# Estimating Model-Level Membership Inference Vulnerability Without Reference Models

**Euodia Dodd**
Imperial College London
e.dodd24@imperial.ac.uk

**Nataša Krčo**
Imperial College London
n.krco23@imperial.ac.uk

**Igor Shilov**
Imperial College London
i.shilov23@imperial.ac.uk

**Matthew Wicker**
Imperial College London
m.wicker@imperial.ac.uk

**Yves-Alexandre de Montjoye**
Imperial College London
demontjoye@imperial.ac.uk

## Abstract

Membership inference attacks (MIAs) have emerged as the standard tool for evaluating the privacy risks of AI models. However, state-of-the-art attacks require training numerous, often computationally expensive, reference models, limiting their practicality. We present a novel approach for estimating model-level vulnerability to the Likelihood Ratio Attack (LiRA), the strongest available attack, directly from the train and test loss distributions of the target model and without training any reference models. We show that LiRA's per-sample signal decomposes into a variance-ratio term and a residual mean-shift term, with the relative contribution of each determined by how much training collapses model uncertainty at the trained sample. This places models on a continuum, with different regimes calling for different reference-free loss-based statistics as proxies for LiRA TPR. The shapes of the loss distributions themselves indicate which proxy applies. We instantiate the framework with two natural proxies. At the heavy-tailed end, the LOSS attack TNR predicts LiRA TPR@FPR=$10^{-3}$ with RMSE 0.036 across 10 image classification architectures and 4 datasets, outperforming low-cost reference-model attacks such as RMIA. At the symmetric end, the LOSS attack AUC predicts LiRA TPR with RMSE 0.018 across five GPT-2 sizes from 10M to 1B parameters.

## 1 Introduction

Large-scale machine learning systems have seen rapid adoption across healthcare, finance, legal, and other domains, with many models now being fine-tuned on sensitive and confidential data. A large body of research has shown that these models may inadvertently memorize and leak samples within their training data (often outliers) (Carlini et al., 2022b, 2019), raising strong privacy concerns.

Membership inference attacks (MIAs) have become the primary tool to measure the empirical privacy leakage of a model, providing both a measure of practical risk and establishing a bound on the success of other types of attacks (Ponomareva et al., 2023). The privacy risk of a model is quantified by the attack's True Positive Rate (TPR) at a low False Positive Rate (FPR) (Carlini et al., 2022a; Ye et al., 2022; Zarifzadeh et al., 2024; Watson et al., 2022), focusing on the members an attack can confidently infer. This is also aligned with legal requirements, such as the EU GDPR "reasonably likely" standard

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

for singling out and the UK Information Commissioner's Office's recent guidance (European Data Protection Board, 2024; Information Commissioner's Office, 2025).

### 1.1 Problem Statement

The most effective MIAs, such as the Likelihood Ratio Attack (LiRA) (Carlini et al., 2022a), provide an estimate of the upper bound of the privacy risk of a model. However, they require training tens to hundreds of reference models, each requiring computational resources equivalent to the original model. It has also been shown that the success of an attack increases with the number of reference models, with recent work showing continuous improvement with 256 reference GPT-2 models (Hayes et al., 2025), and that reference models need to be trained with the same architecture and setup as the model under attack to be optimal (Carlini et al., 2022a; Zarifzadeh et al., 2024). The high computational cost of SOTA attacks limits their practicality as privacy risk evaluation tools, especially for complex models and LLMs. Importantly, this cost becomes even more prohibitive in realistic workflows such as hyperparameter and architecture selection, where practitioners must compare privacy-utility tradeoffs across many training configurations (Ponomareva et al., 2023), as well as continuous monitoring of routinely retrained production models and rapid pre-deployment risk assessments.

A recent line of research has aimed to decrease the number of reference models needed to perform attacks. The state-of-the-art for this, RMIA, however still requires the training of at least one reference model (Zarifzadeh et al., 2024). Instead, another line of research focused on methods to score the vulnerability of individual training samples to MIAs. These however do not provide an overall model-level risk metric (Pollock et al., 2025; Leemann et al., 2024).

**We here propose a new framework to estimate the vulnerability of a model to state-of-the-art membership inference attacks directly from the shape of its train and test loss distributions, without training any reference models.**

### 1.2 Our Approach

Membership inference attacks take advantage of the fact that models tend to be more confident and exhibit lower loss in their predictions for data samples they have seen during training. This differential behavior is a direct consequence of the optimization process that aims to minimize the loss. Some samples are more difficult to learn than others due to their difficulty, rarity or position as outliers in the dataset distribution, and are prone to being memorized.

We focus on LiRA as it represents, to our knowledge, the strongest membership inference attack currently available for both classical deep learning models and language models (Hayes et al., 2025). Estimating LiRA's TPR@low FPR therefore provides a tight estimate of the practical privacy risk from the strongest known attack. While we acknowledge that the attack landscape continues to evolve, focusing on the strongest attack allows for reproducible and interpretable auditing, and for tracking progress in the future as stronger attacks are developed.

We argue that LiRA's per-sample signal decomposes into two components (Section 3): one that captures training-induced collapse of the model's predictive uncertainty, and one that captures a population-level mean shift. The relative importance of these two components varies continuously across models and datasets, placing models on a continuum of uncertainty collapse. At one end, training strongly reduces uncertainty for training (member) samples, producing a heavy-tailed loss distribution for non-members; at the other, uncertainty remains largely unchanged, and only a residual mean-shift signal remains.

Importantly, this continuum is directly observable from the shape of the loss distributions. In the strong-collapse regime, the absence of a heavy tail in the training distribution reflects a population of samples that have migrated from high loss (had they not been seen) to low loss through training. In the weak-collapse regime, where distributions remain symmetric, the signal reduces to a population-level mean separation.

We also empirically confirm both ends across a wide variety of setups: Across 10 image classification architectures on 4 datasets, image classifiers sit near the heavy-tailed end. Across five GPT-2 models from 10M to 1B parameters, LLMs sit near the symmetric end.

Leveraging these insights, we propose a family of loss-based proxies for model-level MIA risk, computed from readily available train and test set loss distributions. In the strong-collapse regime, tail-based statistics capture the fraction of samples that have undergone tail-to-head migration, while in the weak-collapse regime, mean-separation statistics capture the remaining signal. The shape of the loss distribution itself serves as a diagnostic for selecting the appropriate estimator.

We instantiate the framework with two natural proxies, the LOSS attack TNR and the LOSS attack AUC, each appropriate at a different end of the loss-distribution-shape spectrum. We show that, in the strong-collapse regime, LOSS TNR (a tail statistic) provides an accurate estimate of LiRA TPR, while in the weak-collapse regime, the LOSS AUC (a mean-separation statistic) is more appropriate.

This work targets developer-side privacy auditing. The practitioner has access to the target model and to both its training and held-out test sets, and wants a low-cost estimate of how vulnerable the model would be to a state-of-the-art MIA mounted by an external adversary.

## 2 Preliminaries

### 2.1 Notation

We denote with $f_\theta : \mathcal{X} \to \mathcal{Y}$ a machine learning classification model parameterized by $\theta$, with input space $\mathcal{X}$ and output space $\mathcal{Y} = [0,1]^n$. Model $f_\theta \leftarrow \mathcal{T}(D_{\text{train}})$ is trained on dataset $D_{\text{train}} \sim \mathcal{X} \times \mathcal{Y}$ using algorithm $\mathcal{T}$ and loss function $\ell$. A separate test dataset $D_{\text{test}} \sim \mathcal{X} \times \mathcal{Y}$ is reserved for evaluation and is never used during training. For sample $(x, y)$, we write $f_\theta(x)_y$ for the predicted probability for class $y$ and $\ell(x, y) = \ell(f_\theta(x)_y, y)$ for the corresponding loss.

Membership inference attacks (MIAs) aim to determine whether a given record $(x, y)$ was used to train target model $f_\theta$. Specifically, given target model $f_\theta$ and target record $(x, y) \in \mathcal{X} \times \mathcal{Y}$, an MIA $A(x, y, f_\theta)$ aims to infer whether $(x, y) \in D_{\text{train}}$ ($(x, y)$ is a *member*) or not ($(x, y)$ is a *non-member*). We denote the target model's loss on $(x, y)$ with $\ell_{\text{target}}(x, y)$. The attack outputs a membership score that quantifies the likelihood of the target sample $(x, y)$ being part of the training data of the target model. The attacker then makes a binary membership prediction according to threshold $\tau$:

$$A_\tau(x, y, f_\theta) = \begin{cases} 1 & \text{if } A(x, y, f_\theta) \geq \tau \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

A simple and widely used example is the LOSS attack $A^{\text{LOSS}}$ (Yeom et al., 2018), where a sample's membership score is the loss of the target model on it and $\mathbb{1}$ is the indicator function:

$$A^{\text{LOSS}}_\tau(x, y) = \mathbb{1}[-\ell_{\text{target}}(x, y) > \tau] \tag{2}$$

We report the model-level risk as the True Positive Rate (TPR) at a fixed False Positive Rate (FPR)=$\alpha$ (Carlini et al., 2022a; Zarifzadeh et al., 2024; Ye et al., 2022) of an attack.

SOTA attacks rely on *reference models*, models trained on auxiliary datasets $D^i_{\text{reference}} \sim \mathcal{X} \times \mathcal{Y}$ using the same architecture and training process as the target model. For a sample $(x, y)$, an "OUT" reference model $f^i_{\theta_{OUT}}$ is trained on data excluding $(x, y)$, while an "IN" reference model $f^i_{\theta_{IN}}$ is trained with it. This distinction enables the attacker to compare model behavior in the presence and absence of the target sample, improving its power. For a sample $(x, y)$ and a dataset $D^i_{\text{reference}}$, we denote with $\ell^i_{\text{OUT}}(x, y)$ the loss of an OUT model on $(x, y)$, and analogously for an IN model. For K IN reference models, we define the mean loss:

$$\bar{\ell}_{in}(x, y) = \frac{1}{K} \sum_{k=1}^{K} \ell(f^k_{\text{IN}}(x, y)) \tag{3}$$

and analogously for $\bar{\ell}_{out}(x, y)$.

### 2.2 SOTA Reference Model-Based Membership Inference Attacks

The current state-of-the-art reference-model based attack is the Likelihood Ratio Attack (LiRA) (Carlini et al., 2022a). It estimates the loss distributions of models trained with and without a target

sample $(x, y)$, and computes their likelihood ratio as a membership score. In the *online* attack, the attacker trains both "IN" and "OUT" reference models, and in the more efficient *offline* version, only trains "OUT" models. We adopt the strongest online attack in our experiments.

Recent work has explored methods to reduce the number of reference models needed for an effective attack. The state-of-the-art approach in this setting is RMIA (Zarifzadeh et al., 2024). It computes membership scores by comparing the target sample to samples drawn from the population using simple pairwise tests, and has been shown to outperform LiRA when few reference models are available. We compare our method to it.

## 3 Intuition

**Why the tail separates members from non-members.** Figure 1 shows the member and non-member $\ell_{\text{target}}$ distributions on a semilog scale for ResNet-18 on CINIC-10; complete results are shown in Appendix E. In both cases, most probability mass lies at low loss (the "head"), with a smaller fraction at high loss (the "tail"). For members, the head corresponds to learned or memorized examples, while the tail comprises harder-to-learn samples (Pollock et al., 2025). For non-members, the head reflects samples the model generalizes to, whereas the tail contains difficult or outlier cases that the model fails to fit. The two distributions overlap heavily at the head, and what separates them is the tail, which is dominated by non-members. Training has shrunk the member tail by pushing high-loss examples to low loss.

**Vulnerable members are those that are missing from the tail.** Assuming i.i.d. sampling, the absence of a heavy tail for members suggests samples that would have been high-loss (if unseen) have shifted to the low-loss region after training. These tail-to-head samples are now difficult to distinguish from easy non-members, but are exactly those that state-of-the-art membership inference attacks such as LiRA target. Figure 1 supports this hypothesis: samples with the highest LiRA scores exhibit low $\ell_{target}$, yet high $\overline{\ell}_{out}$, aligning with the heavy tail of the non-member distribution (see Appendix E for results for all setups). This pattern indicates that training shifted these examples from the tail to the head. This observation is not specific to a particular attack, but reflects a general property of how training reshapes the loss distribution.

**When tail-to-head migration occurs.** Tail-to-head migration requires that training pin down the model's behavior at the trained sample: a sample's loss becomes reliably low and distinguishable from a non-member at the same $x$ only when including $(x, y)$ in training meaningfully changes how the model behaves at $x$. When the training set carries heavy near-duplication, the model's behavior at $x$ is already determined by similar examples elsewhere, and including a specific $(x, y)$ does not change it. Section 3.1 formalizes this and shows that real models lie on a continuum determined by how much training pins down behavior on individual samples. The signal that an MIA can extract differs at the two ends of this continuum, and so does the appropriate reference-free proxy.

### 3.1 Theoretical Analysis via Gaussian Process Regression

We study membership inference attacks against Gaussian Process (GP) regression models, which, while simpler than deep networks, provide an analytically tractable setting to examine the relationship between LiRA and LOSS attacks. This setting is particularly instructive as it provides a closed-form solution for the LiRA attack—specifically, the performance of LiRA as the number of reference models tends towards the number of data points. Let $\mu_{\mathcal{D}}(x)$ and $\sigma^2_{\mathcal{D}}(x)$ denote the GP's predictive mean and variance, respectively. We denote the corresponding properties for members as $\mu_{\text{mem}}$ and $\sigma_{\text{mem}}$, and for non-members as $\mu_{\text{non}}$ and $\sigma_{\text{non}}$. Under the GP, the likelihood ratio of LiRA, $\Lambda(s) = \ln\left(p(y|x, H_{mem})/p(y|x, H_{non})\right)$, is computed by substituting the Gaussian log-density:

$$\Lambda = \underbrace{\tfrac{1}{2}\ln\left(\frac{\sigma^2_{\text{non}}}{\sigma^2_{\text{mem}}}\right)}_{\text{variance-ratio term}} + \underbrace{\frac{(y-\mu_{\text{non}})^2}{2\sigma^2_{\text{non}}} - \frac{(y-\mu_{\text{mem}})^2}{2\sigma^2_{\text{mem}}}}_{\text{residual term}}. \tag{4}$$

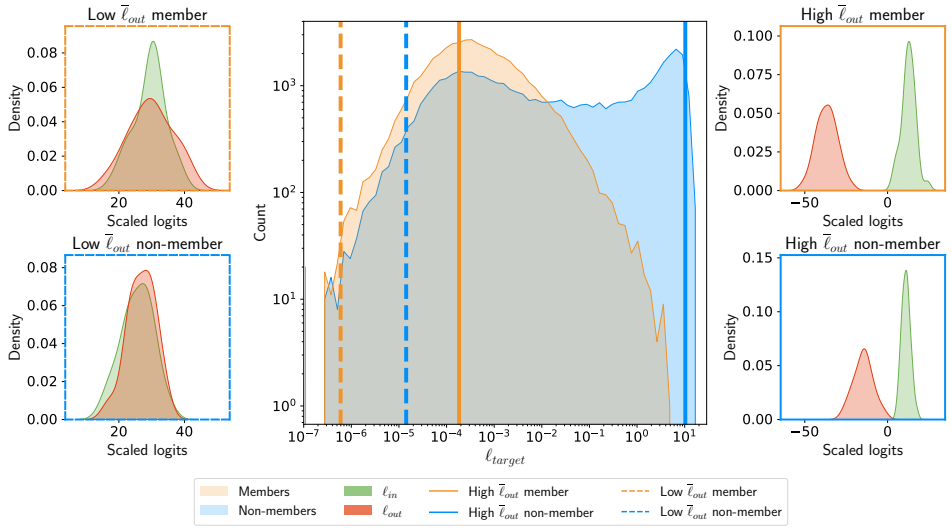


Figure 1: **Vulnerable members are mostly those that moved from tail to head.** The central panel shows member (orange) and non-member (blue) loss distributions on the target model. **Non-members dominate the high-loss tail** (right side), while members concentrate at low loss. Vertical lines mark $\ell_{target}$ values for representative sample groups. Side panels display the IN and OUT scaled logit distributions that LiRA uses to compute membership scores for these samples: left panels show samples with low $\ell_{out}$, right panels show samples with high $\ell_{out}$. In particular, note how the **LiRA-vulnerable members** (orange solid line, top-right panel) have low $\ell_{target}$ but high $\ell_{out}$ indicating these samples would naturally fall in the high-loss tail but moved to the low-loss $\ell_{target}$ head through training. Results shown are for ResNet-18 trained on CINIC-10.

LiRA's advantage over LOSS, which thresholds the squared error $\ell(x, y) = (y - \mu_{\mathcal{D}}(x))^2$, lives in the variance-ratio term i.e the per-sample uncertainty signal. Its size is dependent on how much $\sigma^2$ collapses on the training sample, giving two limits.

**Strong-collapse limit $\sigma^2_{\text{mem}}(x) \ll \sigma^2_{\text{non}}(x)$.** In the interpolating limit ($\mu_{\text{mem}} \to y$, $\sigma^2_{\text{mem}} \to \sigma^2_\epsilon$), Eq. (4) reduces to an uncertainty-calibrated residual:

$$A_{\text{LiRA}} = \mathbb{1}\left[\frac{(y - \mu_{\text{non}})^2}{2\sigma^2_{\text{non}}} + \frac{1}{2}\ln\left(\frac{\sigma^2_{\text{non}}}{\sigma^2_\epsilon}\right) > \tau_{\text{lira}}\right]. \tag{5}$$

Samples with high $\sigma^2_{\text{non}}$ receive a large LiRA bonus. These samples are exactly the tail-to-head population. At the population level this produces heavy-tailed non-member loss distributions, and LOSS TNR at low FNR (the proportion of non-members correctly classified as such by the LOSS attack), which counts exactly this migrated population, is the natural summary. We provide more details on the strong collapse regime in Appendix B.2.

**Weak-collapse limit $\sigma^2_{\text{mem}}(x) \approx \sigma^2_{\text{non}}(x) \equiv \sigma^2(x)$.** The variance-ratio term vanishes and Eq. (4) reduces to a mean-shift:

$$\Lambda \approx \frac{(\mu_{\text{mem}} - \mu_{\text{non}})\big(2y - \mu_{\text{mem}} - \mu_{\text{non}}\big)}{2\sigma^2(x)}. \tag{6}$$

LiRA can now only detect shifts in the model's mean prediction under training. LOSS uses the same signal at the population level, but LiRA retains an advantage as it measures the per-sample mean shift directly via $\mu_{\text{non}}$ from reference models, while LOSS can only threshold the target model's raw loss. This limit produces near-symmetric loss distributions, and LOSS AUC (a measure of population-level mean separation) is the natural summary.

**Real models live between these limits.** Eqs. 5 and 6 are the endpoints of a continuum, not a classification of models. For any real target, $\sigma^2_{\text{mem}}/\sigma^2_{\text{non}}$ varies across samples and sits somewhere between 0 and 1, with the population-level distribution of this ratio determining where the model

falls along the axis. As the ratio shrinks on average, the variance-ratio term contributes more to $\Lambda$, the non-member loss distribution develops a heavier tail, and TNR becomes a stronger proxy for LiRA TPR than AUC. As the ratio grows toward 1, the residual term carries the signal, the loss distribution becomes more symmetric, and AUC overtakes TNR. Both proxies are reference-free, and the loss-distribution shape, itself reference-free, tells the practitioner which one is more appropriate. We provide a more detailed analysis of the efficacy of both summary statistics as proxies under different regimes in Appendix B.

**The continuum is visible in a controlled toy.** We illustrate this directly using a piece-wise regression problem where "easy" examples (linear) occupy the center and "hard" examples (non-linear) form the tails. As shown in Figure 2A, we purposefully sample fewer training points from the tails to induce high epistemic uncertainty, reflecting the conditions for tail-to-head migration.

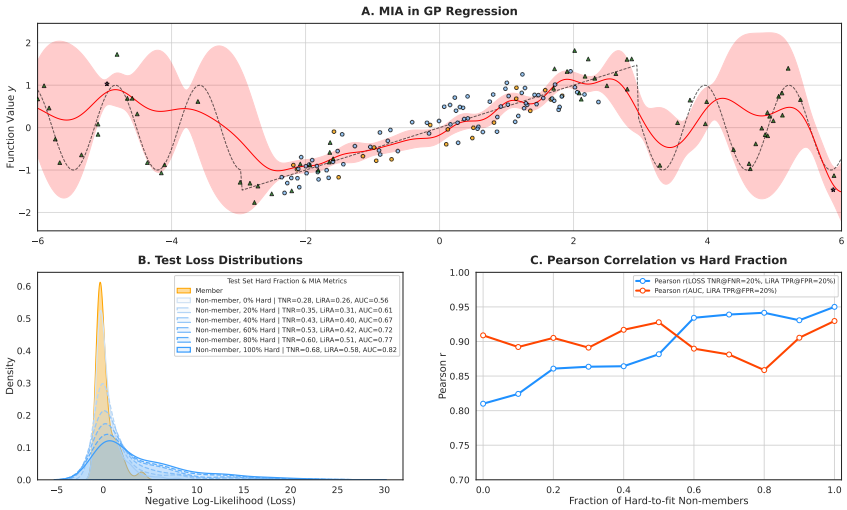


Figure 2: **Relationship between loss-distribution shape and the best loss-based proxy for model-level MIA risk. (A)** Toy GP regression setting with easy-to-fit and hard-to-fit members (orange) and non-members (blue), showing how sparse, high-uncertainty regions induce large losses for hard non-members. **(B)** As the fraction of hard-to-fit non-members increases, the non-member loss distribution becomes increasingly heavy-tailed, increasing both LiRA TPR and the predictive power of tail-sensitive metrics. **(C)** Pearson correlation with LiRA TPR shows that LOSS AUC is the stronger predictor when the loss distributions are relatively symmetric, whereas LOSS TNR overtakes it once the non-member distribution is sufficiently heavy-tailed.

By varying the fraction of hard-to-fit non-members continuously from $0\%$ to $100\%$, we slide the model along the continuum from the no-collapse end toward the full-collapse end. As this fraction increases, the non-member loss distribution shifts from symmetric to increasingly heavy-tailed (Figure 2B), tracing the path from the weak-collapse limit to the strong-collapse limit. The bottom-right panel shows the consequence: Pearson correlation between LOSS AUC and LiRA TPR is highest when the distribution is symmetric, while LOSS TNR overtakes it once the tail develops. We empirically show this to also be the case for image models and report results in Appendix D.

## 4 Experimental Setup

**Datasets.** We select 4 image classification datasets widely used in the literature: MNIST, CIFAR-10, CINIC-10 (Darlow et al., 2018) and CIFAR-100 (Krizhevsky et al., 2009). MNIST contains 70,000 grayscale 28×28 images of handwritten digits (0-9). CIFAR-10 has 60,000 32×32 color images across 10 natural object classes, and CINIC-10 combines CIFAR-10 with downsampled ImageNet data (Deng et al., 2009), creating 270,000 32×32 images across the same 10 classes as CIFAR-10 but with greater data volume and diversity. Finally, CIFAR-100 contains the same images as CIFAR10, but is divided into 100 fine-grained classes.

**Models.** We train 10 neural network architectures commonly used for image classification, ranging between 270K and 86M parameters on all 4 datasets. We train ResNet-20 (He et al., 2016), WRN28-2 (Zagoruyko & Komodakis, 2016), MobileNetV2 (Sandler et al., 2018), DenseNet121 (Huang et al., 2017), WRN40-4, ResNet-18, WRN28-10, VGG11 and VGG16 (Simonyan & Zisserman, 2015). Additionally, we finetune ViT-B-16 (Dosovitskiy et al., 2021) pretrained on ImageNet on CIFAR-10.

**Training.** We follow the setup from Carlini et al. (2022a) where each model is trained on a randomly selected subset consisting of 50% of the samples from the training split as members, with the other 50% reserved as non-members. We use SGD, momentum 0.9, with a cosine annealing learning rate schedule and a batch size of 256. Models are trained for 50 to 200 epochs using weight decay and data augmentation to achieve a high training accuracy of over 90%.

**Attacks.** We instantiate LiRA and RMIA in the online setting, and follow the "mirror" setup from Carlini et al. (2022a) with 2 augmentations. For the image models, we use $K = 64$ reference models (32 IN and 32 OUT). For the GPT-2 models, we follow the implementation from Hayes et al. (2025), using $K = 127$ reference models. RMIA is instantiated with $K = 2$ reference models. For the LOSS attack, we collect per-sample loss values at the final epoch of training for the target model. Appendix P reports the computational cost of each attack.

## 5 Estimating Vulnerability to SOTA MIAs

The GP analysis in Section 3 predicts that two reference-free metrics track LiRA TPR depending on where the target model sits on the variance-collapse continuum: TNR at the heavy-tailed end, where the variance-ratio term dominates and tail-to-head migration is the operative mechanism, and AUC at the symmetric end, where only the residual mean-shift signal remains. We now confirm both empirically.

### 5.1 Image Recognition Models

Image classifiers are expected to lie in the strong-collapse regime, where membership inference signal is concentrated in the tail. Consistent with the theory, we observe a heavy-tailed non-member loss distribution, suggesting that tail statistics should govern attack performance.

The TNR measures how good the LOSS attack is at confidently identifying non-members, providing a measure of how "separable" the tail is from the member distribution. For a set of non-members $D_{\text{test}}$, model $f_\theta$ and threshold $\tau$, we compute the LOSS TNR as the fraction of correctly identified non-members:

$$\text{TNR}_{\text{LOSS}}(D_{\text{test}}, f_\theta, \tau) = \frac{\sum_{(x,y)\in D_{\text{test}}} A_\tau^{\text{LOSS}}(x, y, f_\theta)}{|D_{\text{test}}|} \tag{7}$$

We select $\tau$ to achieve a False Negative Rate equal to the FPR of the LiRA attack for which we are estimating the TPR. In line with the literature, we focus on FPR=0.001, but present results for different FPR values (see Appendix F).

We select FNR=FPR as the most natural choice, but show our method is robust to the choice of FNR threshold: as shown in Appendix G, LOSS TNR maintains strong predictive performance across a wide range of FNR values, with RMSE remaining relatively stable.

Figure 3 shows the LOSS TNR to be a very good predictor of LiRA's TPR@FPR=$10^{-3}$ across models and datasets, obtaining a low RMSE of $0.036$ with a simple linear predictor with one parameter. Bootstrapping with 1000 iterations provides narrow confidence intervals, confirming that the linear model maintains a high degree of certainty over the full range of values. This allows practitioners to accurately estimate the LiRA TPR@FPR without reference models. This is particularly useful for many scenarios that require evaluating many candidate models such as hyperparameter tuning, neural architecture search, or comparing training strategies.

We then compare our approach against an alternate metric, (i) LOSS attack AUC, and baselines: (ii) train-test accuracy gap of the target model; and (iii) LT-IQR AUC (Pollock et al., 2025). LT-IQR is

Table 1: Performance of reference-free statistics and RMIA as predictors of LiRA TPR@FPR$= 10^{-3}$, across the 10 image-classification architectures and 4 datasets of Section 4. Each statistic is mapped to LiRA TPR by a one-parameter least-squares fit anchored at that statistic's no-signal point: the origin for tail statistics, gaps and distances, and chance (0.5) for AUC-type statistics. Baselines are ordered by RMSE; our estimator is set off below them. LOSS TNR is the strongest predictor, ahead even of RMIA, which requires training two reference models.

| Metric | $R^2$ | RMSE | MAE |
|---|---|---|---|
| RMIA (2 reference models) | 0.909 | 0.046 | 0.035 |
| LT-IQR AUC | 0.803 | 0.068 | 0.046 |
| LOSS AUC | 0.802 | 0.068 | 0.049 |
| KS distance | 0.787 | 0.070 | 0.052 |
| Train-test accuracy gap | 0.681 | 0.086 | 0.058 |
| **LOSS TNR** (ours) | **0.944** | **0.036** | **0.025** |

the SOTA for identifying the samples most at risk to LiRA, but is not a measure of overall model vulnerability.

Table 1 reports goodness-of-fit metrics for linear functions fit on our approach and baselines, showing it to outperform the baselines and to be a good estimator of vulnerability to LiRA.

Aware of the prohibitive cost of SOTA MIA methods, a recent line of research aims to develop low-cost attacks which can be used by model developers to estimate the model-level risk. We also instantiate the SOTA low-cost attack, online RMIA, with two reference models and compare its predictive power to our method. As shown in Table 1, our method outperforms RMIA at estimating LiRA TPR, despite RMIA requiring at least 2 reference models.

LOSS AUC, which measures distributional separability across all loss values, would be more effective if memorization were evenly distributed. The train-test gap, a metric studied in the literature, captures only mean differences between loss distributions. While it performs moderately well, we observe that it cannot detect tail differences caused by the asymmetric non-member distribution.

**The relationship holds on unseen architectures and datasets.** The fit shown in Figure 3 also holds on unseen setups. It achieves RMSE $=$ 0.038 vs. 0.036 in sample under leave-one-architecture out. We also leave-one-dataset out (RMSE $=$ 0.064), which is more challenging as holding out CIFAR-100 forces the fit to extrapolate. The ranking within datasets is unaffected, with $\rho = 0.92$–$0.95$. We report the full results in Appendix I.

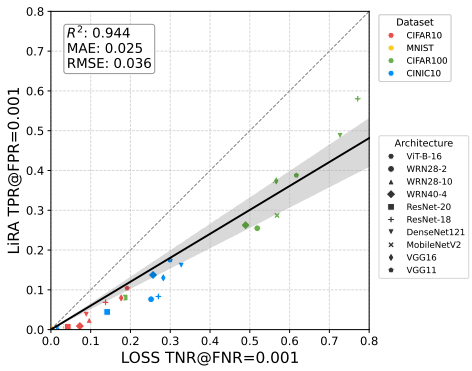


Figure 3: LOSS TNR reliably predicts LiRA TPR@FPR=0.001 across varied model architectures and datasets. The linear relationship (solid black line) exhibits strong performance with RMSE=0.036. The 97.5% confidence interval is shown as the shaded region.

### 5.2 LLMs

Section 3 identifies a second reference-free proxy at the symmetric end of the continuum, where training does not meaningfully collapse $\sigma^2$ at the trained sample and only the residual mean-shift term of Eq.4 survives.

Recent evidence suggests that LLMs exhibit fundamentally different memorization patterns than traditional models. The size of LLMs, for instance, allows them to achieve excellent generalization while simultaneously memorizing training data (Tirumala et al., 2022) and redundancy in large-scale text corpora means that information at $(x, y)$ is typically already represented through near-duplicates elsewhere in training, what Shilov et al. (2024) describe as **mosaic memory**. Including a specific

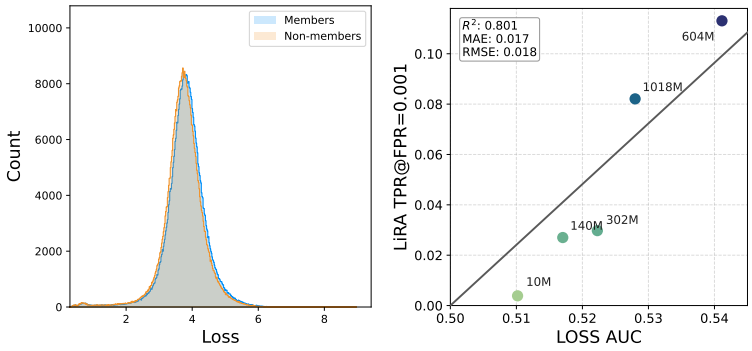


Figure 4: $\ell_{\text{target}}$ **distributions and predictive relationship for large language models**. Left: $\ell_{target}$ distributions for members (orange) and non-members (blue) show high overlap and symmetry, unlike the asymmetric, heavy-tailed distributions observed in image classification models (Figure 1). Right: Despite symmetric distributions, LOSS AUC reliably predicts LiRA TPR@FPR=0.001 across five GPT-2 models (10M to 1018M parameters), achieving RMSE=0.018. Points are labeled by model size.

$(x, y)$ therefore does not collapse the model's uncertainty at $x$ to the extent it does in image classifiers, where each sample carries roughly unique information. Furthermore, the sequence-level loss is an average over many tokens, most carrying no membership signal, which dilutes whatever signal the relevant tokens do carry, weakening the membership signal (Tao & Shokri, 2025).

We test this directly. Following the setup proposed in recent work (Hayes et al., 2025), we evaluate our method on 5 GPT-2 models (Radford et al., 2019) with 10M, 140M, 302M, 604M, and 1018M parameters. We train each model on a $2^{19}$-sample subset of the C4 dataset (Raffel et al., 2020) and instantiate online LiRA with $K = 127$ reference models.

Figure 5 shows that across all five GPT-2 sizes, $\sigma^2_{\text{in}}$ and $\sigma^2_{\text{out}}$ concentrate close to $y = x$, with visibly less member variance collapse than in any image setup. The corresponding loss distributions (Fig. 4, left) are near symmetric and identical in contrast to the heavy-tailed image-model distributions. With no heavy tail for members to be missing from, TNR has no signal to pick up. The mean-shift residual remains, and Fig. 4(right) shows LOSS AUC to be a strong predictor of LiRA TPR@FPR=$10^{-3}$ across all five GPT-2 sizes, with RMSE 0.018.

We also test this on another family of LLMs, eight Pythia models (14M to 2.8B parameters) trained from scratch on the Pile. Due to the prohibitive cost of running LiRA, we use RMIA as a lower cost proxy for LiRA and find that the same relationship holds. LOSS AUC predicts RMIA at TPR@FPR$= 10^{-3}$ with $R^2 = 0.862$ and $\text{RMSE} = 0.024$. We expand on this in Appendix O.

A practitioner selects a proxy for a given target model using the model's train and test losses. Let $s$ denote the KS distance between the non-member loss distribution and a mean-shifted Gaussian fit to the member losses. Use LOSS AUC if $s \leq 0.5$ and LOSS TNR otherwise. We validate this in Appendix C.

## 6 Conclusion

We present a framework for estimating model-level vulnerability to SOTA membership inference attacks without reference models. We show that LiRA's per-sample signal decomposes into a variance-ratio term and a residual mean-shift term, with the relative contribution of each determined by how much training collapses model uncertainty at the trained sample. This places models on a continuum, with the shape of the loss distribution itself acting as a reference-free diagnostic for where a given model sits and which loss-based proxy of LiRA TPR is appropriate.

We instantiate the framework with two natural proxies. At the heavy-tailed end of the continuum, where image classifiers sit, the LOSS attack TNR captures the proportion of high-loss outliers absent from the training side, and predicts LiRA TPR@FPR=$10^{-3}$ with RMSE 0.036 across 10 architectures and 4 datasets, outperforming the train-test accuracy gap, LT-IQR, and low-cost reference-model

attacks. The method is robust across generalization gaps from near-zero to over 50 percentage points, across FNR threshold choices, and across reference-model attack families: it predicts the TPR of RMIA and Attack R with similar accuracy, confirming that it captures signal shared across this family of MIAs rather than a LiRA-specific artifact. At the symmetric end of the continuum, where LLMs sit, the LOSS attack AUC captures the surviving population-level mean-shift signal, and predicts LiRA TPR with RMSE 0.018 across five GPT-2 sizes from 10M to 1B parameters.

While our estimator is empirically calibrated, it is not an arbitrary regression. Both LOSS TNR and LOSS AUC arise as population-level summaries of the same quantities that LiRA thresholds in the two limiting regimes. The observed linear relationship reflects shared dependence rather than incidental correlation.

Where SOTA attacks require training tens to hundreds of reference models, our framework uses only the target model's loss values on its train and test sets. We believe this will substantially help enable privacy risk assessments in practice, in particular during iterative development workflows and scenarios where multiple candidate models require assessment.

**Limitations and future work.** While we evaluate three reference-model MIAs (LiRA, RMIA, Attack R), we have not assessed transferability to attacks from fundamentally different families or to broader privacy threats such as reconstruction or inversion. Our experiments span diverse setups across the continuum, but the framework remains an estimator and may not generalize to all scenarios. Our LLM study covers five GPT-2 models with LiRA and eight Pythia models with RMIA, spanning 10M to 2.8B parameters and two corpora. Scaling to larger models and more datasets is a natural next step. The framework is also proxy-agnostic, and other cheap MIA signals such as Min-K%++ (Zhang et al., 2024) are natural candidates for additional instances.

## Acknowledgments and Disclosure of Funding

We thank Jamie Hayes for his support on the experiments related to LLMs, and for providing valuable feedback.

Additional revenues related to this work: Stipend and compute from MATS Research; Stipend and compute from Anthropic Fellows Program.

## References

Becker, B. and Kohavi, R. Adult. UCI Machine Learning Repository, 1996.

Bertran, M., Tang, S., Roth, A., Kearns, M., Morgenstern, J. H., and Wu, S. Z. Scalable membership inference attacks via quantile regression. *Advances in Neural Information Processing Systems*, 36: 314–330, 2023.

Biderman, S., Schoelkopf, H., Anthony, Q., Bradley, H., O'Brien, K., Hallahan, E., Khan, M. A., Purohit, S., Prashanth, U. S., Raff, E., Skowron, A., Sutawika, L., and van der Wal, O. Pythia: A suite for analyzing large language models across training and scaling, 2023. URL `https://arxiv.org/abs/2304.01373`.

Carlini, N., Liu, C., Erlingsson, Ú., 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., Chien, S., Nasr, M., Song, S., Terzis, A., and Tramer, F. Membership inference attacks from first principles. In *2022 IEEE symposium on security and privacy (SP)*, pp. 1897–1914. IEEE, 2022a.

Carlini, N., Terzis, A., Jagielski, M., Tramer, F., Papernot, N., and Zhang, C. The privacy onion effect: memorization is relative. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022b. Curran Associates Inc. ISBN 9781713871088.

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., Zhang, Y., and Fritz, M. Gan-leaks: A taxonomy of membership inference attacks against generative models. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, CCS '20, pp. 343–362, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450370899. doi: 10.1145/3372297.3417238. URL https://doi.org/10.1145/3372297.3417238.

Chen, D., Yu, N., and Fritz, M. Relaxloss: Defending membership inference attacks without losing utility. In *International Conference on Learning Representations*, 2022.

Darlow, L. N., Crowley, E. J., Antoniou, A., and Storkey, A. J. Cinic-10 is not imagenet or cifar-10. *arXiv preprint arXiv:1810.03505*, 2018.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

European Data Protection Board. Opinion 28/2024 on certain data protection aspects related to the processing of personal data in the context of ai models. Adopted on 17 December 2024, December 2024. URL https://www.edpb.europa.eu/system/files/2024-12/edpb_opinion_202428_ai-models_en.pdf. Version 1.0.

Fanaee-T, H. and Gama, J. Event labeling combining ensemble detectors and background knowledge. *Progress in Artificial Intelligence*, 2(2):113–127, 2014.

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, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450338325. doi: 10.1145/2810103.2813677. URL https://doi.org/10.1145/2810103.2813677.

Gao, L., Biderman, S., Black, S., Golding, L., Hoppe, T., Foster, C., Phang, J., He, H., Thite, A., Nabeshima, N., Presser, S., and Leahy, C. The pile: An 800gb dataset of diverse text for language modeling, 2020. URL https://arxiv.org/abs/2101.00027.

Hanley, J. A. and McNeil, B. J. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143(1):29–36, 1982.

Hayes, J., Shumailov, I., Choquette-Choo, C. A., Jagielski, M., Kaissis, G., Lee, K., Nasr, M., Ghalebikesabi, S., Mireshghallah, N., Annamalai, M. S. M. S., Shilov, I., Meeus, M., de Montjoye, Y.-A., Boenisch, F., Dziedzic, A., and Cooper, A. F. Strong membership inference attacks on massive datasets and (moderately) large language models, 2025. URL https://arxiv.org/abs/2505.18773.

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*, pp. 770–778, 2016.

He, Y., Li, B., Wang, Y., Yang, M., Wang, J., Hu, H., and Zhao, X. Is difficulty calibration all we need? towards more practical membership inference attacks, 2024. URL https://arxiv.org/abs/2409.00426.

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): e1000167, 2008.

Huang, G., Liu, Z., Van Der Maaten, L., and Weinberger, K. Q. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, 2017.

Information Commissioner's Office. How should we assess security and data minimisation in ai?, 2025. URL https://ico.org.uk/for-organisations/uk-gdpr-guidance-and-resources/artificial-intelligence/guidance-on-ai-and-data-protection/how-should-we-assess-security-and-data-minimisation-in-ai/. Accessed: 2025-09-17.

Kelley Pace, R. and Barry, R. Sparse spatial autoregressions. *Statistics & Probability Letters*, 33(3): 291–297, 1997. ISSN 0167-7152. doi: https://doi.org/10.1016/S0167-7152(96)00140-X. URL https://www.sciencedirect.com/science/article/pii/S016771529600140X.

Kim, Y. Convolutional neural networks for sentence classification. In Moschitti, A., Pang, B., and Daelemans, W. (eds.), *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1746–1751, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1181. URL https://aclanthology.org/D14-1181/.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images, 2009.

Kulynych, B., Yaghini, M., Cherubin, G., Veale, M., and Troncoso, C. Disparate vulnerability to membership inference attacks. *Proceedings on Privacy Enhancing Technologies*, 1:460–480, 2022.

Leemann, T., Prenkaj, B., and Kasneci, G. Is my data safe? predicting instance-level membership inference success for white-box and black-box attacks. In *ICML 2024 Next Generation of AI Safety Workshop*, 2024.

Li, H., Li, Z., Wu, S., Hu, C., Ye, Y., Zhang, M., Feng, D., and Zhang, Y. Seqmia: sequential-metric based membership inference attack. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, pp. 3496–3510, 2024.

Liu, Y., Zhao, Z., Backes, M., and Zhang, Y. Membership inference attacks by exploiting loss trajectory. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, CCS '22, pp. 2085–2098, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450394505. doi: 10.1145/3548606.3560684. URL https://doi.org/10.1145/3548606.3560684.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. Learning word vectors for sentiment analysis. In Lin, D., Matsumoto, Y., and Mihalcea, R. (eds.), *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL https://aclanthology.org/P11-1015/.

Mattern, J., Mireshghallah, F., Jin, Z., Schölkopf, B., Sachan, M., and Berg-Kirkpatrick, T. Membership inference attacks against language models via neighbourhood comparison, 2023. URL https://arxiv.org/abs/2305.18462.

Németh, G. D., Ángel Lozano, M., Quadrianto, N., and Oliver, N. Privacy and accuracy implications of model complexity and integration in heterogeneous federated learning, 2025. URL https://arxiv.org/abs/2311.17750.

Pollock, J., Shilov, I., Dodd, E., and de Montjoye, Y.-A. Free record-level privacy risk evaluation through artifact-based methods. In *34th USENIX Security Symposium (USENIX Security 25)*, pp. 5525–5544, 2025.

Ponomareva, N., Vassilvitskii, S., Xu, Z., McMahan, B., Kurakin, A., and Zhang, C. How to dp-fy ml: A practical tutorial to machine learning with differential privacy. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '23, pp. 5823–5824, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400701030. doi: 10.1145/3580305.3599561. URL https://doi.org/10.1145/3580305.3599561.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4510–4520, 2018.

Shilov, I., Meeus, M., and de Montjoye, Y.-A. The mosaic memory of large language models. *arXiv preprint arXiv:2405.15523*, 2024.

Shokri, R., Stronati, M., Song, C., and Shmatikov, V. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, pp. 3–18. IEEE, 2017.

Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations (ICLR 2015)*. Computational and Biological Learning Society, 2015.

Song, L. and Mittal, P. Systematic evaluation of privacy risks of machine learning models, 2020. URL https://arxiv.org/abs/2003.10595.

Song, L., Shokri, R., and Mittal, P. Privacy risks of securing machine learning models against adversarial examples. In *Proceedings of the 2019 ACM SIGSAC conference on computer and communications security*, pp. 241–257, 2019.

Tao, J. and Shokri, R. (token-level) informia: Stronger membership inference and memorization assessment for llms. *arXiv preprint arXiv:2510.05582*, 2025.

Tirumala, K., Markosyan, A., Zettlemoyer, L., and Aghajanyan, A. Memorization without overfitting: Analyzing the training dynamics of large language models. *Advances in Neural Information Processing Systems*, 35:38274–38290, 2022.

Tobaben, M., Ito, H., Jälkö, J., He, Y., and Honkela, A. Impact of dataset properties on membership inference vulnerability of deep transfer learning, 2025. URL https://arxiv.org/abs/2402.06674.

Watson, L., Guo, C., Cormode, G., and Sablayrolles, A. On the importance of difficulty calibration in membership inference attacks. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=3eIrli0TwQ.

Ye, J., Maddi, A., Murakonda, S. K., Bindschaedler, V., and Shokri, R. Enhanced membership inference attacks against machine learning models. In *Proceedings of the 2022 ACM SIGSAC conference on computer and communications security*, pp. 3093–3106, 2022.

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)*, pp. 268–282. IEEE, 2018.

Zagoruyko, S. and Komodakis, N. Wide residual networks. *Proceedings of the British Machine Vision Conference 2016*, 2016.

Zarifzadeh, S., Liu, P., and Shokri, R. Low-cost high-power membership inference attacks. In Salakhutdinov, R., Kolter, Z., Heller, K., Weller, A., Oliver, N., Scarlett, J., and Berkenkamp, F. (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 58244–58282. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/zarifzadeh24a.html.

Zhang, J., Sun, J., Yeats, E., Ouyang, Y., Kuo, M., Zhang, J., Yang, H. F., and Li, H. Min-k%++: Improved baseline for detecting pre-training data from large language models. *arXiv preprint arXiv:2404.02936*, 2024.

Zhang, X., Zhao, J. J., and LeCun, Y. Character-level convolutional networks for text classification. In *NeurIPS*, 2015.

# A Related Work

**Membership inference attacks.** MIAs aim to infer whether a data sample was used to train a given target model. Homer et al. (2008) originally introduced membership inference attacks (MIA) to identify the presence of an individual's genome data among aggregate summary statistics. Shokri et al. (2017) proposed the first MIA on machine learning models and demonstrated their efficacy in the black-box setting. MIAs have since become the de-facto standard to empirically measure the privacy risk posed by a model and providing an upper bound on the risk of more severe empirical attacks such as attribute inference or model inversion (Fredrikson et al., 2015).

Simple MIAs compare metrics like loss (Yeom et al., 2018) or confidence (Song et al., 2019) against thresholds, while others compare samples against neighboring samples (Mattern et al., 2023). While these attacks can obtain high AUC values, they have been shown to be unable to confidently identify members as they struggle to distinguish between low-loss samples that have been memorized and those that are easy to predict. Difficulty-calibrated attacks (Watson et al., 2022; Song & Mittal, 2020; He et al., 2024) aim to address this but still lack high confidence.

State-of-the-art attacks rely on reference models (Shokri et al., 2017) to calibrate the attack with respect to sample difficulty. The Likelihood Ratio Attack (LiRA) (Carlini et al., 2022a) estimates loss distributions for IN and OUT models and performs a likelihood ratio test to determine membership of a given sample. Other methods such as Attack R use only OUT reference models (Ye et al., 2022), while others employ knowledge distillation to train reference models (Liu et al., 2022; Li et al., 2024).

**Low-cost attacks and free identification of vulnerable samples.** Reference model-based attacks depend heavily on the number and quality of reference models. On CIFAR-10, LiRA TPR@$0.01\%$ drops from $2.83\%$ with 254 reference models to $0.76\%$ with 4 (Zarifzadeh et al., 2024). Performance declines further with fewer models or smaller or mismatched architectures. In practice, dozens to hundreds of identical reference models are often needed for high accuracy, making these attacks difficult to scale to iterative workflows and large-scale models such as LLMs (Carlini et al., 2022a).

Recent work has explored ways to estimate model vulnerability to LiRA at lower computational cost or even for free. The state-of-the-art low-cost attack, RMIA (Zarifzadeh et al., 2024), combines reference models with population samples to construct an attack they show to match LiRA performance with as few as 2 reference models. Quantile regression-based attacks (Bertran et al., 2023) use a single regression model to match LiRA's TPR, but their effectiveness is limited to scenarios with only one reference model.

Another direction of research focuses on free identification of the samples that are most vulnerable to SOTA reference model-based MIAs. Predictors such as sample loss, confidence, and training loss dynamics have been shown to successfully identify highly vulnerable samples (Pollock et al., 2025; Leemann et al., 2024) for free. In particular, LT-IQR has been shown to achieve high precision at identifying samples vulnerable to LiRA (Pollock et al., 2025). These methods, however, do not readily provide model-level risk scores as their outputs are inherently relative.

**Factors influencing membership inference vulnerability.** Previous work has explored the connection between a model's training and its vulnerability to MIAs. Yeom et al. (2018) proposed overfitting as a sufficient but not necessary condition for MIA success, whilst other works demonstrate a correlation between the train-test gap (a measure of overfitting) and MIA success (Yeom et al., 2018; Carlini et al., 2022a). Other work has studied the impact of dataset properties on model vulnerability: models trained on smaller datasets are more vulnerable (Tobaben et al., 2025; Chen et al., 2020; Németh et al., 2025), and, within a dataset, classes with fewer samples contain more vulnerable examples (Kulynych et al., 2022; Chang & Shokri, 2021).

# B Why each proxy is appropriate at its end of the continuum

Section 3.1 places models on a continuum between two limits, with the loss-distribution shape as the reference-free diagnostic for where a model sits. Figure 5 confirms this directly: image classifiers exhibit substantial member variance collapse and sit near the strong-collapse end of the continuum, while LLMs exhibit much less and sit near the weak-collapse end.

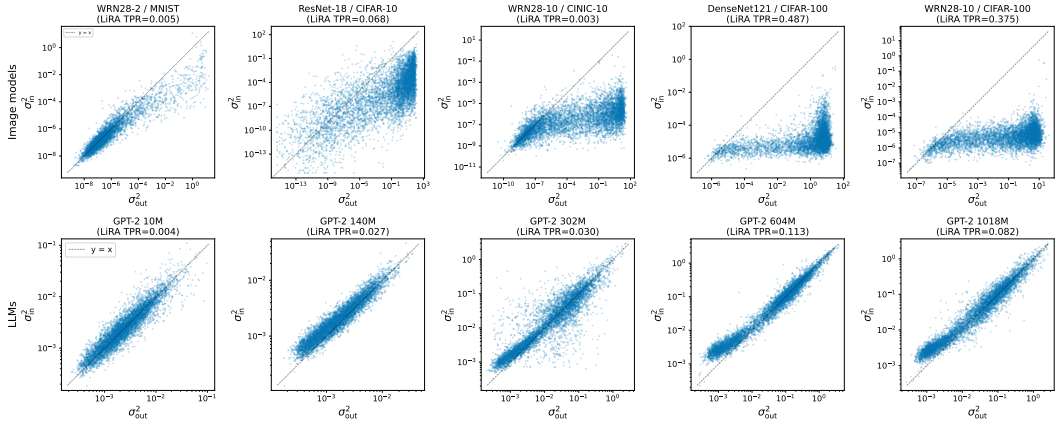


Figure 5: **Training collapses per-sample uncertainty in image models but not in LLMs.** Each panel plots $\sigma^2_{\text{in}}$ against $\sigma^2_{\text{out}}$ per sample. **Top row (image models):** the members are scattered below $y = x$, indicating that training has collapsed the predictive variance at trained samples. **Bottom row (GPT-2):** the points sit close to $y = x$ at all scales. $\sigma^2_{\text{out}}$ itself spans several orders of magnitude indicating that training on a specific sample simply does not meaningfully reduce it rather than it being uniformly low. This places image models near the strong-collapse end of the continuum (Section 3) and LLMs near the weak-collapse end.

We now formalize the corresponding proxy selection. The LiRA decomposition (Eq. (4)) splits the per-sample score into a variance-ratio term and a residual term, with one or the other dominating at each end of the continuum. Whichever term dominates determines a population-level parameter that drives LiRA TPR. In the weak-collapse limit, LOSS TNR and LOSS AUC are functions of the same parameter, and the question is which estimates it more efficiently from a finite sample. In the strong-collapse limit, LOSS TNR tracks the quantity exploited by LiRA directly, while LOSS AUC also absorbs a bulk term that LiRA largely ignores.

We work at the population level. $\mathcal{L}_{\text{mem}}$ and $\mathcal{L}_{\text{non}}$ denote the population distributions of $\ell_{\text{target}}$ on $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{test}}$ respectively. The two proxies are

$$\text{AUC} = \Pr_{X\sim\mathcal{L}_{\text{mem}},\, Y\sim\mathcal{L}_{\text{non}}}(X < Y), \qquad \text{TNR}(\beta) = \Pr_{\ell\sim\mathcal{L}_{\text{non}}}(\ell > \tau_\beta), \tag{8}$$

with $\tau_\beta$ chosen so that $\Pr_{\ell\sim\mathcal{L}_{\text{mem}}}(\ell > \tau_\beta) = \beta$ (the LOSS attack false negative rate). Following the main text, we set this FNR equal to the LiRA FPR at which we predict TPR, so $\beta$ denotes both. Both are computed from $\ell_{\text{target}}$ on $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{test}}$ alone and require no reference models. We relate each proxy to LiRA TPR at fixed FPR $\beta$, in each of the two regimes derived in Section 3.1, and close with the unifying principle.

### B.1 Weak-collapse limit

In the weak-collapse limit of Eq. (6), per-sample variance is unaffected by training and the population-level loss distributions are unimodal and near-symmetric. We model them as $\mathcal{L}_{\text{mem}} \sim \mathcal{N}(\bar{\mu}_{\text{mem}}, \bar{\sigma}^2)$ and $\mathcal{L}_{\text{non}} \sim \mathcal{N}(\bar{\mu}_{\text{non}}, \bar{\sigma}^2)$ with shared variance, and write

$$d = \frac{\bar{\mu}_{\text{non}} - \bar{\mu}_{\text{mem}}}{\bar{\sigma}} \tag{9}$$

for the standardized mean separation. (Bars distinguish these population-level moments from the per-sample moments below)

The FNR constraint $\Pr_{\ell\sim\mathcal{L}_{\text{mem}}}(\ell > \tau_\beta) = \beta$ gives $\tau_\beta = \bar{\mu}_{\text{mem}} + z_{1-\beta}\bar{\sigma}$, where $\Phi$ is the standard normal CDF and $z_q = \Phi^{-1}(q)$. Substituting into the TNR expression, splitting the fraction, and applying $1 - \Phi(x) = \Phi(-x)$:

$$\text{TNR}(\beta) = 1 - \Phi\left(\frac{\tau_\beta - \bar{\mu}_{\text{non}}}{\bar{\sigma}}\right) = 1 - \Phi(z_{1-\beta} - d) = \Phi(d - z_{1-\beta}). \tag{10}$$

Variances add for independent random variables, so $Y - X \sim \mathcal{N}(\bar{\mu}_{\text{non}} - \bar{\mu}_{\text{mem}}, 2\bar{\sigma}^2)$. Then

$$\text{AUC} = \Pr(Y - X > 0) = \Phi(d/\sqrt{2}). \tag{11}$$

While LOSS thresholds a population-level statistic, LiRA is a per-sample attack, so the two operate at different noise scales. For each sample $x$, let $\mu_{\text{mem}}(x)$ and $\mu_{\text{non}}(x)$ denote the mean of its loss over training runs that include and exclude $x$, and $\sigma^2(x)$ its variance over training runs, which is unchanged by training in this limit. Under the GP, these play the role of the predictive moments of Section 3.1. We decompose the population variance of the loss into a within-sample and a between-sample component:

$$\bar{\sigma}^2 = \sigma_w^2 + \sigma_b^2, \qquad \sigma_w^2 := \mathbb{E}_x\big[\sigma^2(x)\big], \qquad \sigma_b^2 := \text{Var}_x\big[\mu_{\text{mem}}(x)\big], \tag{12}$$

where $\sigma_b^2$ captures variation in sample difficulty. For a homogeneous shift $\delta = \mu_{\text{non}}(x) - \mu_{\text{mem}}(x)$, $\sigma_b^2$ is the same for members and non-members. LiRA standardizes each sample's loss against its own reference distribution, so its score is a monotone function of $(\ell - \mu_{\text{non}}(x))/\sigma_w$. This is distributed as $\mathcal{N}(-d_{\text{LiRA}}, 1)$ on members and $\mathcal{N}(0, 1)$ on non-members, with $d_{\text{LiRA}} = \delta/\sigma_w$. LOSS instead sees the population separation

$$d = \frac{\delta}{\bar{\sigma}} = \rho\, d_{\text{LiRA}}, \qquad \rho := \Big(1 + \sigma_b^2/\sigma_w^2\Big)^{-1/2} \leq 1. \tag{13}$$

By the same Gaussian-tail computation as Eq. (10), the three quantities are

$$\text{TPR}_{\text{LiRA}}(\beta) \approx \Phi(d_{\text{LiRA}} - z_{1-\beta}), \quad \text{TNR}(\beta) = \Phi(\rho\, d_{\text{LiRA}} - z_{1-\beta}), \quad \text{AUC} = \Phi\big(\rho\, d_{\text{LiRA}}/\sqrt{2}\big). \tag{14}$$

**Signal-to-noise.** The proxies separate in finite samples. Both see only $d = \rho\, d_{\text{LiRA}}$, and any proxy noise propagates through the calibration into the predicted LiRA TPR. What matters is how much noise each proxy carries relative to how strongly it moves with $d$.

TNR is a binomial proportion. The empirical estimator counts $\mathcal{D}_{\text{test}}$ samples above $\tau_\beta$, with standard error

$$\text{SE}(\widehat{\text{TNR}}) = \sqrt{\frac{\text{TNR}(1 - \text{TNR})}{|\mathcal{D}_{\text{test}}|}} \approx \sqrt{\frac{\text{TNR}}{|\mathcal{D}_{\text{test}}|}} \tag{15}$$

when TNR is small. At the low FNR of interest ($\beta = 10^{-3}$) and $d = 0.5$, Eq. (10) gives $\text{TNR}(10^{-3}) \approx 5 \times 10^{-3}$. With $|\mathcal{D}_{\text{test}}| = 10^4$ the standard error is roughly $7 \times 10^{-4}$. The slope in $d$ is only $\varphi(d - z_{1-\beta}) \approx 0.014$, so the noise on $d$ is about $0.05$.

As AUC is a U-statistic over pairs, every (member, non-member) pair contributes, but pairs sharing a sample are correlated, so

$$\text{SE}(\widehat{\text{AUC}}) \sim \frac{1}{\sqrt{\min(|\mathcal{D}_{\text{train}}|, |\mathcal{D}_{\text{test}}|)}} \tag{16}$$

is independent of $\beta$ (Hanley & McNeil, 1982). For $|\mathcal{D}_{\text{train}}| = |\mathcal{D}_{\text{test}}| = 10^4$ and $d = 0.5$, AUC is around $0.638$ and the standard error is roughly $4 \times 10^{-3}$. The slope in $d$ is $\varphi(d/\sqrt{2})/\sqrt{2} \approx 0.27$, so the noise on $d$ is about $0.015$.

The two estimators have different effective sample sizes. TNR uses only the samples that fall above $\tau_\beta$ (a tiny fraction of the test set at low), and sits in the tail where it barely moves with $d$. AUC, on the other hand, uses every pair. Its noise on $d$ is about three times smaller making it the more reliable predictor of LiRA TPR in this limit.

### B.2 Strong-collapse limit

In the strong-collapse limit of Eq. (5), training collapses the per-sample variance on members. The non-member distribution develops a heavy tail, and the member distribution loses part of that tail to training-induced migration into the head. The two distributions now differ in shape rather than location, so a mean-shift model no longer applies.

We instead use the structure of the train/test split. Members and non-members are drawn i.i.d., so $\mathcal{L}_{\text{non}}$ is also the distribution each member's loss would have followed had it not been trained on. We

denote this counterfactual loss as $\ell_{\text{out}}(x)$. As before, $\tau_\beta$ is the $(1-\beta)$-quantile of $\mathcal{L}_{\text{mem}}$. We assume training does not push members up across the threshold: if $\ell_{\text{out}}(x) \le \tau_\beta$, then $\ell_{\text{target}}(x) \le \tau_\beta$. We call this (A1). The migrated fraction

$$m(\tau) = \Pr_{x \in \mathcal{D}_{\text{train}}} \big(\ell_{\text{out}}(x) > \tau,\ \ell_{\text{target}}(x) \le \tau\big) \tag{17}$$

is the fraction of members that would have sat above $\tau$ unseen but sit below it after training.

**TNR.** Since non-members are drawn from the same distribution, a fraction $\text{TNR}(\beta)$ of members have $\ell_{\text{out}} > \tau_\beta$. By construction, a fraction $\beta$ of members are still above $\tau_\beta$ after training, and by (A1) these belong to the first group. The rest migrated:

$$\text{TNR}(\beta) = m(\tau_\beta) + \beta. \tag{18}$$

TNR is a direct estimate of the migrated fraction. This holds at any $\beta$ and makes no assumption on the shape of either distribution.

**LiRA.** Migrated members have high $\bar{\ell}_{\text{out}}$ and low $\ell_{\text{target}}$. Both terms of Eq. (5) are large for them, and they are the samples LiRA flags at low FPR (Figure 1). Non-migrated members shift too little relative to $\sigma_{\text{out}}(x)$ to clear the threshold. Let $\kappa(\beta, K)$ be the fraction of migrated members that LiRA with $K$ reference models identifies at FPR $\beta$. Then

$$\text{TPR}_{\text{LiRA}}(\beta) \approx \kappa(\beta, K)\, m(\tau_\beta) = \kappa(\beta, K)\big(\text{TNR}(\beta) - \beta\big). \tag{19}$$

This predicts a linear fit with intercept close to zero, and a slope that grows with attack strength. Appendix H confirms that the slope increases with $K$. That $\kappa$ is roughly constant across architectures and datasets for fixed $K$ is an empirical finding, not a consequence of the model.

**AUC.** Let $X \sim \mathcal{L}_{\text{mem}}$ and $Y \sim \mathcal{L}_{\text{non}}$, and condition on whether $Y$ lies above $\tau_\beta$. A non-member above $\tau_\beta$ exceeds all but a fraction $\beta$ of members, so

$$\text{AUC} \approx \text{TNR}(\beta) + \big(1 - \text{TNR}(\beta)\big)\, a_{\text{bulk}}, \qquad a_{\text{bulk}} = \Pr(X < Y \mid Y \le \tau_\beta). \tag{20}$$

AUC depends on the migrated fraction through TNR, but also on $a_{\text{bulk}}$, the ordering of members and non-members within the head.

Comparing Eqs. (18), (19), and (20): TNR and LiRA TPR depend on the same migrated fraction. AUC depends on it too, plus a bulk term.

**Why TNR is preferred.** $a_{\text{bulk}}$ reflects how much training lowers the loss of typical samples. This varies with architecture, augmentation, and training length, independently of tail migration. LiRA at FPR $= 10^{-3}$ largely ignores it, since small shifts on easy samples do not clear a strict threshold. For AUC it is noise: it adds variation across models that LiRA TPR does not share. TNR excludes it. This is consistent with Table 1, where the train-test loss gap (which only measures the shift in mean loss) is a weak predictor compared to LOSS TNR.

**Unifying view.** Eq. (20) holds in both regimes. In the weak-collapse limit, training shifts every member's loss by a small amount, so only members already near $\tau_\beta$ cross it. The migrated fraction $m$ is small, and TNR stays close to $\beta$, where it is dominated by noise. The difference between members and non-members then sits almost entirely in $a_{\text{bulk}}$, which is the mean shift of Appendix B.1. In the strong-collapse limit, training moves a substantial fraction of members out of the tail, and TNR carries the signal. The continuum of Section 3.1 is a continuum in how much of the member-non-member difference sits in the tail term of Eq. (20) versus the bulk term.

## C Selecting a proxy in practice

In Section 5, we apply different proxies to different model families and elaborate on why each proxy is more appropriate in that regime. A practitioner evaluating a given model must decide which regime their model is in, and consequently which proxy is most suitable. A practitioner has access to the target model and its losses on $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{test}}$, but no ground-truth LiRA TPR to validate against. Proxy selection must therefore rely on diagnostics computed directly from these loss distributions.

We test whether the non-member loss distribution is well-approximated by a mean-shifted version of the member distribution. Concretely, we fit a constrained Gaussian model $N(\mu+\Delta, \sigma_{\text{mem}})$ to the non-member losses, where $\mu$ and $\sigma_{\text{mem}}$ are estimated from $\mathcal{L}_{\text{mem}}$ and $\Delta$ is a fitted shift. We then compute the Kolmogorov–Smirnov (KS) statistic $s$ between the empirical CDF of $\mathcal{L}_{\text{non}}$ and this constrained Gaussian. A small KS value indicates that the mean-shift assumption holds (weak-collapse regime), and LOSS AUC is appropriate. A large KS value indicates a systematic deviation from this model e.g., heavy tails, skew.

**The Decision Rule.** Use LOSS AUC if $s \leq \tau$ and LOSS TNR otherwise, with $\tau \approx 0.5$. The two regimes separate cleanly on our models: the five GPT-2 models have $\text{KS} \in [0.047, 0.068]$ and all 37 image classifiers have $\text{KS} \in [0.35, 0.89]$, so any threshold within this gap recovers the proxy selections from Section 5.

To validate the rule in the intermediate range where the choice is ambiguous, we also add 41 RelaxLoss-defended models ($\text{KS} \in [0.07, 0.81]$) and 626 intermediate training checkpoints ($\text{KS} \in [0.012, 0.89]$), with LiRA instantiated against each. $s$ rises smoothly with training accuracy, empirically confirming the continuum of Section 3.

We report the held-out RMSE on the 83 independent configurations (37 image classifier, 5 GPT-2 models, 41 RelaxLoss models). At the error-minimising threshold $\tau^\star = 0.569$:

- Our rule: $0.060$
- Always TNR: $0.074$
- Always AUC: $0.078$
- Per-model oracle: $0.052$

The rule outperforms both single-proxy baselines for any threshold in the interval $\tau \in [0.066, 0.587]$, making $\tau \approx 0.5$ a safe recommendation. For borderline models, computing both proxies on a small calibration set and keeping the lower-error one closes most of the remaining gap at negligible cost.

## D Metric performance along the no-collapse–collapse continuum

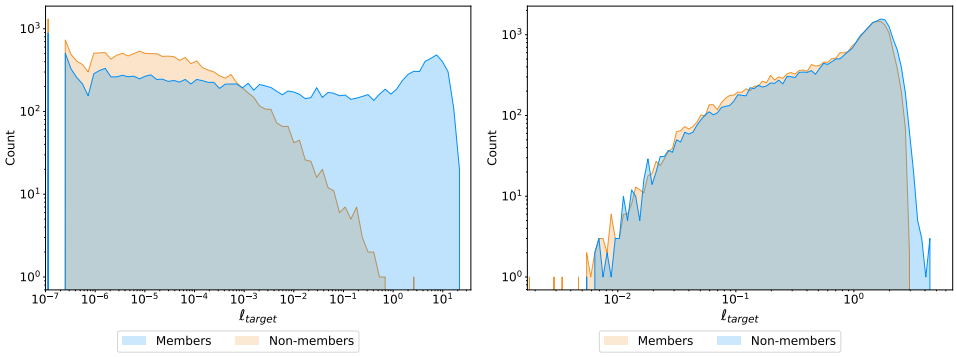


Figure 6: Member and non-member loss distributions for VGG-16 trained on CIFAR-10 with (a) no defense, $s(\mathcal{L}_{\text{mem}}, \mathcal{L}_{\text{non}}) = 0.87$, and (b) RelaxLoss with $\alpha = 1.0$, $s(\mathcal{L}_{\text{mem}}, \mathcal{L}_{\text{non}}) = 0.14$.

We study how the predictive performance of LOSS TNR and LOSS AUC varies as the member and non-member loss distributions of the target model become more similar. To simulate this, we train target models using RelaxLoss (Chen et al., 2022), a privacy defense that explicitly encourages the member and non-member loss distributions to converge during training, across 5 architectures (ResNet-20, DenseNet-121, WRN-28-2, VGG-16, VGG-11) and 2 datasets (CIFAR-10, CIFAR-100). We vary the defense strength $\alpha \in \{0, 0.25, 0.5, 0.75, 1.0\}$ and instantiate LiRA against each resulting target model.

Since the same $\alpha$ value has differing effects across datasets, we use the KS value described in C to measure the distance between the member and non-member loss distributions. Specifically, we fit a constrained Gaussian $N(\mu + \Delta, \sigma_{\text{mem}})$ to the non-member losses. $\mu$ and $\sigma_{\text{mem}}$ are estimated from

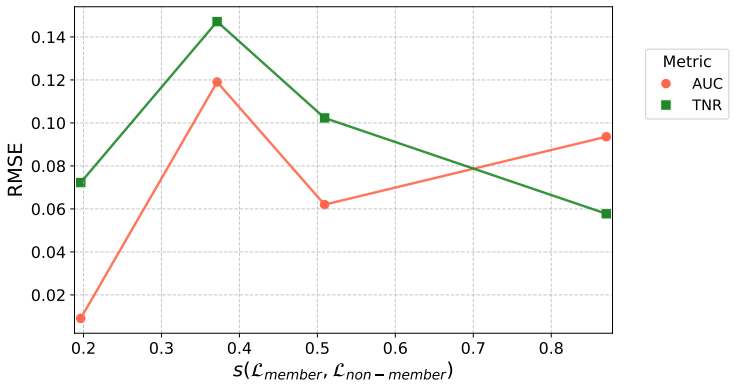


Figure 7: RMSE for predicting LiRA TPR@FPR=$10^{-2}$ with LOSS AUC and LOSS TNR across different member–non-member training distributions. Higher $s$ indicates that the two distributions are more different (strong-collapse limit), and lower values indicate that they are more similar (low-collapse limit).

$\mathcal{L}_{\text{mem}}$ and $\Delta = E[\mathcal{L}_{\text{non}}] - \mu$ is a fitted shift. We use the KS statistic between the empirical CDF of $\mathcal{L}_{\text{non}}$ and this Gaussian directly as a distance measure,

$$s(\mathcal{L}_{\text{mem}}, \mathcal{L}_{\text{non}}) = \sup_{x \in \mathbb{R}} \left| \hat{F}_{\text{non}}(x) - \Phi\left( \frac{x - (\mu + \Delta)}{\sigma_{\text{mem}}} \right) \right|,$$

where $\hat{F}_{\text{non}}$ is the empirical CDF of $\mathcal{L}_{\text{non}}$ and $\Phi$ is the standard normal CDF. A low value of $s$ indicates highly similar distributions (weak-collapse limit), while higher values indicate distributions with different shapes (strong-collapse limit).

Figure 7 shows the RMSE of LOSS TNR and LOSS AUC as predictors of LiRA TPR@FPR=$10^{-2}$ across this range. When the non-member distribution is substantially heavier-tailed than the member distribution (high $s$), LOSS TNR is the stronger predictor. As the two distributions converge, LOSS AUC takes over.

## E Loss distributions across all setups

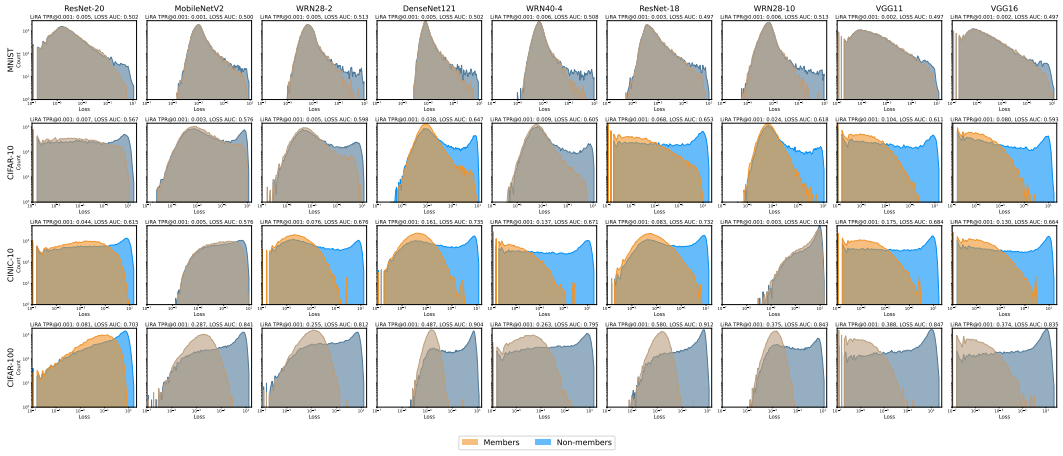


Figure 8: Histograms showing $\ell_{target}$ distributions for training set members (orange) and non-members (blue) in log-log scale across all setups. The distributions demonstrate clear separation between members and non-members, with members typically having lower loss values.

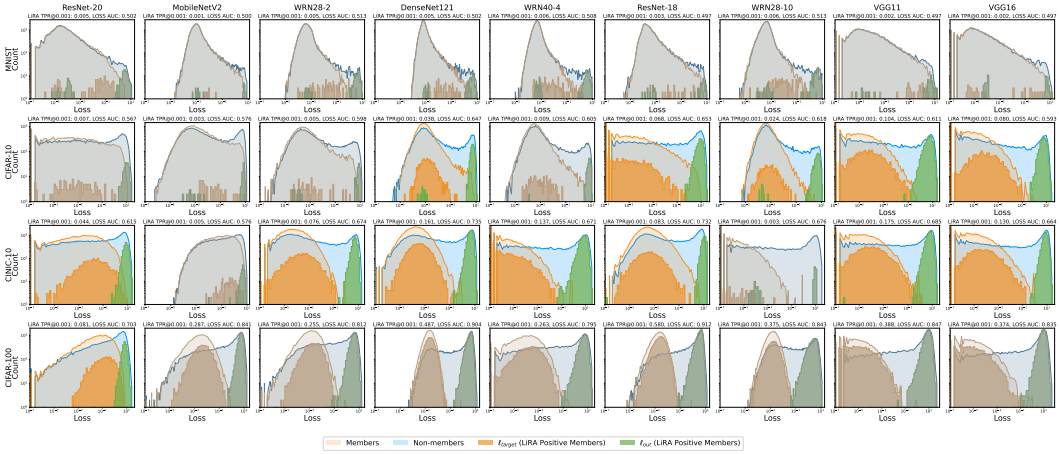


Figure 9: $\ell_{target}$ distributions for training set members and non-members across all setups. Orange histogram shows member losses, blue shows non-member losses, with density curves overlaid. Green bars indicate the OUT model mean ($\ell_{out}$) for points identified by LiRA at FPR=0.001.

# F Varying FPRs

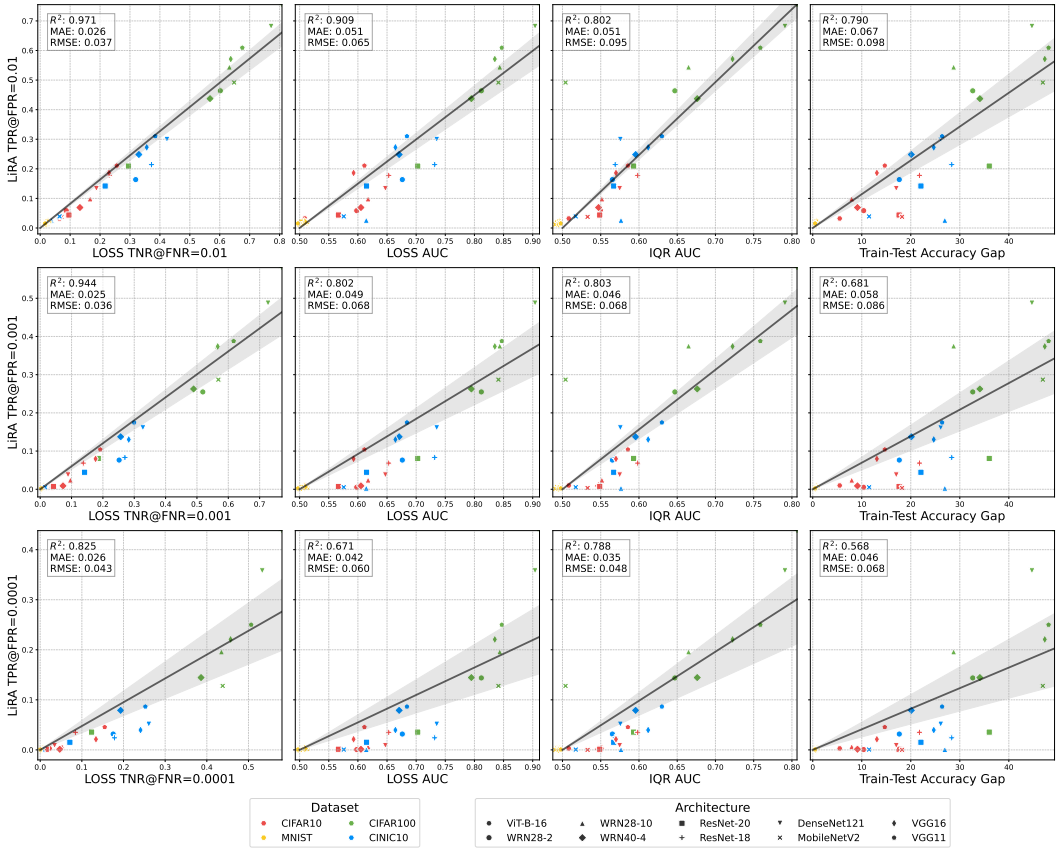


Figure 10: Comparison of different metrics for predicting LiRA TPR at FPRs of 0.01, 0.001, 0.0001. Each panel shows the relationship between a predictor metric (x-axis) and LiRA TPR (y-axis) across all model-dataset combinations. Linear fits (solid lines) with 97.5% confidence intervals (shaded regions) show goodness-of-fit.

## G Sensitivity to selection of LOSS FNR threshold

We select FPR=FNR across the paper, as it is a natural choice. We here study the sensitivity of LOSS TNR's predictive capability to the choice of FNR threshold. Following the experimental setup described in Section 4, we evaluate how well LOSS TNR predicts LiRA's TPR across four different false positive rate thresholds: 0.1, 0.01, 0.001, and 0.0001. For each LiRA FPR setting, we fit a linear function to predict the corresponding LiRA TPR from LOSS TNR values computed at varying FNR thresholds, and measure prediction quality using the RMSE.

Figure 11 shows that RMSE remains relatively stable across a broad range of FNR values, demonstrating that LOSS TNR's predictive power is robust to the specific threshold selection. Notably, setting the FNR equal to the target LiRA FPR (green stars) achieves a similar performance compared to the FNR that minimizes RMSE (yellow stars), showing it to be a good and principled choice. The sharp degradation in RMSE only occurs at very high FNR values (approaching 1.0), where the LOSS threshold becomes too relaxed to provide a meaningful signal.

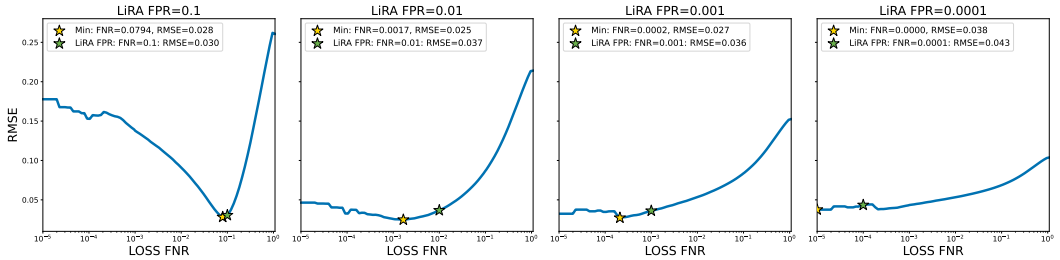


Figure 11: RMSE of LOSS TNR@different FNR values as a predictor of LiRA TPR at a fixed FPR. The yellow star indicates the FNR that achieves the lowest RMSE, while the green star shows performance when setting the FNR to the LiRA FPR. Across all settings, the RMSE difference between the optimal selection and FNR=FPR is minimal.

## H Varying the number of reference models

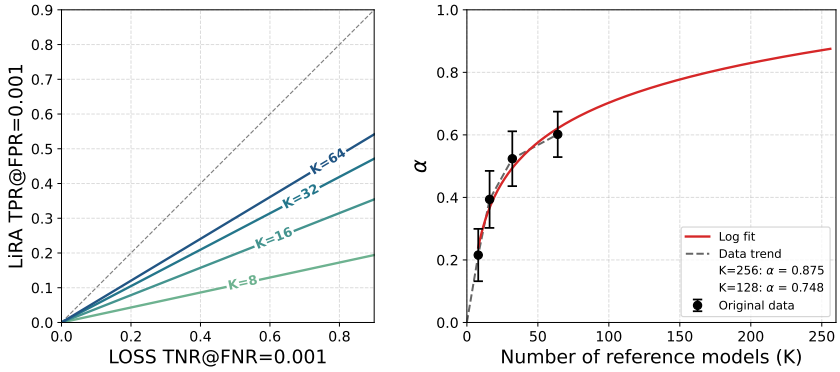


Figure 12: Performance of linear models for predicting LiRA with varying numbers of reference models (K). Left panel shows the correlation between LOSS TNR (x-axis) and LiRA TPR@$0.001$ (y-axis) across different K values, with linear regression fits shown for each condition. Right panel shows slope parameter ($\alpha$) for different K. The error bars show the 97.5% confidence intervals obtained with bootstrapping.

The slope of the linear map from LOSS TNR to LiRA TPR is dependent on the strength of the attack being estimated. We instantiate LiRA with 8, 16, 32 and 64 reference models (half IN, half OUT) on each of our image setups, and fit a line $\mathrm{TPR}_{\mathrm{LiRA}} = \alpha_K \cdot \mathrm{TNR}_{\mathrm{LOSS}}$ for each $K$. Figure 12 shows the fitted lines (left) and the slopes $\alpha_K$ with 97.5% bootstrap confidence intervals (right).

The risk steadily increases with $K$ as the attack becomes more confident at identifying members. As noticed in previous work (Carlini et al., 2022a; Hayes et al., 2025), TPR@low FPR experiences decreasing returns as the number of reference models increase. We can now model this with a logarithmic fit $\alpha_K = a \log K + b$ with $\alpha_{128} = 0.748$ and $\alpha_{256} = 0.875$, giving practitioners a way to estimate risk against much stronger adversaries than they could afford to simulate.

# I Held-out evaluation

In the main body, we report in-sample fit with bootstrap confidence intervals to quantify the uncertainty of that fit. To demonstrate our method also transfers to unseen architectures and datasets, we also report the cross-validated metrics here. We define two groupings: leave-one-architecture-out where we hold out each architecture in turn (10 folds), and leave-one-dataset-out where we do the same for each dataset (4 folds). In each fold, we refit the line on the retained configurations and predict the TPR on the held-out ones. We pool the RMSE over all held-out predictions.

Table 2 reports the results. Under leave-one-architecture-out, the held-out RMSE is within $0.004$ of the in-sample fit at every FPR. We note that the RMSE is higher under leave-one-dataset-out. As CIFAR-100 defines the high end of the range, leaving it out forces the fit to extrapolate beyond the values it was trained on which is the expected cost of extrapolation. As we show below, the ordering of models within each dataset is preserved.

Table 2: Held-out RMSE under leave-one-group-out cross-validation, pooled over all 37 image configurations, compared with the in-sample fit.

| | FPR $= 10^{-2}$ | FPR $= 10^{-3}$ | FPR $= 10^{-4}$ |
|---|---|---|---|
| In-sample | 0.037 | 0.036 | 0.043 |
| Leave-one-architecture-out | 0.038 | 0.038 | 0.047 |
| Leave-one-dataset-out | 0.062 | 0.064 | 0.076 |

**Within-dataset ranking.** Many of the workflows applicable to this work such as hyperparameter tuning, monitoring of trained models and architecture selection only require comparing a set of candidate models rather than estimating the absolute risk. This ranking does not depend on calibrating the slope. Table 3 reports the rank correlation between LOSS TNR and LiRA TPR across architectures within each dataset. The correlation is above $0.91$ on every dataset, demonstrating that the LOSS TNR reliably ranks models by vulnerability, allowing practitioners to use it to compare models without any LiRA calibration.

Table 3: Spearman rank correlation between LOSS TNR@FNR$= 10^{-3}$ and LiRA TPR@FPR$= 10^{-3}$ across architectures, computed within each dataset.

| Dataset | $n$ | $\rho$ | $p$ |
|---|---|---|---|
| MNIST | 9 | 0.950 | $8.8 \times 10^{-5}$ |
| CIFAR-10 | 10 | 0.915 | $2.0 \times 10^{-4}$ |
| CINIC-10 | 9 | 0.933 | $2.4 \times 10^{-4}$ |
| CIFAR-100 | 9 | 0.917 | $5.1 \times 10^{-4}$ |

# J Fitting different functions

So far we have estimated LiRA TPR with a simple linear model. We now study whether non-linear models lead to better risk estimates. Identifying non-members is indeed a matter of finding a good threshold to separate the tail of the loss distribution from the rest, while identifying members is dependent on reference models and is likely to be a more difficult task. We indeed find empirically that the LOSS TNR is typically higher than LiRA TPR at the same FPR.

We thus compute goodness-of-fit metrics for convex functions such as a two-parameter polynomial, power-law, and exponential functions.

Figure 13 shows that the exponential function $a(e^{bx}-1)$ achieves the best fit, outperforming the linear model. An exponential fit would imply that as LOSS TNR increases, member identification becomes easier as the model memorizes more difficult samples. Initial gains in LOSS TNR produce modest improvements in LiRA TPR, but these improvements accelerate as memorization increases. We nonetheless report the linear fit throughout the paper as it performs strongly and is more interpretable.

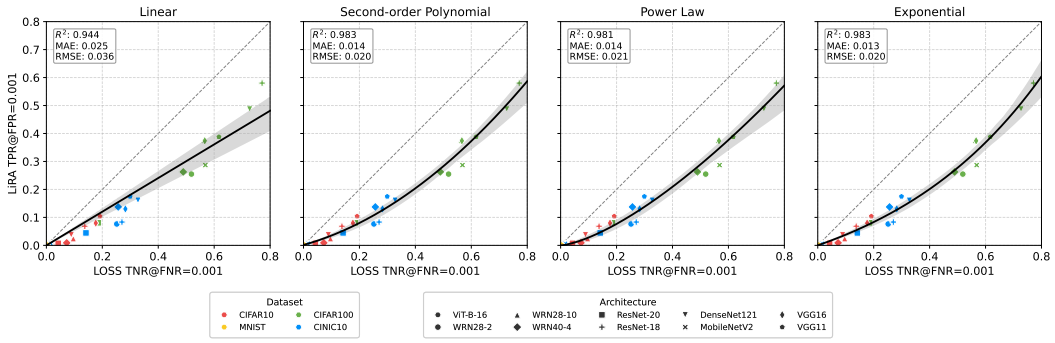


Figure 13: Comparison of regression models for predicting TPR@0.001 with LOSS TNR@0.001 evaluated on image recognition setups. The exponential function achieves the lowest RMSE of 0.02, compared to 0.036 achieved by the linear fit.

## K Additional baselines

Table 4 reports additional reference-free statistics. Each statistic is mapped to LiRA TPR with a linear fit, with the origin fixed at the no-signal point (0.5 for AUC, the origin for tail statistics etc.).

Table 4: The full set of reference-free statistics evaluated, extending Table 1. Fitting convention as in Table 1.

| Metric | $R^2$ | RMSE | MAE |
|---|---|---|---|
| RMIA (2 reference models) | 0.909 | 0.046 | 0.035 |
| LT-IQR AUC | 0.803 | 0.068 | 0.046 |
| LOSS AUC | 0.802 | 0.068 | 0.049 |
| KS distance | 0.787 | 0.070 | 0.052 |
| Train-test accuracy gap | 0.681 | 0.086 | 0.058 |
| Wasserstein-1 | 0.571 | 0.100 | 0.061 |
| Train-test loss gap | 0.562 | 0.101 | 0.061 |
| Quantile gap ($p_{50}$) | 0.503 | 0.108 | 0.072 |
| Quantile gap ($p_{90}$) | 0.456 | 0.113 | 0.068 |
| Quantile gap ($p_{99}$) | 0.204 | 0.136 | 0.086 |
| Partial AUC (FPR $\leq 10^{-3}$) | 0 | 0.153 | 0.120 |
| **LOSS TNR** (ours) | **0.944** | **0.036** | **0.025** |

## L Robustness across levels of generalization gaps

Prior work has identified overfitting, typically measured by the train-test accuracy gap, as a key factor influencing membership inference vulnerability (Yeom et al., 2018; Carlini et al., 2022a).

We now study the effectiveness of our method for predicting the vulnerability of models with widely varying train-test gaps, ranging from near-zero (minimal overfitting) to over 50 percentage points (severe overfitting). Specifically, we treat intermediate training checkpoints from each setup in our main experiment as additional target models, instantiate LiRA against each target, and assess how well our method predicts its vulnerability. Figure 14 shows that LOSS TNR remains a reliable predictor of model vulnerability across this full range of generalization gaps.

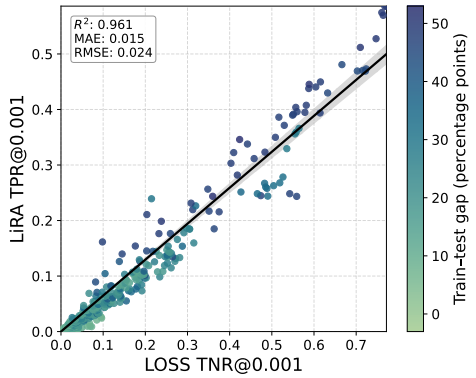


Figure 14: LOSS TNR reliably predicts LiRA TPR across varying overfitting levels.

This consistency demonstrates that LOSS TNR captures memorization patterns much richer than that of the overall generalization performance. While models with larger train-test gaps tend to have higher vulnerability (evidenced by the color gradient), TNR provides accurate risk estimates across a wide spectrum.

## M Predicting vulnerability to different attacks

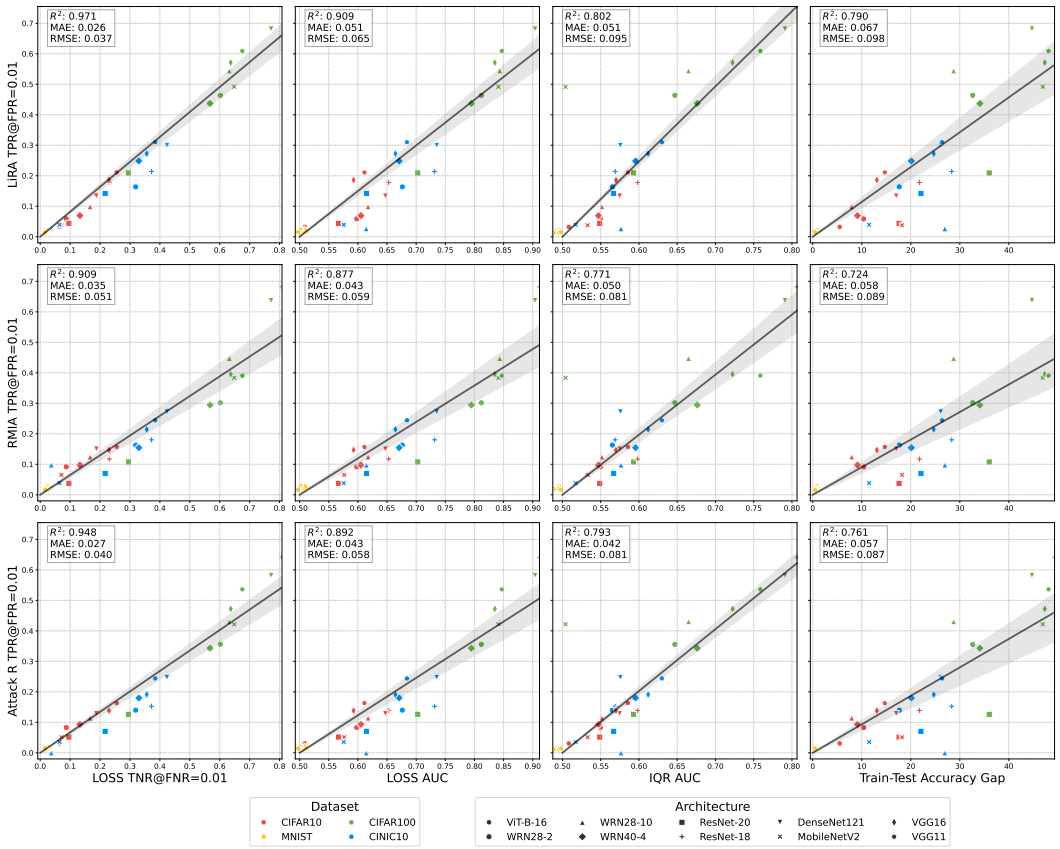


Figure 15: Comparison of different metrics for predicting the TPR at FPR=0.01 for LiRA, RMIA and Attack-R with 64 reference models. Each panel shows the relationship between a predictor metric (x-axis) and the attack TPR (y-axis) across all model-dataset combinations. Linear fits (solid lines) with 97.5% confidence intervals (shaded regions) show goodness-of-fit.

To evaluate whether our method generalizes to other attacks within the same family, we estimate the TPR@0.01 for RMIA and Attack R using 64 reference models (for RMIA: 32 IN and 32 OUT; for Attack R: 64 OUT). Figure 15 demonstrates that TNR serves as an effective predictor for both RMIA and Attack R (two reference-model-based MIAs). Since these attacks share a common approach of using reference models to estimate counterfactuals for individual samples, it follows logically that TNR captures the same underlying signal across attacks in this family.

## N Predicting vulnerability across different tasks

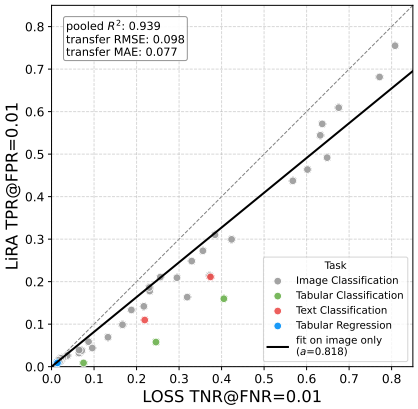


Figure 16: LOSS TNR successfully predicts model-level vulnerability to LiRA across tasks. Our metric achieves a pooled $R^2 = 0.939$. Fitting the prediction line on the image tasks alone and applying it to the non-image tasks gives $\mathrm{RMSE} = 0.098$, against $0.037$ within the image family.

To evaluate how our method performs for tasks beyond image classification, we instantiate it for 3 distinct tasks, specifically:

1. **Text classification**
   (a) IMDb movie reviews dataset (Maas et al., 2011), TextCNN following Kim (2014).

   (b) AGNews dataset (Zhang et al., 2015), TextCNN following Kim (2014).

2. **Tabular classification**
   (a) US Census dataset (Becker & Kohavi, 1996), 3-layer feedforward neural network with ReLU activation

   (b) Texas hospital stays dataset [1], 4-layer feedforward neural network with Tanh activation

   (c) Purchase dataset [2] as processed by Shokri et al. (2017), 4-layer feedforward neural network with Tanh activation

3. **Tabular regression**
   (a) California housing prices dataset, 3-layer feedforward neural network with ReLU activation (Kelley Pace & Barry, 1997)

   (b) Bike sharing dataset (Fanaee-T & Gama, 2014), 3-layer feedforward neural network with ReLU activation

[1] https://www.dshs.texas.gov/THCIC/Hospitals/Download.shtm
[2] https://kaggle.com/c/acquire-valued-shoppers-challenge/data

Figure 16 shows that LOSS TNR remains a strong predictor of LiRA TPR, and, importantly, that the same fitted line is valid for these setups as for the image classification ones. However, the mapping moderately overestimates the vulnerability to these new task families which is the conservative approach for auditing. To achieve an even better fit, we recommend recalibrating the slope once per task family.

## O Generalization to a second language-model family

Section 5.2 evaluates five GPT-2 models. To test whether the weak-collapse behaviour and the LOSS AUC relationship are specific to that architecture and corpus, we also run experiments on Pythia models (8 models, 14M-2.8B parameters) (Biderman et al., 2023), training on samples of the Pile (Gao et al., 2020).

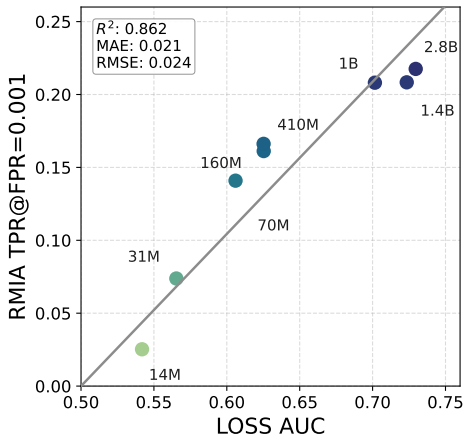


Figure 17: LOSS AUC against RMIA TPR@FPR$= 10^{-3}$ for eight Pythia models trained from scratch on the Pile.

Running LiRA on Pythia-scale models is prohibitively expensive, as it would require training many reference models. We thus use RMIA with $K = 4$ as a lower-cost surrogate. This is motivated by our image model results, where RMIA, while less accurate than our loss-attack-based proxies, correlates well with LiRA as a model-level vulnerability estimator (Table 1, $R^2 = 0.91$, with only two reference models). Using RMIA as a weak proxy for LiRA on Pythia, we observe the same monotonic relationship between LOSS AUC and attack performance that we see for the GPT-2 models: $R^2 = 0.862$, $\mathrm{RMSE} = 0.024$, Pearson $r = 0.941$ ($p < 10^{-3}$) compared to $R^2 = 0.281$ and $\mathrm{RMSE} = 0.055$ when using LOSS TNR to estimate RMIA performance. This provides evidence that the weak-collapse behaviour is not specific to GPT-2, and that the LOSS AUC/LiRA relationship is likely to generalize to other model families.

## P Computational cost

Table 5: Cost of training the required number of reference models to run LiRA and RMIA for each image configuration, measured on a single NVIDIA A100 80GB. Each row reports the per-architecture cost for training on CIFAR-10 (25,000 members, batch size 256) at that architecture's training length. The last row spans all 10 architectures and 4 datasets from Section 4. We also report the cost of our method, which only requires one forward pass over the member and non-member sets of a model.

| Architecture | Params (M) | Epochs | LiRA ($K$=64) (GPU-h) | RMIA ($K$=2) (GPU-h) | Ours (s) |
|---|---|---|---|---|---|
| ResNet-18 | 11.2 | 200 | 5.8 | 0.18 | 0.5 |
| MobileNetV2 | 2.24 | 200 | 11.1 | 0.35 | 1.2 |
| ResNet-20 | 0.27 | 200 | 11.7 | 0.36 | 0.6 |
| VGG11 | 128.8 | 200 | 18.0 | 0.56 | 3.0 |
| WRN28-2 | 1.47 | 200 | 22.2 | 0.69 | 2.2 |
| VGG16 | 134.3 | 200 | 23.7 | 0.74 | 2.6 |
| WRN40-4 | 8.95 | 100 | 23.8 | 0.74 | 5.0 |
| DenseNet121 | 6.96 | 200 | 27.8 | 0.87 | 4.7 |
| WRN28-10 | 36.5 | 100 | 53.2 | 1.66 | 16.7 |
| Range (37 configs) | | | 1.8–95.8 | 0.06–3.00 | 0.5–28.8 |
| median | | | 20.6 | 0.64 | 3.0 |

Table 6: Estimated language-model cost. No training-time logs exist; figures are $6ND$ for one training epoch and $2ND$ for the eval pass, at $40\%$ model-flop utilisation on an A100 80GB (312 TFLOP/s, bfloat16). GPT-2 is trained on $2^{19}$ sequences of 1024 tokens and attacked with LiRA at $K = 127$. Pythia is trained on a $50\%$ split of $2^{18}$ packed 256-token chunks and attacked with RMIA at $K = 4$ (2 IN, 2 OUT): four reference models, each on a random half of the pool. Attack cost is the training of those references; ours is one forward pass over members and non-members of a model that is already trained.

| Model | Attack | Attack cost | Ours (eval) |
|---|---|---|---|
| GPT-2 10M | LiRA ($K$=127) | 9 GPU-h | 3 min |
| GPT-2 140M | LiRA ($K$=127) | 126 GPU-h | 40 min |
| GPT-2 302M | LiRA ($K$=127) | 271 GPU-h | 1.4 GPU-h |
| GPT-2 604M | LiRA ($K$=127) | 551 GPU-h | 2.9 GPU-h |
| GPT-2 1018M | LiRA ($K$=127) | 932 GPU-h | 4.9 GPU-h |
| Pythia 14M | RMIA ($K$=4) | 0.03 GPU-h | 2 s |
| Pythia 31M | RMIA ($K$=4) | 0.06 GPU-h | 5 s |
| Pythia 70M | RMIA ($K$=4) | 0.13 GPU-h | 12 s |
| Pythia 160M | RMIA ($K$=4) | 0.29 GPU-h | 27 s |
| Pythia 410M | RMIA ($K$=4) | 0.73 GPU-h | 1.1 min |
| Pythia 1B | RMIA ($K$=4) | 1.8 GPU-h | 2.8 min |
| Pythia 1.4B | RMIA ($K$=4) | 2.5 GPU-h | 3.9 min |
| Pythia 2.8B | RMIA ($K$=4) | 5.0 GPU-h | 7.6 min |

## NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims are supported by theoretical and experimental results.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.

- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We cover limitations in future works in the Conclusion, including the transferability of our method to other attacks.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: Appendix B states all assumptions explicitly, and derives each result from them. The GP analysis in Section 3.1 is an illustrative model rather than a theorem.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.

- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide all details required to reproduce our experimental results, including datasets, architectures and training setups.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: We have not yet made the code to reproduce our results public, but plan to do so in the future.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.

- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: All results are computed using publicly-available datasets, and we provide sufficient information within the paper to be able to reproduce our results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We compute confidence intervals using bootstrapping and present these in the main plots.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).

- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide an estimate of compute resources used for our experiments in Appendix P.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: This paper adheres to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We highlight the positive social impacts of this work, specifically the enabling of low-cost privacy-risk assessment for model developers, and do not foresee any negative societal impact.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper has no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We include a table detailing all assets used in Appendix **??**.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not release new assets.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.