ON RELIABILITY OF MEMBERSHIP INFERENCE VULNERABILITY EVALUATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Membership inference attacks (MIAs) are popular methods for empirically assessing the leakage of sensitive information in the training data through models or statistics learned from the data. The MI vulnerability is often evaluated through a binary classifier that tries to predict whether a particular sample was in the training data. In order to evaluate the effectiveness of MIAs multiple *shadow models* are trained using random partitions of a larger dataset. After training the shadow models the MI vulnerability can be evaluated for all the samples for which we obtained shadow models. In order to evaluate the MI vulnerability reliably one needs a lot of shadow models which can be computationally infeasible. Therefore instead of reporting the actual sample level vulnerabilities aggregates over multiple samples are often reported in practice. We demonstrate two key weaknesses in typical MIA evaluation pipeline. First, we show that sampling the shadow datasets from a fixed superset leads to finite sample bias inflating the vulnerability estimates. Second, we show that evaluating the true positive rate (TPR) by concatenating MIA scores across multiple individuals, commonly used in the very low false positive rate (FPR) regime, is not calibrated across the per-sample FPRs. For both weaknesses we propose fixes that in the most simple approximate form do not incur any additional computation cost. We show that with additional computation one can further improve the reliability of the vulnerability estimation.

1 INTRODUCTION

Efficiently evaluating the privacy risk associated with a particular machine learning algorithm applied to a particular dataset is critical for calibrating the level of privacy protection, but no good general solutions exist for this problem. The difficulty of the problem arises from many sources: 1) the privacy risk is a statistical phenomenon and evaluating it requires repeated observations; 2) identifying the most vulnerable samples needed to evaluate maximum risk can be difficult; and 3) it is difficult to know if the applied attack is optimal.

Privacy auditing methods (Annamalai et al., 2026) address each of these by 1) developing specific one-run-auditing (ORA) techniques (Steinke et al., 2023; Mahloujifar et al., 2025); 2) crafting canary samples to represent maximally vulnerable samples in the entire data universe (Boglioni et al., 2026); and 3) potentially using more easily auditable internals of the algorithm to ensure strong attacks. These methods have weaknesses: the reliability of ORA is fundamentally limited (Xiang et al., 2025; Keinan et al., 2025), and the actual samples may be more or less vulnerable than the canaries.

To evaluate the vulnerability of the actual samples, we focus on membership inference attacks (MIAs) for privacy risk estimation. MIA is based on a binary classification problem: was a given sample part of the training dataset or not. MIAs are useful for this purpose because unlike stronger privacy attacks like attribute inference and training data reconstruction, the setting is highly standardised. MIA vulnerability is related to vulnerability to these other attacks by reductions (Salem et al., 2023). The most widely used privacy enhancing technology in machine learning, differential privacy (Dwork et al., 2006), can be naturally interpreted as a bound on MIA success (Wasserman and Zhou, 2010).

Following Carlini et al. (2022), MIA evaluation has focused on evaluating the true positive rate (TPR) of membership identification at fixed low false positive rate (FPR) averaged over multiple members of the training data. This is most sensitive to the adversary's ability to confidently detect even a small number of true members of the training dataset.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

When we evaluate the MIA success over multiple data points, we are measuring the average vulnerability instead of the sample-level vulnerability. Aerni et al. (2024), Tobaben et al. (2025) and Knolle et al. (2026) have shown how the average vulnerability can significantly fail to communicate the vulnerability of the samples. However, reliably estimating the per-sample vulnerability becomes expensive for *shadow-model-based* MIA where we need to train multiple models to estimate the effect the sample has to the learning outcomes.

In this paper, we study both the per-sample and average estimates of MIA vulnerability. We demonstrate that estimating the per-sample vulnerability by learning the shadow models by sampling their training data from a finite superset leads to inflated vulnerability. This bias also affects the average vulnerability estimation. We demonstrate how to mitigate the bias with a simple correction method. We also demonstrate that the reporting the average vulnerability through a ROC curve computed from MIA scores combined over multiple target samples is not calibrated in terms of the per-sample FPR. We demonstrate a simple post-processing fix for this issue.

Contributions

- We demonstrate that the most common approaches for performing shadow-model-based MIA suffer from finite population bias in estimating the in/out-distribution variances. This leads to inflated vulnerability estimates (Sections 3 and 5). We propose ways to fix this.
- We show that concatenated MIA aggregation reports an average over per-sample vulnerabilities, but that a single global threshold induces heterogeneous per-sample FPRs. This miscalibration can distort conclusions about which learning algorithms or defenses are more private (Section 4.1).
- We show that offline LiRA (Carlini et al., 2022) avoids the calibration problem. Based on this we form a post-processing-based approach that calibrates the per-sample FPRs and allows the concatenation to report average per-sample TPR with the targeted FPR under mild Gaussian assumptions (Sections 4.2 and 5).

Related Work

Many works have focused on MIA evaluation through the ROC-curve for the binary classification task. Carlini et al. (2022) proposed using TPR at low FPR as a metric for the vulnerability, and proposed two MIA attacks: the online and offline LiRA attack which are both based on approximating the shadow distributions with Gaussians. Both of their attacks compare the target point to reference models and devise a threshold classifier to make the decision whether the point is in the training data or not. Ye et al. (2022) also focuses on TPR at low FPR, and shows how the decision threshold for the classifier should be chosen to match different types of membership inference games, e.g. how to choose the threshold if we want to target the MIA for fixed sample or model. In our work we study the per-sample vulnerability, which matches their Attack R setting. While these works focus on vulnerability evaluation against decision threshold classifiers, neither of the works discuss how the vulnerability should be aggregated over multiple models or samples. This is crucial in order to understand the vulnerability over the population reliably, as sample-wise vulnerability evaluation becomes expensive due to massive amounts of shadow models needed to achieve statistically significant evaluation.

While Carlini et al. (2022) do not study the aggregation of the MIA vulnerabilities formally, implementations of the LiRA algorithm e.g. in the ML Privacy Meter library (Murakonda and Shokri, 2020) compute an aggregate of the vulnerability by concatenating the LiRA scores over multiple individuals and target models. In many cases, this is the only feasible way to reach very small FPR values without training an excessive number of models. In this work, we explain mathematically what this strategy actually measures, and propose an alternative that more closely relates to the per-sample vulnerability.

Aerni et al. (2024) showed how the average vulnerability can significantly underestimate the vulnerability of a single data point. They demonstrated this by computing the per-sample vulnerabilities for multiple samples and showed that for a fixed per-sample $\mathrm{FPR} = 0.1\%$ the TPR can reach 99.9% while the population average remains at 4%. Later, Tobaben et al. (2025) and Knolle et al. (2026) have similarly shown that individual points can be significantly more vulnerable than the population average. Our study on the aggregated vulnerability is closely related, but instead of learning the per-sample decision thresholds which is computationally expensive, we study the average vulnerability of the fixed threshold classifier arising from the concatenated MIA scores.

2 BACKGROUND

2.1 MEMBERSHIP INFERENCE ATTACKS

The goal of the MIA is to infer whether a particular training sample belongs to the training dataset of a particular learning outcome (Shokri et al., 2017). Denoting the target sample with x_{target} and the learning outcome with $\mathcal{M}_{\text{target}}$, MIA can be formally stated as the following binary hypothesis test

$$H_0 : x_{\text{target}} \notin D_{\text{train}} \quad \text{vs.} \quad H_1 : x_{\text{target}} \in D_{\text{train}}, \tag{1}$$

based on observing $\mathcal{M}_{\text{target}} \leftarrow \mathcal{A}(D_{\text{train}})$ where D_{train} is latent. In this work we will focus on *shadow-model*-based MIA where the test statistic for the above hypothesis test is based on comparing the observed $\mathcal{M}_{\text{target}}$ to the distributions of learning outcomes when x_{target} is in or out of the D_{train}. For a data distribution $\mathcal{P}$, the in and out samples are distributed as

$$D \sim \mathcal{P}^N \tag{2}$$

$$\mathcal{M}_{\text{out}} = \mathcal{A}(D \setminus \{x_{\text{target}}\}) \tag{3}$$

$$\mathcal{M}_{\text{in}} = \mathcal{A}(D \cup x_{\text{target}}). \tag{4}$$

We will denote the in and out distributions for a sample x respectively as $p_{\text{in},x}$ and $p_{\text{out},x}$. Instead of performing the attack directly on the $\mathcal{M}_{\text{target}}$, it is common to apply some deterministic function, e.g. a logit, to the model and target point before the attack to make the distribution 1D and approximately Gaussian. We denote such function that takes the sample and the learning outcome as arguments with $\mathcal{S}(x_{\text{target}}, \mathcal{M})$.

Ye et al. (2022) studied MIA under various different threat models, i.e. with different levels of attacker's knowledge. They proposed a unified threshold classifier type of attack, where the decision threshold is adapted to the targeted threat model, e.g. if we want to attack particular target samples or models. In this paper we will focus on the per-sample attacks, and the MIA hypothesis becomes

$$H_0 : \mathcal{S}(x_{\text{target}}, \mathcal{A}(D_{\text{train}})) \sim p_{\text{out},x} \quad \text{vs.} \quad H_1 : \mathcal{S}(x_{\text{target}}, \mathcal{A}(D_{\text{train}})) \sim p_{\text{in},x}. \tag{5}$$

In other words, we are studying the distinguishability of x_{target} from the learning outcome over the stochasticity arising from any inherent randomness of $\mathcal{A}$ and from the latent training dataset $D \sim \mathcal{P}^n$ that may or may not include x_{target}.

The LiRA algorithm by Carlini et al. (2022) has proved to be a highly effective. From the trained shadow models, LiRA infers normal approximations for the in/out-distributions for each target point separately. The LiRA score for a target point x and model $\mathcal{M}_{\text{target}}$ is given as the log-likelihood ratio (LLR)

$$\mathcal{L}(x, \mathcal{M}_{\text{target}}) := \log \frac{p_{\text{in},x}(\mathcal{S}_{\text{target}})}{p_{\text{out},x}(\mathcal{S}_{\text{target}})} = \log \frac{\mathcal{N}(\mathcal{S}_{\text{target}};\, \mu_{\text{in},x}, \sigma^2_{\text{in},x})}{\mathcal{N}(\mathcal{S}_{\text{target}};\, \mu_{\text{out},x}, \sigma^2_{\text{out},x})}, \tag{6}$$

where the parameters $\mu_{\text{in},x}, \mu_{\text{out},x}, \sigma_{\text{in},x}, \sigma_{\text{out},x}$ are learned from the shadow models and $\mathcal{S}_{\text{target}} := \mathcal{S}(x, \mathcal{M}_{\text{target}})$. Carlini et al. (2022) also proposed a computationally cheaper alternative to LiRA called *offline* LiRA that does not require learning the IN distribution. In offline LiRA, the membership of the observation $\mathcal{S}_{\text{target}}$ is classified based on the following statistic

$$\Lambda_{\text{off}}(\mathcal{S}_{\text{target}}) = \Pr(S_x^{\text{out}} \leq \mathcal{S}_{\text{target}}) = \Phi\left(\frac{\mathcal{S}_{\text{target}} - \mu_{\text{out},x}}{\sigma_{\text{out},x}}\right), \tag{7}$$

where Φ denotes the CDF of the standard normal distribution.

2.2 MEASURING VULNERABILITY

Since MIA is a binary hypothesis test, the vulnerability can be measured through the properties of the distinguisher performing this test. Carlini et al. (2022) suggested measuring the vulnerability in terms of the attack's TPR at low fixed FPR. This metric has been since widely adopted in the literature (see, e.g. Zarifzadeh et al., 2024; Liu et al., 2022). Empirically, the FPR and TPR are commonly obtained from a ROC-curve built from the MIA classifiers' scores and corresponding ground truth labels.

Trade-off function. The FPR is an empirical estimate of a classifier's type I error, while the $\mathrm{TPR} = 1 - \mathrm{FNR}$ with FNR being the estimate of the type II error. *Trade-off function* is an important tool that allows to understand these two errors for arbitrary classifier. It gives a lower bound for any classifiers $(\mathrm{FPR}, \mathrm{FNR})$ curve, i.e. the inverse ROC-curve, and is defined as follows

Definition 2.1 (Dong et al. (2022))**.** For any distributions P and Q on a measurable space $(\mathcal{X}, \mathcal{F})$, the trade-off function $T(P, Q) : [0, 1] \to [0, 1]$ is given as

$$T(P, Q)(\alpha) = \inf_{\phi} \{\beta_\phi \,:\, \alpha_\phi \leq \alpha\} \tag{8}$$

where $\phi : \mathcal{X} \to [0, 1]$ is a rejection rule with

$$\alpha_\phi = \mathbb{E}_P[\phi],\ \beta_\phi = 1 - \mathbb{E}_Q[\phi]. \tag{9}$$

The next Lemma shows that the trade-off function is invariant to bijective post-processing.

Lemma 2.1. *Let P and Q be distributions on $\mathbb{R}$, and let $f(x)$ be a bijective function. Denote by P^f and Q^f the pushforward distributions of P and Q under f, i.e., $P^f(B) = P(f^{-1}(B))$ for all Borel sets $B \subseteq \mathbb{R}$, and similarly for Q^f. Then, for all $\alpha \in [0, 1]$,*

$$T(P, Q)(\alpha) \;=\; T\big(P^f, Q^f\big)(\alpha).$$

The proof for this Lemma can be found in Section A.1 of the Appendix.

Measuring vulnerability using a parametric model Recently Knolle et al. (2026) proposed using the area under the curve (AUC) of the MI classifier as the vulnerability measure. They showed that if the in/out distributions are Gaussians as $\mathcal{N}(\mu_{\text{in}}, \sigma^2_{\text{in}})$ and $\mathcal{N}(\mu_{\text{out}}, \sigma^2_{\text{out}})$, then a threshold classifier applied directly on the in/out samples will have an AUC of

$$\mathrm{AUC} = \Phi\left(\frac{\mu_{\text{in}} - \mu_{\text{out}}}{\sqrt{\sigma^2_{\text{in}} + \sigma^2_{\text{out}}}}\right). \tag{10}$$

In Section A.1 of the Appendix we show similar analytical forms for the LLR test and the ROC curve. Note that while the analytical vulnerability estimates such as the AUC above are convenient to compute based on the in/out statistics, they can be biased due to the parametric model used for the shadow distributions.

3 Finite Population Bias in MIA

When implementing a MIA in practice on a real dataset, we do not have access to the underlying data generating process $\mathcal{P}$ but only a finite superset $D_{\text{full}} = \{x_1, \ldots, x_{N_+}\} \sim \mathcal{P}^{N^+}$ from which the training sets of each model must be sampled. For each target model $\mathcal{M}$, the corresponding training set D of N elements is sampled without replacement from D_{full}. This approach comes with two important differences compared to the "ideal" LiRA setting where shadow datasets are independently drawn as $D \sim \mathcal{P}^N$:

1. All shadow and target models are trained on overlapping subsets of the *same* finite D_{full}.
2. Each D is sampled without replacement from D_{full}, rather than from an infinite population.

As a consequence, estimators of the in/out variances of the learning outcomes (or LiRA scores) are subject to a finite population bias. Intuitively, reusing the same finite set D_{full} multiple times makes the empirical variability of the shadow models smaller than what one would observe if each training set were drawn independently from $\mathcal{P}$.

3.1 Finite Population Correction

To see the effect at a high level, consider any scalar statistic $T(D)$ of a dataset D of size N drawn *without replacement* from a finite population of size N_+. Classical sampling theory (Cochran, 1977) shows that

$$\mathrm{Var}[T(D)] \approx \mathrm{Var}_{\text{iid}}[T(D)] \cdot \Big(1 - \frac{N}{N_+}\Big), \tag{11}$$

where $\mathrm{Var}_{\mathrm{iid}}[T(D)]$ is the variance that would be obtained if the elements of D were sampled independently from the underlying population (i.e., with replacement). The factor $\mathrm{FPC} = 1 - {}^{N}/_{N_+}$ is the standard finite population correction. When N/N_+ is small, the correction is negligible, but for larger N the effect becomes significant. When training shadow models for MIA, it is common to choose N to be a large fraction of N_+ (e.g. $N = 0.5\,N_+$ in the Privacy meter library (Murakonda and Shokri, 2020)), in which case $\mathrm{FPC} \approx 0.5$ and the variance may be substantially underestimated.

In our setting, the statistic is the learning outcome (or its LiRA score) for a fixed point x across shadow models. We estimate the in/out distributions for x from $\{\mathcal{M}^{(m)}\}_{m=1}^{M}$, but all $\mathcal{M}^{(m)}$ are trained on overlapping subsets of a fixed finite D_{full}. Therefore the empirical in/out variances estimated from the shadow models are biased towards *smaller* values compared to the variances one would obtain from independent draws $D \sim \mathcal{P}^N$.

Since LiRA's distinguishability is driven in large part by the separation between the in and out distributions relative to their variances, underestimating these variances can lead to inflated estimates of membership inference vulnerability. The effect is easy to see from Eq. (10). Consider both the estimated in and out stds $\hat{\sigma}_{\text{in}}$, $\hat{\sigma}_{\text{out}}$ are equally effected by the bias, i.e. $\hat{\sigma}_{\text{in}} = \sqrt{f}\sigma_{\text{in}}$ and $\hat{\sigma}_{\text{out}} = \sqrt{f}\sigma_{\text{out}}$ where $f = \mathrm{FPC} \leq 1$. We get that

$$\widehat{\mathrm{AUC}} = \Phi\left(\frac{1}{\sqrt{f}}\frac{\mu_{\text{in}} - \mu_{\text{out}}}{\sqrt{\sigma_{\text{in}}^2 + \sigma_{\text{out}}^2}}\right) > \Phi\left(\frac{\mu_{\text{in}} - \mu_{\text{out}}}{\sqrt{\sigma_{\text{in}}^2 + \sigma_{\text{out}}^2}}\right) = \mathrm{AUC}. \tag{12}$$

The ROC-curve-based vulnerability evaluations are also affected by the finite population bias. However, in addition to the FPC fixes to the IN/OUT distributions that correct the LiRA score computation, we also need to correct the bias on the sampled target models that we use to compute the ROC curve. A simple way to correct the test set is to rescale each observed target model by the FPC factor within its true class as

$$g(\mathcal{S}, y) = \mu_y + \frac{\mathcal{S} - \mu_y}{\sqrt{f}}, \tag{13}$$

where the $y \in \{\text{in}, \text{out}\}$. This maps the test points to the population distribution. Note that the map depends on knowing the label y, and therefore cannot be considered as part of an actual attack. However, we are simply using the transformation in the evaluation stage to remove the bias from the test points, not to implement an actual attack.

3.2 Variance Decomposition

Above we demonstrated the bias on vulnerability estimation by assuming the variance is affected as $\sigma_{\mathrm{obs}}^2 = f\sigma_{\mathrm{pop}}^2$ which we will call *Simple* FPC. However, there might be population-independent sources of randomness in the in/out distributions arising e.g. from the learning algorithm. These sources of variation are not subject to the finite population correction, and therefore correcting the observed total variance by dividing with f might over amplify some parts of the variance.

In order to separate the population-independent sources of randomness, we will use the following simple variance decomposition for both the in and out distributions

$$\sigma_{\mathrm{obs}}^2(f) = f\sigma_{\mathrm{pop}}^2 + \sigma_{\mathrm{rem}}^2, \tag{14}$$

where σ_{pop}^2 is the component affected by finite-population sampling and σ_{rem}^2 plays the role of the other variability. We will call this variance decomposition based correction the *Decomposed* FPC.

Estimating the decomposition We can estimate the σ_{pop} and σ_{rem} by computing the observed variance arising from multiple different ratios N/N_+, leading to different FPC factors f, and fitting the observations to the line. Note that computing each observed variance requires multiple shadow models to be trained. Furthermore with small N/N_+, the probability of x_{target} belonging in the shadow dataset is small, and therefore we need a lot of repeats to estimate the IN variance. Conversely, with large N/N_+ we need a lot of repeats to estimate the OUT variance for each x_{target}. In Algorithm 1, we show how to estimate the corrected variance from 2 sampling ratios (r_a, r_b) and use them to correct the observed IN/OUT variances. In Figures A4 and A5, we compare two-ratio decompositions against all-ratio estimates and select the pair minimizing mean parameter RMSE across 3 repeats.

4 REPORTING AGGREGATES OF THE VULNERABILITY

Recall that our goal is to estimate the vulnerability of the samples x_{target}. In order to empirically assess this vulnerability, we learn multiple shadow models trained with or without x_{target} and build the ROC-curve with the corresponding scores and labels. The in/out scores can be written as

$$\ell^{(\text{out})}_{x_{\text{target}}} := \mathcal{L}(x_{\text{target}}, \mathcal{A}(D)), \quad \ell^{(\text{in})}_{x_{\text{target}}} := \mathcal{L}(x_{\text{target}}, \mathcal{A}(D \cup x_{\text{target}})), \tag{15}$$

where $D \sim \mathcal{P}^n$. The FPR and TPR for a threshold classifier with a decision threshold $\tau \in \mathbb{R}$ become

$$\text{FPR}_x(\tau) = \Pr(\ell^{(\text{out})}_x > \tau) = \mathbb{E}_{D,\xi}[\mathbf{1}[\ell^{(\text{out})}_x > \tau]] \tag{16}$$

$$\text{TPR}_x(\tau) = \Pr(\ell^{(\text{in})}_x > \tau) = \mathbb{E}_{D,\xi}[\mathbf{1}[\ell^{(\text{in})}_x > \tau]] \tag{17}$$

where the $D \sim P^n$ and ξ denotes any additional stochasticity in $\mathcal{A}$.

4.1 CONCATENATING THE MIA SCORES

Consider now that we have MIA scores evaluated over M target models and N target data points. Each of the N samples x_{target} was either in our out of the training data for each of the M models. Since reliably estimating per-sample TPR at low FPR requires many target models, practical evaluations often aggregate scores over samples and models.

Concatenating MIA scores over multiple target models has been used in the past to learn aggregates of the vulnerability (Carlini et al., 2022; Murakonda and Shokri, 2020). In concatenated MIA evaluation, we are merging the MIA scores computed for the N individuals over M target models into a single vector. The vulnerability is then evaluated based on a ROC-curve built on these raveled scores and their corresponding labels.

The FPR and TPR for the concatenated approach are given as

$$\text{FPR} = \Pr(\ell^{(\text{out})} > \tau), \quad \text{TPR} = \Pr(\ell^{(\text{in})} > \tau), \tag{18}$$

where the dependence of both the model and sample to the scores is marginalized away. The next proposition demonstrates how the concatenated approach is related to the per-sample vulnerability.

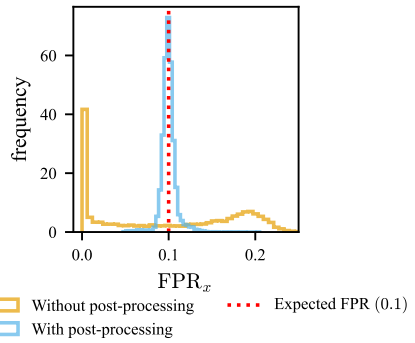


Figure 1: **Per-sample FPR distributions under concatenation-based MIA evaluation.** Naive concatenation yields heterogeneous FPRs, while post-processing calibrates samplewise OUT distributions before aggregation, reducing FPR variation. Results are shown for TabPFN models trained on Adult dataset.

Proposition 4.1. *For any fixed threshold τ, the concatenated FPR and TPR are equal to the expectation of per-sample FPR$_x$ and TPR$_x$:*

$$\text{FPR}(\tau) = \mathbb{E}_x[\text{FPR}_x(\tau)], \qquad \text{TPR}(\tau) = \mathbb{E}_x[\text{TPR}_x(\tau)].$$

Proof can be found in Section A.1 of the Appendix. Thus, choosing a single decision threshold τ that yields $\text{FPR} = \alpha$ at the concatenated level corresponds to choosing τ such that the average per-sample FPR_x equals α, while individual FPR_x may vary widely. Hence calibrating the concatenated FPR does not necessarily serve as a good proxy of understanding the vulnerability.

4.2 POST-PROCESSING FOR BETTER AGGREGATION

We start by showing that the offline LiRA attack, originally proposed by Carlini et al. (2022) as a computationally more feasible version of LiRA, avoids the FPR calibration problem. For a fixed sample x, suppose the learning outcome $\mathcal{S}(x, A(D))$ satisfies

$$S^{\text{out}}_x \sim \mathcal{N}(\mu_{\text{out},x}, \sigma^2_{\text{out},x}), \qquad S^{\text{in}}_x \sim \mathcal{N}(\mu_{\text{in},x}, \sigma^2_{\text{in},x}), \tag{19}$$

and define the mean gap $\Delta_x = \mu_{\text{in},x} - \mu_{\text{out},x}$. The transformation Λ_{off} in Eq. (7) is a bijection since it is a composition of the bijection Φ and an affine mapping $f_x(t) = (t - \mu_{\text{out},x})/\sigma_{\text{out},x}$. Therefore Lemma A.2 tells us that

$$T\big(S_x^{\text{out}}, S_x^{\text{in}}\big) = T\big(\Lambda_{\text{off}}(S_x^{\text{out}}), \Lambda_{\text{off}}(S_x^{\text{in}})\big) = T\left(\mathcal{N}\left(\frac{\Delta_x}{\sigma_{\text{out},x}}, \frac{\sigma_{\text{in},x}^2}{\sigma_{\text{out},x}^2}\right), \mathcal{N}(0,1)\right). \tag{20}$$

We can further make this transformation independent of the ordering of in/out distributions by multiplying with $\text{sign}(\Delta_x)$

$$f_x(t) = \frac{\text{sign}(\Delta_x)(t - \mu_{\text{out},x})}{\sigma_{\text{out},x}}. \tag{21}$$

Again by Lemma A.2, we get

$$T\big(\mathcal{N}(\mu_{\text{in},x}, \sigma_{\text{in},x}^2), \mathcal{N}(\mu_{\text{out},x}, \sigma_{\text{out},x}^2)\big) = T\left(\mathcal{N}\left(\frac{|\Delta_x|}{\sigma_{\text{out},x}}, \frac{\sigma_{\text{in},x}^2}{\sigma_{\text{out},x}^2}\right), \mathcal{N}(0,1)\right). \tag{22}$$

We can see that both of the transformations normalize all the out scores into $N(0,1)$. Therefore no matter which decision rule we apply to the out scores, we will get the same FPR across all the samples and as a result the average gets calibrated. Figure 1 shows this in practice: naive concatenation leads to large spread in FPR values while post-processing concentrates them on the target FPR.

When $\sigma_{\text{in},x} = \sigma_{\text{out},x}$, we can see from (22) that all per-sample in/out pairs can be viewed as testing between a standard normal and a shifted unit-variance normal with mean $|\Delta_x|/\sigma_x$. In this case the LLR is monotonic, and therefore we can perform a Neyman-Pearson optimal test directly on the post-processed samples. When $\sigma_{\text{in},x} \neq \sigma_{\text{out},x}$, the LLR is no longer a bijection and a threshold test on the post-processed samples is no longer Neyman-Pearson optimal. However the FPR will still remain calibrated.

Finally, we can estimate the average TPR as

$$\overline{\text{TPR}}(\tau) = \frac{1}{NM} \sum_{\substack{i=1\ldots N \\ m=1\ldots M}} \mathbf{1}[\hat{\mathcal{S}}_{i,j}^{\text{in}} > \tau], \tag{23}$$

where $\hat{\mathcal{S}}^{\text{in}}$ denotes the post-processed in samples. We want to emphasize that $\overline{\text{TPR}}$ is not suitable for assessing the worst-case per-sample nor the per-model vulnerability, but should be seen as a measure of how much the learning algorithm leaks about the individuals on average. However, as we are averaging over NM values in total, we can actually have a rather confident estimate of the underlying average per-sample vulnerability, even in the low FPR regime.

5 Results

Next, we test how our theoretical findings and the proposed fixes work in practice. We experiment on both tabular Adult (Becker and Kohavi, 1996) dataset and computer vision (CIFAR10 (Krizhevsky, 2009) and PatchCamelyon (Veeling et al., 2018)) datasets. For the tabular data we use the TabPFN as the target model architecture and for image classification we fine-tune ResNet50 (Kolesnikov et al., 2020) and ViT-B-16 (Dosovitskiy et al., 2021) models. A detailed experimental setup can be found in the Appendix (Section A.3).

5.1 Finite Population Correction

We evaluate how sampling training sets from a finite data pool affects MIA vulnerability evaluation. In our experiments, we fix the training set size to N while varying the size of the finite superset $N_+(= |D_{\text{full}}|)$. This varies the sampling fraction N/N_+ without changing the number of samples used to train each model. For the population-level MIA estimate, we train M shadow models for each sample using the leave-one-out (LOO) approach. In the LOO setting, for each target x, each of the IN models contain x together with $N-1$ samples chosen without replacement from the population dataset, D_{pop}. For the OUT models, we replace the x with another sample from D_{pop}.

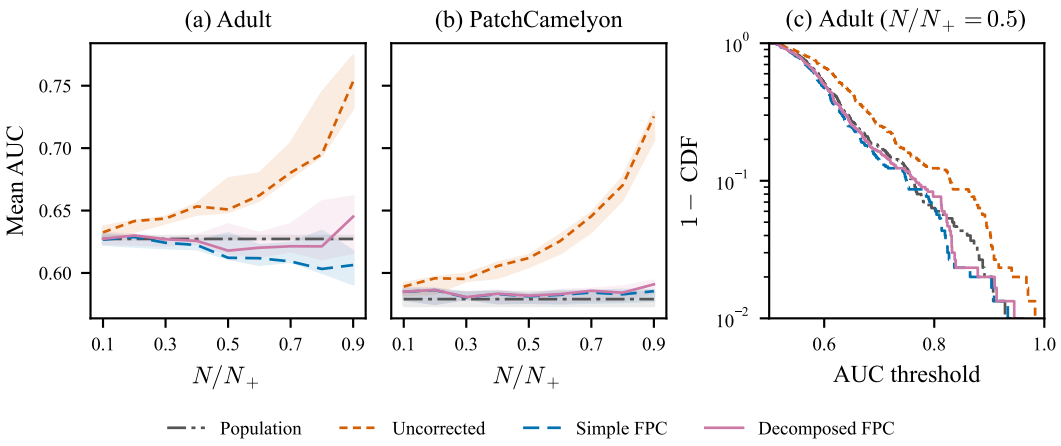


Figure 2: **Finite population bias in MIA.** (a–b) Median of per-seed mean AUCs across three repeats, with minimum–maximum ranges, for PatchCamelyon + ViT-B-16 and Adult + TabPFN. (c) Empirical survival functions of per-target AUCs pooled across 3 repeats for Adult at $N/N_+ = 0.5$. Uncorrected estimates increasingly overstate mean AUC as the N/N_+ grows. Both corrections reduce this discrepancy; on Adult, simple FPC tends to underestimate mean AUC at larger fractions, while decomposed FPC, using variance decomposition with two ratios, (r_a, r_b), generally remains closer to the population reference. Survival function plots for additional N/N_+ are in Figures A1 and A2

In Figure 2, we evaluate MIA AUC for 100 target samples for (a) TabPFN models fitted to the Adult dataset using 1000 examples, (b) ViT-B-16 models fine-tuned on PatchCamelyon dataset using 500 examples/class. We observe that as $N/N+$ increases, the uncorrected mean AUC consistently overstates MIA vulnerability w.r.t. the population-reference estimate. For the image classification task on PatchCamelyon, we observe that the FPC (with or without variance decomposition) helps recover the population-level MIA vulnerability. On the Adult dataset, Simple FPC tends to underestimate mean AUC at larger fractions, whereas decomposed FPC generally stays closer to the population reference.

We also test the validity of our variance decomposition of Section 3.2. Figure 3 shows the linear fits for the IN variances of 20 samples randomly sampled from the pool of 100 evaluation samples from Adult and PatchCamelyon datasets, as well as the R^2 of the fits across the 100 targeted samples and $N/N_+ \in \{0.1, \ldots, 0.9\}$. We see that the linear fits capture large proportion of the observed variance for a large proportion of the samples. Fits for OUT variance decomposition are in Figure A3.

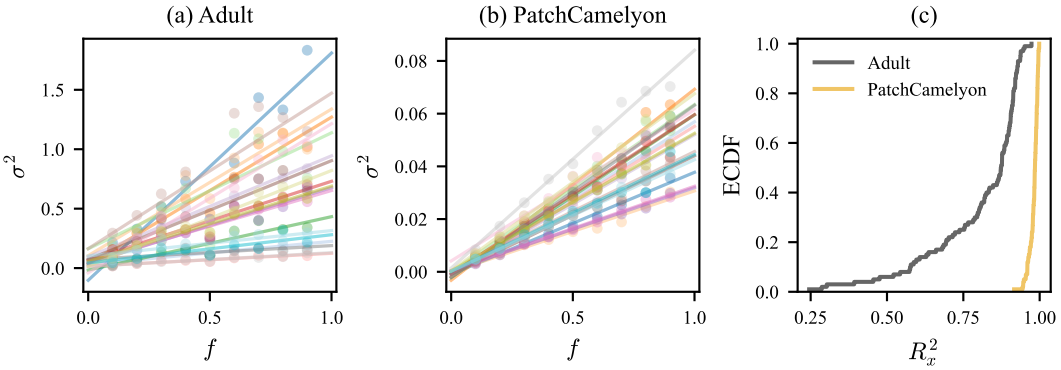


Figure 3: **Variance decomposition for Adult and PatchCamelyon datasets for IN distribution variances** (a-b) Observed variances and OLS fits $\sigma_x^2(f) = f\sigma_{\text{pop},x}^2 + \sigma_{\text{rem},x}^2$ for 20 randomly selected target samples as computed per Section 3.2 for Adult and PatchCamelyon datasets.(c) Empirical CDF of per-target R_x^2 across all 100 evaluation targets using observed variances for all ratios $N/N_+ \in \{0.1, \ldots, 0.9\}$ for both the datasets. We train 2048 models per N/N_+, with $N = 1000$, and use the collected statistics to estimate each target's IN variance.

5.2 Effect of Post-Processing On MIA Evaluation

Next, we will focus on the average per-sample vulnerability estimation and the post-processing strategy of 4.2. Figure 4 compares three strategies: (1) the **Average TPR/Sample** which computes the actual average per-sample using $M = 4096$ shadow models, (2) the **Concatenated (Naive)** which concatenates the plain LiRA scores for the ROC computation and (3) **Concatenated (PP)** where we apply the post-processing of Eq. (21) to the logits. We observe that across all settings, the naive concatenation overestimates the average vulnerability whereas the post-processed TPR closely follows the average per-sample vulnerability even with a moderate ($M = 64$) number of shadow models. These results demonstrate that the post-processed approach can be a computationally economical proxy for understanding the average vulnerability. After applying the FPC to remove the finite population bias from the experiments, the same trends persist, but in general with a reduced vulnerability.

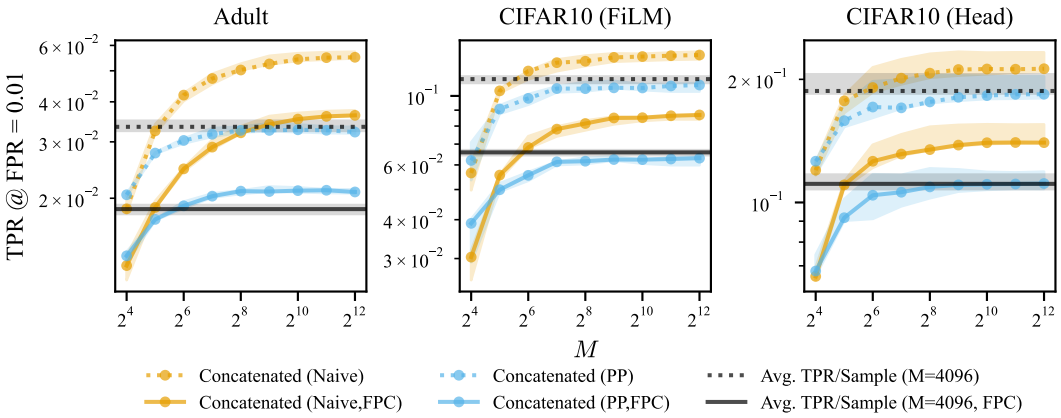


Figure 4: **Effect of finite population correction and post-processing on MIA performance.** Assessing TPR at FPR $= 0.01$ against the number of reference models M, for Adult + TabPFN and CIFAR10 + FiLM/Head experimental settings. Plots show median TPR over 3 seeds, with minimum–maximum ranges; We use average per-sample FPR at $M = 4096$ as our baseline. As M grows, naive concatenation exceeds the corresponding per-sample reference, while post-processing (Section 4.2) brings estimates closer. FPC lowers the estimated TPR with/without post-processing.

6 Conclusion

Our results show how to evaluate both the per-sample as well as the average MIA vulnerability more reliably. As pointed out by Aerni et al. (2024), vulnerability evaluation should focus on the most vulnerable data points, which can be watered down in the average. This means that the average TPR values need to be interpreted with care. However with TPR at low FPR as a metric for the vulnerability, the search for the most vulnerable sample would become computationally infeasible, as reliable estimation would require a large number of shadow models. As an alternative one could consider metrics such as the parametric per-example AUC suggested by Knolle et al. (2026). While the AUC might not be able to separate low probability privacy leaks, it can capture the overall worst-case individual privacy risk well which still is a very important privacy measure. However, while the parametric AUC estimation is likely more stable than the TPR at low FPR, its reliability is still affected by the number of shadow models. Studying this is an interesting direction for future work.

Recommendations

- For any shadow-model-based MIA vulnerability analysis, apply at least simple FPC
- When computing average vulnerability by concatenation, apply post-processing to the concatenated scores
- If compute allows, repeat experiment with multiple N/N_+ for decomposed FPC

Reproducibility statement

- Source code for our experiments is available at `https://anonymous.4open.science/r/mi_vulnerbility_evaluation`.
- we elaborate on the proofs missing from the main paper in Section A.1,
- We share the details of the experimental setup in Section A.3. In Table A1, we list the hyperparameters used in our experiments.

AI Usage Disclosure

In this work, we used generative AI tools (OpenAI's ChatGPT) to,

- improve readability of the manuscript,
- format the mathematical content for LaTeX, and
- create and modify scientific figures.

We did not use generative AI tools for any tasks with required disclosure, including generating synthetic datasets, developing theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, proposing or refining hypotheses, designing or providing feedback on research methodology or experiments, implementing methods, assisting with translation, cleaning and reformatting datasets, supporting qualitative and thematic data analysis, or interpreting results. We have reviewed all AI-assisted work and verified the generated text, code, and figures for correctness and consistency with the underlying results. We take responsibility for the final content of this work, including text, claims, code, and artifacts produced with the aid of generative AI.

References

M. Aerni, J. Zhang, and F. Tramèr. Evaluations of machine learning privacy defenses are misleading. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security, (CCS 2024)*, pages 1271–1284, 2024.

M. S. M. S. Annamalai, B. Balle, J. Hayes, G. Kaissis, and E. D. Cristofaro. The hitchhiker's guide to efficient, end-to-end, and tight DP auditing. In *The Fourth IEEE Conference on Secure and Trustworthy Machine Learning (SaTML 2026)*, 2026.

B. Becker and R. Kohavi. Adult. UCI Machine Learning Repository, 1996. DOI: https://doi.org/10.24432/C5XW20.

M. Boglioni, T. Liu, A. Ilyas, and S. Wu. Optimizing canaries for privacy auditing with metagradient descent. In *The Fourteenth International Conference on Learning Representations (ICLR 2026)*, 2026.

H. Buss. Membership inference attacks with uncertainty quantification. Master's thesis, University of Helsinki, May 2026.

N. Carlini, S. Chien, M. Nasr, S. Song, A. Terzis, and F. Tramèr. Membership inference attacks from first principles. In *43rd IEEE Symposium on Security and Privacy, SP*, pages 1897–1914, 2022.

W. G. Cochran. Sampling techniques. *Johan Wiley & Sons Inc*, 1977.

J. Dong, A. Roth, and W. J. Su. Gaussian differential privacy. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 84(1):3–37, 2022. doi: 10.1111/rssb.12454.

A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *9th International Conference on Learning Representations, ICLR*, 2021.

C. Dwork, F. McSherry, K. Nissim, and A. D. Smith. Calibrating Noise to Sensitivity in Private Data Analysis. In *Theory of Cryptography, Third Theory of Cryptography Conference, TCC, Proceedings*, volume 3876 of *Lecture Notes in Computer Science*, pages 265–284, 2006.

S. Garg, D. Tsipras, P. Liang, and G. Valiant. What can transformers learn in-context? A case study of simple function classes. In *Advances in Neural Information Processing Systems 35 (NeurIPS 2022)*, 2022.

N. Hollmann, S. Müller, L. Purucker, A. Krishnakumar, M. Körfer, S. B. Hoo, R. T. Schirrmeister, and F. Hutter. Accurate predictions on small data with a tabular foundation model. *Nature*, 637 (8044):319–326, 2025. doi: 10.1038/S41586-024-08328-6. URL https://doi.org/10.1038/s41586-024-08328-6.

A. Keinan, M. Shenfeld, and K. Ligett. How well can differential privacy be audited in one run? In *Advances in Neural Information Processing Systems 38 (NeurIPS 2025)*, 2025.

M. A. Knolle, M. J. Menten, F. Jungmann, F. Meissen, B. Glocker, D. Rueckert, and G. Kaissis. Disparate privacy risks from medical AI. *Nature*, 656(8126):192—198, August 2026. ISSN 0028-0836. doi: 10.1038/s41586-026-10688-0. URL https://doi.org/10.1038/s41586-026-10688-0.

A. Kolesnikov, L. Beyer, X. Zhai, J. Puigcerver, J. Yung, S. Gelly, and N. Houlsby. Big Transfer (BiT): General Visual Representation Learning. In *Computer Vision - ECCV 2020 - 16th European Conference*, volume 12350 of *Lecture Notes in Computer Science*, pages 491–507. Springer, 2020. doi: 10.1007/978-3-030-58558-7_29.

A. Krizhevsky. Learning Multiple Layers of Features From Tiny Images. Master's thesis, University of Toronto, 2009.

Y. Liu, Z. Zhao, M. Backes, and Y. Zhang. Membership Inference Attacks by Exploiting Loss Trajectory. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security, CCS*, pages 2085–2098, 2022.

S. Mahloujifar, L. Melis, and K. Chaudhuri. Auditing f-Differential Privacy in One Run. In *Forty-second International Conference on Machine Learning ICML*, 2025.

S. K. Murakonda and R. Shokri. ML privacy meter: Aiding regulatory compliance by quantifying the privacy risks of machine learning. 2020. arXiv:2007.09339.

E. Perez, F. Strub, H. de Vries, V. Dumoulin, and A. C. Courville. FiLM: Visual Reasoning with a General Conditioning Layer. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18)*, pages 3942–3951. AAAI Press, 2018. doi: 10.1609/AAAI.V32I1.11671.

A. Salem, G. Cherubin, D. Evans, B. Köpf, A. Paverd, A. Suri, S. Tople, and S. Zanella-Béguelin. SoK: Let the Privacy Games Begin! A Unified Treatment of Data Inference Privacy in Machine Learning. In *44th IEEE Symposium on Security and Privacy, SP*, pages 327–345, 2023.

R. Shokri, M. Stronati, C. Song, and V. Shmatikov. Membership Inference Attacks Against Machine Learning Models. In *2017 IEEE Symposium on Security and Privacy, SP*, pages 3–18, 2017.

T. Steinke, M. Nasr, and M. Jagielski. Privacy Auditing with One (1) Training Run. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS*, 2023.

M. Tobaben, H. Ito, J. Jälkö, Y. He, and A. Honkela. Impact of Dataset Properties on Membership Inference Vulnerability of Deep Transfer Learning. In *Advances in Neural Information Processing Systems 38 (NeurIPS)*, 2025.

B. S. Veeling, J. Linmans, J. Winkens, T. Cohen, and M. Welling. Rotation Equivariant CNNs for Digital Pathology. In A. F. Frangi, J. A. Schnabel, C. Davatzikos, C. Alberola-López, and G. Fichtinger, editors, *Medical Image Computing and Computer Assisted Intervention - MICCAI 2018 - 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part II*, volume 11071 of *Lecture Notes in Computer Science*, pages 210–218. Springer, 2018. doi: 10.1007/978-3-030-00934-2_24. URL https://doi.org/10.1007/978-3-030-00934-2_24.

L. Wasserman and S. Zhou. A statistical framework for differential privacy. *Journal of the American Statistical Association*, 105(489):375–389, 2010. doi: 10.1198/jasa.2009.tm08651.

Z. Xiang, T. Wang, and D. Wang. Privacy audit as bits transmission: (im)possibilities for audit by one run. In *Proceedings of the 34th USENIX Conference on Security Symposium*, 2025.

J. Ye, A. Maddi, S. K. Murakonda, V. Bindschaedler, and R. Shokri. Enhanced Membership Inference Attacks against Machine Learning Models. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security, CCS*, pages 3093–3106, 2022.

S. Zarifzadeh, P. Liu, and R. Shokri. Low-cost High-power Membership Inference Attacks. In *Forty-first International Conference on Machine Learning, ICML 2024*, 2024.

A Appendix

A.1 Missing proofs

Proposition 4.1. *For any fixed threshold τ, the concatenated FPR and TPR are equal to the expectation of per-sample FPR$_x$ and TPR$_x$:*

$$\mathrm{FPR}(\tau) = \mathbb{E}_x[\mathrm{FPR}_x(\tau)], \qquad \mathrm{TPR}(\tau) = \mathbb{E}_x[\mathrm{TPR}_x(\tau)].$$

Proof. For the concatenated FPR we have

$$\mathrm{FPR} = \Pr(\ell^{(\text{out})} > \tau) = \mathbb{E}_{x,D,\xi}[\mathbf{1}[\ell_x^{(\text{out})} > \tau]] = \mathbb{E}_x[\mathbb{E}_{D,\xi}[\mathbf{1}[\ell_x^{(\text{out})} > \tau]]] = \mathbb{E}_x[\mathrm{FPR}_x], \tag{24}$$

which similarly follows for the TPR. □

Lemma A.1 (Dong et al., 2022, Lemma 1)**.** *For any distributions P and Q and a function f, for any $\alpha \in [0,1]$ we have*

$$T(P^f, Q^f)(\alpha) \geq T(P,Q)(\alpha). \tag{25}$$

Lemma A.2. *Let P and Q be distributions on $\mathbb{R}$, and let $f(x)$ be a bijective function. Denote by P^f and Q^f the pushforward distributions of P and Q under f, i.e., $P^f(B) = P(f^{-1}(B))$ for all Borel sets $B \subseteq \mathbb{R}$, and similarly for Q^f. Then, for all $\alpha \in [0,1]$,*

$$T(P,Q)(\alpha) \;=\; T\big(P^f, Q^f\big)(\alpha).$$

Proof. According to Lemma A.1, for any functions f and g, for any $\alpha \in [0,1]$ we have

$$T(P,Q)(\alpha) \leq T(P^f, Q^f)(\alpha) \leq T(P^{g\circ f}, Q^{g\circ f})(\alpha), \tag{26}$$

where $(g \circ f)(x) = g(f(x))$. Now, if f is a bijection it is invertible and we can chooce $g = f^{-1}$, which yields

$$T(P,Q)(\alpha) \leq T(P^f, Q^f)(\alpha) \leq T(P^{f^{-1}\circ f}, Q^{f^{-1}\circ f})(\alpha) = T(P,Q)(\alpha) \tag{27}$$

$$\Rightarrow \quad T(P,Q)(\alpha) = T(P^f, Q^f)(\alpha). \tag{28}$$

□

Analytical Estimation of TPR at Select FPR Using LLRs Following Buss (2026),

$$\mathrm{TPR}_{\mathrm{LLR}}(\alpha) = \begin{cases} 1 - F_{\mathrm{in}}\left(\dfrac{\sigma_{\mathrm{out}}^2}{\sigma_{\mathrm{in}}^2} F_{\mathrm{out}}^{-1}(1-\alpha)\right), & \sigma_{\mathrm{in}}^2 > \sigma_{\mathrm{out}}^2, \\ F_{\mathrm{in}}\left(\dfrac{\sigma_{\mathrm{out}}^2}{\sigma_{\mathrm{in}}^2} F_{\mathrm{out}}^{-1}(\alpha)\right), & \sigma_{\mathrm{in}}^2 < \sigma_{\mathrm{out}}^2, \\ \Phi\left(\Phi^{-1}(\alpha) + \dfrac{|\mu_{\mathrm{in}} - \mu_{\mathrm{out}}|}{\sigma_{\mathrm{in}}}\right), & \sigma_{\mathrm{in}}^2 = \sigma_{\mathrm{out}}^2. \end{cases}$$

Here Φ is the standard normal CDF and, for unequal variances, F_j is the CDF of $\chi_1^2((\mu_j - m)^2/\sigma_j^2)$, where

$$m = \frac{\mu_{\mathrm{out}}\sigma_{\mathrm{in}}^2 - \mu_{\mathrm{in}}\sigma_{\mathrm{out}}^2}{\sigma_{\mathrm{in}}^2 - \sigma_{\mathrm{out}}^2}, \qquad j \in \{\mathrm{in}, \mathrm{out}\}.$$

Analytical Estimation of AUC Using LLRs Let $X \sim \mathcal{N}(\mu_{\text{in}}, \sigma_{\text{in}}^2)$ and $Y \sim \mathcal{N}(\mu_{\text{out}}, \sigma_{\text{out}}^2)$ be independent, with positive variances. Then,

$$\ell(x) = \log \frac{p_{\text{in}}(x)}{p_{\text{out}}(x)}.$$

$$\text{AUC}_{\text{LLR}} = \Pr(\ell(X) > \ell(Y)).$$

Expanding the Gaussian log densities gives

$$\ell(X) - \ell(Y) = \frac{UV}{2\sigma_{\text{in}}^2\sigma_{\text{out}}^2},$$

where

$$U = X - Y, \qquad V = (\sigma_{\text{in}}^2 - \sigma_{\text{out}}^2)(X + Y) + 2(\mu_{\text{in}}\sigma_{\text{out}}^2 - \mu_{\text{out}}\sigma_{\text{in}}^2).$$

These variables are jointly Gaussian, with

$$\begin{aligned}
\mathbb{E}[U] &= \mu_{\text{in}} - \mu_{\text{out}}, & \text{Var}(U) &= \sigma_{\text{in}}^2 + \sigma_{\text{out}}^2, \\
\mathbb{E}[V] &= (\mu_{\text{in}} - \mu_{\text{out}})(\sigma_{\text{in}}^2 + \sigma_{\text{out}}^2), & \text{Var}(V) &= (\sigma_{\text{in}}^2 - \sigma_{\text{out}}^2)^2(\sigma_{\text{in}}^2 + \sigma_{\text{out}}^2), \\
\text{Cov}(U, V) &= (\sigma_{\text{in}}^2 - \sigma_{\text{out}}^2)^2.
\end{aligned}$$

Thus, for $\sigma_{\text{in}}^2 \neq \sigma_{\text{out}}^2$,

$$\begin{aligned}
\text{AUC}_{\text{LLR}} &= \Pr(UV > 0) \\
&= 1 - \Pr(U \leq 0) - \Pr(V \leq 0) + 2\Pr(U \leq 0, V \leq 0) \\
&= 1 - \Phi\left(-\frac{\mu_{\text{in}} - \mu_{\text{out}}}{\sqrt{\sigma_{\text{in}}^2 + \sigma_{\text{out}}^2}}\right) - \Phi\left(-\frac{(\mu_{\text{in}} - \mu_{\text{out}})\sqrt{\sigma_{\text{in}}^2 + \sigma_{\text{out}}^2}}{|\sigma_{\text{in}}^2 - \sigma_{\text{out}}^2|}\right) \\
&\quad + 2\Phi_2\left(-\frac{\mu_{\text{in}} - \mu_{\text{out}}}{\sqrt{\sigma_{\text{in}}^2 + \sigma_{\text{out}}^2}}, -\frac{(\mu_{\text{in}} - \mu_{\text{out}})\sqrt{\sigma_{\text{in}}^2 + \sigma_{\text{out}}^2}}{|\sigma_{\text{in}}^2 - \sigma_{\text{out}}^2|}; \frac{|\sigma_{\text{in}}^2 - \sigma_{\text{out}}^2|}{\sigma_{\text{in}}^2 + \sigma_{\text{out}}^2}\right),
\end{aligned}$$

where Φ and $\Phi_2(\cdot, \cdot; \rho)$ are the standard univariate and bivariate normal CDFs, respectively. Gaussianity is assumed for X, Y, not for their LLR scores.

For equal variances, the LLR becomes equivalent to AUC as formulated by (Knolle et al., 2026),

$$\text{AUC}_{\text{LLR}} = \Phi\left(\frac{|\mu_{\text{in}} - \mu_{\text{out}}|}{\sqrt{\sigma_{\text{in}}^2 + \sigma_{\text{out}}^2}}\right).$$

A.2 Additional Plots

A.3 Experimental Setup

- **Models** We use TabPFN (Hollmann et al., 2025) for tabular binary classification. For image classification experiments, we fine-tune (a) final linear layer (Head) of ViT-B-16 (Dosovitskiy et al., 2021)/ ResNet50 (Kolesnikov et al., 2020) model pretrained on ImageNet21K, (b) parameter-efficient FiLM adapters (Perez et al., 2018) across ViT-B-16 model.
- **Datasets** We use UCI's Adult (Becker and Kohavi, 1996) dataset for tabular binary classification task to predict whether the Income is $< \$50\text{K}$ or $\geq \$50\text{K}$. For the image classification experiments, we use CIFAR10 (Krizhevsky, 2009) and PatchCamelyon (Veeling et al., 2018) datasets.
- **Training Paradigm** TabPFN leverages transformer-based in-context learning (Garg et al., 2022) to predict labels for the training samples. For image classification, we adopt a few-shot transfer learning setting, using a subset of training data to optimize the classification linear layer and, FiLM parameters. Table A1 lists the hyperparameters used to train different models.

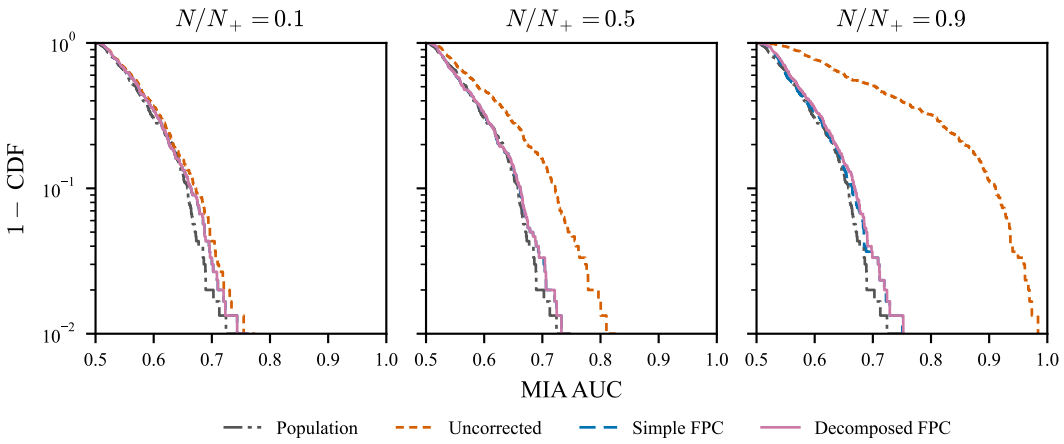


Figure A1: **Survival functions of per-sample MIA AUC demonstrating finite population bias for PatchCamelyon dataset**. We train 2048 ViT-B-16 models with final layer fine-tuned on PatchCamelyon dataset for varying N/N_+ (with $N = 1000$). Per-sample MIA AUCs are pooled across 3 seeds and shown for $N/N_+ \in \{0.1, 0.5, 0.9\}$. We observe that finite population bias increasingly inflates estimated MIA vulnerability as N/N_+ grows. FPC (Section 3.1) substantially restores the population-level MIA vulnerability estimates.

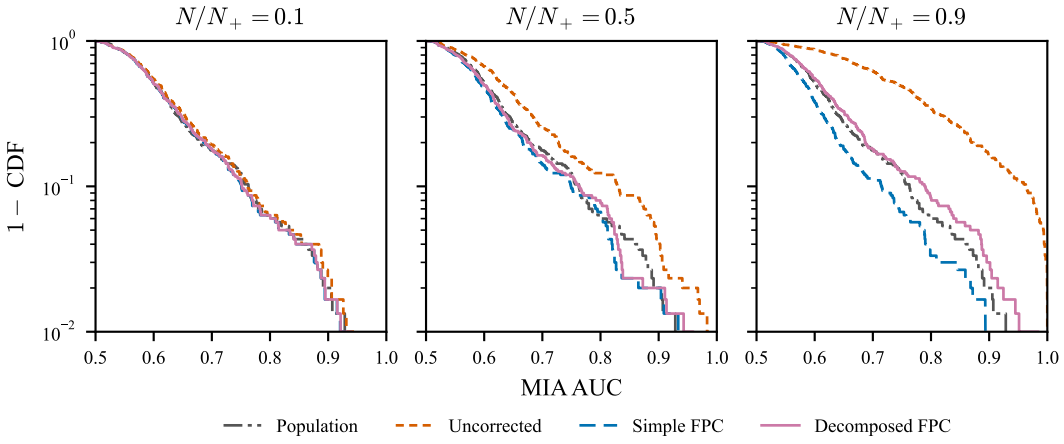


Figure A2: **Survival functions of per-sample MIA AUC demonstrating finite population bias for Adult dataset**. We train 2048 TabPFN models fine-tuned on Adult dataset for varying N/N_+ (with $N = 1000$). Per-sample MIA AUCs are pooled across 3 seeds and shown for $N/N_+ \in \{0.1, 0.5, 0.9\}$. We observe that finite population bias increasingly inflates estimated MIA vulnerability as N/N_+ grows. Here, applying simple FPC (Section 3.1) tend to underestimate mean AUC w.r.t the population reference whereas whereas decomposed FPC (Section 3.2) yields estimates closer to population-level MIA vulnerability estimates.

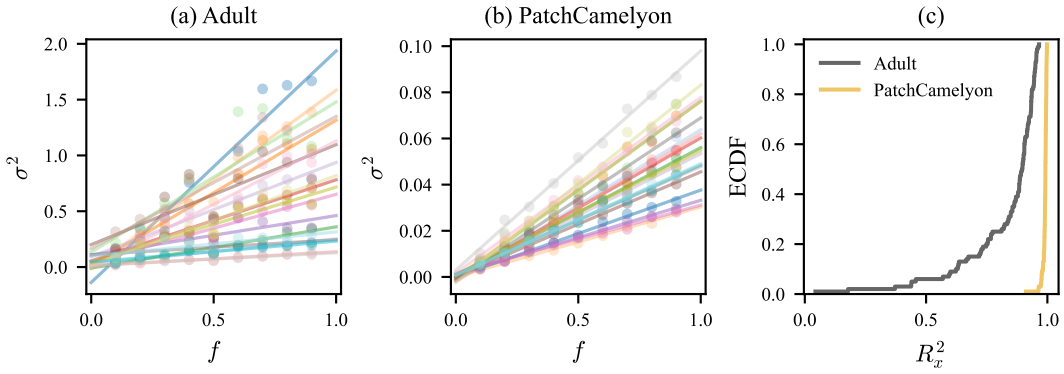


Figure A3: **Variance Decomposition for Adult and PatchCamelyon datasets for OUT distribution variances** (a-b) Observed variances and OLS fits $\sigma_x^2(f) = f\sigma_{\text{pop},x}^2 + \sigma_{\text{rem},x}^2$ for 20 randomly selected target samples as computed per Section 3.2 for Adult and PatchCamelyon datasets.(c) Empirical CDF of per-target R_x^2 across all 100 evaluation targets using observed variances for all ratios $N/N_+ \in \{0.1, \ldots, 0.9\}$ for both the datasets. We train 2048 models per N/N_+, with $N = 1000$, and use the collected statistics to estimate each target's OUT variance.

Table A1: List of hyperparameters used in the experiments.

Hyperparameter	**Adult**	**PatchCamelyon**	**CIFAR10**
Model	TabPFN	ViT-B-16	ResNet50
Adaptation	In-context	Linear	FiLM/Linear
Training Set Size			
FPC (Section 3.1)	1000	1000	–
Post-Processing (Section 4.2)	10000	–	500
Optimizer	–	Adam	Adam
Epochs	–	500	50
Batch size	–	128	64
Learning rate	–	5×10^{-3}	8×10^{-3}
Weight decay	–	10^{-3}	–

Algorithm 1 Decomposed FPC for IN variances. The algorithm can be repurposed for OUT variances by using models excluding the target and $f(r) = 1 - N/(N_+(r) - 1)$

Require: Training size N, targets $\mathcal{X}$, ratios r_a, r_b, shadow models M, learning algorithm $\mathcal{A}$

for $r \in \{r_a, r_b\}$ **do**
  Construct $D^{(r)}_{\text{full}} \supseteq \mathcal{X}$ with $N_+(r) = \text{round}(N/r)$ records
  for $m = 1, \ldots, M$ **do**
    Sample N records without replacement: $D^{(r,m)} \subseteq D^{(r)}_{\text{full}}$
    Train $\mathcal{M}^{(r,m)} \leftarrow \mathcal{A}(D^{(r,m)})$
    Record target statistics and membership signals for $\forall x \in \mathcal{X}$
  end for
  Estimate $\widehat{\mu}_x(r)$ and $\widehat{\sigma}^2_{\text{obs},x}(r)$ using models whose training sets contain x
  $f(r) \leftarrow 1 - \dfrac{N-1}{N_+(r)-1}$
end for
$\mathcal{V}_{\text{corr}} \leftarrow \varnothing$
for each target $x \in \mathcal{X}$ **do**
  $\widehat{\sigma}^2_{\text{pop},x} \leftarrow \dfrac{\widehat{\sigma}^2_{\text{obs},x}(r_a) - \widehat{\sigma}^2_{\text{obs},x}(r_b)}{f(r_a) - f(r_b)}$
  $\widehat{\sigma}^2_{\text{rem},x} \leftarrow \dfrac{f(r_a)\widehat{\sigma}^2_{\text{obs},x}(r_b) - f(r_b)\widehat{\sigma}^2_{\text{obs},x}(r_a)}{f(r_a) - f(r_b)}$
  $V \leftarrow \widehat{\sigma}^2_{\text{pop},x} + \widehat{\sigma}^2_{\text{rem},x}$
  if $V \le 0$ **then**
    Mark the decomposition invalid and **continue**
  end if
  $B \leftarrow \text{clip}(\widehat{\sigma}^2_{\text{rem},x}, 0, V)$
  for $r \in \{r_a, r_b\}$ **do**
    $\widehat{\sigma}^2_{\text{corr},x}(r) \leftarrow \widehat{\sigma}^2_{\text{obs},x}(r) \dfrac{V}{f(r)(V-B)+B}$
    $\mathcal{V}_{\text{corr}} \leftarrow \mathcal{V}_{\text{corr}} \cup \{(x, r, \widehat{\sigma}^2_{\text{corr},x}(r))\}$
  end for
end for
return $\mathcal{V}_{\text{corr}}$

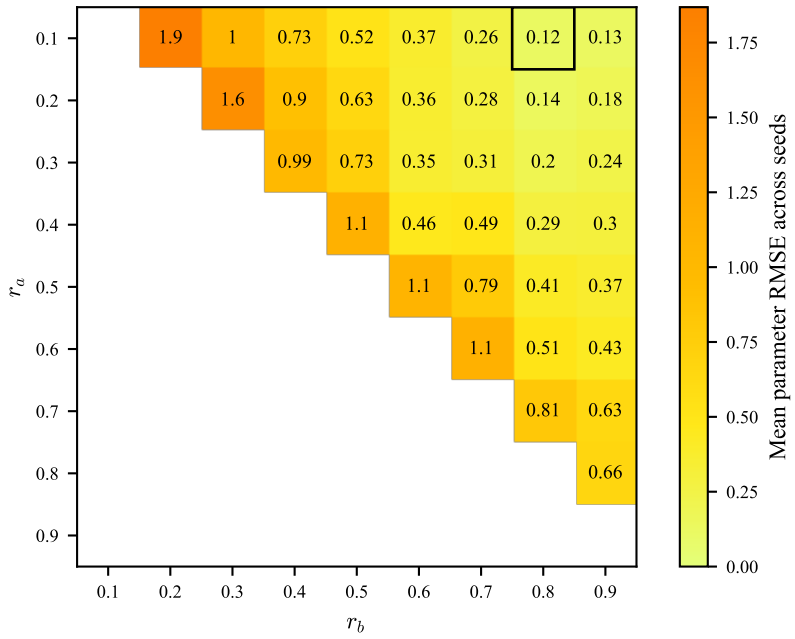


Figure A4: **Selecting ratios for variance decomposition on Adult dataset.** Each cell of the heatmap compares parameter estimates obtained from a pair of sampling ratios (r_a, r_b) with the corresponding all-ratio OLS estimates. RMSE is computed for the 2 settings across all targets and then averaged over 3 repeats. Highlighted values indicate closest parameter alignment between decomposition using (r_a, r_b) and all ratios.

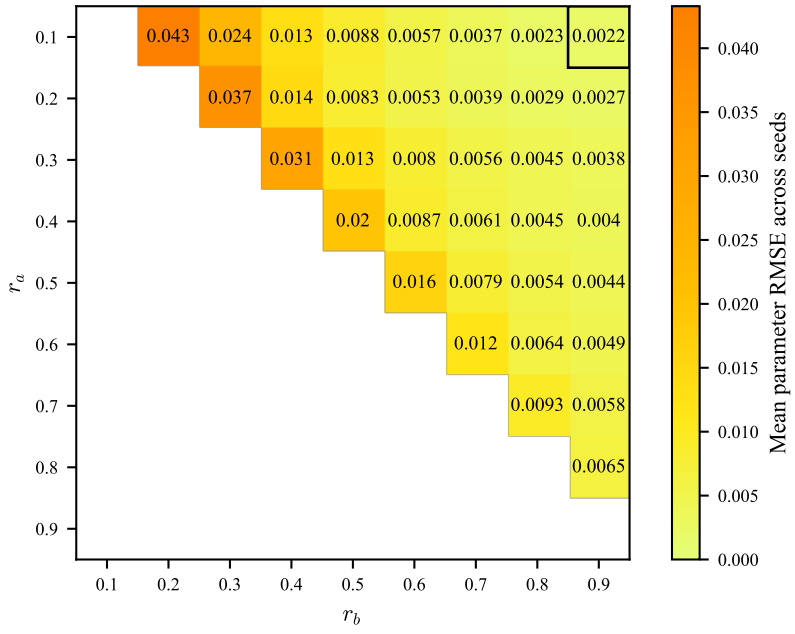


Figure A5: **Selecting ratios for variance decomposition on PatchCamelyon dataset.** Each cell of the heatmap compares parameter estimates obtained from a pair of sampling ratios (r_a, r_b) with the corresponding all-ratio OLS estimates. RMSE is computed for the 2 settings across all targets and then averaged over 3 repeats. Highlighted values indicate closest parameter alignment between decomposition using (r_a, r_b) and all ratios.