation 10. Then, we calculate the ratio of prediction probabilities between the target model and the reference models for sample x as $Ratio_x$ (line 15). The same routine is applied to obtain the ratio for each reference sample $z \in Z$ (line 16-18).

According to equation 3, the division of the computed ratio for sample x by the obtained ratio for sample z determines $\text{LR}_\theta(x, z)$ to assess if z is γ-dominated by x (line 19). The fraction of γ-dominated reference samples establishes the MIA score of the target sample x in RMIA (line 23).

B.2. Likelihood Ratio Computation in equation 3

B.2.1. Computing $\Pr(x|\theta)$

Let θ be a classification (neural network) model that maps each input data x to a probability distribution across d classes. Assume data point x is in class y. Let $c(x) = \langle c_1, \cdots, c_d \rangle$ be the output vector (logits) of the neural network for input x, before applying the final normalization. We denote the normalized prediction probability of class y for the input x as

$f_\theta(x)_y$. For the Softmax function, the probability is given by

$$f_\theta(x)_y = \frac{e^{\frac{c_y}{T}}}{\sum_{i=1}^{d} e^{\frac{c_i}{T}}},$$

where T is a temperature constant. We can use this to estimate $\Pr(x|\theta)$. However, there are many alternatives to Softmax probability proposed in the literature to improve the accuracy of estimating $\Pr(x|\theta)$, using the Taylor expansion of the exponential function, and using heuristics to refine the relation between the probabilities across different classes (De Brebisson & Vincent, 2016; Liu et al., 2016; Liang et al., 2017; Banerjee et al., 2021). A more recent method, proposed by Banerjee et al. (2021), combines the Taylor expansion with the soft-margin technique to compute the confidence as:

$$f_\theta(x)_y = \frac{apx(c_y - m)}{apx(c_y - m) + \sum_{i \neq y} apx(c_i)}, \tag{7}$$

where $apx(a) = \sum_{i=0}^{n} \frac{a^i}{i!}$ is the nth order Taylor approximation of e^a, and m is a hyper-parameter that controls the separation between probability of different classes.

In Table 6, we compare the result of RMIA obtained with four different confidence functions, i.e. Softmax, Taylor-Softmax (De Brebisson & Vincent, 2016), Soft-Margin Softmax (SM-Softmax) (Liang et al., 2017), and the combination of last two functions (SM-Taylor-Softmax) (Banerjee et al., 2021), formulated in equation 7. Based on our empirical results, we set the soft-margin m and the order n in Taylor-based functions to 0.6 and 4, respectively. The temperature (T) is set to 2 for CIFAR-10 models. While the performance of all functions is closely comparable, SM-Taylor-Softmax stands out by achieving a slightly higher AUC and a rather better TPR at low FPRs. Therefore, we use this function for CIFAR and CINIC-10 datasets. In Figure 5, we present the performance sensitivity of our attack in terms of AUC concerning three hyper-parameters in this function: order n, soft-margin m, and temperature T. The AUC of RMIA appears to be robust to variations in n, m and T. For $n \geq 3$, the results are consistent, but employing lower orders leads to a poor Taylor-based approximation for Softmax, as reported by Banerjee et al. (2021). We use Softmax (with temperature of 1) for our ImageNet and Purchase-100 datasets. We do not apply confidence functions to other attacks, as they do not use confidence as their signal. For instance, LiRA (Carlini et al., 2022) operates with rescaled logit, while Attack-P and Attack-R (Ye et al., 2022) use the loss signal. In fact, we follow the original algorithms and codes to reproduce their results.

Confidence Function	AUC	TPR@FPR	
		0.01%	0.0%
Softmax	68.96 ± 0.31	2.25 ± 0.44	1.69 ± 0.37
SM-Softmax	69.02 ± 0.34	$\mathbf{2.27 \pm 0.43}$	1.67 ± 0.44
Taylor-Softmax	68.57 ± 0.29	2.06 ± 0.49	1.43 ± 0.49
SM-Taylor-Softmax	$\mathbf{69.15 \pm 0.35}$	2.26 ± 0.43	$\mathbf{1.80 \pm 0.55}$

Table 6: Performance of RMIA obtained with various confidence functions. The number of reference models, trained with CIFAR-10, is 254. The soft-margin m and the order n in Taylor-based functions are 0.6 and 4, respectively.

B.2.2. COMPUTING $\Pr(x)$

Recall that $\Pr(x)$ is the normalizing constant for the Bayes rule in computing our LR. See equation 4. We compute it as the empirical average of $\Pr(x|\theta')$ on reference models θ' trained on random datasets D drawn from the population distribution π. This is then used in pairwise likelihood ratio equation 3.

In order to compute $\Pr(x)$, we need to train reference models θ'. Note that the reference models must be sampled in an unbiased way with respect to whether x is part of their training data. This is because the summation in equation 4 is over all θ', which can be partitioned to the set of models trained on x (IN models), and the set of models that are not trained on x (OUT models). Let $\bar{x}$ denote that a training set does not include x. Let θ'_x denote an IN model trained on dataset D_x

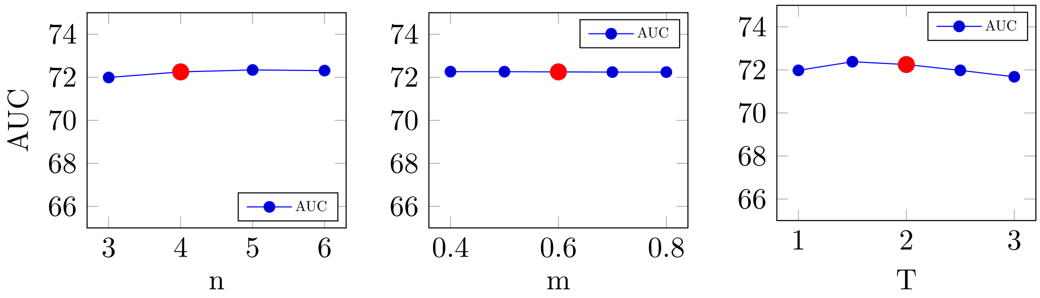


Figure 5: AUC of our attack (RMIA) obtained by using different values of n (order in Taylor function), m (soft-margin) and T (temperature) in SM-Taylor-Softmax function. When modifying one parameter, we hold the values of the other two parameters constant at their optimal values. Here, we use 254 reference models trained on CIFAR-10. Results are averaged over 10 target models. The red points indicate the default values used in our experiments.

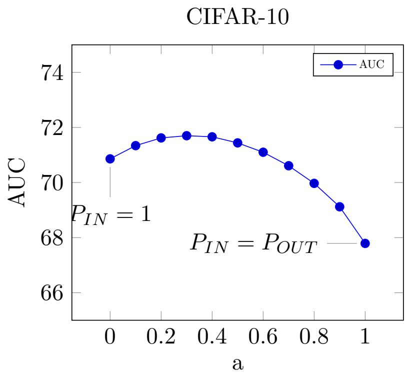


Figure 6: AUC of offline RMIA obtained by using different values of a in the linear approximation function. Here, we use 127 reference models (OUT) trained on CIFAR-10.

($x \in D_x$) and $\theta'_{\bar{x}}$ be an OUT model trained on dataset $D_{\bar{x}}$ ($x \notin D_{\bar{x}}$). Then, from equation 4, we have:

$$\begin{aligned}\Pr(x) &= \sum_{\theta',D} \Pr(x|\theta')\Pr(\theta'|D)\Pr(D)\\ &= \sum_{\theta'_x,D_x} \Pr(x|\theta'_x)\Pr(\theta'_x|D_x)\Pr(D_x) + \sum_{\theta'_{\bar{x}},D_{\bar{x}}} \Pr(x|\theta'_{\bar{x}})\Pr(\theta'|D_{\bar{x}})\Pr(D_{\bar{x}})\end{aligned} \tag{8}$$

The two sums on the right-hand side of the above equation can be computed empirically using sampling methods. Instead of integrating over all possible datasets and models, we sample datasets D_x and $D_{\bar{x}}$ and models θ'_x and $\theta'_{\bar{x}}$, and compute the empirical average of $\Pr(x|\theta'_x)$ and $\Pr(x|\theta'_{\bar{x}})$ given the sampled models. We sample $D_{\bar{x}}$ from the probability distribution $\Pr(D_{\bar{x}})$, which is the underlying data distribution π. For sampling D_x, we sample a dataset from π and add x to the dataset. We sample θ'_x and $\theta'_{\bar{x}}$ by training models on D_x and $D_{\bar{x}}$, respectively.

In the online setting for MIA, we can empirically estimate $\Pr(x)$ by computing the average $\Pr(x|\theta')$ over 50% IN models and 50% OUT models (using $2k$ models), i.e.:

$$\Pr(x) \approx \frac{1}{2}\Big(\underbrace{\frac{1}{k}\sum_{\theta'_x}\Pr(x|\theta'_x)}_{\Pr(x)_{IN}} + \underbrace{\frac{1}{k}\sum_{\theta'_{\bar{x}}}\Pr(x|\theta'_{\bar{x}})}_{\Pr(x)_{OUT}}\Big) \tag{9}$$

However, in the offline setting, we do not have access to IN models. Thus, we need to exclusively use $\Pr(x|\theta'_{\bar{x}})$. We now introduce a simple heuristic to obtain a less biased estimate of $\Pr(x)$ without having access to IN models, through an offline pre-computation to approximate the shift of probability between IN and OUT models. Essentially, we approximate the sensitivity of models (the gap between probability of member and non-member points in reference models). See Figure 1 in (Zhang et al., 2021) for such computation.

We use our existing reference models to compute the rate at which $\Pr(x)$ for any sample x changes between reference models that include x versus the others. We approximate the gap with a linear function, so we obtain $\Pr(x)_{IN} = a. \Pr(x)_{OUT} + b$, and finally can obtain $\Pr(x) = (\Pr(x)_{IN} + \Pr(x)_{OUT})/2$. Given that both $\Pr(x)_{IN}$ and $\Pr(x)_{OUT}$ fall within the range of 0 to 1, it follows that $a + b = 1$. Consequently, we can simplify the linear function as $\Pr(x)_{IN} = a.(\Pr(x)_{OUT} - 1) + 1$ which results in:

$$\Pr(x) \approx \frac{1}{2}\big((1 + a) \Pr(x)_{OUT} + (1 - a)\big) \tag{10}$$

In Figure 6, we present the AUC obtained by our offline RMIA on CIFAR-10 models, varying the value of a from 0 to 1. As a approaches 1, there is a degradation in AUC (by up to 5.5%), because we heavily rely on $\Pr(x)_{OUT}$ to approximate $\Pr(x)_{IN}$. Lower values of a result in an enhancement of $\Pr(x)_{OUT}$ to approximate $\Pr(x)_{IN}$, leading to improved performance. The AUC appears to be more robust against lower values of a, particularly those below 0.5. Notably, even with $a = 0$, where $\Pr(x) = (\Pr(x)_{OUT} + 1)/2$, a considerable improvement in results is observed. In this case, we are alleviating the influence of very low $\Pr(x)_{OUT}$ for atypical/hard samples.

To determine the best value of a for models trained with each dataset, we use the following procedure. It is executed only once, independent of test queries, without the need to train any new models. We choose two existing models and then, select one as the temporary target model and subject it to attacks from the other model using varying values of a. Finally, we select the one that yields the highest AUC as the optimal a. In the case of having only one reference model, we simulate an attack against the reference model and use the original target model as the reference model for the simulated attack to obtain the best a. Based on the result of our experiments, this optimal a remains roughly consistent across random selections of reference models. In our experiments, we empirically derived the following values for our models: $a = 0.3$ for CIFAR-10 and CINIC-10, $a = 0.6$ for CIFAR-100, $a = 1$ for ImageNet, and $a = 0.2$ for Purchase-100 models.

B.3. Direct Computation of Likelihood Ratio in equation 2

In Section 3, we introduced two separate approaches for computing the fundamental likelihood ratio in equation 2: I) a Bayesian method, as shown by equation 3, and II) a direct method, expressed in equation 6. While our empirical results are primarily derived using the Bayesian method, we anticipate that the outcomes of these two approaches will converge closely when a sufficient number of reference models is employed, although their performance may vary with a limited number of models. In this section, we attempt to assess these two methods and possibly invalidate our initial anticipation.

The direct approach involves employing Gaussian modeling over logits. Our attack is then simplified to the estimation of the mean and variance for the logits of samples x and z in two distinct distributions of reference models; One distribution comprises models trained with x in their training set but not z (denoted by $\theta'_{x,\bar{z}}$), while the other comprises models trained with z in their training set but not x (denoted by $\theta'_{\bar{x},z}$). Let $\mu_{x,\bar{z}}(x)$ and $\sigma_{x,\bar{z}}(x)$ be the mean and variance of $f_{\theta'}(x)$ in the distribution of $\theta'_{x,\bar{z}}$ models, respectively (a similar notation can be defined for $\theta'_{\bar{x},z}$ models). We can approximate the likelihood ratio in equation 6 as follows:

$$\mathrm{LR}_\theta(x, z) \approx \frac{\Pr(f_\theta(x), f_\theta(z)|x)}{\Pr(f_\theta(x), f_\theta(z)|z)} \approx \frac{\Pr(f_\theta(x)|\mathcal{N}(\mu_{x,\bar{z}}(x),\, \sigma^2_{x,\bar{z}}(x)))}{\Pr(f_\theta(x)|\mathcal{N}(\mu_{\bar{x},z}(x),\, \sigma^2_{\bar{x},z}(x)))} \times \frac{\Pr(f_\theta(z)|\mathcal{N}(\mu_{x,\bar{z}}(z),\, \sigma^2_{x,\bar{z}}(z)))}{\Pr(f_\theta(z)|\mathcal{N}(\mu_{\bar{x},z}(z),\, \sigma^2_{\bar{x},z}(z)))} \tag{11}$$

where $f_\theta(x)$ represents the output (logits) of the target model θ on sample x. Although the direct method may appear to be more straightforward and accurate, it comes with a significantly higher computational cost, since we must train online reference models, i.e. $\theta'_{x,\bar{z}}$, to compute the probabilities in the above relation.

We now demonstrate the performance of our RMIA when the likelihood ratio is computed using equation 3 (hereafter, called RMIA-Bayes), in comparison to using the aforementioned likelihood ratio in equation 11 (referred to as RMIA-direct). Figure 19 compares the ROCs achieved by the two methods when 64 reference models are employed to estimate the probabilities. We show the results obtained from models trained with different datasets (we use no augmented queries). In this case, RMIA-direct slightly outperforms in terms of both AUC and TPR at low FPRs, yet RMIA-Bayes closely matches its pace, even at low FPR values. On the other hand, Figure 20 presents the scenario where 4 reference models are used. When utilizing fewer models, RMIA-Bayes exhibits a better performance across all datasets (e.g., 6.6% higher AUC in CIFAR-10). It appears that RMIA-direct struggles to accurately estimate the parameters of Gaussian models with only four models available. Given the substantial processing cost of RMIA-direct, associated with training online models relative to sample pairs, RMIA-Bayes emerges as a more reasonable choice due to its ability to operate with a reduced number of models trained in an offline context.

B.4. Analyzing γ and β Parameters and their Relation

We here investigate the impact of selecting γ on the efficacy of our attack. In Figure 7, we illustrate the sensitivity of our attack, measured in terms of AUC and FPR@TPR, to changes in the value of γ. We do not show the result for $\gamma < 1$, because it implies that a target sample x is allowed to have a lower chance of being member than reference samples to pass our pairwise likelihood ratio test, causing lots of non-members to be wrongly inferred as member. As it can be seen from the figure, our attack's performance is consistent against changes in the value of γ; both AUC and FPR@TPR remain relatively stable with increasing γ (except for a considerably high value of γ). As γ increases, the need arises to decrease the value of β in the hypothesis test equation (equation 1) to strike a balance between the power and error of the attack. By appropriately adjusting the value of β, we achieve a roughly same result across different γ values. In extreme cases where a very high γ is employed, the detection of γ-dominated reference samples between a limited set of z records becomes exceedingly challenging. We consistently note the same trend in results across all our datasets and with varying numbers of reference models. Throughout our experiments, we set γ to 2, as it yields a slightly higher TPR@FPR.

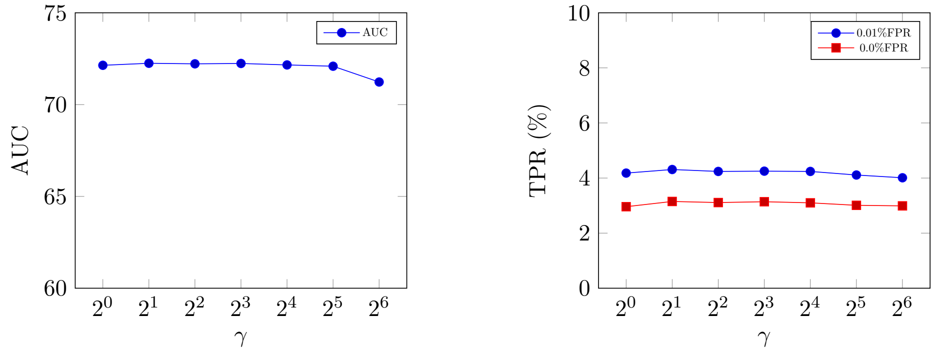


Figure 7: The performance sensitivity of our attack (RMIA) with respect to γ parameter. Here, we use 254 reference models trained on CIFAR-10. The left plot shows the AUC obtained when using different γ values, while the right plot demonstrates the TPR at 0.01% and 0% FPR values versus γ (corresponding to blue and red lines, respectively).

As we discussed in Section 3, β is a common threshold among all attacks, and will be used to generate the ROC curve. In our method (as opposed to e.g., Carlini et al. (2022)), the exact value of β is interpretable. Specifically, when γ equals 1, the value of β shows a strong correlation with the value of 1 - FPR. To show this, Figure 8 illustrates the impact of selecting β from the range [0, 1] on the TPR and FPR of RMIA. We use two distinct γ values: $\gamma = 1$ (shown in the left plot) and $\gamma = 2$ (depicted in the right plot). Both TPR and FPR approach zero, as we increase β to 1 (under both γ values) and the reason is that it is unlikely that a target sample can dominate most of reference records. However, FPR decreases more rapidly than TPR in two plots, especially with $\gamma = 2$, which allows us to always have a higher power gain than error. When $\gamma = 1$, FPR decreases in a calibrated manner, proportional to $1 - \beta$.

B.5. Boosting RMIA with Augmented Queries

We can further enhance the effectiveness of attacks by augmenting the input query x with multiple data samples that are similar to x (Carlini et al., 2022; Choquette-Choo et al., 2021). These data samples can be simple transformations of x (for example, using shift or rotation in case of image data). To consolidate the results in our multi-query setting, we use majority voting on our hypothesis test in equation 2: x is considered to dominate z if more than half of all generated transformations of x dominate z.

In (Carlini et al., 2022), the authors discussed how to extend LiRA to support multiple queries. We here compare the performance of online and offline attacks when we increase the number of augmented queries from 1 to 50. Note that

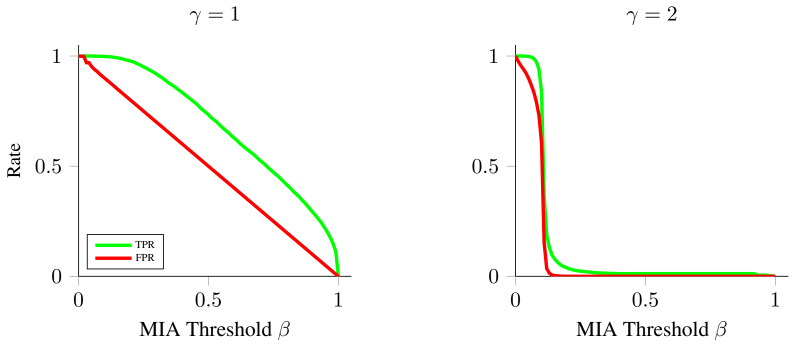


Figure 8: TPR and FPR achieved by RMIA for different values of β. The left and right plots correspond to $\gamma = 1$ and $\gamma = 2$, respectively. The number of reference models, trained with CIFAR-10, is 254.

Attack-P and Attack-R have no result for multiple queries, as they do not originally support query augmentations. In this experiment, we use 254 reference models (for offline attacks, we only use 127 OUT models).

We first compare the offline attacks, shown in the Offline column of Table 7. With no augmented queries, RMIA presents a clear advantage over Attack-R (with 6.7% higher AUC and 116% more TPR at zero FPR) and the performance gap between two attacks widens with using more queries. As we increase queries, RMIA gets a better result (for example, a 4x improvement in TPR at zero FPR and also about 4.6% higher AUC as queries go from 1 to 50), while LiRA cannot benefit from the advantage of more queries to improve its AUC. Note that we use the same technique to generate augmentations, as proposed in (Carlini et al., 2022).

In the Online column of Table 7, we show the result of LiRA and RMIA when all 254 IN and OUT reference models are available to the adversary. Compared to LiRA, RMIA always has a slightly higher AUC and at least 48% better TPR at zero FPR. Note that in this case, even minor AUC improvements are particularly significant when closely approaching the true leakage of the training algorithm through hundreds of models. Both LiRA and RMIA work better with increasing augmented queries, e.g. around 2× improvement in TPR@FPR when going from 1 query to 50 queries. Unless explicitly stated otherwise, the experimental results in this paper are obtained using 18 augmented queries for both LiRA and RMIA.

# Queries	Attack	Online			Offline		
		AUC	TPR@FPR		AUC	TPR@FPR	
			0.01%	0.0%		0.01%	0.0%
1	Attack-R	-	-	-	64.41 ± 0.41	1.52 ± 0.33	0.80 ± 0.43
	Attack-P	-	-	-	58.19 ± 0.33	0.01 ± 0.01	0.00 ± 0.01
	LiRA	68.92 ± 0.42	1.78 ± 0.70	0.92 ± 0.48	56.12 ± 0.41	0.46 ± 0.18	0.28 ± 0.17
	RMIA	**69.15 ± 0.35**	**2.26 ± 0.43**	**1.80 ± 0.55**	**68.74 ± 0.34**	**2.42 ± 0.57**	**1.73 ± 0.61**
2	LiRA	71.28 ± 0.46	2.83 ± 0.49	1.73 ± 0.70	55.77 ± 0.46	1.16 ± 0.41	0.59 ± 0.29
	RMIA	**71.46 ± 0.43**	**3.69 ± 0.36**	**2.55 ± 0.71**	**71.06 ± 0.39**	**3.64 ± 0.61**	**2.46 ± 0.98**
18	LiRA	72.04 ± 0.47	3.39 ± 0.86	2.01 ± 0.78	55.18 ± 0.37	1.37 ± 0.32	0.72 ± 0.31
	RMIA	**72.25 ± 0.46**	**4.31 ± 0.47**	**3.15 ± 0.61**	**71.71 ± 0.43**	**4.18 ± 0.61**	**3.14 ± 0.87**
50	LiRA	72.26 ± 0.47	3.54 ± 0.50	2.19 ± 0.83	55.00 ± 0.36	1.52 ± 0.34	0.75 ± 0.36
	RMIA	**72.51 ± 0.46**	**4.47 ± 0.44**	**3.25 ± 0.36**	**71.95 ± 0.44**	**4.39 ± 0.54**	**3.22 ± 0.81**

Table 7: Performance of attacks when we use different number of augmented queries for LiRA (Carlini et al., 2022) and RMIA. We evaluate attacks in two different settings, shown in Online and Offline columns. In the online setting, we use 254 models where half of them are IN models and half are OUT (for each sample). The offline setting uses only 127 OUT models. Both Attack-P and Attack-R (Ye et al., 2022) do not originally support multiple augmented queries. The Attack-P does not work with reference models, thus we consider it offline. Models are trained with CIFAR-10. Results are averaged over 10 random target models.

C. Supplementary Empirical Results

C.1. Robustness of Attacks against Noise and OOD Non-member Queries

To evaluate the performance of attacks against OOD non-member samples, we consider two strategies for generating test queries: incorporating samples from a dataset different from the training dataset and using pure noise. Figure 9 illustrates the ROC curves of three offline attacks using models trained on CIFAR-10 (all member queries are from in-distribution). We use 127 (OUT) reference models. The ROCs are presented in normal scale to highlight the gap between attacks. To generate OOD samples, we employ samples from CINIC-10 with the same label as CIFAR-10. When using OOD samples as test queries (depicted in the left plot), we observe a substantial performance gap (at least 21% higher AUC) between RMIA and other attacks, with LiRA not performing much better than random guessing. In the case of using noise as non-member test queries (depicted in the right plot), RMIA once again outperforms the other two two attacks by at least 61% in terms of AUC, while LiRA falls significantly below the random guess (i.e., it has a very large false positive rate).

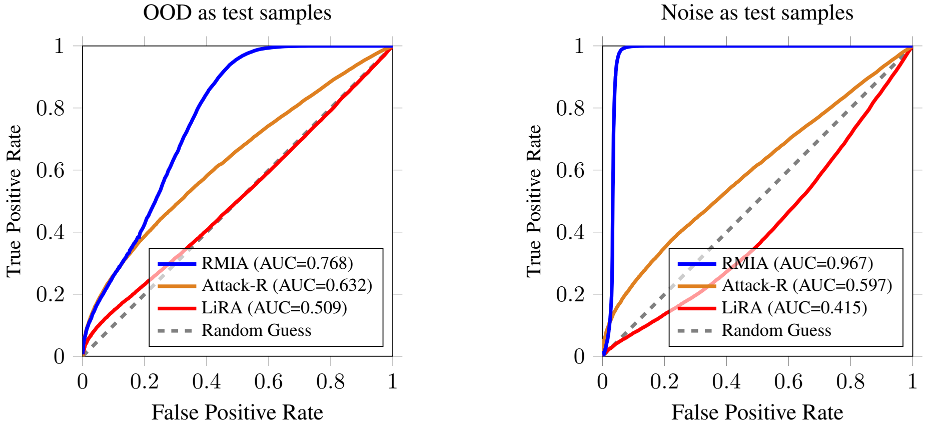


Figure 9: ROC of three offline attacks using models trained on CIFAR-10 when non-member test queries come from a distribution different from the population data π. The result is obtained on one random target model. The left plot uses out-of-distribution (OOD) samples from CINIC-10 (with the same label as CIFAR-10) as non-member test queries, while the right plot uses random noise. We here use 127 reference models (OUT). For RMIA, we use $a = 0$ and $\gamma = 2$.

Figure 10 illustrates the performance of various attacks when using different ratios of OOD non-member samples as test queries (clearly, all member queries are in-distribution). When the ratio is zero, only in-distribution non-member queries are used, and when it is one, all non-member test queries are OOD. As depicted in the left plot, increasing the ratio of OOD samples results in an increase in the AUC of RMIA and Attack-P, whereas the performance of the other two attacks decreases. This is primarily because the former two attacks are capable of filtering OOD samples by comparing the MIA score of the test query with in-distribution population data. When comparing attacks based on their TPR at zero FPR (the left plot), the population attack yields nearly zero TPR, while RMIA outperforms all other attacks in this regard. We observe a decreasing trend for TPR at zero FPR in all attacks, indicating the presence of OOD cases that are difficult to detect via comparison of their MIA score with population data.

To gain a deeper insight into RMIA's response to OOD MIA queries, Figure 11 presents a comparison of the FPRs when using two distinct sets of non-members: out-of-distribution (OOD) versus in-distribution (ID) samples. The left plot illustrates the TPR and FPR achieved for various MIA thresholds β, separately for ID and OOD samples. Notably, as β increases from zero, the FPR for OOD samples diminishes significantly faster in contrast to the FPR and TPR for ID samples. This observation underscores RMIA's efficacy in discerning a majority of OOD queries as non-members. Furthermore, as illustrated in the right plot, when comparing the FPRs obtained from OOD and ID non-member queries, a striking discrepancy emerges. While the FPR for ID samples can exceed 0.8, the FPR for OOD samples (at the same β value) is almost zero. This discrepancy implies that RMIA finds it easier to detect OOD non-members compared to ID non-members.

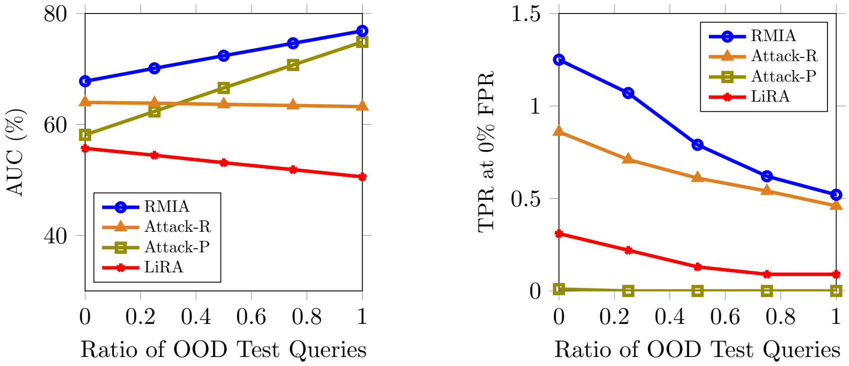


Figure 10: Performance of attacks, when testing against different ratios of OOD non-member queries (if ratio=0, we use no OOD sample as test query, i.e. all non-members are in-distribution, while for ratio=1, all non-member queries are OOD). The left plot shows the AUC obtained by different attacks, and the right plot depicts the TPR at zero FPR. All attacks are offline and the number of reference models is 127. All models are trained with CIFAR-10 and OOD queries are from CINIC-10.

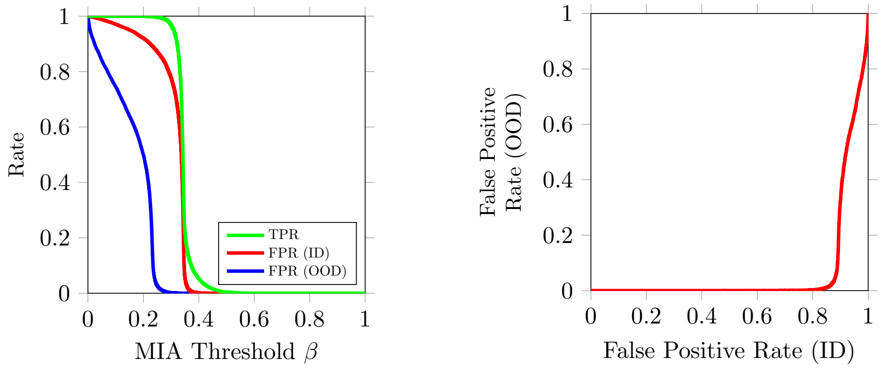


Figure 11: Comparing FPR of RMIA (Offline) when two separate sets of non-member test queries, i.e. in-distribution (ID) and out-of-distribution (OOD) samples, are used. The number of reference models is 127 and $\gamma = 2$. All models are trained with CIFAR-10 and OOD queries are from CINIC-10. The left plot shows TPR and FPR obtained for each MIA threshold β, while the right plot compares obtained FPRs when OOD vs ID samples are used as non-member test queries.

C.2. Data Distribution Shift

Table 8 compares the result of attacks when the target models are trained on different datasets than the reference models. More specifically, the target models are trained on CIFAR-10, while the reference models are trained on CINIC-10 (all test queries are from CIFAR-10). We use images with common class labels between two datasets. We here concentrate on offline attacks, as the reference models are trained on a completely different dataset than the target model. We report the results obtained with different number of reference models (1, 2 and 4). The shift in distribution of training data between the target model and the reference models affects the performance of all attacks. Compared with other two attacks, RMIA always obtains a higher AUC (e.g. by up to 25% in comparison with LiRA) and a better TPR at low FPRs.

C.3. Variations in Neural Network Architectures

Figure 12 illustrates the performance of attacks when models are trained with different architectures, including CNN and Wide ResNet (WRN) of various sizes, on CIFAR-10. In this scenario, both target and reference models share the same architecture. RMIA consistently outperforms other attacks across all architectures (e.g. 7.5%-16.8% higher AUC compared with LiRA). Using network architectures with more parameters can lead to increased leakage, as shown by Carlini et al. (2019).

# Ref Models	Attack	AUC	TPR@FPR	
			0.01%	0.0%
1	Attack-R, (Ye et al., 2022)	61.41 ± 0.23	0.02 ± 0.01	0.01 ± 0.01
	LiRA (Offline), (Carlini et al., 2022)	54.48 ± 0.21	0.06 ± 0.03	0.01 ± 0.01
	RMIA (Offline)	$\mathbf{64.75 \pm 0.27}$	$\mathbf{0.06 \pm 0.02}$	$\mathbf{0.02 \pm 0.01}$
2	Attack-R, (Ye et al., 2022)	61.45 ± 0.3	0.03 ± 0.01	0.01 ± 0.01
	LiRA (Offline), (Carlini et al., 2022)	52.58 ± 0.18	0.01 ± 0.01	0 ± 0
	RMIA (Offline)	$\mathbf{66.05 \pm 0.29}$	$\mathbf{0.13 \pm 0.07}$	$\mathbf{0.04 \pm 0.04}$
4	Attack-R, (Ye et al., 2022)	61.54 ± 0.3	0.04 ± 0.02	0.02 ± 0.01
	LiRA (Offline), (Carlini et al., 2022)	55.15 ± 0.25	0.03 ± 0.02	0.01 ± 0.01
	RMIA (Offline)	$\mathbf{66.78 \pm 0.32}$	$\mathbf{0.22 \pm 0.06}$	$\mathbf{0.09 \pm 0.09}$

Table 8: Performance of offline attacks when different datasets are used for training target and reference models. Specifically, the target models are trained on **CIFAR-10**, while the reference models are trained on **CINIC-10**. We use different numbers of reference models (1, 2 and 4 OUT models). Results are averaged over 10 random target models.

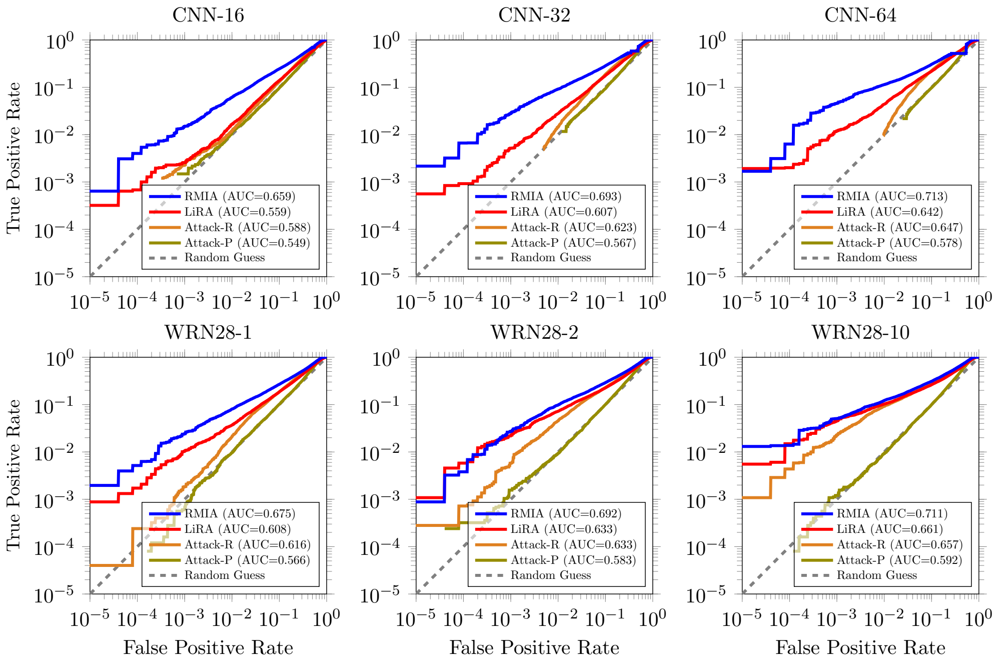


Figure 12: ROC of attacks using different neural network architectures for training models on CIFAR-10. Here, both target and reference models share the same architecture. We use 2 reference models (1 IN, 1 OUT).

Figure 13 presents the performance of attacks when different architectures are used to train reference models, while keeping the structure of the target model fixed as WRN28-2. So, the target and reference models may have different architectures. The optimal performance for attacks is observed when both target and reference models share similar architectures. However, RMIA outperforms other attacks again, and notably, the performance gap widens under architecture shifts.

C.4. Performance of Attacks on Models Trained with a Larger Subset of CINIC-10

Analyzing the effectiveness of attacks against models with larger training set is crucial, as they might experience performance deterioration, as highlighted by Bertran et al. (2023). We here evaluate attacks when tested against models trained with an expanded training set from CINIC-10. Specifically, we use a random subset consisting of 90k samples which is larger than the size of training sets in our previous experiments (i.e. 25k). We train a Wide ResNet network for 200 epochs on half of the randomly chosen dataset (with batch size, learning rate, and weight decay set to 64, 0.1, and 0.0005, respectively). The test accuracy of CINIC-10 models is 79%. For our offline attack, we use the original Softmax with temperature of 1 as the confidence function. Moreover,the best value of a in equation 10 is obtained 1. Figure 14 illustrates the ROC curves of various attacks conducted on CINIC-10 when using only 1 (OUT) reference model. Additionally, it presents the

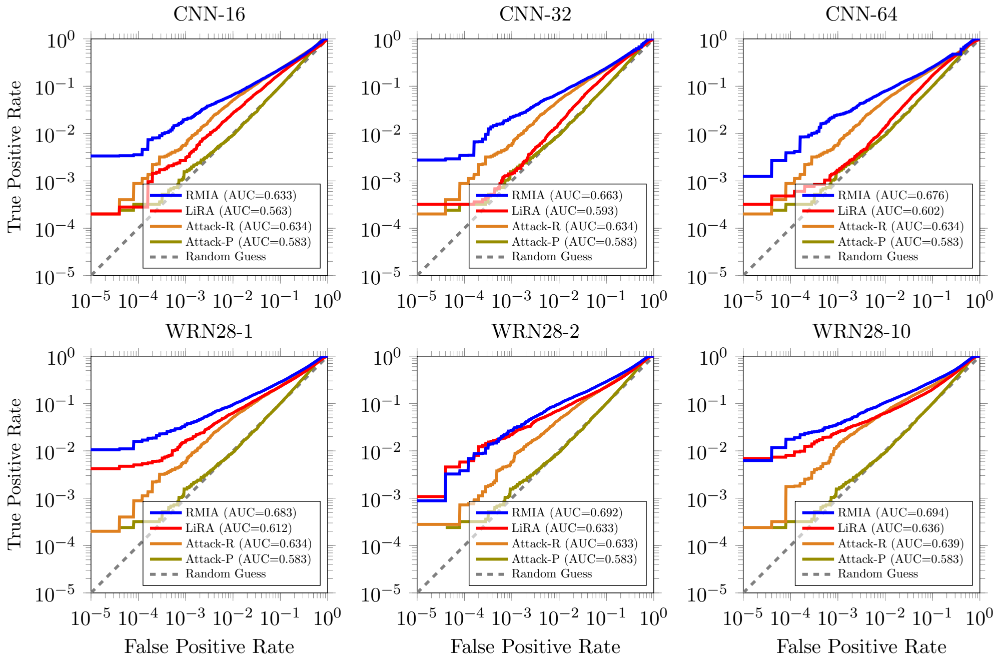


Figure 13: ROC of attacks using different neural network architectures for training reference models on CIFAR-10. The target model is always trained using WRN28-2. We here use 2 reference models (1 IN, 1 OUT).

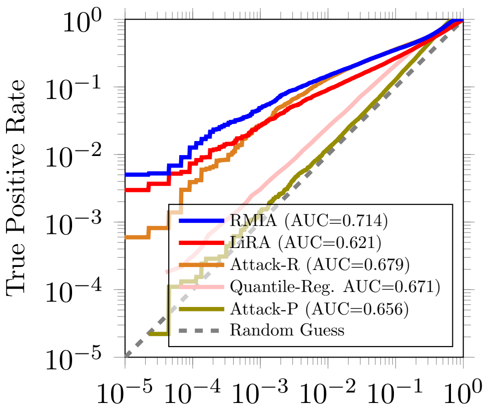


Figure 14: ROC of attacks using models trained with a larger subset of CINIC-10 (i.e. 90k samples). The result is obtained on one random target model. We here use **1 reference model** (OUT). We do not use augmented queries for any attacks.

outcome of the Quantile Regression attack by Bertran et al. (2023), which trains regression-based attack models instead of reference models. Our attack demonstrates superior performance compared to other attacks in terms of both AUC and TPR at low FPRs. For instance, it achieves approximately 15% higher AUC than LiRA and an order of magnitude higher TPR at 0.01% FPR than the Quantile Regression attack. Although the AUC obtained by the Quantile Regression attack is notably higher than that of LiRA (with only 1 reference model), it was unable to achieve any true positives at zero FPR.

C.5. Performance of Attacks using Fewer Reference Models

In Table 9, we show the average result of attacks with their standard deviations obtained on three different datasets, i.e. CIFAR-10, CIFAR-100 and CINIC-10, under low computation budget where we use a few reference models.

# Ref	Attack	CIFAR-10			CIFAR-100			CINIC-10		
		AUC	TPR@FPR		AUC	TPR@FPR		AUC	TPR@FPR	
			0.01%	0.0%		0.01%	0.0%		0.01%	0.0%
0	Attack-P	58.19 ± 0.33	0.01 ± 0.01	0.0 ± 0.0	75.91 ± 0.36	0.01 ± 0.01	0.0 ± 0.0	66.91 ± 0.3	0.01 ± 0.01	0.0 ± 0.0
*	Quantile-Reg.	61.45 ± 0.29	0.08 ± 0.05	0.03 ± 0.03	83.32 ± 0.34	0.26 ± 0.15	0.04 ± 0.04	73.48 ± 0.5	0.35 ± 0.09	0.12 ± 0.08
1	Attack-R	63.65 ± 0.27	0.07 ± 0.04	0.02 ± 0.02	81.61 ± 0.17	0.06 ± 0.04	0.02 ± 0.02	72.04 ± 0.35	0.07 ± 0.05	0.02 ± 0.01
	LiRA	53.2 ± 0.23	0.48 ± 0.1	0.25 ± 0.11	68.95 ± 0.28	0.54 ± 0.34	0.27 ± 0.25	59.93 ± 0.4	0.32 ± 0.15	0.07 ± 0.07
	RMIA	**68.64 ± 0.43**	**1.19 ± 0.28**	**0.51 ± 0.33**	**87.18 ± 0.14**	**2.06 ± 0.87**	**0.77 ± 0.74**	**79 ± 0.29**	**0.86 ± 0.4**	**0.32 ± 0.32**
2	Attack-R	63.35 ± 0.3	0.32 ± 0.15	0.08 ± 0.06	81.52 ± 0.21	0.31 ± 0.2	0.06 ± 0.08	72.02 ± 0.32	0.21 ± 0.17	0.07 ± 0.06
	LiRA	54.42 ± 0.34	0.67 ± 0.24	0.27 ± 0.12	72.21 ± 0.28	1.52 ± 0.61	0.76 ± 0.58	62.18 ± 0.47	0.57 ± 0.24	0.26 ± 0.08
	LiRA (Online)	63.97 ± 0.35	0.76 ± 0.24	0.43 ± 0.21	84.55 ± 0.16	1.15 ± 0.51	0.55 ± 0.36	73.17 ± 0.29	0.53 ± 0.24	0.12 ± 0.12
	RMIA	**70.13 ± 0.37**	**1.71 ± 0.23**	**0.91 ± 0.3**	**88.92 ± 0.2**	**4.9 ± 1.86**	**1.73 ± 1.23**	**80.56 ± 0.29**	**2.14 ± 0.53**	**0.98 ± 0.67**
4	Attack-R	63.52 ± 0.29	0.65 ± 0.21	0.21 ± 0.2	81.78 ± 0.19	0.63 ± 0.31	0.19 ± 0.2	72.18 ± 0.27	0.4 ± 0.19	0.14 ± 0.12
	LiRA	54.6 ± 0.25	0.97 ± 0.44	0.57 ± 0.4	73.57 ± 0.31	2.26 ± 1.21	1.14 ± 0.84	63.07 ± 0.41	1.03 ± 0.35	0.45 ± 0.24
	LiRA (Online)	67 ± 0.33	1.38 ± 0.37	0.51 ± 0.35	87.82 ± 0.2	3.64 ± 0.96	2.19 ± 0.99	77.06 ± 0.29	1.34 ± 0.4	0.51 ± 0.35
	RMIA	**71.02 ± 0.37**	**2.91 ± 0.64**	**2.13 ± 0.47**	**89.81 ± 0.17**	**7.05 ± 1.29**	**3.5 ± 1.1**	**81.46 ± 0.31**	**3.2 ± 0.71**	**1.39 ± 0.72**

Table 9: Performance of attacks where a **few reference models** are used. All attacks, except LiRA (Online), are **low-cost and offline** (do not need new reference models to be trained per query). Results are averaged over 10 random target models.

C.6. Performance of Attacks on Models Trained with other ML Algorithms

While the majority of contemporary machine learning (ML) models rely on neural networks, understanding how attacks generalize in the presence of other machine learning algorithms is intriguing. Although it is beyond the scope of this paper to comprehensively analyze attacks across a wide range of ML algorithms on various datasets, we conduct a simple experiment to study their impact by training models with a Gradient Boosting Decision Tree (GBDT) algorithm. Figure 15 illustrates the ROC of attacks when GBDT (with three different max depths) is employed to train models on our non-image dataset, i.e. Purchase-100. The hyper-parameters of GBDT are set as n_estimators=250, lr=0.1 and subsample=0.2, yielding a test accuracy of around 53%. For this experiment, we use two reference models (1 IN, 1 OUT). Since the output of GBDT is the prediction probability (not logit), we use this probability as the input signal for all attacks. The TPR obtained by our attack consistently outperforms all other attacks, particularly by an order of magnitude at zero FPR.

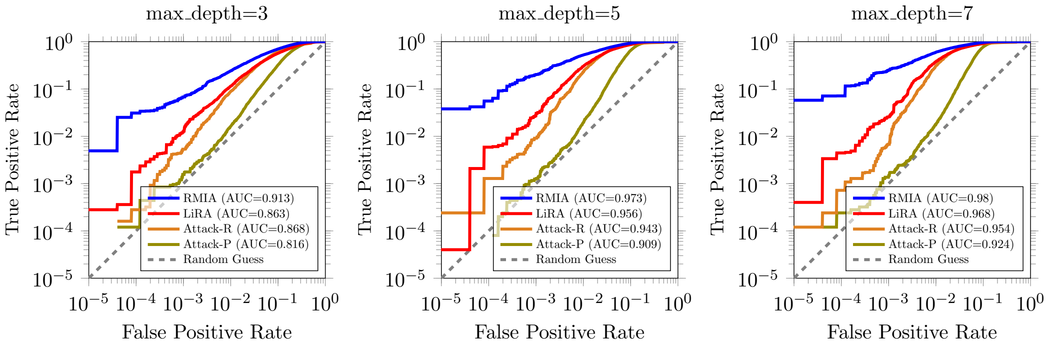


Figure 15: ROC of attacks on models trained with Gradient Boosted Decision Tree (GBDT) on the Purchase-100 dataset. Three different values of max depth parameter are used in GBDT. The test accuracy of models is around 53%. We use 2 reference models (1 IN, 1 OUT). In LiRA (Carlini et al., 2022), we use the output probability p to compute the rescaled-logit signal as $\log(\frac{p}{1-p})$. In other attacks, we use the output probability as the input signal of attacks.

C.7. Performance of Attacks Obtained on Different Datasets

We here compare the ROC of attacks using reference models trained on four different datasets, i.e. CIFAR-10, CIFAR-100, CINIC-10 and Purchase-100. We evaluate attacks when different number of reference models is used. We provide a depiction of the ROC curves in both log and normal scales. In Figure 21, we show the ROC of attacks obtained with using 1 reference model. Since we only have 1 (OUT) model, the result of offline attacks is reported here. RMIA works remarkably better than other three attacks across all datasets. Although LiRA has a rather comparable TPR at low FPR for CIFAR-10 and Purchase-100 models, but it yields a much lower AUC (e.g. 22% lower AUC in CIFAR-10), when compared with RMIA. In other two datasets, RMIA results in around 3 times better TPR at zero FPR, than its closest rival.

Figure 22 displays the ROC of attacks resulted from employing 2 reference model (1 IN, 1 OUT). Again, RMIA works much better than other three attacks (in terms of both AUC and TPR@FPR) across all datasets. For example, it has a 10% higher AUC in CIFAR-10 models and at least 3 times better TPR at zero FPR in CIFAR-100 and CINIC-10, than other attacks. Figure 23 presents the ROC of offline attacks when using 127 OUT models. RMIA works much better than other three attacks across all datasets. For example, it leads to at least 20% higher AUC than LiRA. Figure 24 illustrates the ROC of attacks obtained when using all 254 models (127 IN, 127 OUT models). With the help of training hundreds of IN and OUT models, LiRA can work close to our attack in terms of AUC, but the TPR at zero FPR of RMIA is considerably higher, e.g. by at least 50% in CIFAR-10, CINIC-10 and Purchase-100 models, as compared with LiRA.

For a better comparison between attack performances concerning the number of reference models, Figure 25 presents the AUC results for both offline and online attacks across varying reference model counts (ranging from 1 to 254). The left plots in this figure showcase the outcomes of offline attacks, while the right plots highlight the performance of online attacks. A consistent trend emerges, revealing that an increase in the number of reference models yields an improvement in AUC across all attacks. Notably, in both offline and online scenarios and across all datasets, RMIA consistently outperforms other attacks, particularly when employing a limited number of models.

C.8. Performance of Attacks on Purchase-100 Models

Table 10 presents the attack results when evaluating models trained on the Purchase-100 dataset. The observed superiority aligns with findings from other datasets: employing more reference models enhances the performance of all attacks. However, across all scenarios, RMIA consistently outperforms other attacks in terms of both AUC and TPR at low FPR metrics. Notably, RMIA with just 1 (OUT) reference model achieves over 30% higher AUC than LiRA. The offline RMIA with 2 or 4 reference models can surpass online LiRA (using the same number of reference models). RMIA can maintain its superiority even in the presence of hundreds of reference models.

# Ref	Attack	AUC	TPR@FPR	
			0.01%	0.0%
0	Attack-P	66.62 ± 0.27	0 ± 0.01	0 ± 0
1	Attack-R	74.24 ± 0.28	0 ± 0.01	0 ± 0
	LiRA	59.3 ± 0.35	0.2 ± 0.09	0.1 ± 0.06
	RMIA	$\mathbf{77.8 \pm 0.2}$	$\mathbf{0.33 \pm 0.2}$	$\mathbf{0.14 \pm 0.09}$
2	Attack-R	74.8 ± 0.26	0.05 ± 0.05	0.01 ± 0.02
	LiRA (Online)	72.97 ± 0.46	0.37 ± 0.09	0.2 ± 0.07
	RMIA	$\mathbf{80.44 \pm 0.24}$	$\mathbf{0.89 \pm 0.38}$	$\mathbf{0.23 \pm 0.19}$
4	Attack-R	75.68 ± 0.36	0.09 ± 0.07	0.01 ± 0.02
	LiRA (Online)	77.57 ± 0.52	0.63 ± 0.1	0.27 ± 0.07
	RMIA	$\mathbf{81.94 \pm 0.29}$	$\mathbf{1.58 \pm 0.67}$	$\mathbf{0.41 \pm 0.4}$
127	Attack-R	77.80 ± 0.41	1.02 ± 0.35	0.42 ± 0.36
	LiRA	65.82 ± 0.58	0.31 ± 0.13	0.11 ± 0.07
	RMIA	$\mathbf{83.21 \pm 0.33}$	$\mathbf{2.35 \pm 0.92}$	$\mathbf{0.69 \pm 0.67}$
254	LiRA (Online)	83.23 ± 0.37	1.99 ± 0.88	0.82 ± 0.59
	RMIA (Online)	$\mathbf{83.90 \pm 0.36}$	$\mathbf{2.56 \pm 0.79}$	$\mathbf{1.29 \pm 0.91}$

Table 10: Performance of attacks obtained for attacking models trained on **Purchase-100** using varying number of reference models. All attacks are **low-cost and offline** by default (do not need new reference models to be trained per query). However, we also provide the result of online versions of LiRA and RMIA to showcase their highest performance. We do not use augmented queries. Results are averaged over 10 random target models.

C.9. Performance of RMIA with Different Number of z Samples

Table 11 presents the performance of RMIA obtained with 1 (OUT) reference model, as we change the number of reference samples. We here observe roughly the same trend of results as the one we saw in Table 4 where we employed 254 reference models. Specifically, the AUC increases when we use more reference samples, but using 2500 population samples, equivalent to 10% of the size of the models' training set, brings results comparable to those obtained with a 10 fold larger population set.

# population samples z	AUC	TPR@FPR	
		0.01%	0.0%
25 samples	58.92 ± 0.22	0 ± 0	0 ± 0
250 samples	63.28 ± 0.2	0.29 ± 0.16	0.12 ± 0.09
1250 samples	64.75 ± 0.27	0.31 ± 0.15	0.17 ± 0.1
2500 samples	65.08 ± 0.22	0.31 ± 0.16	0.19 ± 0.14
6250 samples	65.35 ± 0.24	0.34 ± 0.18	0.19 ± 0.13
12500 samples	65.45 ± 0.25	0.32 ± 0.17	0.2 ± 0.16
25000 samples	65.46 ± 0.16	0.61 ± 0.16	0.47 ± 0.1

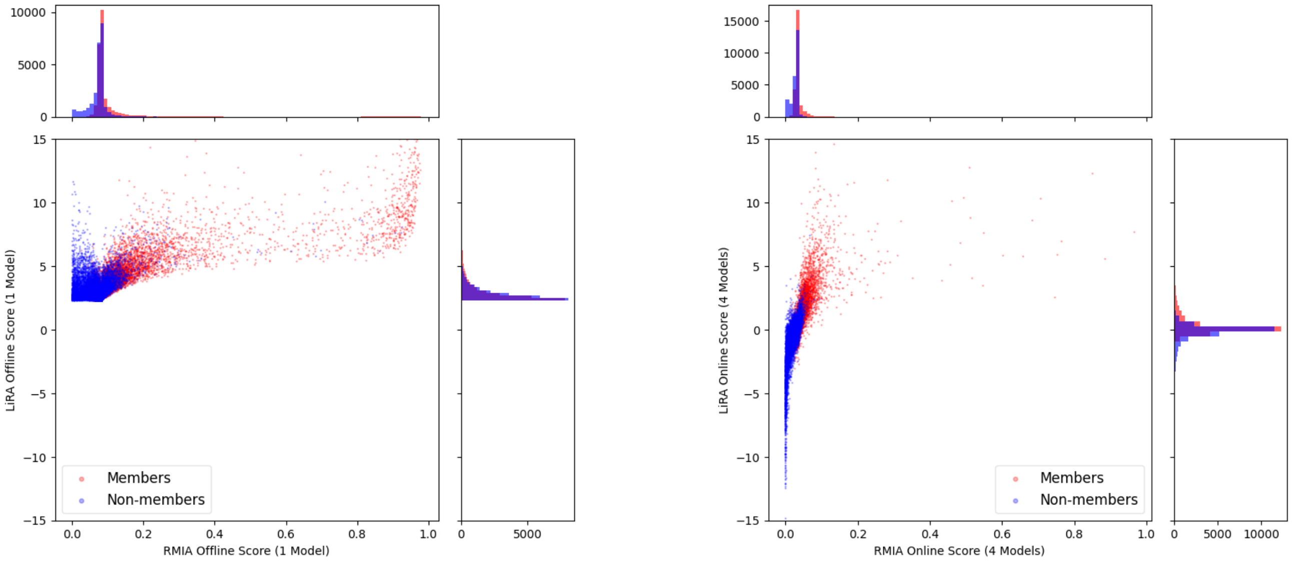
Table 11: Performance of RMIA (Offline) using different number of random z samples. We only use **1 (OUT) reference model**, trained on CIFAR-10. Here, we do not use augmented queries. Results are averaged over 10 random target models.

C.10. MIA Score Comparison between Attacks

To better understand the difference between the performance of our attack with others', we compare the MIA score of member and non-member samples obtained in RMIA and other attacks. Figure 16 displays RMIA scores versus LiRA scores in two scenarios: one using only 1 reference model (shown on the left) and another using 4 reference models (shown on the right). With just 1 reference model, RMIA provides clearer differentiation between numerous member and non-member samples, as it assigns distinct MIA scores in the $[0, 1]$ range, separating many members on the right side and non-members on the left side. In contrast, LiRA scores are more concentrated towards the upper side of the plot, lacking a distinct separation between member and non-member scores. When employing more reference models, we observe a higher degree of correlation between the scores of the two attacks.

Similarly, Figure 17 shows RMIA scores compared to Attack-R scores in the same two scenarios. RMIA can better separate members from non-members via assigning distinct MIA scores to them (member scores apparently tend to be larger than non-member scores for lots of samples). In contrast, there is no such clear distinction in Attack-R.

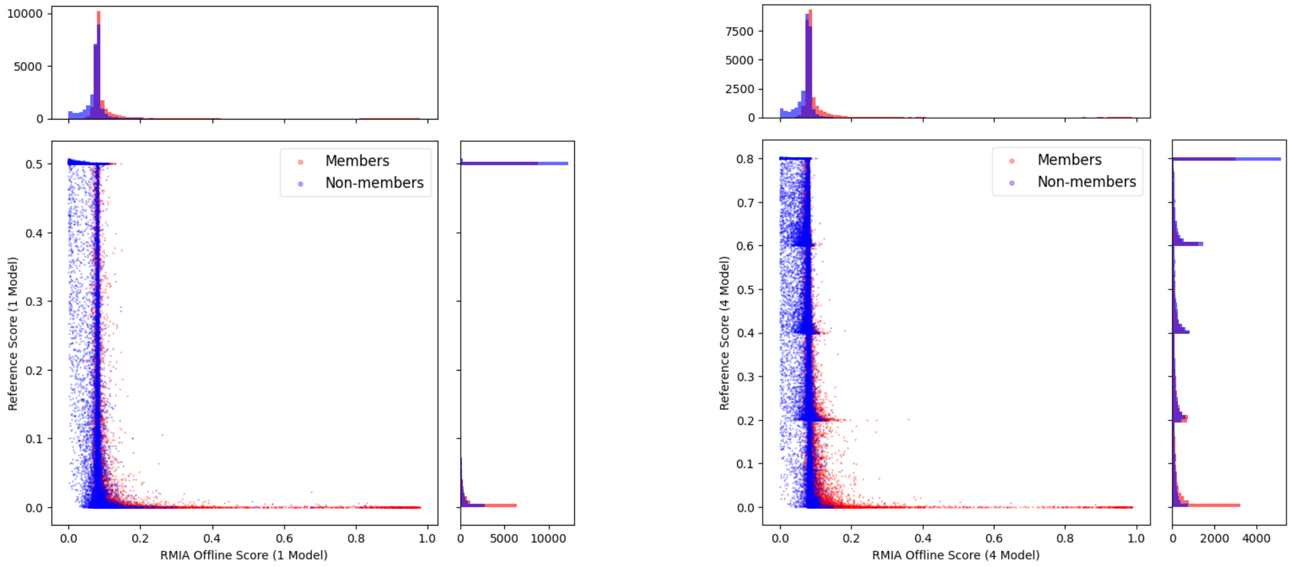
We also show RMIA scores versus Attack-P scores in Figure 18. In this experiment, we only use 1 reference model (OUT) for RMIA, because Attack-P does not work with reference models. As opposed to RMIA, Attack-P clearly fails to provide a good separation between member and non-member scores.



(a) LiRA against RMIA with 1 model

(b) Against LiRA against RMIA with 4 models

Figure 16: MIA score comparison between RMIA and LiRA (Carlini et al., 2022) over all member and non-member samples of a random target model. The left plot is obtained when only 1 (OUT) model is used, while the right plot uses 4 reference models (2 IN and 2 OUT). In both plots, the x-axis and y-axis represent RMIA and LiRA scores, respectively.



(a) Attack-R against RMIA with 1 model

(b) Attack-R against RMIA with 4 models

Figure 17: MIA score comparison between RMIA and Attack-R (Ye et al., 2022). The left plot is obtained when only 1 reference model (OUT) is used, while the right plot uses 4 models (OUT). In both plots, the x-axis and y-axis represent RMIA and Attack-R scores, respectively.

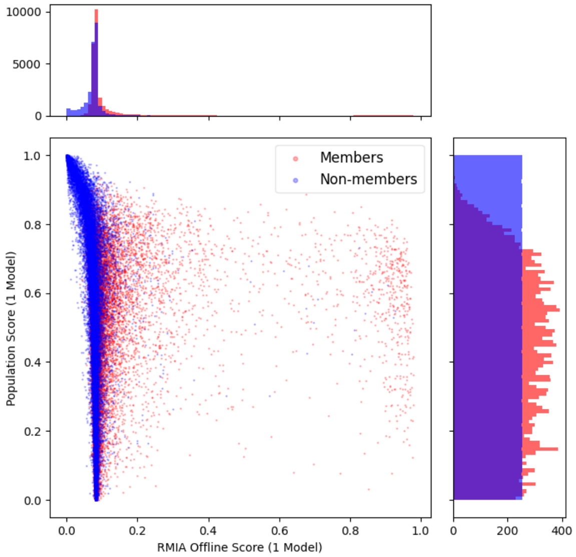


Figure 18: MIA score comparison between RMIA and Attack-P (Ye et al., 2022) obtained for a random target model. The x-axis represents RMIA scores, while the y-axis depicts Attack-P scores. For RMIA, we use only 1 OUT model.

Strong inference attacks compute $\mathrm{Score}_{\mathrm{MIA}}(x;\theta)$ in such a way that it accurately reflects the distinguishability between models that are trained on a target data point and the ones that are not. For a better understanding of the distinctions between attacks, Figure 26 illustrates the distribution of MIA scores obtained from various attacks for a set of test samples. We compute $\mathrm{Score}_{\mathrm{MIA}}(x;\theta)$ on 254 target models, with the sample being a member to half of them and a non-member to the other half. The attacks are conducted using 127 OUT models. The MIA score produced by RMIA contributes to a more apparent separation between members and non-members (member scores tend to concentrate in the right part of the plot, as opposed to other attacks).

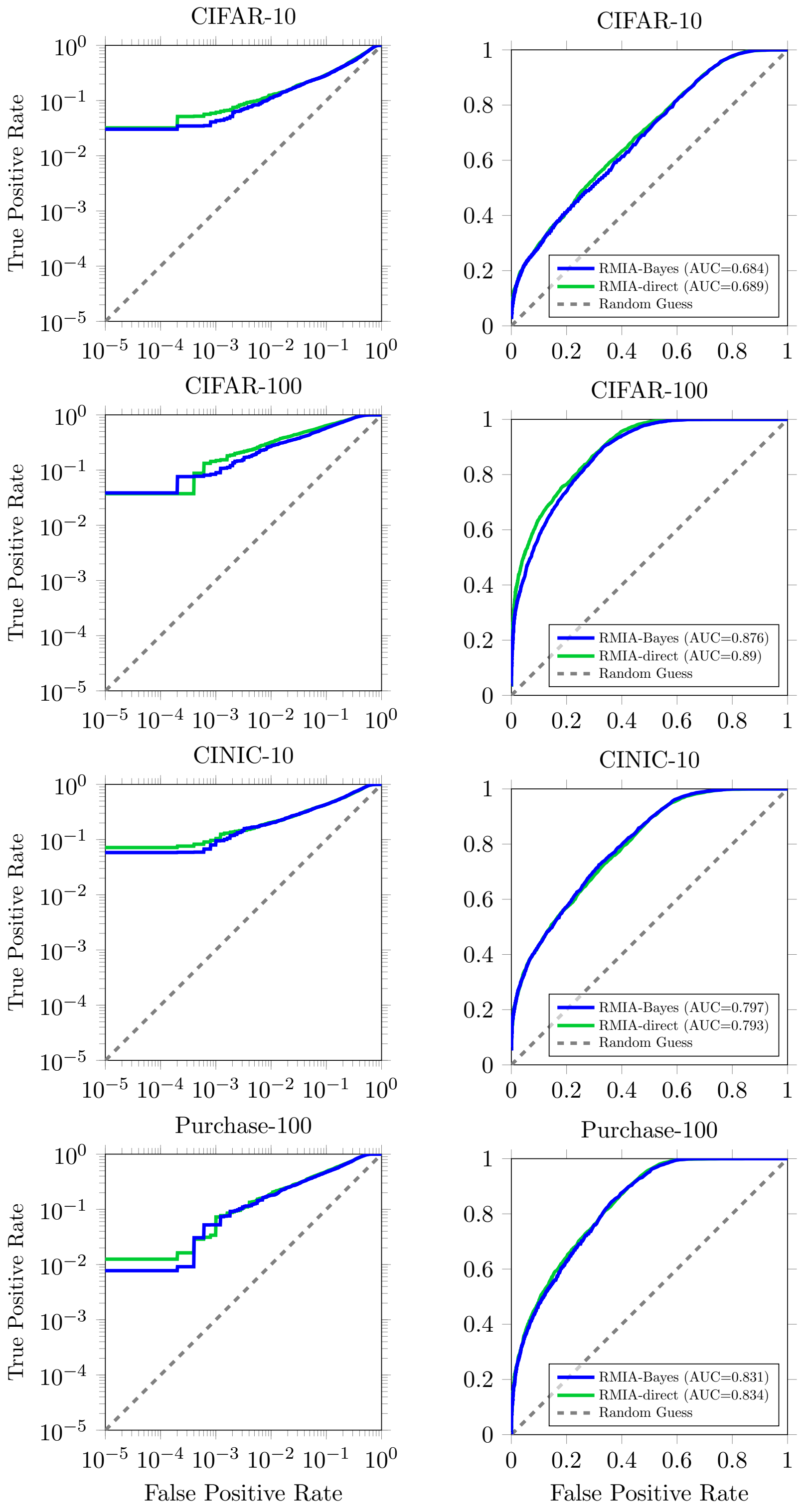


Figure 19: ROC of RMIA when two different methods, i.e. RMIA-direct (as formulated in equation 11) and RMIA-Bayes (based on equation 3), are used to approximate the likelihood ratio. ROCs are shown in both log and normal scales. Here, we use **64 reference models**.

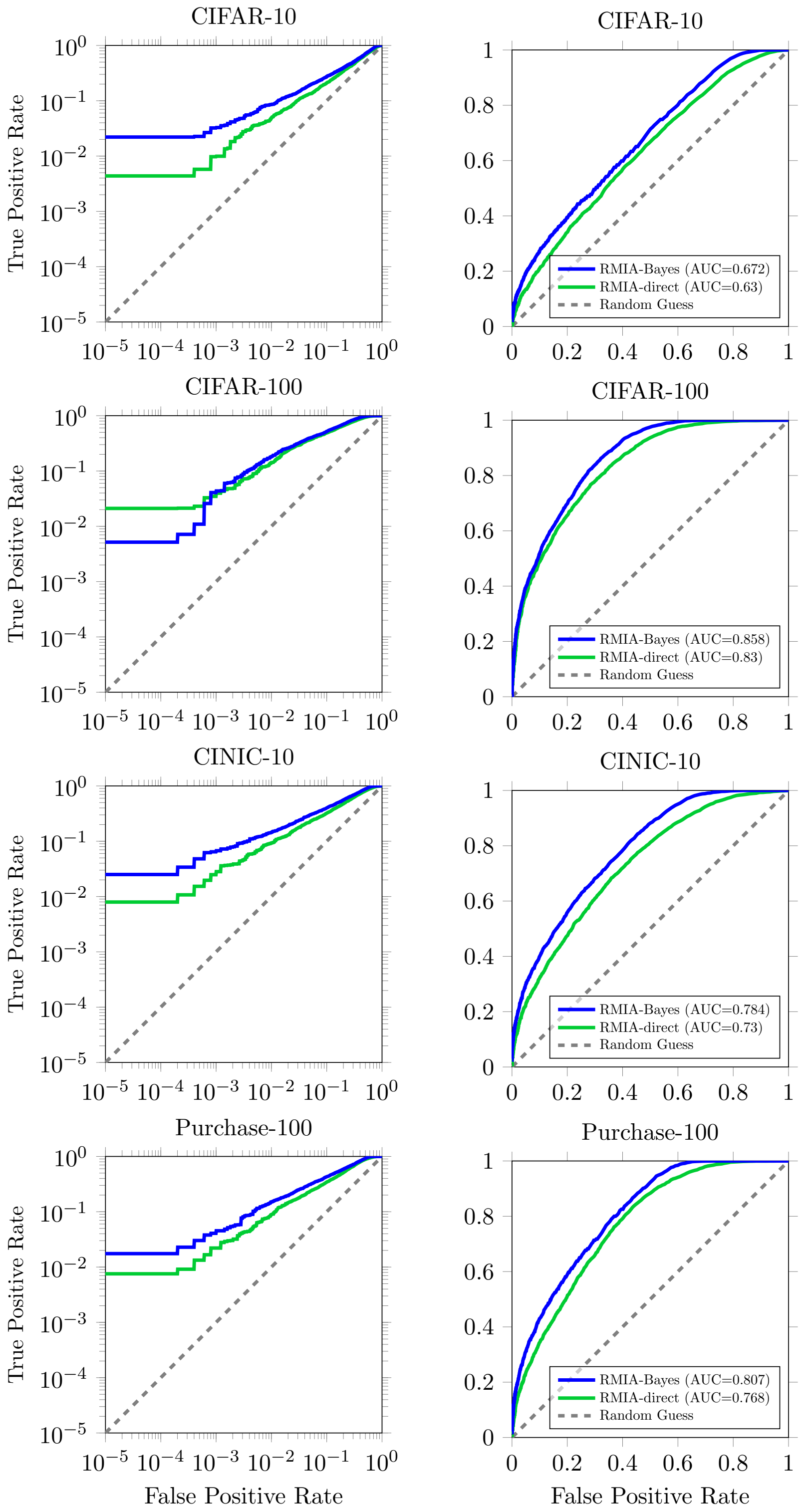


Figure 20: ROC of RMIA when two different methods, i.e. RMIA-direct (as formulated in equation 11) and RMIA-Bayes (based on equation 3), are used to approximate the likelihood ratio. ROCs are shown in both log and normal scales. Here, we use **4 reference models**.

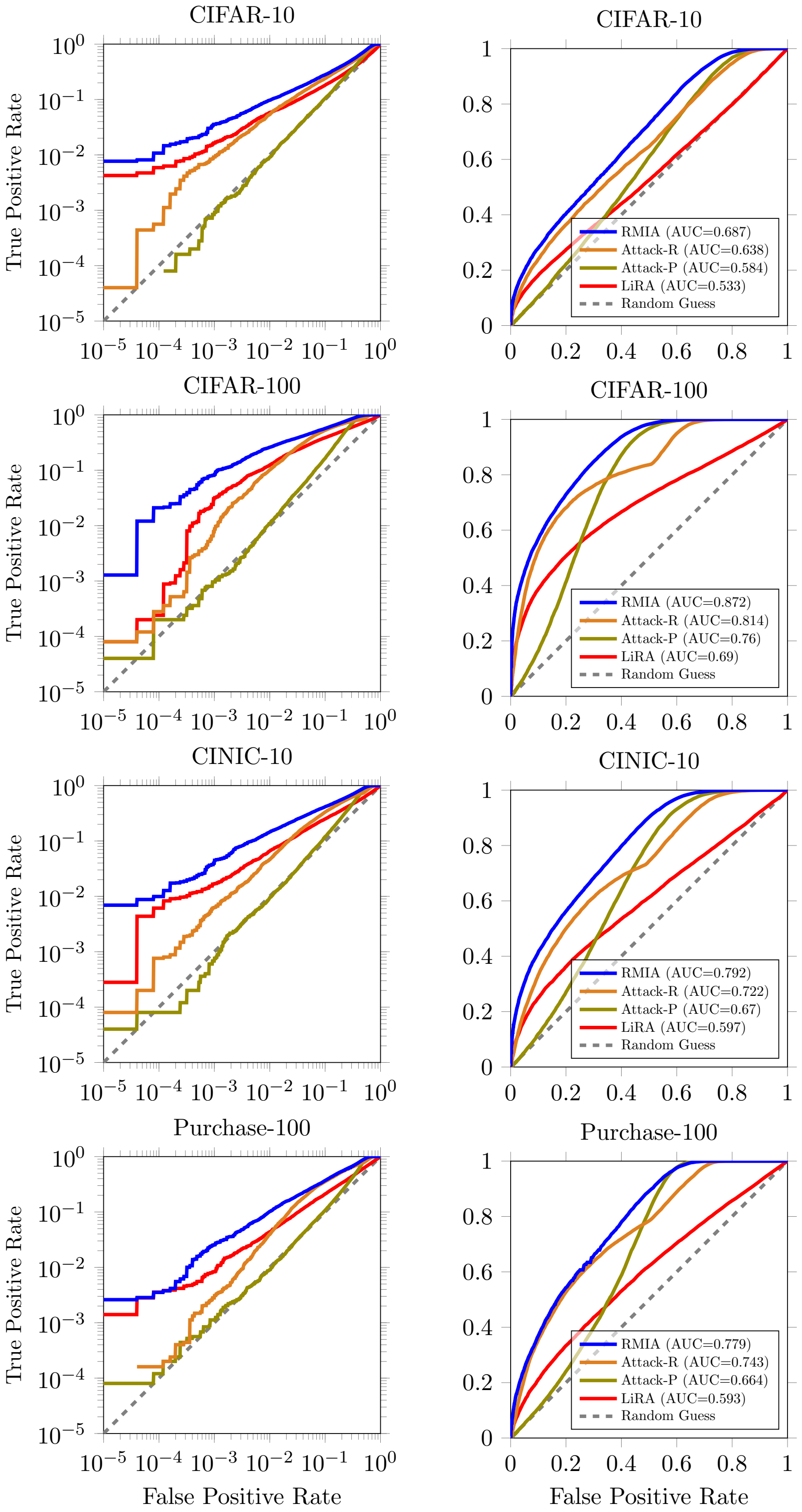


Figure 21: ROC of attacks using models trained on different datasets (ROCs are shown in both log and normal scales). The result is obtained on one random target model. We here use **1 reference model** (OUT).

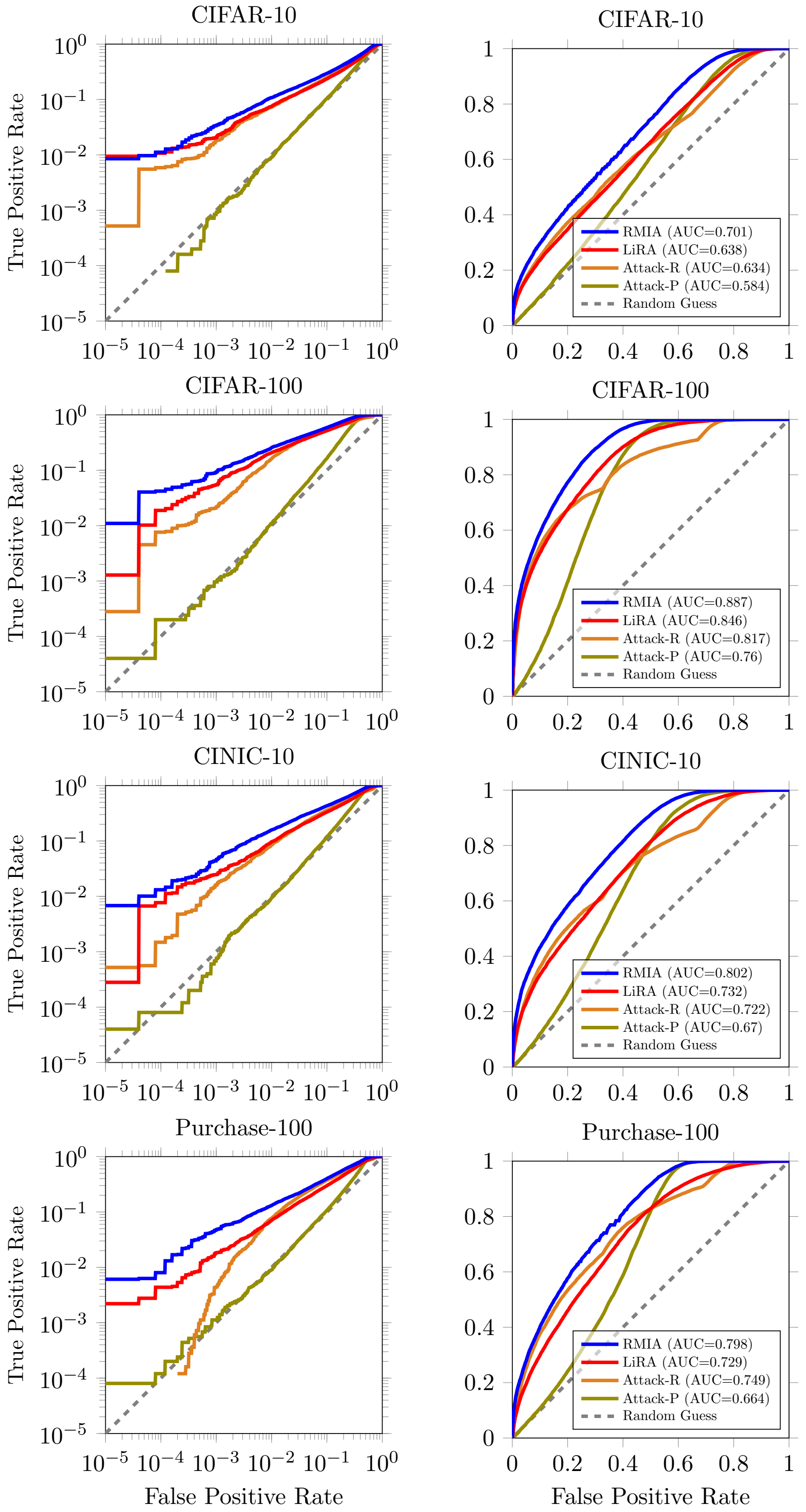


Figure 22: ROC of attacks using models trained on different datasets (ROCs are shown in both log and normal scales). The result is obtained on one random target model. We here use **2 reference models** (1 IN, 1 OUT).

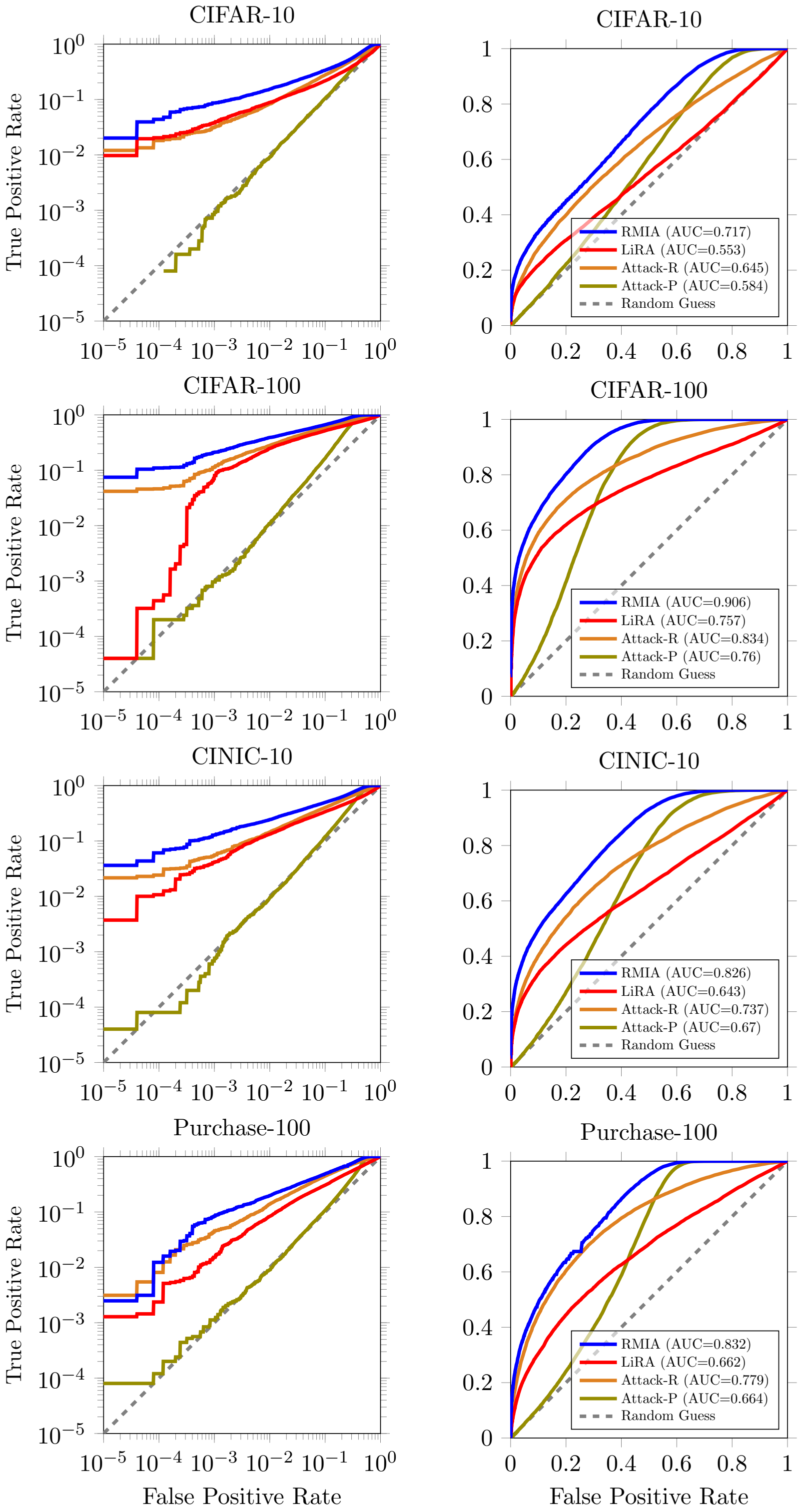


Figure 23: ROC of offline attacks using models trained on different datasets (ROCs are shown in both log and normal scales). The result is obtained on one random target model. We use **127 reference models** (OUT).

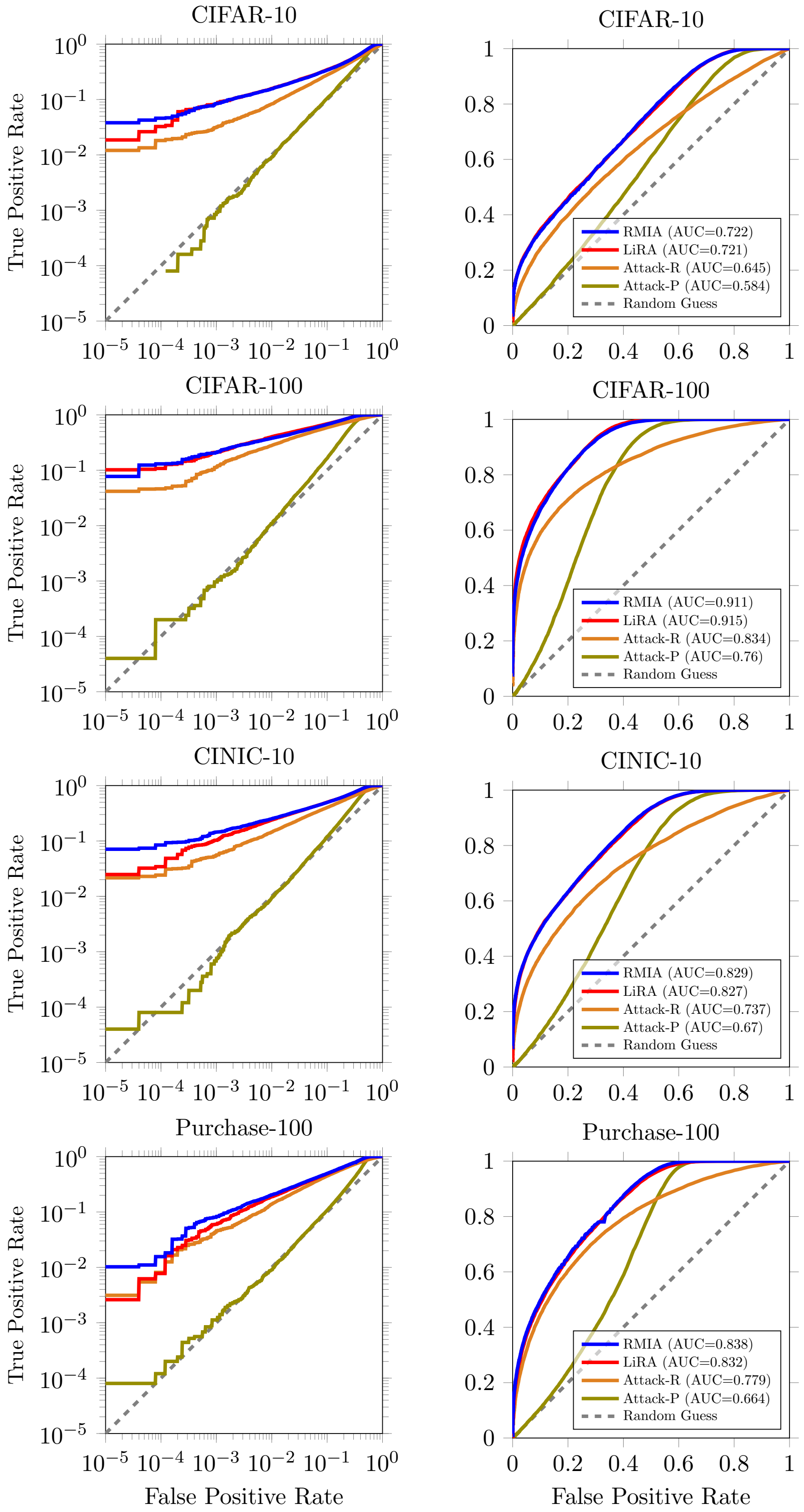
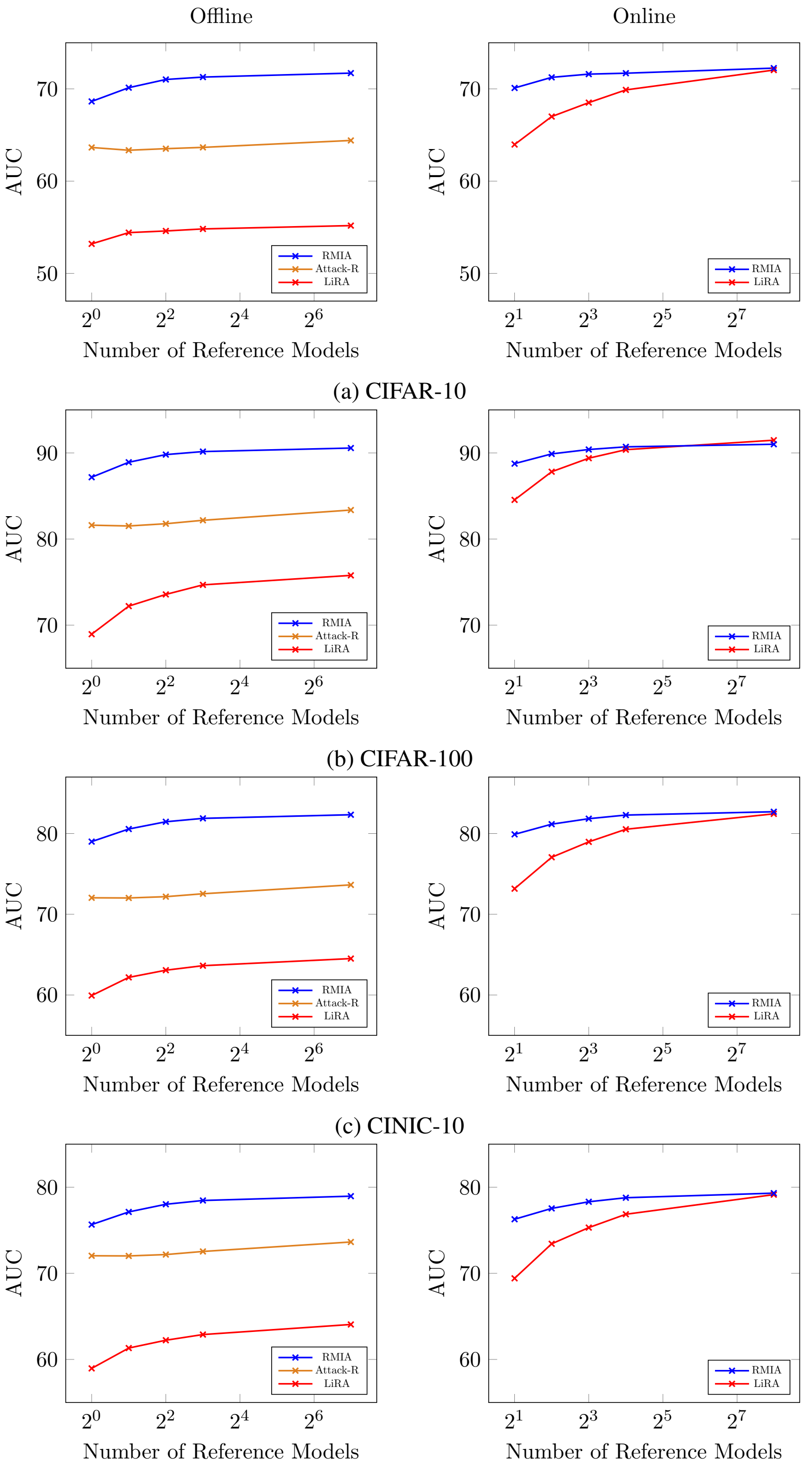


Figure 24: ROC of attacks using models trained on different datasets (ROCs are shown in both log and normal scales). The result is obtained on one random target model. We here use **254 reference models** (127 IN, 127 OUT).



(a) CIFAR-10

(b) CIFAR-100

(c) CINIC-10

(d) Purchase-100

Figure 25: AUC of various attacks obtained with using different number of reference models. The left plots illustrate the results of offline attacks, while the right ones depict the AUC scores obtained by online attacks. For online attacks, half of reference models are OUT and half are IN.

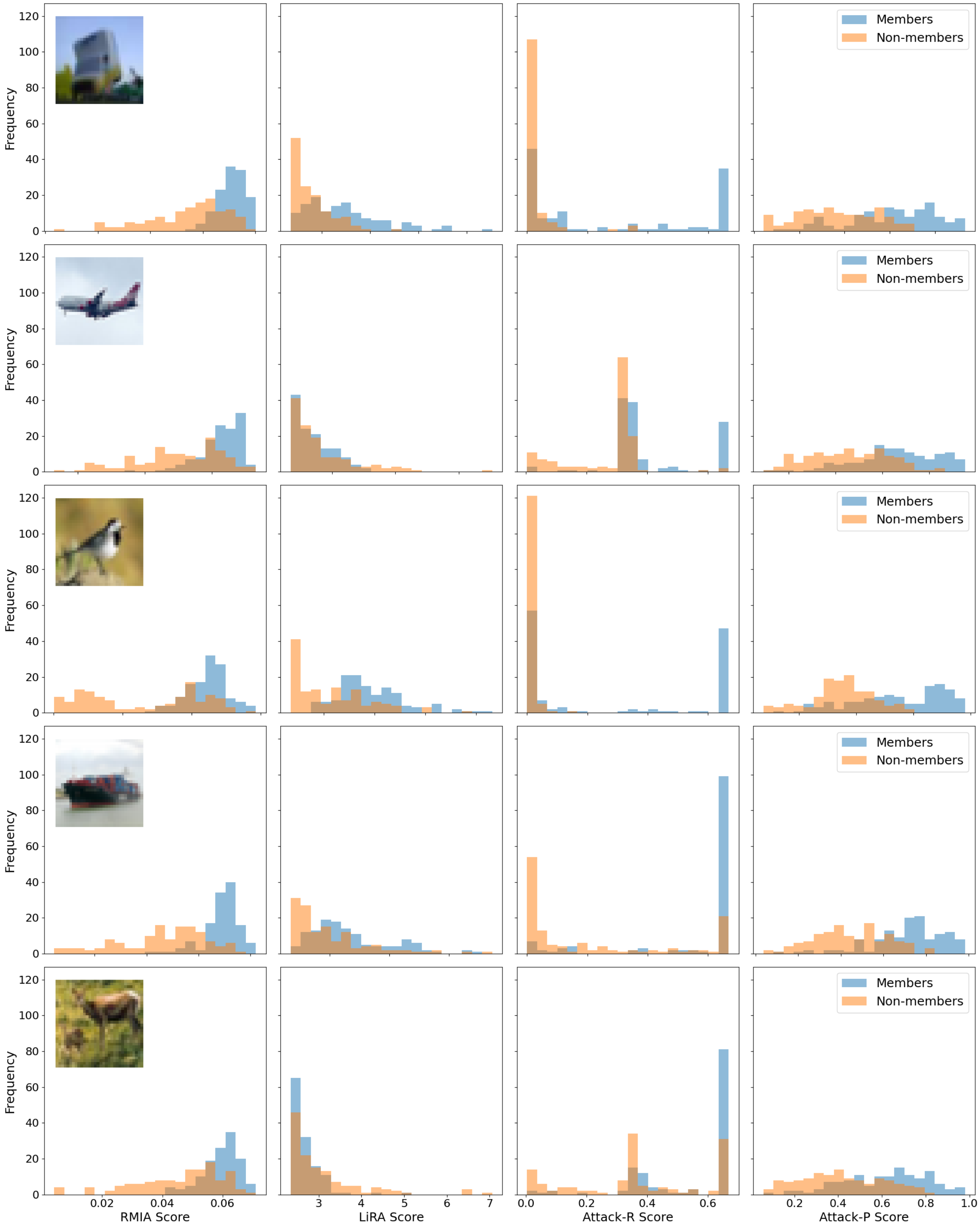


Figure 26: Distribution of MIA scores obtained from different attacks for some test sample. We calculate the MIA score of each sample across 254 target models where half of them are IN models, and the remaining half are OUT models. To perform attacks, we use 127 reference models (OUT), trained on CIFAR-10.

D. Related Work

Neural networks, particularly when trained with privacy-sensitive datasets, have been proven to be susceptible to leaking information about their training data. A variety of attacks have been designed to gauge the degree of leakage and the subsequent privacy risk associated with training data. For instance, *data extraction attacks* attempt to recreate individual samples used in training the model (Carlini et al., 2021), whereas *model inversion attacks* focus on extracting aggregate information about specific sub-classes instead of individual samples (Fredrikson et al., 2015). In contrast, *property inference attacks* derive non-trivial properties of samples in a target model's training dataset (Ganju et al., 2018). This paper, however, is concerned with *membership inference attacks (MIAs)*, which predict whether a particular sample was used in the training process of the model (Shokri et al., 2017). MIAs, due to their simplicity, are commonly utilized as auditing tools to quantify data leakage in trained models.

MIAs first found their use in the realm of genome data to identify the presence of an individual's genome in a mixed batch of genomes (Homer et al., 2008). Backes et al. (2016) went on to formalize the risk analysis for identifying an individual's genome from aggregate statistics on independent attributes, with this analysis later extended to include data with dependent attributes (Murakonda et al., 2021).

Algorithms featuring differential privacy (DP) are designed to limit the success rate of privacy attacks when distinguishing between two neighboring datasets (Nasr et al., 2021). Some researches, such as Thudi et al. (2022), provide upper limits on the average success of MIAs on general targets. Other studies evaluate the effectiveness of MIAs on machine learning models trained with DP algorithms (Rahman et al., 2018).

Shokri et al. (2017) introduced membership inference attacks for machine learning algorithms. The work demonstrated the efficacy of membership inference attacks against machine learning models in a setting where the adversary has query access to the target model. This approach was based on the training of reference models, also known as shadow models, with a dataset drawn from the same distribution as the training data. Subsequent works extended the idea of shadow models to different scenarios, including white-box (Leino & Fredrikson, 2020; Nasr et al., 2019; Sablayrolles et al., 2019) and black-box settings (Song & Mittal, 2021; Hisamoto et al., 2020; Chen et al., 2021), label-only access (Choquette-Choo et al., 2021; Li & Zhang, 2021), and diverse datasets (Salem et al., 2019). However, such methods often require the training of a substantial number of models upon receiving an input query, making them unfeasible due to processing and storage costs, high response times, and the sheer amount of data required to train such a number of models. MIA has been also applied in other machine learning scenarios, such as federated learning (Nasr et al., 2019; Melis et al., 2019; Truex et al., 2019) and multi-exit networks (Li et al., 2022).

Various mechanisms have been proposed to defend against MIAs, although many defense strategies have proven less effective than initially reported (Song & Mittal, 2021). Since the over-fitting issue is an important factor affecting membership leakage, several regularization techniques have been used to defend against membership inference attacks, such as L2 regularization, dropout and label smoothing (Shokri et al., 2017; Salem et al., 2019; Liu et al., 2022a). Some recent works try to mitigate membership inference attacks by reducing the target model's generalization gap (Li et al., 2021; Chen et al., 2022) or self-distilling the training dataset (Tang et al., 2022). Abadi et al. (2016) proposed DP-SGD method which adds differential privacy (Dwork, 2006) to the stochastic gradient descent algorithm. Subsequently, some works concentrated on reducing the privacy cost of DP-SGD through adaptive clipping or adaptive learning rate (Yu et al., 2019; Xu et al., 2020). In addition, there are defense mechanisms, such as AdvReg (Nasr et al., 2018) and MemGuard (Jia et al., 2019), that have been designed to hide the distinctions between the output posteriors of members and non-members.

Recent research has emphasized evaluating attacks by calculating their true positive rate (TPR) at a significantly low false positive rate (FPR) (Carlini et al., 2022; Ye et al., 2022; Liu et al., 2022b; Long et al., 2020; Watson et al., 2022b). For example, Carlini et al. (2022) found that many previous attacks perform poorly under this evaluation paradigm. They then created an effective attack based on a likelihood ratio test between the distribution of models that use the target sample for training (IN models) and models that do not use it (OUT models). Despite the effectiveness of their attack, especially at low FPRs, it necessitates the training of many reference models to achieve high performance. Watson et al. (2022b) constructed a membership inference attack incorporating sample hardness and using each sample's hardness threshold to calibrate the loss from the target model. Ye et al. (2022) proposed a template for defining various MIA games and a comprehensive hypothesis testing framework to devise potent attacks that utilize reference models to significantly improve the TPR for any given FPR. Liu et al. (2022b) presented an attack which utilizes trajectory-based signals generated during the training of distilled models to effectively enhance the differentiation between members and non-members. The recent paper (Wen et al., 2023) has improved the performance of likelihood test-driven attacks by estimating a variant of the target sample through minimally perturbing the original sample, which minimizes the fitting loss of IN and OUT shadow models. Lately, Leemann et al. (2023) have introduced a novel privacy notion called f-MIP, which allows for bounding the trade-off between the power and error of attacks using a function f. This is especially applicable when models are trained using gradient updates. The authors demonstrated the use of DP-SGD to achieve this f-MIP bound. Additionally, they proposed a cost-effective attack for auditing the privacy leakage of ML models, with the assumption that the adversary has white-box access to gradients of the target model.

A number of related work tune the threshold β in MIA based on the data point. Chang & Shokri (2021) improve the attack by determining the threshold for a data point, based on its attributes (e.g., data points corresponding to the same gender are used as the reference population to determine the threshold corresponding to a given FPR.) When it is not very easy to identify the reference population, Bertran et al. (2023) train an attack model for each FPR threshold to determine the threshold associated with the reference population. The attack slightly improves over vanilla population attack (Attack-P or loss attack).

Some recent studies have focused on low-cost auditing of differentially private deep learning algorithms. This is a different setting in terms of the problem statement, yet it is aligned with the spirit of our work that aims at reducing the auditing cost. Steinke et al. (2023) outline an efficient method for auditing differentially private machine learning algorithms. Their approach achieves this with a single training run, leveraging parallelism in the addition or removal of multiple training examples independently.

In alignment with MIA research (Sankararaman et al., 2009; Murakonda et al., 2021; Ye et al., 2022; Carlini et al., 2022), we highlight the power (TPR) of MIAs at extremely low errors (FPR). We propose a different test that allows us to devise a new attack that reaps the benefits of various potent attacks. The attack proposed in this paper demonstrates superior overall performance and a higher TPR at zero FPR than the attacks introduced by Carlini et al. (2022); Ye et al. (2022), especially when only a limited number of reference models are trained. Additionally, it maintains high performance in an offline setting where none of the reference models are trained with the target sample. This characteristic renders our attack suitable for practical scenarios where resources, time, and data are limited.