WHEN MEMBERSHIP SIGNALS LIE: PER-SAMPLE MEMBERSHIP INFERENCE VIA INTERMEDIATE CHECKPOINTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Membership Inference Attacks (MIAs) are the primary tool for privacy auditing and machine unlearning evaluation. Population-based MIAs, which typically derive membership scores solely from the target model's output, are computationally cheap and widely adopted under the assumption that they provide a weak but informative signal. Per-sample MIAs such as LiRA, in contrast, account for individual sample hardness but require training tens to hundreds of shadow (reference) models. Challenging the weak-but-informative assumption, we introduce a vulnerability-stratified evaluation protocol and show that population-based MIAs can produce an *inverted signal* on vulnerable samples—those at greatest risk of privacy leakage—assigning them lower membership scores than typical non-members. Avoiding both the inverted signal of population-based MIAs and the prohibitive cost of per-sample MIAs, we propose **Ghost-LiRA**, a per-sample membership inference that requires no additional model training. Motivated by the observation that vulnerable samples tend to be memorized in the later stages of training, we find that on these samples, mid-stage checkpoints, naturally saved during training, behave almost as if the sample had been excluded, providing the proxies LiRA needs. Ghost-LiRA thus replaces LiRA's spatial ensemble of shadow models with a temporal ensemble drawn from a single training trajectory. Across four datasets and architectures, Ghost-LiRA matches the privacy-critical TPR@0.1%FPR of online LiRA with 16–64 shadow models at a roughly $1600\times$ reduction in computational cost. Applied to unlearning evaluation, Ghost-LiRA exposes substantial residual leakage on vulnerable samples—leakage that population-based MIAs fail to detect.

1 INTRODUCTION

Modern neural networks are increasingly trained on datasets containing personal and sensitive information (Carlini et al., 2022a; Du et al., 2025). To address the resulting privacy risks, regulations such as the GDPR require model providers to audit their models (Murakonda & Shokri, 2020) and uphold users' right to be forgotten, giving rise to the field of machine unlearning (Cao & Yang, 2015; Bourtoule et al., 2021). Together, these requirements have established privacy auditing as a core responsibility of organizations that develop and deploy AI systems (Koshiyama et al., 2024).

Originally introduced to infer whether a given sample was part of a model's training set (Shokri et al., 2017), membership inference attacks (MIAs) have served as the primary tool for both privacy auditing (Murakonda & Shokri, 2020; Song & Marn, 2020; Nasr et al., 2021; Carlini et al., 2022a; Steinke et al., 2023) and unlearning evaluation (Chundawat et al., 2023; Jia et al., 2023; Fan et al., 2024; Golatkar et al., 2020; 2021). Existing MIAs fall into two broad categories that differ in how they calibrate membership decisions. Population-based MIAs apply a shared decision rule across samples (Yeom et al., 2018; Salem et al., 2019; Duan et al., 2023; Zhang et al., 2024; Zhang et al., 2025) and typically derive membership scores from the target model's output alone, making them computationally cheap but unable to account for the inherent hardness of individual samples (Watson et al., 2022; Carlini et al., 2022a). Per-sample MIAs address this limitation by constructing a dedicated decision rule per sample, calibrating each prediction against the sample's own reference distribution via auxiliary models (Carlini et al., 2022a; Watson et al., 2022; Ye et al., 2022; Mireshghallah et al., 2022; Zarifzadeh et al., 2024). These attacks are considerably more powerful but at prohibitive

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

computational cost, requiring the training of at least one auxiliary model—often tens to hundreds in practice. Constrained by this cost, many studies have continued to rely on population-based MIAs for unlearning evaluation (Zhao et al., 2024; Cheng et al., 2026; Spartalis et al., 2025; Zhou et al., 2025; Kim et al., 2025), despite Hayes et al. (2025) demonstrating that such evaluation can underestimate residual privacy leakage. This pattern reflects an implicit assumption that population-based MIAs provide a weak but informative signal.

Prior work (Long et al., 2020; Carlini et al., 2022a) has shown that aggregate metrics such as AUC (area under the ROC curve) obscure worst-case privacy risk. Consequently, reporting TPR at low FPR (true-positive rate at a fixed low false-positive rate) has become standard in MIA evaluation (Carlini et al., 2022a; Zarifzadeh et al., 2024). Yet TPR at low FPR remains an aggregate metric, diluting the signal on the vulnerable subset where worst-case risk concentrates. To expose how MIAs behave across different privacy risk levels, our *vulnerability-stratified evaluation protocol* partitions samples by their susceptibility to privacy risk and reports MIA performance on each subgroup separately. Under this protocol, we formally and empirically demonstrate that population-based MIAs can produce an *inverted* signal on vulnerable samples, assigning them lower membership scores than typical non-members; per-sample MIAs such as LiRA (Carlini et al., 2022a) detect these at-risk samples, while population-based MIAs do the opposite. The root cause is what we term endpoint blindness: any score derived solely from the target model's final output predominantly reflects sample difficulty. Yet vulnerable members can remain atypical to the model even at convergence, making them indistinguishable from—or ranked below—easy non-members.

This raises a natural question: *can we achieve per-sample membership inference—without the inverted signal of population-based MIAs and without the auxiliary-model cost of per-sample MIAs?* We show this is possible by exploiting an overlooked byproduct of standard training: intermediate checkpoints. Our approach builds on two strands of prior work—models fit easy samples early and hard samples late (Arpit et al., 2017; Mangalam & Prabhu, 2019; Jiang et al., 2021), and privacy risk concentrates on atypical samples (Carlini et al., 2019; Feldman & Zhang, 2020; Carlini et al., 2022b)—which together suggest that privacy-vulnerable samples tend to be memorized late in training.

We empirically confirm this prediction: vulnerable samples are memorized notably later than typical ones, with a temporal gap large enough to be practically exploitable. Mid-stage checkpoints of the target model therefore behave closer to models that have not yet seen these samples than to the final trained model, allowing them to serve as shadow-model proxies for LiRA at no additional training cost. Building on this observation, we propose **Ghost-LiRA**, a per-sample membership inference that requires no additional model training. Whereas LiRA constructs its reference distribution by averaging over a *spatial* ensemble of independently trained shadow models, Ghost-LiRA draws its reference from a *temporal* ensemble along a single training trajectory. This approximation is most effective on the vulnerable subset, because mid-stage checkpoints precede the bulk of memorization for these samples. By reusing checkpoints that model owners already possess, Ghost-LiRA enables per-sample MIA at negligible marginal cost. As we will show, Ghost-LiRA matches the privacy-critical TPR@0.1%FPR of online LiRA equipped with 16–64 shadow models at a roughly $1600\times$ reduction in computational cost on average, across four benchmarks spanning diverse architectures and training regimes. Applied to machine unlearning evaluation, Ghost-LiRA exposes substantial residual leakage on vulnerable samples that population-based MIAs declare fully forgotten, suggesting that prior evaluations relying on such attacks may have systematically overestimated the degree of unlearning.

2 Background and Preliminaries

Membership Inference Attacks (MIAs). We distinguish two families of MIAs based on whether they perform per-sample calibration. *Population-based MIAs* derive the membership score solely from the target model's final output[1] (e.g., loss, Yeom et al., 2018; or entropy, Song & Mittal, 2021) and apply a shared decision rule across samples, using either a single global threshold (Yeom et al.,

[1]We restrict the term *population-based* to attacks whose membership score is derived solely from the target model's final output (also called *score-based*; Watson et al., 2022). A separate line of work (Liu et al., 2022; Li et al., 2024) augments this signal with intermediate-stage features from auxiliary models; while these use auxiliary models, they apply a shared rule across samples without per-sample calibration, and thus fall in a grey area between our two categories. See Appendix B.1 for details.

2018; Salem et al., 2019; Song et al., 2019; Duan et al., 2023) or per-class thresholds calibrated via shadow models (Choquette-Choo et al., 2021; Song & Mittal, 2021). However, such attacks fail to capture individual sample difficulty (Watson et al., 2022; Carlini et al., 2022a), motivating the development of *per-sample MIAs*. These methods train multiple shadow models on subsets of a reference dataset to estimate how each sample's confidence varies between models that include the sample in training (IN models) and those that do not (OUT models). LiRA (Carlini et al., 2022a), a canonical instance, fits Gaussians to these per-sample confidence distributions of IN and OUT models and performs a likelihood ratio test. Its offline variant reduces this to a one-sided test against the distribution of OUT models. Other per-sample frameworks (Watson et al., 2022; Ye et al., 2022; Mireshghallah et al., 2022; Zarifzadeh et al., 2024; Du et al., 2025; 2026) likewise require training auxiliary models. These methods typically assume the attacker has access to a dataset of comparable size drawn from the target's training distribution, along with knowledge of the training recipe.

Training dynamics of easy and hard samples. It is well established that DNNs fit easy samples early in training and fit harder samples later in training (Arpit et al., 2017; Mangalam & Prabhu, 2019; Jiang et al., 2021). Pollock et al. (2025) refine this picture by tracking full loss trajectories, identifying two types of outliers: *easy-to-fit outliers*, whose loss stays high through most of training before collapsing in the final stages, and *hard-to-fit outliers*, whose loss never reaches the low values attained by typical samples. They further show that both types of outliers are substantially more vulnerable to membership inference attacks, such as LiRA (Carlini et al., 2022a), than typical samples.

Vulnerable samples. Carlini et al. (2022a) quantify how much a model's behavior on (x, y) shifts depending on whether it was included in training, via the *privacy score*

$$d = \frac{|\mu_{\text{in}} - \mu_{\text{out}}|}{\sigma_{\text{in}} + \sigma_{\text{out}}}, \tag{1}$$

where μ_{in} and σ_{in} (resp. μ_{out} and σ_{out}) denote the mean and standard deviation of logit-scaled confidences on a sample (x, y) across shadow models trained with (resp. without) it. Since this quantity is defined independently of any specific MIA framework, we call a sample **vulnerable** if its privacy score is high. We adopt from Carlini et al. (2022a) and Pollock et al. (2025) the distinction between *hard-to-fit* (loss never reaches typical levels) and *easy-to-fit* (loss eventually collapses to typical levels) samples, and use it throughout as a sub-categorization of vulnerable samples; both carry high privacy scores.

3 Why population-based MIAs fail: signal inversion on vulnerable samples

In this section, we formalize signal inversion on vulnerable samples under the canonical loss attack (Yeom et al., 2018; Ye et al., 2022) as a representative population-based MIA. We first establish pointwise inversion on the *hard-to-fit* subset (Section 3.1), where the inversion is most pronounced, and then discuss how the same argument extends to *easy-to-fit* vulnerable samples in Section 3.2. The argument extends to any attack that depends only on the target model's output on the queried sample.

3.1 Formalizing signal inversion

Let $\mathcal{A}$ denote a (possibly randomized) training algorithm, D a training set, and $\theta = \mathcal{A}(D)$ the resulting trained model. For a sample $z = (x, y)$, we write $\ell(\theta; z)$ for the loss of θ on z. We say z is a *member* if $z \in D$; and a *non-member* otherwise.

Definition 1 (Loss-threshold MIA)**.** *Given a trained model θ and a threshold $\tau \geq 0$, the* loss-threshold membership inference attack *predicts membership via*

$$M_\tau(z; \theta) \;=\; \mathbb{1}[\ell(\theta; z) \leq \tau]\,. \tag{2}$$

This is the canonical Attack-P formulation described in Appendix C.1.

Definition 2 (Hard-to-fit member)**.** *A member $z \in D$ is* hard-to-fit at level η *with respect to $\theta = \mathcal{A}(D)$ if $\ell(\theta; z) > \eta$. We denote the set of such members by*

$$H_\eta(\theta) \;=\; \{\, z \in D : \ell(\theta; z) > \eta \,\}. \tag{3}$$

Definition 3 (Easy non-member)**.** *A non-member $z \notin D$ is* easy at level η *with respect to θ if $\ell(\theta; z) \leq \eta$. We denote the set of such non-members by $E_\eta(\theta)$.*

The following assumption is the structural ingredient that drives signal inversion. It states that, for some loss level η, both hard-to-fit members and easy non-members exist simultaneously.

Assumption 1 (Coexistence of hard members and easy non-members)**.** *There exists $\eta > 0$ such that $H_\eta(\theta) \neq \emptyset$ and $E_\eta(\theta) \neq \emptyset$.*

This assumption follows directly from two complementary observations. Carlini et al. (2022a) and Pollock et al. (2025) identify hard-to-fit members whose loss remains elevated even at convergence, and Watson et al. (2022) report that attacks relying only on the target model's output incur high FPR on easy non-members, which indicates that such samples receive low loss on the final model. Figure 1 confirms their coexistence on CIFAR-100, with the same overlap persisting in modified entropy (Song & Mittal, 2021). We further verify that the same pattern holds across all four benchmarks in Section 6 and Appendix G.2. We now show that, under Assumption 1, there exist individual member–non-member pairs on which the loss-threshold MIA assigns the *opposite* of the correct membership label.

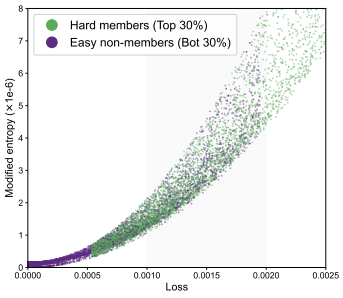


Figure 1: **Coexistence is not a restrictive assumption (CIFAR-100, ResNet-34).** Top-30% highest-loss members and bottom-30% lowest-loss non-members overlap substantially in both loss and modified entropy.

Proposition 1 (Pointwise inversion)**.** *Suppose Assumption 1 holds at level η. Then there exist $\tau \leq \eta$, a member $z_h \in D$, and a non-member $z_e \notin D$ such that*

$$\ell(\theta; z_e) \;\leq\; \tau \;<\; \ell(\theta; z_h), \tag{4}$$

and consequently

$$M_\tau(z_h; \theta) = 0 \quad \textit{and} \quad M_\tau(z_e; \theta) = 1. \tag{5}$$

That is, the attack predicts "non-member" on a true member and "member" on a true non-member.

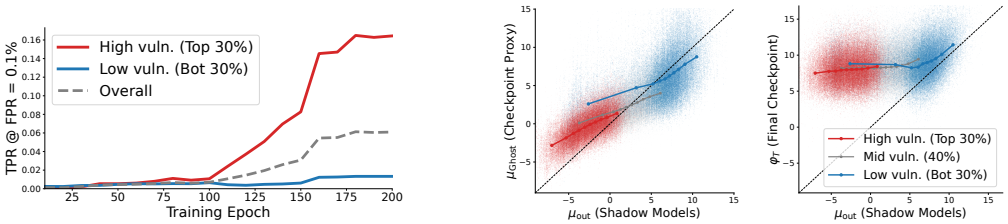
Proof. By Assumption 1, $H_\eta(\theta)$ is non-empty; pick any $z_h \in H_\eta(\theta)$. By Definition 2, $\ell(\theta; z_h) > \eta \geq \tau$, so $M_\tau(z_h; \theta) = 0$. Similarly, $E_\eta(\theta)$ is non-empty; pick any $z_e \in E_\eta(\theta)$. By Definition 3, $\ell(\theta; z_e) \leq \eta$. Choosing $\tau = \eta$ (or any $\tau \in [\ell(\theta; z_e), \eta]$) gives $\ell(\theta; z_e) \leq \tau$, hence $M_\tau(z_e; \theta) = 1$. □

3.2 Discussion

Proposition 1 shows that, under Assumption 1, the loss-threshold MIA can invert the membership signal. Appendix D.1 extends this: Corollary 1 shows no threshold recovers the correct ordering, and Proposition 2 strengthens it to a distributional statement of zero TPR with strictly positive FPR on the hard subset. Figure 1 makes this ranking inversion visually apparent.

The same argument extends to easy-to-fit vulnerable samples. Proposition 1 requires only $\ell(\theta; z_e) < \ell(\theta; z_h)$, a condition satisfied whenever an easy-to-fit vulnerable member is paired with a non-member that generalizes even more readily. Population-based MIAs can therefore invert, rather than merely weaken, the membership signal across vulnerable samples broadly. While prior work noted that attacks relying only on the target's output are confounded by sample difficulty (Watson et al., 2022), Assumption 1 yields a stronger failure: on the vulnerable subset, the signal can be not merely weakened but ranking-inverted. This failure mode captures *endpoint blindness*: any score derived solely from the target model's final output predominantly reflects sample difficulty. The resulting inversion on vulnerable samples is masked in aggregate metrics, which average it out with typical samples.

This limitation is feature-agnostic. In Appendix D.2, we operationalize the overlap region of Figure 1 as members and non-members whose loss falls in $[0.001, 0.002]$. Within this region, four standard MIA signals—loss (Yeom et al., 2018), max softmax confidence (Salem et al., 2019), modified entropy (Song & Mittal, 2021), and logit-scaled confidence (Carlini et al., 2022a)—all yield Cohen's d in the range 0.10 ± 0.04, well below the small-effect threshold ($d = 0.2$): no signal meaningfully separates members from non-members in this region. They are also nearly rank-equivalent on this region (pairwise Spearman $\rho \geq 0.97$), so the four signals induce nearly identical sample rankings.



(a) Vulnerability becomes detectable only late in training

(b) Mid-stage tracks μ_{out}; final drifts above it

Figure 2: **Vulnerable samples tend to be memorized late in training, and mid-stage checkpoints behave OUT-like on these samples** (CIFAR-100, ResNet-34, step-decay schedule; see Figures 5 and 6 for remaining benchmarks). **(a)** TPR@0.1%FPR of LiRA online against intermediate checkpoints, stratified by vulnerability. **(b)** Target model's mean logit-scaled confidence averaged across mid-stage checkpoints (left) and final-checkpoint confidence (right), plotted against μ_{out} from true shadow models. Solid lines show means over 10 equal-sized vulnerability bins within each subgroup.

4 From late memorization to OUT-model proxies

4.1 Atypicality implies privacy vulnerability

Feldman (2020) showed that natural data distributions exhibit a long-tailed structure, and that label memorization is necessary to achieve near-optimal generalization error. They define the memorization of a sample (x_i, y_i) as the change in the training algorithm's prediction due to its inclusion (see Appendix B.2 for a fuller discussion). Feldman & Zhang (2020) empirically demonstrated that atypical examples lying in the long tail consistently exhibit higher memorization values. Rewriting Feldman (2020)'s memorization in logit space yields

$$\widetilde{\text{mem}}(x_i, y_i) := \mathbb{E}_{h\sim\mathcal{A}(S)}[\varphi(h(x_i)_{y_i})] - \mathbb{E}_{h\sim\mathcal{A}(S^{\setminus i})}[\varphi(h(x_i)_{y_i})] = \mu_{\text{in}} - \mu_{\text{out}}, \tag{6}$$

so that the privacy score in Eq. 1 is this logit-space memorization normalized by $\sigma_{\text{in}} + \sigma_{\text{out}}$, yielding an effect-size (Cohen's-d-style) reformulation of memorization. Atypical samples therefore tend to receive high privacy scores, consistent with the empirical findings of Carlini et al. (2022a) that out-of-distribution samples carry higher privacy scores. In the terminology of Section 2, such samples tend to be *vulnerable*.

4.2 Mid-stage checkpoints approximate OUT models on vulnerable samples

Observation. Section 4.1 establishes that vulnerable samples are associated with atypical, memorized outliers, while Section 2 recalls that hard samples tend to be memorized in the later stages of training (Arpit et al., 2017; Mangalam & Prabhu, 2019; Jiang et al., 2021). Bridging these two threads, we hypothesize that vulnerable samples are likewise fit late in training. If so, mid-stage checkpoints of the target model should behave, on a vulnerable sample, closer to a model that has not seen it than to the final IN model. Figure 2a matches this prediction: LiRA against intermediate checkpoints recovers near-zero TPR@0.1%FPR on the top-30% vulnerable subset until the late stages of training, while the bottom-30% stays near the FPR floor throughout.

Approximation. Building on this observation, we posit that the output distribution at mid-stage checkpoints of the target model closely approximates the OUT-model output distribution on vulnerable samples. Figure 2b illustrates this: at mid-stage checkpoints, the target model's mean confidence (μ_{Ghost}) tracks μ_{out} across all vulnerability groups (left); by the final checkpoint, high- and mid-vulnerability samples have drifted well above the diagonal as they are progressively memorized, while low-vulnerability samples, which are fit early, remain on it (right). The approximation is necessarily imperfect. Memorization of vulnerable samples has already begun at the mid-stage, leaving a small residual bias. This offset, however, is substantially smaller than final-checkpoint drift, and mid-stage checkpoints can therefore serve as OUT-model proxies. A quantitative comparison of the Ghost proxy against the true OUT mean is reported in Appendix J.1.

5 GHOST-LIRA

Section 3 established that any MIA depending only on the target model's output is structurally limited, and escaping this regime has required training reference models. Section 4.2 showed that mid-stage checkpoints can serve as a proxy for this reference as a byproduct of standard training. We instantiate this observation as Ghost-LiRA, a framework that approximates the LiRA offline attack (Carlini et al., 2022a) without training any auxiliary models. The approximation is most effective on the late-memorized vulnerable tail identified in Section 4. On easy samples, which are fit early and exhibit little gap between intermediate and final checkpoints, Ghost-LiRA is not expected to recover the LiRA signal. These samples, however, are also the ones on which IN and OUT distributions largely overlap, since low-vulnerability points sit near the diagonal at both mid and final checkpoints (Figure 2b). They are therefore the samples that per-sample MIAs cannot attack in the first place (Figure 2a, bottom-30% curve).

Threat model. Ghost-LiRA requires only the pre-softmax logits on (x, y) from the final model and from at least one intermediate training checkpoint. The natural use case is privacy auditing and unlearning verification by model owners. Even with white-box access, white-box signals provide no membership information beyond the loss under a posterior model of training (Sablayrolles et al., 2019) and thus do not improve Ghost-LiRA (Appendix I). Ghost-LiRA can also be applied by external adversaries who either obtain a leaked intermediate checkpoint or, in its absence, train a *single* model under the standard assumption of per-sample MIAs and use its checkpoints as the OUT proxy (Appendix J.4).

5.1 THE GHOST-LIRA ALGORITHM

Setup. Let f denote a target model trained for T epochs on a dataset $\mathcal{D}_{\text{train}}$, and let $\mathcal{C} = \{t_1, \ldots, t_M\} \subseteq \{1, \ldots, T\}$ with $t_1 < \cdots < t_M = T$ denote the epochs at which the model owner saves checkpoints. For each $t \in \mathcal{C}$, let $z_t(x) \in \mathbb{R}^K$ be the pre-softmax logits and $\text{acc}_t^{\text{val}}$ the validation accuracy. We work with the logit-scaled confidence

$$\varphi_t(x, y) \;:=\; z_t(x)_y \;-\; \underset{y' \neq y}{\text{LogSumExp}}\; z_t(x)_{y'}, \tag{7}$$

which is numerically equivalent to the probability-space form $\phi(f(x)_y) = \log f(x)_y - \log \sum_{y' \neq y} f(x)_{y'}$ used in LiRA (Carlini et al., 2022a). Ghost-LiRA uses only $\{z_t(x)\}_{t \in \mathcal{C}}$ and $\{\text{acc}_t^{\text{val}}\}_{t \in \mathcal{C}}$—artifacts already produced by standard training—to return a per-sample membership score $s_{\text{Ghost}}(x, y)$. Full pseudocode is provided in Appendix E.

From ensemble average to temporal average. Recall the LiRA offline score. Given N shadow models trained on subsets excluding (x, y), LiRA fits a Gaussian $\mathcal{N}(\mu_{\text{out}}, \sigma_{\text{out}}^2)$ to their logit-scaled confidences and scores membership as

$$\Lambda(x, y) = \Phi\left(\frac{\varphi_T(x, y) - \mu_{\text{out}}}{\sigma_{\text{out}}}\right), \tag{8}$$

where Φ is the standard normal CDF. Motivated by the observation in Section 4 that vulnerable samples are fit late in training, we instead approximate μ_{out} by averaging φ across *checkpoints of the target model itself* at which the sample has not yet been memorized, replacing the shadow-model *ensemble average* with a single-trajectory *temporal average*.

Selecting the OUT window via validation accuracy. A natural question is how to identify checkpoints at which a sample is "not yet memorized." We do not know in advance which samples are memorized at which epoch, but we can use the model's aggregate learning progress as a proxy. Concretely, we select an *OUT window*

$$\mathcal{W}_{\text{out}} \;:=\; \left\{\, t \in \mathcal{C} \;:\; \alpha_{\text{lo}} \cdot \text{acc}_{\max}^{\text{val}} \;\leq\; \text{acc}_t^{\text{val}} \;\leq\; \alpha_{\text{hi}} \cdot \text{acc}_{\max}^{\text{val}} \,\right\}, \tag{9}$$

consisting of all saved checkpoints whose validation accuracy falls within a fraction $(\alpha_{\text{lo}}, \alpha_{\text{hi}})$ of the peak validation accuracy $\text{acc}_{\max}^{\text{val}} := \max_{t \in \mathcal{C}} \text{acc}_t^{\text{val}}$. This window is intended to capture the regime between generalization and memorization. In practice, for the image models, we use $(\alpha_{\text{lo}}, \alpha_{\text{hi}}) = (0.90, 0.97)$. For the language model, we use validation perplexity and set $\text{acc}_t^{\text{val}} := 1/\text{ppl}_t^{\text{val}}$ (Appendix F.3). Below 0.90 the model has not fully generalized, while above 0.97 the proxy drifts toward the IN model. This validation-accuracy-based selection is designed for model owners who

have full access to training metadata and yields the strongest performance. External adversaries are unlikely to know validation accuracies, but Ghost-LiRA does not strictly require them. Any intermediate checkpoint that precedes the late training phase suffices as an OUT proxy. Appendix J.3 shows that even a single such checkpoint recovers a substantial fraction of the full-window signal across a broad range of training stages. Beyond the specific choice of $(\alpha_{\mathrm{lo}}, \alpha_{\mathrm{hi}}) = (0.90, 0.97)$, Ghost-LiRA remains stable across a wide plateau of window placements (Appendix J.2).

Ghost-LiRA score. We then estimate

$$\mu_{\mathrm{Ghost}}(x, y) \;:=\; \frac{1}{|\mathcal{W}_{\mathrm{out}}|} \sum_{t \in \mathcal{W}_{\mathrm{out}}} \varphi_t(x, y), \tag{10}$$

and define the Ghost-LiRA membership score as

$$s_{\mathrm{Ghost}}(x, y) \;:=\; \varphi_T(x, y) \;-\; \mu_{\mathrm{Ghost}}(x, y). \tag{11}$$

A larger s_{Ghost} indicates that the final model's confidence on (x, y) has drifted upward from what the OUT-window checkpoints predicted—a signature of memorization—and hence that (x, y) is likely a member. Recalling Section 4.1, $s_{\mathrm{Ghost}}(x, y)$ can be interpreted as a single-trajectory estimator of the logit-space memorization $\widetilde{\mathrm{mem}}(x, y) = \mu_{\mathrm{in}} - \mu_{\mathrm{out}}$, with $\varphi_T(x, y)$ serving as a point estimate of μ_{in} (the final model is, by construction, a member model for (x, y)) and $\mu_{\mathrm{Ghost}}(x, y)$ as a temporal estimate of μ_{out}. Ghost-LiRA thus operationalizes the memorization quantity of Feldman (2020) as a directly computable per-sample score. s_{Ghost} can be shown to be a monotone transformation of the LiRA offline statistic under the global-variance variant of Carlini et al. (2022a), and thus inherits its ranking; we defer the derivation to Appendix E.

6 Experiments

6.1 Setup

Datasets and architectures. We evaluate our method on four datasets widely adopted in the membership inference literature: CIFAR-10, CIFAR-100 (Krizhevsky, 2009), Tiny ImageNet (Le & Yang, 2015), and WikiText-103 (Merity et al., 2017), paired respectively with ResNet-18, ResNet-34 (He et al., 2016), Swin-T (Liu et al., 2021), and GPT-2-Large (Radford et al., 2019). Image models are trained from scratch with randomly initialized weights and the language model is fine-tuned. We cover both step-decay and cosine annealing (Loshchilov & Hutter, 2017) schedules, and additionally include a warm-up and early stopping scenario for Swin-T, demonstrating that Ghost-LiRA remains stable across diverse training regimes. For every (dataset, architecture) pair, we train models under 10 independent random seeds, and all reported numbers correspond to the mean across these runs. Full experimental details are provided in Appendix F.

Baselines. We select two attacks from each family: Attack-P (Ye et al., 2022) and Modified Entropy (Song & Mittal, 2021) from the population-based family, and LiRA (Carlini et al., 2022a) and RMIA (Zarifzadeh et al., 2024) from the per-sample family. Attack-P and Modified Entropy both derive their membership signal solely from the target model's output (loss and modified entropy, respectively). Modified Entropy additionally trains a single shadow model, used only to calibrate per-class thresholds rather than to construct the signal itself. LiRA is a widely adopted baseline, and RMIA is currently regarded as the state-of-the-art among practical per-sample attacks; both require training multiple shadow models. For LiRA, we employ both the online and the more practical offline variant. For RMIA, we use the offline variant, which matches the offline structure of Ghost-LiRA and, per the original paper, performs comparably to its online counterpart. Both LiRA and RMIA are evaluated with 256 shadow models, serving as an upper bound on their performance. We further analyze the effect of varying this number in Figure 4. Full descriptions and configurations of all baselines are provided in Appendix C and F.2.

Shadow-model training. Modified Entropy, LiRA, and RMIA require shadow models for calibration, granting these baselines access to additional data drawn from the target's training distribution. Following standard protocol, we subsample 25k training examples from the full 50k pool for CIFAR-10 and CIFAR-100, and 50k from 100k for Tiny ImageNet. For WikiText-103, we first fix a 50k subset of the full training set and then randomly sample 25k examples from it for each run. Since Ghost-LiRA trains no shadow models, it does not benefit from this additional resource, making the comparison inherently unfavorable to our method.

Vulnerability-stratified evaluation protocol. Standard MIA evaluation aggregates performance over the full training set, concealing how attacks behave on privacy-critical samples. We propose a *vulnerability-stratified evaluation protocol* that partitions samples by vulnerability and reports MIA performance on each subgroup separately. Evaluation is conducted over the members and non-members of the target model, drawn from the same training pool. This protocol directly tests whether an MIA can detect the most at-risk members and exposes false positives on easy non-members that aggregate metrics mask.

6.2 Results

The vulnerability-stratified evaluation reveals that the two attack families exploit different signals, which aggregate evaluation masks entirely. Figure 3 presents ROC curves on CIFAR-100, evaluated on (a) the full training set, (b) the top-30% most vulnerable samples, and (c) the bottom-30% least vulnerable samples. Results on the remaining three benchmarks, as well as on the top/bottom-10% and -20% subgroups, are consistent with the trends described below and are reported in Appendix G.2.

Aggregated evaluation obscures the failure mode. On the full training set (Figure 3a), Ghost-LiRA substantially outperforms the population-based baselines (Attack-P, Modified Entropy) and closely tracks the per-sample MIAs (LiRA, RMIA) across most of the FPR range despite requiring no additional model training. Attack-P and Modified Entropy yield a weak but non-trivial signal in the high-FPR regime, consistent with prior reports. Modified Entropy outperforms Attack-P, reflecting its use of per-class thresholds calibrated via a shadow model.

Stratification exposes complementary—and inverted—signals. The stratified view reveals what the aggregate hides. On the top-30% most vulnerable samples (Figure 3b), per-sample attacks (LiRA, RMIA, Ghost-LiRA) lift above their full-population performance, while population-based attacks drop. Modified Entropy collapses to the random-guess diagonal, and Attack-P falls *below* it, directly exhibiting the signal inversion formalized in Section 3. On the bottom-30% least vulnerable samples (Figure 3c), the roles reverse. Per-sample attacks drop noticeably, while population-based attacks exceed their full-population performance, revealing that their apparent aggregate signal comes primarily from easy samples. Ghost-LiRA inherits the per-sample profile across both extremes without any additional model training. Aggregate evaluation hides this complementarity entirely, producing the misleadingly benign picture in Figure 3a.

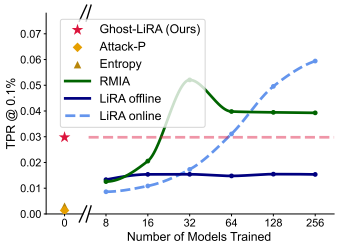
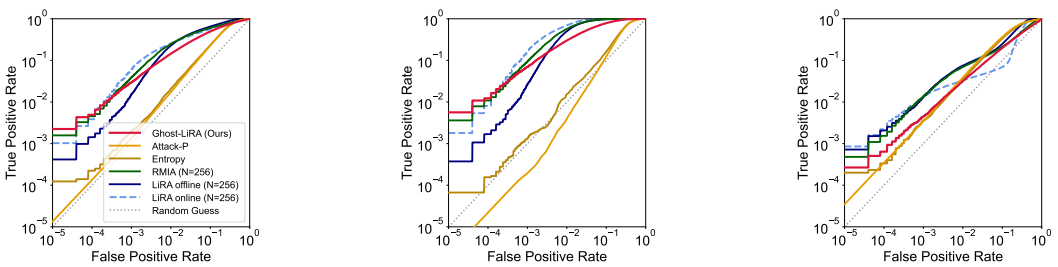


Figure 4: **Ghost-LiRA achieves shadow-model–level attack strength without additional training.**

Effect of the number of shadow models. Figure 4 reports TPR at 0.1% FPR on CIFAR-100, as a function of the number of shadow models. Other FPR thresholds and benchmarks yield consistent trends (Appendix G.3). Since Ghost-LiRA requires no additional training, it is plotted at x=0 as a fixed reference. Both LiRA and RMIA fall below Ghost-LiRA at low shadow-model counts and require tens of shadow models to surpass it. Even at this crossover point, Ghost-LiRA exceeds their performance at $\sim 1600\times$ lower computation cost on average (Table 5), making it a practical alternative under constrained training budgets.



(a) Overall (b) High vuln. (Top 30%) (c) Low vuln. (Bot 30%)

Figure 3: **ROC curves on CIFAR-100 (ResNet-34), evaluated on (a) the full training set, (b) the top-30% most vulnerable, and (c) the bottom-30% least vulnerable samples.** Population-based attacks collapse on the vulnerable subset (Figure 3b) and recover on the safe subset (Figure 3c), while per-sample attacks (including Ghost-LiRA) show the opposite pattern.

Table 1: **Aggregated evaluation conceals residual leakage on vulnerable samples.** CIFAR-10 (ResNet-18, 10 seeds, mean ± std). The forget set comprises 10% of the training set (half from top-5% most vulnerable, half uniformly random). *Aggregated*: full forget-set evaluation as in prior work; *High-Vulnerable*: top-5% most vulnerable samples only. Red marks signal inversion (Section 3); **bold** marks residual leakage hidden by aggregated evaluation. Additional algorithms in Appendix K.

			Aggregated (prior work)			High-Vulnerable (top 5%)		
State	Test Acc. (%)	MIA Attack	AUC	TPR@1%	TPR@0.1%	AUC	TPR@1%	TPR@0.1%
Original	92.66 ± 0.27	Loss	0.536 ± 0.006	1.14 ± 0.22	0.10 ± 0.04	0.441 ± 0.010	0.65 ± 0.39	0.03 ± 0.05
		LiRA online (N=256)	0.788 ± 0.002	46.09 ± 0.35	15.24 ± 2.13	0.987 ± 0.003	91.79 ± 0.66	30.36 ± 4.25
		Ghost-LiRA (Ours)	0.730 ± 0.004	15.43 ± 1.72	3.58 ± 0.95	0.936 ± 0.006	30.32 ± 3.54	7.15 ± 1.89
Retrained	92.11 ± 0.18	Loss	0.323 ± 0.003	0.63 ± 0.11	0.09 ± 0.07	0.051 ± 0.001	0.00 ± 0.00	0.00 ± 0.00
		LiRA online (N=256)	0.338 ± 0.005	0.52 ± 0.27	0.08 ± 0.05	0.113 ± 0.009	0.65 ± 0.26	0.09 ± 0.08
		Ghost-LiRA (Ours)	0.348 ± 0.004	0.15 ± 0.07	0.01 ± 0.03	0.198 ± 0.007	0.06 ± 0.05	0.00 ± 0.00
After SCRUB	91.52 ± 0.36	Loss	0.417 ± 0.007	0.42 ± 0.14	0.07 ± 0.04	0.227 ± 0.013	0.00 ± 0.00	0.00 ± 0.00
		LiRA online (N=256)	0.589 ± 0.016	6.56 ± 1.89	0.67 ± 0.18	$\mathbf{0.610} \pm 0.032$	$\mathbf{12.29} \pm 3.75$	$\mathbf{1.22} \pm 0.38$
		Ghost-LiRA (Ours)	0.597 ± 0.018	3.76 ± 0.88	0.82 ± 0.29	$\mathbf{0.676} \pm 0.037$	$\mathbf{6.67} \pm 1.90$	$\mathbf{1.59} \pm 0.58$

7 Application to Machine Unlearning Evaluation

Machine unlearning aims to remove the influence of a designated *forget set* $D_f \subset D$ from a trained model, ideally producing a model behaviorally indistinguishable from one trained from scratch on $D \setminus D_f$ (the *retrained reference*) (Cao & Yang, 2015; Bourtoule et al., 2021). Since exact retraining is prohibitive, inexact unlearning algorithms have been actively developed, with MIAs serving as a primary tool for evaluating their effectiveness. In practice, population-based MIAs that use only the unlearned model's output have been overwhelmingly dominant, with performance reported in aggregated metrics across all samples (Jia et al., 2023; Fan et al., 2024). Both conventions are inadequate given the failure mode formalized in Section 3 and observed empirically in Section 6.

U-LiRA. Hayes et al. (2025) observe that population-based attacks, which do not account for per-sample difficulty, can yield a false sense of privacy in unlearning evaluation. They propose U-LiRA as a more rigorous per-sample alternative, but its cost exceeds even standard LiRA, precluding its use within the hyperparameter search loop. In practice, unlearning evaluations still rely on population-based MIAs—tacitly assuming that their signal, while weak, is not misleading. Ghost-LiRA provides per-sample evaluation at negligible marginal cost, removing this trade-off.[2]

Setup. We unlearn on CIFAR-10 with SCRUB (Kurmanji et al., 2023), one of the few unlearning algorithms that also reports LiRA-based evaluation. The forget set comprises 10% of the training set, half drawn from the top-5% most vulnerable samples and half drawn uniformly at random. This composition is designed to expose how aggregate metrics dilute leakage on vulnerable samples. We evaluate using Loss attack (Yeom et al., 2018) (the canonical population-based MIA) alongside LiRA online and Ghost-LiRA.

Finding. Table 1 exposes the failure mode of Section 3 at two stages. *On the original model*, Loss attack ranks vulnerable members *below* typical non-members (AUC $0.441 < 0.5$, TPR@1%FPR $= 0.65\%$), while the aggregated AUC (0.536) appears benign as typical samples dominate the average. Per-sample MIAs do the opposite, with a *stronger* vulnerable-subgroup signal than in aggregate. *After SCRUB*, aggregated Loss falls well below random (AUC 0.417), which would conventionally suggest successful unlearning, yet Ghost-LiRA reveals order-of-magnitude residual leakage on the vulnerable subgroup (6.67% vs. 0.06% TPR@1%FPR). SCRUB's test accuracy (91.52%) sits slightly below the retrained reference (92.11%), confirming sufficient unlearning pressure.

8 Conclusion

We showed that population-based MIAs can produce an inverted signal on vulnerable samples, and proposed a vulnerability-stratified protocol to surface this failure. Since vulnerable samples tend to be memorized late, mid-stage checkpoints serve as OUT-model proxies, enabling Ghost-LiRA to approximate per-sample MIA from a single trajectory. Applied to unlearning, it reveals residual leakage that population-based MIAs miss.

[2]Ghost-LiRA and U-LiRA target different questions—whether (x, y) was a training member, vs. whether it was in the forget set given that unlearning was applied—and we accordingly do not benchmark against U-LiRA. See Appendix B.3.

AI USE STATEMENT

In this work, we used generative AI tools to assist with translation, suggest and refine notation used in mathematical proofs, implement baseline methods, and create or format scientific figures and tables. The proofs themselves were written and independently reviewed by the authors; AI assistance was limited to suggestions for notation, from which the authors selected the final notation. AI-generated code was used to implement baseline methods and was reviewed and tested for correctness by two authors. During manuscript preparation, the authors also used generative AI tools to suggest alternative wording and phrasing; the authors wrote, reviewed, and finalized the manuscript and selected all final wording.

All AI-assisted outputs were reviewed by the authors. The authors take responsibility for the final content of this work, including its mathematical claims and proofs, methodology, experimental results, figures, tables, and written text.

ETHICS STATEMENT

We use only publicly available benchmark datasets and do not involve human subjects or personally identifiable information.

Our work enables low-cost privacy auditing and evaluation of machine unlearning, with potential benefits for privacy-aware model development. However, the training-free design may also lower the barrier to membership inference against models with publicly available or leaked checkpoints, as discussed in Section 5.1 and Appendix J.3 and J.4.

We release no new high-risk datasets or models. The authors take responsibility for the integrity and responsible release of the research materials.

REPRODUCIBILITY STATEMENT

We provide sufficient details to reproduce our experimental results. The experimental setup, including datasets, architectures, shadow-model training protocols, and baselines, is described in Section 6. Full experimental details are provided in Appendix F. The proposed Ghost-LiRA method is described in Section 5.1, with the full algorithm provided in Appendix E. All experiments use standard public benchmark datasets, and code and precomputed privacy scores will be released upon publication.

REFERENCES

Devansh Arpit, Stanislaw Jastrzebski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S. Kanwal, Tegan Maharaj, Asja Fischer, Aaron C. Courville, Yoshua Bengio, and Simon Lacoste-Julien. A closer look at memorization in deep networks. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning, ICML*, Proceedings of Machine Learning Research, pp. 233–242. PMLR, 2017.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. Pythia: A suite for analyzing large language models across training and scaling. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *International Conference on Machine Learning, ICML*, Proceedings of Machine Learning Research, pp. 2397–2430. PMLR, 2023.

Lucas Bourtoule, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *42nd IEEE Symposium on Security and Privacy, SP*, pp. 141–159. IEEE, 2021.

Yinzhi Cao and Junfeng Yang. Towards making systems forget with machine unlearning. In *2015 IEEE Symposium on Security and Privacy, SP*, pp. 463–480. IEEE Computer Society, 2015.

Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. The secret sharer: Evaluating and testing unintended memorization in neural networks. In Nadia Heninger and Patrick Traynor (eds.), *28th USENIX Security Symposium, USENIX*, pp. 267–284. USENIX Association, 2019.

Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramèr. Membership inference attacks from first principles. In *43rd IEEE Symposium on Security and Privacy, SP*, pp. 1897–1914. IEEE, 2022a.

Nicholas Carlini, Matthew Jagielski, Chiyuan Zhang, Nicolas Papernot, Andreas Terzis, and Florian Tramèr. The privacy onion effect: Memorization is relative. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022*, 2022b.

Jingpu Cheng, Ping Liu, Qianxiao Li, and Chi Zhang. Machine unlearning under retain–forget entanglement. In *The Fourteenth International Conference on Learning Representations*, 2026.

Christopher A. Choquette-Choo, Florian Tramèr, Nicholas Carlini, and Nicolas Papernot. Label-only membership inference attacks. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML*, Proceedings of Machine Learning Research, pp. 1964–1974. PMLR, 2021.

Vikram S. Chundawat, Ayush K. Tarun, Murari Mandal, and Mohan S. Kankanhalli. Can Bad Teaching Induce Forgetting? Unlearning in Deep Networks Using an Incompetent Teacher. In Brian Williams, Yiling Chen, and Jennifer Neville (eds.), *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI*, pp. 7210–7217. AAAI Press, 2023.

Gilad Cohen and Raja Giryes. Membership inference attack using self influence functions. In *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 4880–4889. IEEE, 2024.

Jacob Cohen. *Statistical power analysis for the behavioral sciences*. Routledge, 2013.

Ekin Dogus Cubuk, Barret Zoph, Jonathon Shlens, and Quoc Le. RandAugment: Practical Automated Data Augmentation with a Reduced Search Space. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems*, 2020.

Terrance DeVries and Graham W. Taylor. Improved regularization of convolutional neural networks with cutout. *CoRR*, 2017. URL http://arxiv.org/abs/1708.04552.

Yuntao Du, Yuetian Chen, Hanshen Xiao, Bruno Ribeiro, and Ninghui Li. Imitative membership inference attack. *CoRR*, abs/2509.06796, 2025. doi: 10.48550/ARXIV.2509.06796.

Yuntao Du, Jiacheng Li, Yuetian Chen, Kaiyuan Zhang, Zhizhen Yuan, Hanshen Xiao, Bruno Ribeiro, and Ninghui Li. Cascading and proxy membership inference attacks. In *Proceedings of the Network and Distributed System Security (NDSS) Symposium*, 2026.

Jinhao Duan, Fei Kong, Shiqi Wang, Xiaoshuang Shi, and Kaidi Xu. Are diffusion models vulnerable to membership inference attacks? In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *International Conference on Machine Learning, ICML*, Proceedings of Machine Learning Research, pp. 8717–8730. PMLR, 2023.

Chongyu Fan, Jiancheng Liu, Yihua Zhang, Eric Wong, Dennis Wei, and Sijia Liu. SalUn: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation. In *The Twelfth International Conference on Learning Representations, ICLR*, 2024.

Vitaly Feldman. Does learning require memorization? A short tale about a long tail. In Konstantin Makarychev, Yury Makarychev, Madhur Tulsiani, Gautam Kamath, and Julia Chuzhoy (eds.), *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC*, pp. 954–959. ACM, 2020.

Vitaly Feldman and Chiyuan Zhang. What neural networks memorize and why: Discovering the long tail via influence estimation. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems*, 2020.

Aditya Golatkar, Alessandro Achille, and Stefano Soatto. Forgetting outside the box: Scrubbing deep networks of information accessible from input-output observations. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (eds.), *Computer Vision - ECCV 2020 - 16th European Conference*, pp. 383–398. Springer, 2020.

Aditya Golatkar, Alessandro Achille, Avinash Ravichandran, Marzia Polito, and Stefano Soatto. Mixed-privacy forgetting in deep networks. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pp. 792–801. Computer Vision Foundation / IEEE, 2021.

Jamie Hayes, Ilia Shumailov, Eleni Triantafillou, Amr Khalifa, and Nicolas Papernot. Inexact unlearning needs more careful evaluations to avoid a false sense of privacy. In *IEEE Conference on Secure and Trustworthy Machine Learning, SaTML*, pp. 497–519. IEEE, 2025.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pp. 770–778. IEEE Computer Society, 2016.

Jinghan Jia, Jiancheng Liu, Parikshit Ram, Yuguang Yao, Gaowen Liu, Yang Liu, Pranay Sharma, and Sijia Liu. Model sparsity can simplify machine unlearning. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems*, 2023.

Ziheng Jiang, Chiyuan Zhang, Kunal Talwar, and Michael C. Mozer. Characterizing structural regularities of labeled data in overparameterized models. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML*, Proceedings of Machine Learning Research, pp. 5034–5044. PMLR, 2021.

Hoki Kim, Keonwoo Kim, Sungwon Chae, and Sangwon Yoon. Unlearning-aware minimization. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Adriano Koshiyama, Emre Kazim, Philip Treleaven, Pete Rai, Lukasz Szpruch, Giles Pavey, Ghazi Ahamat, Franziska Leutner, Randy Goebel, Andrew Knight, et al. Towards algorithm auditing: managing legal, ethical and technological risks of ai, ml and associated algorithms. *Royal Society Open Science*, 11(5), 2024.

Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

Meghdad Kurmanji, Peter Triantafillou, Jamie Hayes, and Eleni Triantafillou. Towards unbounded machine unlearning. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems*, 2023.

Ya Le and Xuan S. Yang. Tiny ImageNet Visual Recognition Challenge. 2015. URL https://cs231n.stanford.edu/reports/2015/pdfs/yle_project.pdf.

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

Hao Li, Zheng Li, Siyuan Wu, Chengrui Hu, Yutong Ye, Min Zhang, Dengguo Feng, and Yang Zhang. SeqMIA: Sequential-Metric Based Membership Inference Attack. In Bo Luo, Xiaojing Liao, Jun Xu, Engin Kirda, and David Lie (eds.), *Proceedings of the 2024 ACM SIGSAC Conference on Computer and Communications Security*, pp. 3496–3510. ACM, 2024.

Yiyong Liu, Zhengyu Zhao, Michael Backes, and Yang Zhang. Membership inference attacks by exploiting loss trajectory. In Heng Yin, Angelos Stavrou, Cas Cremers, and Elaine Shi (eds.), *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security, CCS*, pp. 2085–2098. ACM, 2022.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV*, pp. 9992–10002. IEEE, 2021.

Yunhui Long, Lei Wang, Diyue Bu, Vincent Bindschaedler, XiaoFeng Wang, Haixu Tang, Carl A. Gunter, and Kai Chen. A pragmatic approach to membership inferences on machine learning models. In *IEEE European Symposium on Security and Privacy, EuroS&P*, pp. 521–534. IEEE, 2020.

Ilya Loshchilov and Frank Hutter. SGDR: stochastic gradient descent with warm restarts. In *5th International Conference on Learning Representations, ICLR*, 2017.

Karttikeya Mangalam and Vinay Uday Prabhu. Do deep neural networks learn shallow learnable examples first? In *ICML 2019 Workshop on Identifying and Understanding Deep Learning Phenomena*, 2019.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations (ICLR)*, 2017.

Fatemehsadat Mireshghallah, Kartik Goyal, Archit Uniyal, Taylor Berg-Kirkpatrick, and Reza Shokri. Quantifying privacy risks of masked language models using membership inference attacks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 8332–8347, 2022.

Sasi Kumar Murakonda and Reza Shokri. ML privacy meter: Aiding regulatory compliance by quantifying the privacy risks of machine learning. *CoRR*, abs/2007.09339, 2020. URL https://arxiv.org/abs/2007.09339.

Milad Nasr, Reza Shokri, and Amir Houmansadr. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *2019 IEEE symposium on security and privacy (SP)*, pp. 739–753. IEEE, 2019.

Milad Nasr, Shuang Song, Abhradeep Thakurta, Nicolas Papernot, and Nicholas Carlini. Adversary instantiation: Lower bounds for differentially private machine learning. In *42nd IEEE Symposium on Security and Privacy, SP*, pp. 866–882. IEEE, 2021.

Joseph Pollock, Igor Shilov, Euodia Dodd, and Yves-Alexandre de Montjoye. Free record-level privacy risk evaluation through artifact-based methods. In Lujo Bauer and Giancarlo Pellegrino (eds.), *34th USENIX Security Symposium, USENIX Security*, pp. 5525–5544. USENIX Association, 2025.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, Yann Ollivier, and Hervé Jégou. White-box vs black-box: Bayes optimal strategies for membership inference. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML*, Proceedings of Machine Learning Research, pp. 5558–5567. PMLR, 2019.

Ahmed Salem, Yang Zhang, Mathias Humbert, Pascal Berrang, Mario Fritz, and Michael Backes. ML-Leaks: Model and Data Independent Membership Inference Attacks and Defenses on Machine Learning Models. In *26th Annual Network and Distributed System Security Symposium, NDSS*. The Internet Society, 2019.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy, SP*, pp. 3–18. IEEE Computer Society, 2017.

Liwei Song and Prateek Mittal. Systematic evaluation of privacy risks of machine learning models. In Michael D. Bailey and Rachel Greenstadt (eds.), *30th USENIX Security Symposium, USENIX Security*, pp. 2615–2632. USENIX Association, 2021.

Liwei Song, Reza Shokri, and Prateek Mittal. Privacy risks of securing machine learning models against adversarial examples. In Lorenzo Cavallaro, Johannes Kinder, XiaoFeng Wang, and Jonathan Katz (eds.), *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, CCS*, pp. 241–257. ACM, 2019.

Shuang Song and David Marn. Introducing a new privacy testing library in TensorFlow. `https://blog.tensorflow.org/2020/06/introducing-new-privacy-testing-library.html`, 2020.

Christoforos N. Spartalis, Theodoros Semertzidis, Efstratios Gavves, and Petros Daras. LoTUS: Large-Scale Machine Unlearning with a Taste of Uncertainty. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pp. 10046–10055. Computer Vision Foundation / IEEE, 2025.

Thomas Steinke, Milad Nasr, and Matthew Jagielski. Privacy auditing with one (1) training run. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems*, 2023.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pp. 2818–2826. IEEE Computer Society, 2016.

Eleni Triantafillou, Peter Kairouz, Fabian Pedregosa, Jamie Hayes, Meghdad Kurmanji, Kairan Zhao, Vincent Dumoulin, Júlio C. S. Jacques Júnior, Ioannis Mitliagkas, Jun Wan, Lisheng Sun-Hosoya, Sergio Escalera, Gintare Karolina Dziugaite, Peter Triantafillou, and Isabelle Guyon. Are we making progress in unlearning? Findings from the first NeurIPS unlearning competition. *CoRR*, abs/2406.09073, 2024.

Lauren Watson, Chuan Guo, Graham Cormode, and Alexandre Sablayrolles. On the importance of difficulty calibration in membership inference attacks. In *The Tenth International Conference on Learning Representations, ICLR*, 2022.

Jiayuan Ye, Aadyaa Maddi, Sasi Kumar Murakonda, Vincent Bindschaedler, and Reza Shokri. Enhanced membership inference attacks against machine learning models. In Heng Yin, Angelos Stavrou, Cas Cremers, and Elaine Shi (eds.), *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security, CCS*, pp. 3093–3106. ACM, 2022.

Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy risk in machine learning: Analyzing the connection to overfitting. In *31st IEEE Computer Security Foundations Symposium, CSF*, pp. 268–282. IEEE Computer Society, 2018.

Sangdoo Yun, Dongyoon Han, Sanghyuk Chun, Seong Joon Oh, Youngjoon Yoo, and Junsuk Choe. CutMix: Regularization Strategy to Train Strong Classifiers With Localizable Features. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV*, pp. 6022–6031. IEEE, 2019.

Sajjad Zarifzadeh, Philippe Liu, and Reza Shokri. Low-cost high-power membership inference attacks. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Forty-first International Conference on Machine Learning, ICML*, Proceedings of Machine Learning Research, pp. 58244–58282. PMLR, 2024.

Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018.

Jingyang Zhang, Jingwei Sun, Eric Yeats, Yang Ouyang, Martin Kuo, Jianyi Zhang, Hao Yang, and Hai Li. Min-K%++: Improved baseline for pre-training data detection from large language models. In *International Conference on Learning Representations*, volume 2025, pp. 64845–64862, 2025.

Weichao Zhang, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Yixing Fan, and Xueqi Cheng. Pretraining data detection for large language models: A divergence-based calibration method. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 5263–5274, 2024.

Kairan Zhao, Meghdad Kurmanji, George-Octavian Barbulescu, Eleni Triantafillou, and Peter Triantafillou. What makes unlearning hard and what to do about it. In Amir Globerson, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems*, 2024.

Yu Zhou, Dian Zheng, Qijie Mo, Renjie Lu, Kun-Yu Lin, and Wei-Shi Zheng. Decoupled distillation to erase: A general unlearning method for any class-centric tasks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pp. 20350–20359. Computer Vision Foundation / IEEE, 2025.

APPENDICES

The appendices are organized as follows:

- Appendix A: Limitations and future work
- Appendix B: Related works
- Appendix C: Detailed descriptions of baseline MIAs
- Appendix D: Extended formal and empirical analysis
- Appendix E: Ghost-LiRA algorithm details
- Appendix F: Experimental details
- Appendix G: Additional results across benchmarks
- Appendix H: Comparison with SeqMIA
- Appendix I: Comparison with white-box signals
- Appendix J: Robustness and design space of the OUT proxy
- Appendix K: Hidden leakage across unlearning algorithms

A LIMITATIONS AND FUTURE WORK

Our work has several limitations:

(i) Ghost-LiRA requires intermediate checkpoints of the target model to operate without additional training (Section 5). While a single checkpoint suffices (Appendix J.3), this makes the zero-cost setting suited to self-auditing by model owners or evaluating the success of unlearning, where intermediate checkpoints are available. When they are not, an adversary can train one model of its own and use its checkpoints instead (Appendix J.4), at the cost of one training run and under the data and recipe assumptions of LiRA and RMIA. Reducing this cost further is left for future work.

(ii) Near-convergence checkpoints from a single trajectory can be correlated, which complicates their use as independent IN-model samples. Within the current framework, we do not yet have a principled way to account for this correlation while retaining the single-trajectory construction, and developing a principled treatment of checkpoint dependence is an interesting direction for future work. We therefore adopt the global-variance variant of LiRA offline (Appendix E). Consequently, Ghost-LiRA is designed to capture $\mu_{\text{in}} - \mu_{\text{out}}$ in expectation, while extending the framework to more accurately characterize the per-sample privacy score d (Eq. 1) is left for future work.

(iii) Within the OUT-only regime, mid-stage checkpoints can provide only approximate proxies. Memorization of vulnerable samples has already begun at these stages (Section 4.2), so μ_{Ghost} may slightly overestimate the true μ_{out}, attenuating the Ghost-LiRA score. This bias is small relative to the final-checkpoint drift (Figure 2b), and further improving the temporal proxy is left for future work.

(iv) Our experiments cover models up to GPT-2-Large (774M parameters), spanning both from-scratch training and fine-tuning from a pretrained initialization. Whether Ghost-LiRA remains effective at the billion-parameter scale is not yet verified. The behavior of the temporal proxy at larger scales may differ from that observed in our experiments, which we leave for future work to investigate.

B RELATED WORKS

B.1 OTHER MEMBERSHIP INFERENCE ATTACKS

Section 2 divides MIAs into two families: *population-based* attacks, whose membership score is derived solely from the target model's final output, and *per-sample* attacks, which calibrate per-sample decisions via auxiliary reference models. The baselines compared in Section 6—Attack-P (Ye et al., 2022), Modified Entropy (Song & Mittal, 2021), LiRA (Carlini et al., 2022a), RMIA (Zarifzadeh et al., 2024)—are canonical instances of these two families. Here we discuss MIAs that either lie outside this clean dichotomy or were not covered in the main text, and clarify how Ghost-LiRA is positioned relative to each.

Trajectory-based MIAs. TrajectoryMIA (Liu et al., 2022) extracts the membership signal from the loss trajectory of intermediate distilled models, obtained via knowledge distillation of both the target and a shadow model, combined with the target model's final loss. A single auxiliary classifier (a 4-layer MLP), trained on features from a shadow model, is shared across all target samples. SeqMIA (Li et al., 2024) extends this line of work by broadening the trajectory signal from loss alone to multiple metrics (e.g., loss, max-confidence, modified entropy) and replacing the MLP with an attention-based RNN that processes the snapshot sequence in temporal order. The shared-classifier design itself is retained.[3]

B.2 Memorization and privacy vulnerability

Memorization. Feldman (2020) theoretically shows that memorization of labels is necessary for achieving close-to-optimal generalization error on natural data distributions that exhibit long-tailed structure, arguing that such memorization is not an artifact of overfitting but a fundamental requirement for near-optimal generalization. Feldman (2020) formally defines sample-level memorization as

$$\mathrm{mem}(\mathcal{A}, S, i) := \Pr_{h\sim\mathcal{A}(S)}[h(x_i) = y_i] - \Pr_{h\sim\mathcal{A}(S^{\setminus i})}[h(x_i) = y_i], \tag{12}$$

where $S^{\setminus i}$ denotes the dataset S with the i-th example (x_i, y_i) removed. This quantity captures the extent to which the algorithm's prediction on x_i depends on the inclusion of (x_i, y_i) in the training set: a high memorization value indicates that the model relies on memorization rather than generalization from other examples to correctly predict the label of x_i. Feldman & Zhang (2020) empirically validate this framework and demonstrate that atypical examples residing in the long tail of the data distribution consistently exhibit higher memorization values.

From memorization to privacy scores. Carlini et al. (2022a), influenced by this notion of memorization, designed the LiRA framework (see Section 2) and extended it into the *privacy score* used in the main text:

$$d = \frac{|\mu_{\mathrm{in}} - \mu_{\mathrm{out}}|}{\sigma_{\mathrm{in}} + \sigma_{\mathrm{out}}}. \tag{13}$$

Unlike the memorization score of Feldman & Zhang (2020), which relies on prediction probabilities, the privacy score operates on logit-scaled confidences and normalizes the separation between the IN and OUT distributions by their combined spread. A high privacy score indicates that the model's behavior changes substantially depending on whether the sample is included in training, signaling greater privacy leakage.

Relativity of privacy vulnerability. Carlini et al. (2022b) empirically demonstrated that a sample's vulnerability to privacy attacks such as membership inference is not a fixed property but a relative one that depends on the composition of the training data. Specifically, they used LiRA (Carlini et al., 2022a) to measure each sample's susceptibility and removed the most vulnerable samples, which were observed to match the atypical outliers identified by Feldman & Zhang (2020). After retraining the model on the remaining data, they found that samples previously considered safe became newly vulnerable to the same attack, a phenomenon they termed the *onion effect*. They further verified that this effect is not explained by statistical noise in the privacy metric, reduction in dataset size, limited model capacity, or the presence of duplicate training examples. These findings suggest that the existence of vulnerable samples in a training set is a persistent phenomenon rather than an artifact of a particular data composition.

[3]Both attacks thus occupy a grey area between the population-based and per-sample families defined in Section 2. These attacks share with Ghost-LiRA the high-level intuition that intermediate training stages carry membership information beyond the final output, but differ from it on two axes: (i) they require *additional model training*, namely an auxiliary shadow model plus a sequence of distilled checkpoints, whereas Ghost-LiRA reuses the *target model's own checkpoints* and trains no auxiliary model; and (ii) they aggregate the intermediate-state signal through a single shared classifier, whereas Ghost-LiRA performs *per-sample calibration* in the LiRA family's sense. TrajectoryMIA and SeqMIA also adopt a five-way data partition (target train/test, shadow train/test, and a dedicated knowledge-distillation pool) that is not directly compatible with the 50:50 split used by the LiRA-family attacks in our main benchmark. We therefore conduct a dedicated head-to-head evaluation of SeqMIA against Ghost-LiRA under a matched auxiliary budget in Appendix H.

B.3 MACHINE UNLEARNING EVALUATION

Population-based U-MIA. Membership inference attacks (MIAs) were originally introduced to determine whether a specific data point was included in the training set of a target model (Shokri et al., 2017). In the context of unlearning evaluation, MIAs serve to verify whether the influence of the forget set has been removed, by testing whether the unlearned model can distinguish forget-set samples from unseen test-set samples. A common approach (Jia et al., 2023; Song & Mittal, 2021) trains a binary classifier on prediction-confidence-based features from the unlearned model, labeling the retain set $\mathcal{D}_r$ as members and the test set $\mathcal{D}_t$ as non-members, and then applies the learned predictor to the forget set $\mathcal{D}_f$. Results are reported either as the fraction of $\mathcal{D}_f$ classified as non-members (MIA-Efficacy) (Jia et al., 2023; Fan et al., 2024) or as the fraction still identified as members (attack success rate) (Chundawat et al., 2023; Golatkar et al., 2020; 2021). Both conventions rely on similar features such as prediction confidence, loss, or entropy, and apply a single, shared (or sometimes per-class) decision rule across all target examples. Hayes et al. (2025) define this family of attacks as *population U-MIAs* and argue that, because individual samples differ substantially in learning difficulty and in the distribution of their outputs, such population-level attacks are inherently weak, causing prior unlearning evaluations to convey a *false sense of privacy*. We further note that population U-MIAs rely solely on the outputs of the target (unlearned) model, and therefore also fall under the population-based category as defined in Section 2.

U-LiRA. As an alternative, Hayes et al. (2025) adapt the Likelihood Ratio Attack (LiRA; Carlini et al., 2022a) to the unlearning setting, yielding *U-LiRA*. Standard LiRA distinguishes whether a point z was included in training by comparing two shadow distributions: models trained with z versus models trained without z. U-LiRA modifies the former by additionally applying the unlearning algorithm U to z, so that the comparison becomes models that trained on z and then unlearned it versus models that never trained on z at all. That is, U-LiRA does not ask whether z was a member of the target model's training set, but whether the unlearned model's behavior on z is distinguishable from that of a model retrained without z. This formulation allows U-LiRA to detect not only insufficient forgetting but also *over-forgetting*, where the model's output on z deviates from the retrained distribution in the opposite direction (e.g., by assigning an anomalously high loss). Crucially, U-LiRA is a *per-sample* attack. A separate pair of shadow distributions and decision rule is constructed for each target sample, directly addressing the population-level limitation identified above.

Two limitations of U-LiRA are worth noting. First, U-LiRA inherits the shadow-model training procedure of standard LiRA but additionally requires running the unlearning algorithm on each shadow model, making it more expensive than LiRA itself. Second, U-LiRA assumes that the unlearning algorithm and its hyperparameters are already fixed before evaluation begins, yet tuning those hyperparameters itself requires a cheaper MIA as a proxy, creating a circular dependency. More broadly, when the cost of evaluating an inexact unlearning procedure approaches or exceeds that of retraining from scratch, the practical motivation for inexact unlearning itself is called into question (Triantafillou et al., 2024). U-LiRA therefore remains a rigorous *post hoc* auditing tool for assessing a given unlearning algorithm, but may be difficult to deploy in practical settings where evaluation must be repeated across many configurations.[4]

Memorization-aware unlearning analysis. Zhao et al. (2024) measure how strongly the original model memorizes each training sample using the memorization score of Feldman (2020), and partition the forget set into low-, medium-, and high-memorization groups to compare the performance of approximate unlearning algorithms across them. Building on this analysis, they propose the RUM framework, which divides the forget set into subsets by memorization level and applies a separate unlearning algorithm to each subset. While this is an early study that stratifies unlearning evaluation by sample-level characteristics, two limitations remain. First, their forgetting-quality metric, the binary-classifier MIA of Jia et al. (2023), falls under the population U-MIA category of Hayes et al. (2025). In addition, such an attack relies solely on the output of the target (unlearned) model; it measures the model's prediction difficulty on a target sample rather than its membership (Section 3). Second,

[4]We do not directly benchmark against U-LiRA in our experiments. The two attacks are complementary rather than competing. U-LiRA audits whether the unlearning procedure itself was effective by comparing the unlearned model against a retrained model, while our attack audits whether membership traces of z persist after unlearning.

Feldman's memorization score is defined as the difference between the in- and out-distribution means and therefore does not account for the per-sample variance of these distributions. Two samples with identical memorization scores can thus carry different levels of membership leakage risk, so a memorization-based partition is not guaranteed to align with sample-level vulnerability.

C Detailed descriptions of baseline MIAs

C.1 Attack-P

The most fundamental approach to membership inference attacks is the loss-based attack (Yeom et al., 2018). This method relies on the observation that a target model θ, trained on a private dataset D, tends to exhibit low loss on data used during training and relatively high loss on data not seen during training. Specifically, the loss $\ell(\theta, x_z, y_z)$ of a target record $z = (x_z, y_z)$ is compared against a constant threshold c, and the record is classified as a member if its loss falls below c:

$$\text{If } \ell(\theta, x_z, y_z) \leq c, \quad \text{predict } z \in D. \tag{14}$$

However, this approach does not provide a systematic method for determining a threshold that corresponds to a desired false positive rate requirement α prior to evaluation; instead, it resorts to sweeping over all possible threshold values and evaluating the resulting FPR and TPR post hoc (Ye et al., 2022).

Attack-P (Ye et al., 2022) addresses this limitation by replacing the constant threshold with a model-dependent threshold $c_\alpha(\theta)$. Population data $P^\theta = \{z_i\}_{i=1,\ldots,m}$, sampled i.i.d. from the data distribution π, is queried on the target model θ to estimate the loss distribution of non-members. The threshold is then set to the α-percentile of this distribution:

$$\frac{|\{z \in P^\theta : \ell(\theta, x_z, y_z) \leq c_\alpha(\theta)\}|}{|P^\theta|} = \alpha. \tag{15}$$

This formulation enables Attack-P to approximately guarantee a desired FPR level α prior to observing any evaluation data. Furthermore, since the attack requires only black-box query access to the target model and population data—without the need to train any shadow models—it incurs substantially lower computational cost.

C.2 Modified entropy

Song & Mittal (2021) first point out a limitation of prior prediction-entropy-based attacks. These attacks build on the observation that a model tends to produce more confident predictions on its training samples, leading to a smaller entropy $H(F(x)) = -\sum_i F(x)_i \log F(x)_i$. However, since entropy does not incorporate the ground-truth label y, a correct prediction with probability 1 and a completely incorrect prediction with probability 1 both yield an entropy of zero and thus become indistinguishable. To address this issue, the authors propose a *modified prediction entropy* that incorporates label information:

$$\mathrm{Mentr}(F(x), y) = -(1 - F(x)_y) \log F(x)_y - \sum_{i \neq y} F(x)_i \log\big(1 - F(x)_i\big). \tag{16}$$

This metric is monotonically decreasing in the correct-class probability $F(x)_y$ and monotonically increasing in each incorrect-class probability, so that a correct prediction with probability 1 yields a value of 0 while a wrong prediction with probability 1 yields infinity. In this way, prediction uncertainty is captured consistently with respect to the ground-truth label.

To perform the attack, the authors train a shadow model that mimics the behavior of the target model, and use its predictions on known member and non-member samples to calibrate the decision rule. Rather than applying a single global threshold, they set a class-specific threshold τ_y for each class y, chosen to best separate members from non-members among shadow samples of that class. An input (x, y) is then inferred as a member if $\mathrm{Mentr}(F(x), y) \leq \tau_y$. This class-wise calibration accounts for the fact that models exhibit different prediction confidence levels across classes, particularly under class imbalance, and is reported to improve balanced attack accuracy over a class-independent threshold.

C.3 LIKELIHOOD-RATIO ATTACK (LIRA)

Carlini et al. (2022a) introduced the Likelihood Ratio Attack (LiRA), which trains multiple shadow models under two conditions—either including (IN) or excluding (OUT) the target sample—and performs a likelihood ratio test to determine membership. Formally, given a target model f and a sample (x, y), LiRA performs a hypothesis test based on

$$\Lambda(f; x, y) = \frac{p(f \mid \mathbb{Q}_{\text{in}}(x, y))}{p(f \mid \mathbb{Q}_{\text{out}}(x, y))}, \tag{17}$$

where $\mathbb{Q}_{\text{in}}(x, y)$ and $\mathbb{Q}_{\text{out}}(x, y)$ denote the distributions of models trained on datasets that include or exclude (x, y), respectively. In practice, LiRA trains N shadow models on random subsets of the data, applies the logit scaling $\phi(f(x)_y) = \log(f(x)_y) - \log(\sum_{y' \neq y} f(x)_{y'})$ to obtain approximately Gaussian-distributed scores, and fits separate Gaussians to the IN and OUT shadow-model scores for each sample.

Carlini et al. (2022a) also proposed an *offline* variant (used in the main text) that estimates only the OUT distribution, avoiding the need to train shadow models on the target sample. This reduces the two-sided likelihood ratio test to the one-sided form given in Section 2.

C.4 ROBUST MEMBERSHIP INFERENCE ATTACK (RMIA)

A prominent per-sample baseline, LiRA (Carlini et al., 2022a), estimates the likelihood ratio $\Pr(\theta \mid x)/\Pr(\theta \mid \bar{x})$ by comparing the output distributions of reference models trained with x (IN models) and without x (OUT models). Because LiRA summarizes the null hypothesis as a single aggregate distribution, its test power degrades substantially when the number of available reference models is limited (Zarifzadeh et al., 2024).

RMIA (Zarifzadeh et al., 2024) addresses this limitation by decomposing the null hypothesis at a finer granularity. Rather than collapsing the non-membership world into an average-case distribution, RMIA enumerates the individual worlds in which x is replaced by each population sample $z \sim \pi$, and constructs a pairwise likelihood ratio for every such pair:

$$\text{LR}_\theta(x, z) = \frac{\Pr(\theta \mid x)}{\Pr(\theta \mid z)}. \tag{18}$$

In the black-box setting, applying Bayes' rule yields an equivalent and efficiently computable form:

$$\text{LR}_\theta(x, z) = \frac{\Pr(x \mid \theta) \,/\, \Pr(x)}{\Pr(z \mid \theta) \,/\, \Pr(z)}, \tag{19}$$

where $\Pr(x \mid \theta)$ is the softmax prediction probability of the target model on x, and $\Pr(x)$ is estimated by averaging $\Pr(x \mid \theta')$ over a set of reference models θ'. The membership score is then defined as the fraction of population samples z that are γ-dominated by x:

$$\text{Score}_{\text{MIA}}(x; \theta) = \Pr_{z \sim \pi}\left[\, \text{LR}_\theta(x, z) \geq \gamma \,\right], \tag{20}$$

and x is classified as a member if $\text{Score}_{\text{MIA}}(x; \theta) \geq \beta$.

Two key distinctions separate RMIA from LiRA. First, whereas LiRA averages the likelihood ratio over all possible replacements z, RMIA performs individual pairwise comparisons, enabling a more fine-grained hypothesis test. Second, LiRA's test statistic reduces to $\Pr(x \mid \theta)/\Pr(x)$ and lacks a mechanism to assess whether this quantity is significant relative to non-members; RMIA explicitly calibrates each comparison against $\Pr(z \mid \theta)/\Pr(z)$, thereby contextualizing the target sample's score within the population distribution.

However, these design choices introduce additional hyperparameters that are absent in LiRA. The pairwise threshold γ controls the dominance criterion, and the number of population samples $|Z|$ determines the resolution of the score. In the offline setting, where only OUT reference models are available, RMIA further requires a scaling parameter a to approximate $\Pr(x)_{\text{IN}}$ from $\Pr(x)_{\text{OUT}}$ via a linear heuristic. The confidence function used to compute $\Pr(x \mid \theta)$ also admits additional hyperparameters, including the Taylor expansion order n, soft-margin m, and temperature T. The optimal settings are dataset-dependent, and selecting them requires a preliminary calibration procedure using existing reference models.

D EXTENDED FORMAL AND EMPIRICAL ANALYSIS

Section 3 established, under the Coexistence Assumption (Assumption 1), that the loss-threshold MIA is *pointwise* inverted on hard-to-fit members: Proposition 1 constructs a threshold $\tau \leq \eta$ and a member–non-member pair on which the attack predicts the opposite of the correct membership status. This appendix complements Proposition 1 along two axes. Appendix D.1 extends the formal analysis (ranking-level restatement in Corollary 1, distributional strengthening in Proposition 2). Appendix D.2 provides empirical support by quantifying, via Fisher SNR, the limited separability between hard-to-fit members and easy non-members achievable from any function of the final output.

D.1 FORMAL EXTENSIONS: RANKING AND DISTRIBUTIONAL INVERSION

We reuse the setup of Section 3—the loss-threshold attack M_τ, the hard-to-fit set $H_\eta(\theta)$, and the easy non-member set $E_\eta(\theta)$—as well as Assumption 1 throughout.

Corollary 1 (Ranking inversion)**.** *Suppose Assumption 1 holds at level η. Then there exist a member $z_h \in D$ and a non-member $z_e \notin D$ such that*

$$\ell(\theta; z_e) \;<\; \ell(\theta; z_h). \tag{21}$$

That is, the loss-based ranking places the non-member above *the member in membership likelihood, independently of any threshold choice.*

Proof. Immediate from Proposition 1: the same pair (z_h, z_e) satisfies $\ell(\theta; z_e) \leq \tau < \ell(\theta; z_h)$, hence $\ell(\theta; z_e) < \ell(\theta; z_h)$. □

Empirical operating point. Given a member set S_{mem} and a non-member set S_{non}, we evaluate M_τ via the standard empirical true and false positive rates,

$$\mathrm{TPR}(M_\tau) \;=\; \frac{1}{|S_{\mathrm{mem}}|} \sum_{z \in S_{\mathrm{mem}}} M_\tau(z; \theta), \qquad \mathrm{FPR}(M_\tau) \;=\; \frac{1}{|S_{\mathrm{non}}|} \sum_{z \in S_{\mathrm{non}}} M_\tau(z; \theta). \tag{22}$$

Proposition 2 (Distributional inversion on the hard-to-fit subset)**.** *Suppose Assumption 1 holds at level η. Restrict the member set to the hard-to-fit subset by setting $S_{\mathrm{mem}} = H_\eta(\theta)$, and let S_{non} be any non-member set containing at least one element of $E_\eta(\theta)$. Then for every $\tau \leq \eta$, $\mathrm{TPR}(M_\tau) = 0$. Moreover, setting $\tau = \eta$ yields $\mathrm{FPR}(M_\eta) > 0$, so that*

$$\mathrm{TPR}(M_\eta) \;=\; 0 \qquad \textit{and} \qquad \mathrm{FPR}(M_\eta) \;>\; 0. \tag{23}$$

Consequently, on this restricted member set, the loss-threshold attack is strictly worse, in expectation, than random guessing at the same FPR. *A random classifier that predicts membership independently with probability $p = \mathrm{FPR}(M_\eta)$ achieves expected false positive rate p, matching the attack, while attaining expected true positive rate $p > 0$ on S_{mem}.*

Proof. Every $z \in S_{\mathrm{mem}} = H_\eta(\theta)$ satisfies $\ell(\theta; z) > \eta$ by Definition 2, so for any $\tau \leq \eta$ we have $M_\tau(z; \theta) = 0$, and hence $\mathrm{TPR}(M_\tau) = 0$. By assumption, S_{non} contains some $z_e \in E_\eta(\theta)$, which satisfies $\ell(\theta; z_e) \leq \eta$ by Definition 3. Setting $\tau = \eta$ gives $\ell(\theta; z_e) \leq \eta$, hence $M_\eta(z_e; \theta) = 1$, so $\mathrm{FPR}(M_\eta) > 0$.

To compare against random guessing at the same operating point, consider the trivial classifier that, independently for each sample, predicts "member" with probability $p = \mathrm{FPR}(M_\eta)$. Its expected false positive rate is p, matching the attack, while its expected true positive rate on S_{mem} is also $p > 0$. Thus, in expectation at the same FPR, this random classifier strictly outperforms M_η, whose TPR on S_{mem} is identically zero. □

D.2 EMPIRICAL ANALYSIS: FISHER SNR ON THE OVERLAP REGION

Section 3.2 claimed that no score derived from the target model's final output can separate hard-to-fit members from easy non-members. We quantify this via two complementary analyses on the overlap region: *Fisher SNR* measures, within each feature, how well the two groups can be distinguished;

Spearman correlation measures, between features, the degree to which they induce the same sample ordering.

The Fisher SNR between two groups $\tilde{H}$ (hard-to-fit members) and $\tilde{E}$ (easy non-members) is defined as

$$\mathrm{SNR} = \frac{(\mu_{\tilde{H}} - \mu_{\tilde{E}})^2}{\sigma^2_{\tilde{H}} + \sigma^2_{\tilde{E}}}, \tag{24}$$

which relates to Cohen's $d = \sqrt{2 \cdot \mathrm{SNR}}$; values of $d = 0.2$ and $d = 0.5$ correspond to the conventional thresholds for "small" and "medium" effect sizes (Cohen, 2013). We evaluate four features used in prior MIA work: loss (Yeom et al., 2018), max softmax confidence $\max_{y'} f(x)_{y'}$ (Salem et al., 2019), modified entropy (Song & Mittal, 2021), and logit-scaled confidence (Carlini et al., 2022a)[5].

Setup. We operationalize the overlap region on CIFAR-100 (ResNet-34, 10 seeds) by defining

$$\tilde{H} := \{z \in D : \ell(\theta; z) \in [0.001, 0.002]\},$$
$$\tilde{E} := \{z \notin D : \ell(\theta; z) \in [0.001, 0.002]\},$$

the densest overlap region identified in Figure 1.[6] We then evaluate whether any final-output feature (including loss itself) separates $\tilde{H}$ from $\tilde{E}$ in this region.

Within-feature separability. Table 2 reports Fisher SNR and Cohen's d for each of the four MIA signals. All four features yield Cohen's d in the range 0.10 ± 0.04, falling below even Cohen's small-effect threshold ($d = 0.2$). The final-checkpoint output alone therefore provides little signal for separating members and non-members in this region.

Between-feature ordering similarity. The similarly small SNR values across all four features in Table 2 raise the question of whether they encode independent weak signals or strongly related information. Table 3 reports pairwise Spearman rank correlation between the four features, computed on the union of $\tilde{H}$ and $\tilde{E}$. Three pairs (loss, max softmax confidence, logit-scaled confidence) achieve $\rho = 1.00 \pm 0.00$ across all 10 seeds, and modified entropy achieves $\rho = 0.97$ with each of the other three. Since membership inference attacks classify samples by thresholding a per-sample score, features with strongly correlated rank orderings produce highly similar ROC curves at every threshold. The four features therefore induce nearly identical orderings ($\rho \geq 0.97$), and the uniformly small SNR values in Table 2 reflect the failure of this shared signal to separate members and non-members.

Conclusion. Together, the two tables establish that the standard MIA signals examined provide little signal for separating hard-to-fit members from easy non-members in the overlap region, and induce nearly identical orderings on this region. This extends the structural failure formalized in Proposition 1 beyond the loss-threshold attack to the broader family of MIA signals derived from the final checkpoint.

E GHOST-LIRA ALGORITHM DETAILS

Section 5.1 described the Ghost-LiRA score as a temporal analogue of the LiRA offline test, with the OUT-model ensemble average replaced by a temporal average over mid-stage checkpoints. For completeness, we provide here the full pseudocode (Algorithm 1), together with a summary of the inputs required from the model owner and the per-sample computational cost.

[5]We use only the logit-scaling transformation $\phi(x, y) = \log f(x)_y - \log \sum_{y' \neq y} f(x)_{y'}$ as a feature of the target model's output, without the shadow-model reference distributions that constitute the full LiRA attack.

[6]Formal Definitions 2 and 3 place H_η and E_η on opposite sides of a common threshold η. To probe the regime where their loss distributions overlap, we instead operationalize $\tilde{H}$ and $\tilde{E}$ as members and non-members within a common loss interval $[0.001, 0.002]$. This relaxation captures the overlap zone empirically while preserving the spirit of the definitions.

Table 2: **Within-feature separability between members and non-members on the loss-matched overlap region.** Fisher SNR and Cohen's $d = \sqrt{2 \cdot \mathrm{SNR}}$ (Cohen, 2013) on CIFAR-100 (ResNet-34, 10 seeds, mean $\pm$ std), computed between members and non-members with loss in $[0.001, 0.002]$. Each value quantifies how well a single feature distinguishes members from non-members. All four signals yield Cohen's d in the range 0.10 ± 0.04, smaller than Cohen's small-effect threshold ($d = 0.2$). The final-checkpoint output alone provides little signal for separating members and non-members in this region.

Feature	Fisher SNR	Cohen's d
Loss	0.0061 ± 0.0036	0.101 ± 0.044
Max softmax confidence	0.0061 ± 0.0037	0.101 ± 0.044
Modified entropy	0.0063 ± 0.0042	0.104 ± 0.043
Logit-scaled confidence (φ)	0.0061 ± 0.0038	0.101 ± 0.044

Table 3: **Pairwise rank-equivalence between MIA signals on the loss-matched overlap region.** Spearman rank correlation on the union of members and non-members with loss in $[0.001, 0.002]$ (CIFAR-100, ResNet-34, 10 seeds, mean $\pm$ std; lower triangle omitted by symmetry). All signals are oriented so that higher values indicate likely membership. Each value quantifies, between two features, whether they induce the same sample ordering. Spearman $\rho \geq 0.97$ across all pairs indicates that the four signals are nearly rank-equivalent on this region and therefore produce essentially the same membership ranking at every threshold.

	$-$Loss	Max conf.	Mod. entropy	Logit-scaled
$-$Loss	—	1.000 ± 0.000	0.970 ± 0.014	1.000 ± 0.000
Max conf.		—	0.970 ± 0.014	1.000 ± 0.000
Mod. entropy			—	0.970 ± 0.014
Logit-scaled				—

Relation to LiRA offline. Our score is a principled simplification of the LiRA offline test. Starting from Eq. 8 and rewriting the argument of Φ,

$$\Lambda(x, y) \;=\; \Phi\left(\frac{\phi_T(x, y) - \mu_{\mathrm{out}}(x, y)}{\sigma_{\mathrm{out}}(x, y)}\right), \tag{25}$$

the strict monotonicity of Φ implies that the induced ranking (and therefore the ROC curve) depends only on its argument. Under the global-variance variant of Carlini et al. (2022a) (in which σ_{out} is sample-independent),[7] Λ becomes a monotone transformation of $\varphi_T(x, y) - \mu_{\mathrm{out}}(x, y)$, which is exactly s_{Ghost} with μ_{out} replaced by μ_{Ghost}. Ghost-LiRA thus inherits the ranking of LiRA offline while replacing its N-shadow-model ensemble average with a single-trajectory temporal average over $\mathcal{W}_{\mathrm{out}}$.

Input requirements. Ghost-LiRA requires only the pre-softmax logits of the final target model and of the saved intermediate checkpoints in $\mathcal{W}_{\mathrm{out}}$, together with their validation accuracies[8]. No additional training is performed, and no information about the training procedure or its data is required. Because only per-sample logit vectors need be retained (not full model weights), the storage footprint is orders of magnitude smaller than saving intermediate checkpoints.

[7] Carlini et al. (2022a) report that global variance outperforms per-sample variance with fewer than 64 shadow models, with the gap remaining marginal even at 1024. The distinction is moot for Ghost-LiRA in any case: checkpoints within $\mathcal{W}_{\mathrm{out}}$ are strongly correlated across epochs, so their empirical spread underestimates the true OUT-distribution variance.

[8] When validation accuracies are unavailable, the OUT window can be replaced by any accessible intermediate checkpoint that precedes the late training phase (Section 5.1).

Algorithm 1 Ghost-LiRA membership score

Require: Target model f trained for T epochs on $\mathcal{D}_{\text{train}}$
Require: Query example (x, y)
Require: Set of saved checkpoints $\mathcal{C} \subseteq \{1, \dots, T\}$
Require: Pre-softmax logits $\{z_t(x)\}_{t\in\mathcal{C}}$
Require: Validation accuracies $\{\mathrm{acc}_t^{\mathrm{val}}\}_{t\in\mathcal{C}}$
(For language models, $\mathrm{acc}_t^{\mathrm{val}} := 1/\mathrm{ppl}_t^{\mathrm{val}}$*.)*
Require: Peak validation accuracy $\mathrm{acc}_{\max}^{\mathrm{val}} := \max_{t\in\mathcal{C}} \mathrm{acc}_t^{\mathrm{val}}$
Ensure: Membership score s_{Ghost} (higher $\Rightarrow$ more likely member)
1: **Select OUT window by validation accuracy:**

$$\mathcal{W}_{\mathrm{out}} \leftarrow \left\{ t \in \mathcal{C} \ : \ \alpha_{\mathrm{lo}} \cdot \mathrm{acc}_{\max}^{\mathrm{val}} \le \mathrm{acc}_t^{\mathrm{val}} \le \alpha_{\mathrm{hi}} \cdot \mathrm{acc}_{\max}^{\mathrm{val}} \right\}$$

if $\mathcal{W}_{\mathrm{out}} = \emptyset$ **then**

$$t^* \leftarrow \arg\max_{t\in\mathcal{C},\, t\neq T} \mathrm{acc}_t^{\mathrm{val}}$$
$$\mathcal{W}_{\mathrm{out}} \leftarrow \{t^*\}$$

end if
2: **Compute logit-scaled confidence** for each $t \in \mathcal{W}_{\mathrm{out}} \cup \{T\}$:

$$\varphi_t(x, y) \leftarrow z_t(x)_y \ - \ \underset{y'\neq y}{\mathrm{LogSumExp}}\ z_t(x)_{y'}$$

3: **Estimate** $\mu_{\mathrm{Ghost}}(x, y)$ **from the OUT window:**

$$\mu_{\mathrm{Ghost}}(x, y) \leftarrow \frac{1}{|\mathcal{W}_{\mathrm{out}}|} \sum_{t\in\mathcal{W}_{\mathrm{out}}} \varphi_t(x, y)$$

4: **Membership score:** $s_{\mathrm{Ghost}} \leftarrow \varphi_T(x, y) \ - \ \mu_{\mathrm{Ghost}}(x, y)$
5: **return** s_{Ghost}

F Experimental details

This section provides additional details on our experimental setup. Appendix F.1 reports the full training configurations for target and shadow models across all four benchmarks, and Appendix F.2 describes the implementation of each MIA baseline.

F.1 Training configurations

Table 4 summarizes the training configurations for target and shadow models across the four benchmarks. Within each (dataset, architecture) pair, target and shadow models share identical configurations and differ only in the training subset and the random seed; reported accuracies are averaged over 10 seeds.

The four benchmarks are chosen to span a diverse set of training regimes commonly encountered in the MIA literature. CIFAR-10 (ResNet-18) and CIFAR-100 (ResNet-34) follow the standard SGD-with-momentum recipe and a step-decay schedule, with CIFAR-100 additionally using Cutout (DeVries & Taylor, 2017) to reflect the stronger regularization typically applied to harder vision benchmarks. Tiny ImageNet (Swin-T) follows a modern transformer recipe with AdamW, cosine annealing with linear warm-up, and a strong augmentation stack—RandAugment (Cubuk et al., 2020), MixUp (Zhang et al., 2018), CutMix (Yun et al., 2019), label smoothing (Szegedy et al., 2016); training is terminated at epoch 200 of a 250-epoch cosine schedule, providing an early-stopping scenario that we use to verify Ghost-LiRA's stability across diverse training regimes (Section 6). WikiText-103 (GPT-2-Large) is a text benchmark and uses no augmentation. GPT-2 was pretrained on WebText, from which all Wikipedia documents were removed (Radford et al., 2019); WikiText-103, drawn from Wikipedia, is therefore not part of its pretraining corpus. For the three image benchmarks, an exponential moving average (EMA) of the model weights with decay 0.999 is maintained during training and used as the final evaluated model; WikiText-103 (GPT-2-Large) uses no EMA.

Table 4: **Training configurations and final model performance across the four benchmarks.** Image models report train/test accuracy; the language model reports validation perplexity. Values are mean ± std over 10 random seeds. Shadow models are trained under identical configurations to target models, differing only in the training subset and random seed.

	CIFAR-10	CIFAR-100	Tiny ImageNet	WikiText-103
Architecture	ResNet-18	ResNet-34	Swin-T	GPT-2-Large
Initialization	Random	Random	Random	Pretrained
# classes	10	100	200	–
Epochs	182	200	200 / 250[†]	10
Batch size	128	128	512	16[‡]
Optimizer	SGD (m=0.9, Nesterov)	SGD (m=0.9, Nesterov)	AdamW (β=(0.9, 0.999))	AdamW (β=(0.9, 0.999))
Weight decay	5×10^{-4}	5×10^{-4}	0.05	0
Initial LR	0.1	0.1	2×10^{-3}	10^{-5}
LR schedule	MultiStepLR (milestones = [91, 136], γ=0.1)	MultiStepLR (milestones = [100, 150], γ=0.1)	Cosine decay ($\eta_{\min}$=10^{-5})	Linear decay (to 0)
Warm-up	None	None	Linear, 25 ep ($10^{-6} \rightarrow 2\times10^{-3}$)	None
Augmentation	Crop + HFlip	Crop + HFlip + Cutout (len=16)	Crop + HFlip + RandAug (n=2, m=9) + MixUp (α=0.8) / CutMix (α=1.0) + LS (0.1)	None
Other regularization	–	–	drop_path = 0.15	dropout = 0.1 grad. clip = 1.0
EMA (decay)	✓(0.999)	✓(0.999)	✓(0.999)	–
Train accuracy (%)	99.99 ± 0.00	99.98 ± 0.01	99.75 ± 0.05	–
Test accuracy (%)	92.66 ± 0.27	70.03 ± 0.69	47.72 ± 0.41	–
Val. perplexity	–	–	–	23.39 ± 0.12

[†]Cosine schedule set with $T_{\max} = 250$; training is terminated at epoch 200. [‡]Per-device batch size 4 with 4 gradient-accumulation steps; sequences are single WikiText-103 lines truncated to 1,024 tokens.

All evaluations (validation accuracy and MIA queries) are performed on the EMA-smoothed weights (raw weights for WikiText-103). This applies to all saved checkpoints, both intermediate and final, and to all attacks uniformly. An ablation comparing EMA and non-EMA settings on CIFAR-100 is provided in Appendix J.5.

F.2 Baselines

For all baselines, we adhered to the original implementations as closely as possible. Below, we note several implementation details worth mentioning.

Attack-P. For Attack-P (Ye et al., 2022), we use the test set as population data to estimate the non-member loss distribution on each target model, and sweep the threshold over this distribution to obtain the ROC curve.

Modified entropy. We implement the modified prediction entropy attack of Song & Mittal (2021), using their modified entropy metric (Equation 16) with class-dependent thresholds $\check{\tau}_y$ learned from a single shadow model via the shadow training technique (Shokri et al., 2017), following the original paper. To produce ROC curves and TPR at low FPR, we use $\check{\tau}_y - \mathrm{Mentr}(F(\mathbf{x}), y)$ as a calibrated membership score and sweep a single threshold over this score across all samples; the original attack is recovered at threshold zero.

LiRA. In our experiments, we employ both variants of LiRA (Carlini et al., 2022a) as baselines: the online variant, which is the most powerful but computationally expensive as it requires training both IN and OUT shadow models for each target sample, and the offline variant, which is more practical

and efficient, training only OUT shadow models independently of the target sample and performing a one-sided hypothesis test. For the offline variant, we used a global variance when it yielded better performance.

RMIA. RMIA (Zarifzadeh et al., 2024) introduces additional hyperparameters, including γ, the offline scaling factor a, and the number of reference samples z; we adopt the values reported in the original paper. Since the original paper does not include experiments on Tiny ImageNet, we tune these hyperparameters starting from those used for ImageNet. We adopt only the offline variant of RMIA, both because it aligns with Ghost-LiRA's offline formulation and because the original work reports it to achieve performance on par with the online version.

F.3 Ghost-LiRA configuration

Checkpoint cadence. We save checkpoints every 10 epochs for CIFAR-10, CIFAR-100, and Tiny ImageNet, and every epoch for WikiText-103 (which trains for only 10 epochs in total). At each saved checkpoint, only the pre-softmax logits on the queried samples and the validation accuracy are required by Ghost-LiRA.

OUT window. Across all four benchmarks, we use $(\alpha_{\mathrm{lo}}, \alpha_{\mathrm{hi}}) = (0.90, 0.97)$ in Eq. 9, yielding the following OUT windows $\mathcal{W}_{\mathrm{out}}$:

- CIFAR-10: epochs $\{40, 50, 60\}$
- CIFAR-100: epochs $\{50, 60, 70, 80\}$
- Tiny ImageNet: epochs $\{90, 100\}$
- WikiText-103: epochs $\{5\}$

The same $(\alpha_{\mathrm{lo}}, \alpha_{\mathrm{hi}})$ is used unchanged across benchmarks; we do not tune it per dataset.

Progress metric for WikiText-103. For GPT-2-Large, next-token accuracy on the validation split changes by only a few percent during fine-tuning, which leaves most cells of the $(\alpha_{\mathrm{lo}}, \alpha_{\mathrm{hi}})$ grid without a checkpoint. We therefore use validation perplexity and set $\mathrm{acc}_t^{\mathrm{val}} := 1/\mathrm{ppl}_t^{\mathrm{val}}$, so that Eq. 9 and Algorithm 1 apply unchanged with the ratio $\mathrm{ppl}_{\min}^{\mathrm{val}}/\mathrm{ppl}_t^{\mathrm{val}}$.

Sensitivity and minimal-checkpoint variants. The choice $(\alpha_{\mathrm{lo}}, \alpha_{\mathrm{hi}}) = (0.90, 0.97)$ sits at the center of a broad robustness plateau; sensitivity to window placement is analyzed in Appendix J.2, and the single-checkpoint variant in Appendix J.3.

F.4 Privacy scores

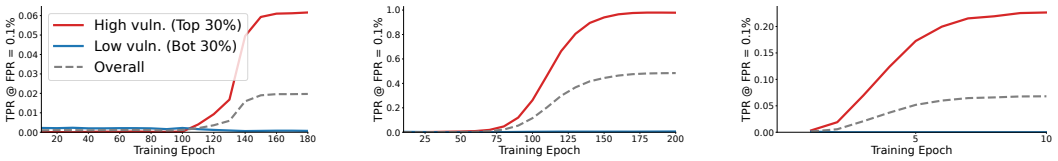
The privacy scores used for vulnerability stratification (Eq. 1) are computed from a dedicated set of 256 shadow models, trained independently of those used for LiRA and RMIA evaluation. Each shadow model follows the same training configuration as in Table 4, with each sample included in training with probability 0.5.

G Additional results across benchmarks

This section reports experimental results omitted from the main text due to space.

G.1 Late memorization and OUT-model approximation across benchmarks

This section extends the analysis of Section 4 to the remaining three benchmarks. Figure 5 shows that vulnerable samples are consistently memorized late across all settings, and Figure 6 confirms that mid-stage checkpoints approximate the true OUT distribution on these samples.



(a) CIFAR-10, ResNet-18. (b) Tiny ImageNet, Swin-T. (c) WikiText-103, GPT-2-Large.

Figure 5: TPR@0.1%FPR of LiRA online against intermediate checkpoints on CIFAR-10, Tiny ImageNet and WikiText-103, stratified by vulnerability (cf. Figure 2a for CIFAR-100). On CIFAR-10 and Tiny ImageNet, the same late-memorization pattern holds: vulnerable samples remain undetectable until the late training phase. On WikiText-103, memorization of vulnerable samples begins earlier, consistent with fine-tuning from a pretrained initialization and the shorter training schedule. Despite this compressed timeline, Ghost-LiRA still achieves competitive performance on this benchmark (Figure 7c), indicating that even a narrow temporal gap suffices for effective OUT-model approximation.

G.2 Vulnerability-stratified evaluation across benchmarks and strata

This section complements the CIFAR-100 stratified ROC curves in Section 6 along two axes: Appendix G.2.1 extends the top/bottom-30% evaluation to the remaining three benchmarks, and Appendix G.2.2 verifies that the same trends persist at finer stratification thresholds (10% and 20%).

G.2.1 Remaining benchmarks (CIFAR-10, Tiny ImageNet, WikiText-103)

Figure 3 in the main text reported the top/bottom-30% stratified ROC curves on CIFAR-100. Figure 7 extends this evaluation to CIFAR-10, Tiny ImageNet, and WikiText-103. The trends mirror those on CIFAR-100: per-sample MIAs (LiRA, RMIA, Ghost-LiRA) lift above their full-population performance on the vulnerable subgroup and drop on the safe one, while population-based MIAs (Attack-P, Modified Entropy) show the opposite pattern.

G.2.2 Extended stratification thresholds (top/bottom-10% and -20%)

Figures 3 and 7 report results on the top/bottom-30% subgroups. Here we extend this analysis to the 10% and 20% thresholds to verify that the same trends hold as the subgroup narrows. As shown in Figure 8, the patterns persist: on the vulnerable (top) subgroups, per-sample MIAs maintain their advantage, while population-based MIAs perform at the random-guess level on CIFAR-10 and below it on the remaining benchmarks; on the safe (bottom) subgroups, population-based MIAs improve over their full-population performance while per-sample MIAs degrade. Ghost-LiRA tracks the per-sample profile across all subgroups and benchmarks.

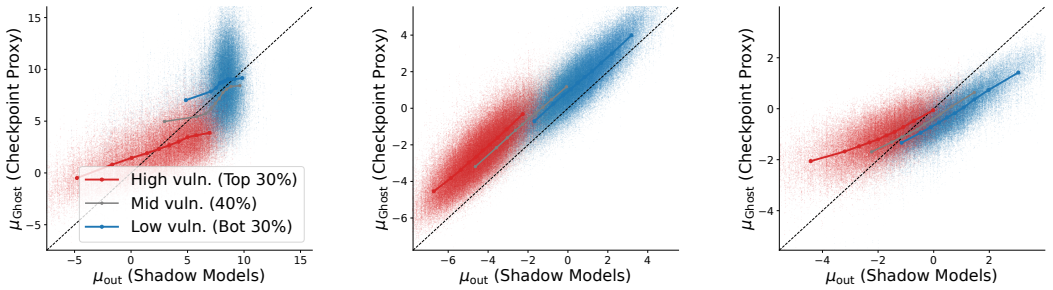
G.3 Effect of the number of shadow models across benchmarks and FPR thresholds

Figure 4 in the main text reports TPR@0.1%FPR on CIFAR-100 as a function of the number of trained shadow models. Here we extend this analysis along two axes: (i) to the remaining three benchmarks (CIFAR-10, Tiny ImageNet, WikiText-103), and (ii) to additional FPR thresholds (1% and 0.01%). Figure 9 reports the results.

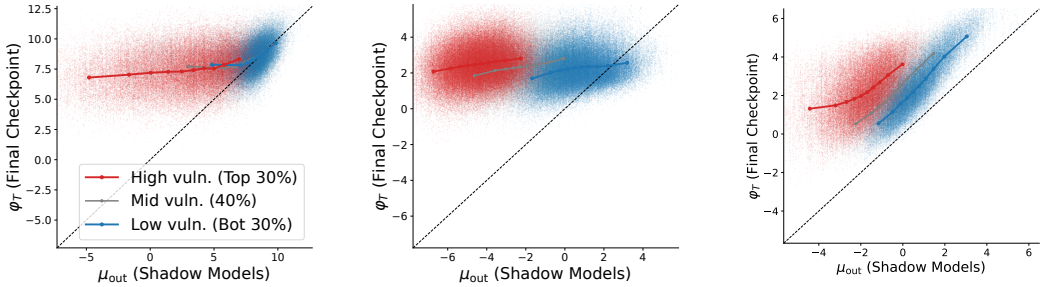
G.4 Computational cost of MIA baselines

Table 5 reports computation cost against all baselines. For each per-sample baseline, we report its cost at the largest number of shadow models at which Ghost-LiRA still strictly outperforms the baseline's TPR@0.1%FPR. The shadow-model counts are: LiRA online uses 16 (CIFAR-10) and 32 (CIFAR-100, Tiny ImageNet, WikiText-103); LiRA offline uses 8 (CIFAR-10), 256 (CIFAR-100) and 32 (Tiny ImageNet, WikiText-103); RMIA uses 8 (CIFAR-10, WikiText-103) and 16 (CIFAR-100, Tiny ImageNet). Population-based baselines have a fixed cost. Attack-P requires no shadow models, and Modified Entropy trains a single shadow model (Appendix C.2).

All measurements include shadow-model training, score computation, and threshold calibration.



(a) **Ghost-LiRA proxy (μ_{Ghost}, y-axis) vs. true OUT mean (μ_{out}, x-axis).** Points near the diagonal indicate that mid-stage checkpoints approximate the true OUT distribution well.



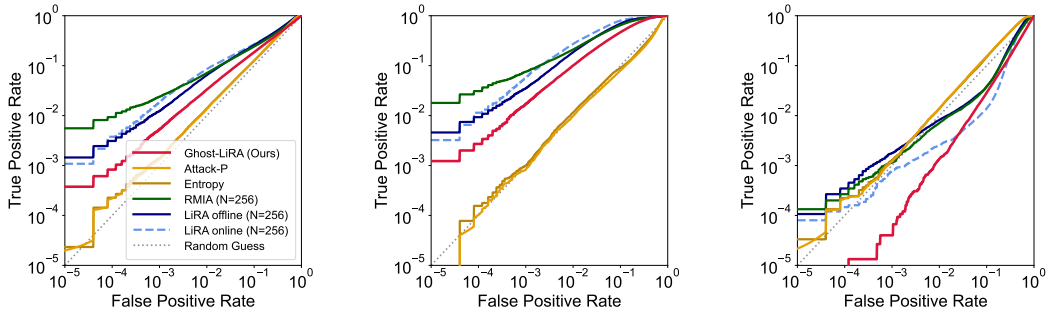
(b) **Final-checkpoint confidence (φ_T, y-axis) vs. true OUT mean (μ_{out}, x-axis).** High-vulnerability samples drift above the diagonal as they are memorized, while low-vulnerability samples remain near it.

Figure 6: **Mid-stage checkpoints behave OUT-like on vulnerable samples across benchmarks** (cf. Figure 2b for CIFAR-100). Columns from left to right: CIFAR-10, Tiny ImageNet, WikiText-103. Solid lines show binned averages: each vulnerability subgroup is further divided into 10 equally sized bins by vulnerability score, and the mean of each bin is plotted. **(a)** Target model's mean logit-scaled confidence over the OUT-window checkpoints ($(\alpha_{\text{lo}}, \alpha_{\text{hi}}) = (0.90, 0.97)$, cf. Eq. 9) vs. μ_{out} from true shadow models. **(b)** Final-checkpoint confidence (φ_T) vs. μ_{out}. The trends are consistent with those observed on CIFAR-100 in the main text: mid-stage proxies track μ_{out} closely across all vulnerability groups (Figure 6a), while high-vulnerability samples drift well above the diagonal by the final checkpoint (Figure 6b). Tiny ImageNet exhibits the tightest alignment between μ_{Ghost} and μ_{out}, which corresponds to its strongest Ghost-LiRA performance across benchmarks (Section 6). On WikiText-103, the earlier onset of memorization (Figure 5) results in a less ideal proxy, yet the gap between mid-stage and final-checkpoint drift remains sufficient for Ghost-LiRA to achieve competitive attack performance.

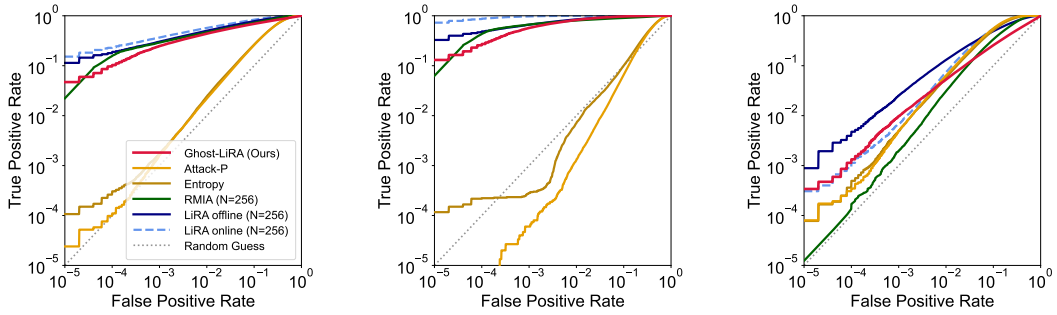
H Comparison with SeqMIA

Target model. We use the same target model as in our main experiments: a ResNet-34 trained on a random 25k subset of CIFAR-100's 50k training set (configuration in Appendix F.1). The remaining 25k samples form the ground-truth non-members.

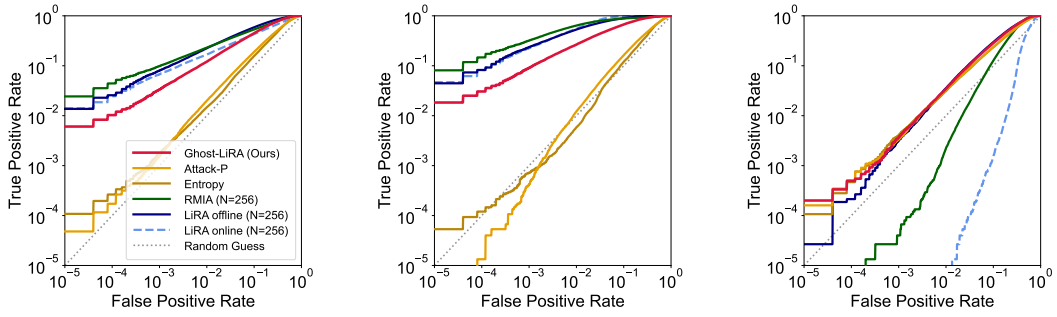
Data split. SeqMIA's original evaluation (Li et al., 2024) partitions the dataset in a 1:1:1:1:2 ratio across target train/eval, shadow train/eval, and a distillation set, yielding an adversary auxiliary budget of 40k samples on CIFAR-100, four times the target training set. This departs from the post-LiRA convention (Carlini et al., 2022a; Zarifzadeh et al., 2024; Du et al., 2026; 2025), where the adversary's auxiliary set is comparable in size to the target training set and evaluation uses a 50:50 member/non-member split. Adopting SeqMIA's original split here would grant the adversary substantially more data than other baselines and would yield too few evaluation samples to reliably estimate low-FPR metrics such as TPR @ 0.001% FPR, which the post-LiRA literature emphasizes. We therefore evaluate SeqMIA's algorithmic design under our framework's standard data budget. Specifically, this



(a) **CIFAR-10.**

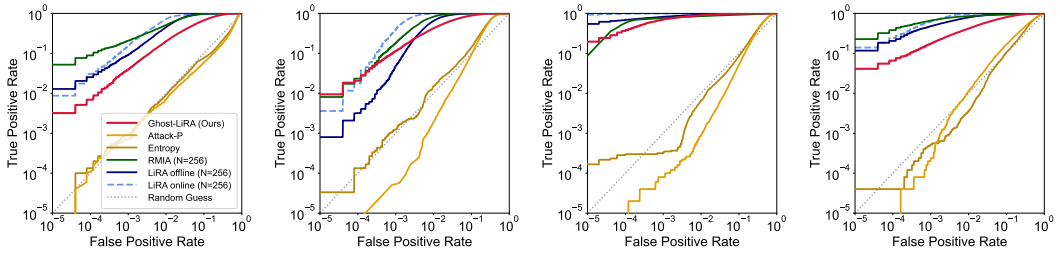


(b) **Tiny ImageNet.**

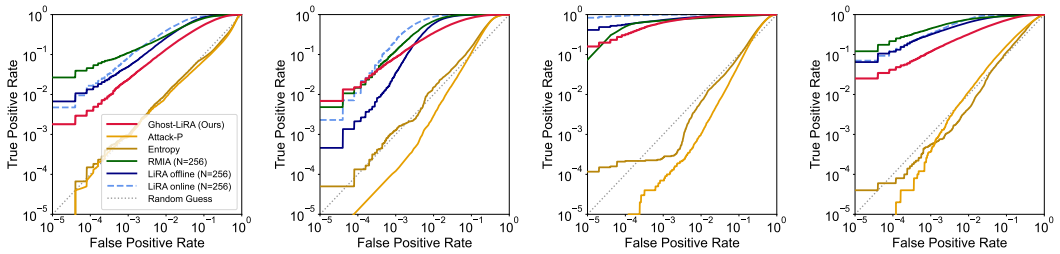


(c) **WikiText-103.**

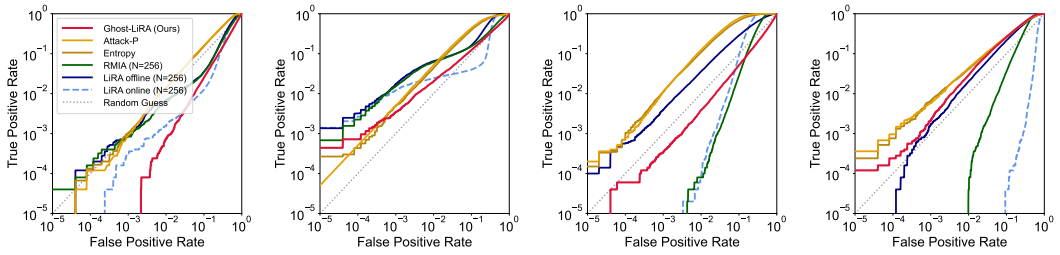
Figure 7: **ROC curves on CIFAR-10, Tiny ImageNet, and WikiText-103** (cf. Figure 3 for CIFAR-100). Each row corresponds to one benchmark; columns show evaluation on (left) the full training set, (middle) the top-30% most vulnerable samples, and (right) the bottom-30% least vulnerable samples. The trends are consistent with those reported on CIFAR-100: population-based attacks collapse on the vulnerable subgroup and recover on the safe one, while per-sample attacks (including Ghost-LiRA) show the opposite pattern.



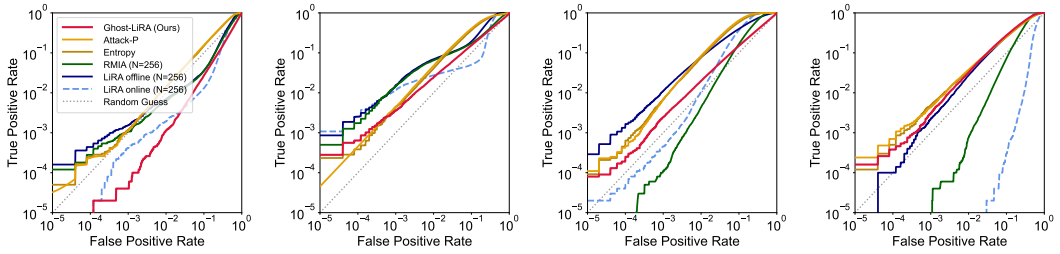
(a) **Top-10% most vulnerable samples.**



(b) **Top-20% most vulnerable samples.**

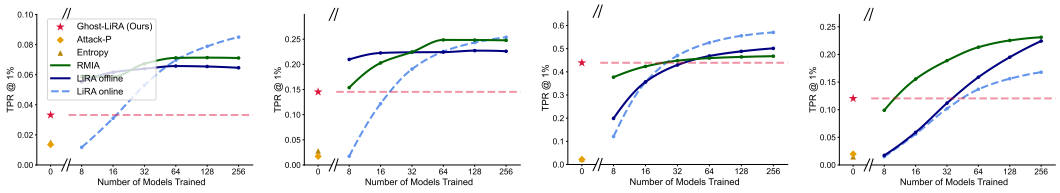


(c) **Bottom-10% least vulnerable samples.**

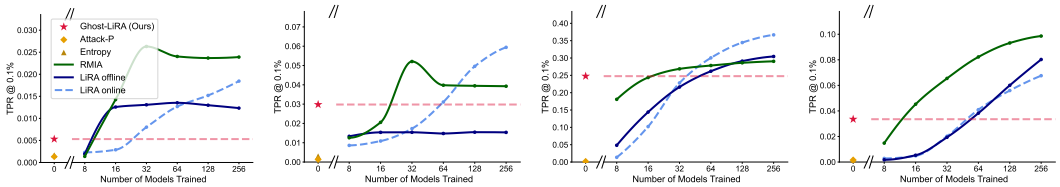


(d) **Bottom-20% least vulnerable samples.**

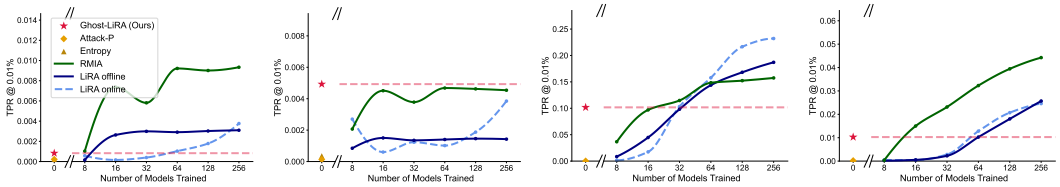
Figure 8: **ROC curves on CIFAR-10, CIFAR-100, Tiny ImageNet, and WikiText-103 (left to right), evaluated on (a) the top-10% most vulnerable samples, (b) the top-20% most vulnerable samples, (c) the bottom-10% least vulnerable samples and (d) the bottom-20% least vulnerable samples.** The trends observed at the 30% threshold in Figures 3 and 7 persist at these narrower subgroups. Population-based MIAs (Attack-P, Modified Entropy) improve over their full-population performance on safe (bottom) subgroups, but degrade to random-guess level on CIFAR-10 or below random-guess on the remaining benchmarks when evaluated on vulnerable (top) subgroups. Ghost-LiRA consistently tracks the per-sample profile across both extremes.



(a) **TPR @ 1% FPR.**



(b) **TPR @ 0.1% FPR.**



(c) **TPR @ 0.01% FPR.**

Figure 9: **Effect of the number of shadow models across benchmarks and FPR thresholds.** TPR as a function of the number of trained shadow models at (a) 1% FPR, (b) 0.1% FPR, and (c) 0.01% FPR on CIFAR-10, CIFAR-100, Tiny ImageNet, and WikiText-103 (left to right). Figure 4 in the main text reports 0.1% FPR on CIFAR-100 only; this figure extends the comparison to all four benchmarks and three FPR thresholds. Ghost-LiRA (dashed red line, plotted at x=0) requires no additional model training yet remains competitive with per-sample attacks using tens of shadow models across all settings. The trends are consistent across FPR thresholds: population-based attacks (Attack-P, Modified Entropy) yield near-zero TPR regardless of ensemble size.

Table 5: **End-to-end computation time of MIA evaluation (seconds).** For each per-sample baseline, the cost is measured at the largest ensemble size at which Ghost-LiRA still strictly outperforms the baseline in TPR@0.1%FPR (see Figure 9); the reference-model counts used are listed in the text. All measurements are on a single NVIDIA L40S GPU.

Method	CIFAR-10	CIFAR-100	Tiny ImageNet	WikiText-103
Attack-P	4.5	6.8	11.9	158.5
Modified Entropy	1,342	1,710	2,164	9,789
LiRA online	21,422	54,551	68,906	312,896
LiRA offline	10,711	436,361	68,907	310,368
RMIA	10,711	27,276	34,465	78,224
Ghost-LiRA (Ours)	15.5	19.3	44.4	316
Speedup vs. LiRA online	1,387×	2,832×	1,554×	990×

design exploits metric sequences extracted from a series of *distilled student snapshots* obtained via knowledge distillation from the final target model.

Auxiliary data for SeqMIA. SeqMIA's auxiliary data is restricted to CIFAR-100's 10k test split, partitioned disjointly into D_s^{train} (4,000 samples; shadow members), D_s^{test} (2,000 samples; shadow non-members), and the distillation set D_k (4,000 samples) used to query both target and shadow teachers. The 50k CIFAR-100 training pool is reserved entirely for evaluation. We follow SeqMIA's reference implementation for all algorithmic components: $\alpha = 1$ (KL loss only), 30 distillation

Table 6: **SeqMIA vs. Ghost-LiRA on CIFAR-100 (ResNet-34).** SeqMIA uses every-epoch distilled checkpoints, with the CIFAR-100 test split (10k) as its auxiliary data; Ghost-LiRA uses the sparse checkpoint pool (every 10 epochs) of the target model itself and trains no auxiliary model. Per-target attack cost on a single L40S GPU: 19.3 s for Ghost-LiRA vs. 1,170 s for SeqMIA (including shadow training and distillation). *Overall* aggregates the 50k evaluation pool (25k members, 25k non-members); *High vuln. (Top 30%)* and *Low vuln. (Bot 30%)* stratify by vulnerability. Values are mean $\pm$ std over 10 seeds. TPR in %. Best per (subset, metric) in **bold**.

Subset	Method	AUC	TPR@10%	TPR@1%	TPR@0.1%	TPR@0.01%
Overall	SeqMIA	$\mathbf{0.835 \pm 0.004}$	44.25 ± 0.65	7.12 ± 0.66	0.79 ± 0.21	0.10 ± 0.08
	Ghost-LiRA	0.788 ± 0.009	$\mathbf{49.43 \pm 1.19}$	$\mathbf{14.49 \pm 0.84}$	$\mathbf{2.98 \pm 0.47}$	$\mathbf{0.49 \pm 0.26}$
High vuln. (Top 30%)	SeqMIA	0.892 ± 0.004	61.43 ± 1.48	11.45 ± 1.26	1.25 ± 0.34	0.17 ± 0.13
	Ghost-LiRA	$\mathbf{0.932 \pm 0.004}$	$\mathbf{78.59 \pm 1.43}$	$\mathbf{30.08 \pm 1.82}$	$\mathbf{7.02 \pm 1.12}$	$\mathbf{1.23 \pm 0.67}$
Low vuln. (Bot 30%)	SeqMIA	$\mathbf{0.767 \pm 0.009}$	$\mathbf{25.30 \pm 0.96}$	3.03 ± 0.42	0.33 ± 0.12	0.05 ± 0.03
	Ghost-LiRA	0.590 ± 0.020	19.85 ± 1.47	$\mathbf{3.19 \pm 0.36}$	$\mathbf{0.45 \pm 0.11}$	$\mathbf{0.06 \pm 0.03}$

epochs with per-epoch student snapshots, the five metrics (Loss, Max, SD, Entropy, M-Entropy), and the attention-based LSTM attack model. The shadow model uses the same architecture and hyperparameters as the target.

Evaluation protocol. Our vulnerability-stratified evaluation (Section 6) is designed to test two distinct properties of an MIA framework: (i) whether it can *distinguish* between samples with genuinely high membership leakage and those without, rather than scoring all members uniformly; and (ii) whether it avoids assigning spuriously high scores to low-vulnerability samples, typically easy non-members, that are inherently difficult to memorize and should not be flagged. Verifying property (ii) requires a sufficiently diverse non-member population in the evaluation pool, since training-only evaluation cannot expose false positives on such samples. Accordingly, attack scores are computed for all 50k CIFAR-100 training samples (25k members and 25k non-members per the target's training subset), and ROC analysis is performed over this pool. This is the same evaluation pool as in our main experiments; the test split, unused there, serves as SeqMIA's auxiliary data.

Findings. Table 6 compares SeqMIA and Ghost-LiRA under a setup that allocates SeqMIA's required auxiliary data, and the results are consistent with the main claims of this paper. First, SeqMIA's overall TPR at low FPR exceeds that of population-based baselines (Section 6), confirming that signals from training trajectories carry information beyond the final target output. We note that SeqMIA's higher overall AUC (0.835 vs. Ghost-LiRA's 0.788) does not indicate stronger overall attack capability in the privacy-relevant sense. AUC integrates TPR uniformly across the entire FPR range, including the high-FPR regime that recent MIA literature deems uninformative for worst-case privacy risk (Carlini et al., 2022a; Zarifzadeh et al., 2024). In the low-FPR regime where it matters most, Ghost-LiRA in fact outperforms SeqMIA on the overall pool: TPR @ 1% FPR is 14.49% vs. 7.12%, TPR @ 0.1% FPR is 2.98% vs. 0.79%, and TPR @ 0.01% FPR is 0.49% vs. 0.10%. Second, on the high-vulnerability subset, both attacks improve over their overall performance, but Ghost-LiRA amplifies its TPR by $\sim 2.3\times$ on average against SeqMIA's $\sim 1.6\times$, and this gap widens monotonically as FPR tightens: Ghost-LiRA's Top-30% TPR is $2.6\times$ that of SeqMIA at 1% FPR, $5.6\times$ at 0.1% FPR, and $7.2\times$ at 0.01% FPR, the regime where it matters most. Third, on Bottom-30%, where in/out behavior is similar and discrimination is intrinsically information-limited, SeqMIA's TPR @ 10% FPR (25.30%) noticeably exceeds Ghost-LiRA's (19.85%), but the two converge at tight FPRs. This pattern is the signature of uncalibrated scores inflating on easy samples whose absolute metrics resemble those of members, rather than genuine signal recovery. Together, these observations separate two distinct contributions: *trajectory signals* (SeqMIA) improve over final-output-only attacks, while *per-sample calibration* (Ghost-LiRA) is an orthogonal axis that controls how those signals are interpreted relative to each sample's difficulty. Both are necessary for clean vulnerability stratification, and Ghost-LiRA achieves both without auxiliary models or distillation.

Table 7: **White-box attacks vs. Ghost-LiRA on CIFAR-100 (ResNet-34).** Each white-box attack is applied to the final target model as defined by its authors. Nasr (supervised) and Leino train an additional attack or proxy model; Nasr (grad-norm) and Cohen (SIF) do not. Ghost-LiRA uses only the pre-softmax logits of the target model and its checkpoints, and trains no auxiliary model. *Overall* aggregates the full evaluation pool; *High vuln. (Top 30%)* and *Low vuln. (Bot 30%)* stratify by vulnerability. Values are mean ± std over 10 seeds. TPR in %. Best per (subset, metric) in **bold**.

Subset	Method	AUC	TPR@10%	TPR@1%	TPR@0.1%	TPR@0.01%
Overall	Nasr (grad-norm)	0.741 ± 0.004	22.36 ± 0.70	1.71 ± 0.23	0.15 ± 0.03	0.01 ± 0.01
	Nasr (supervised)	0.762 ± 0.004	31.64 ± 0.70	6.16 ± 0.38	1.19 ± 0.22	0.18 ± 0.05
	Leino	0.821 ± 0.004	42.65 ± 1.05	8.13 ± 1.03	1.30 ± 0.49	0.19 ± 0.14
	Cohen (SIF)	$\mathbf{0.830 \pm 0.003}$	29.20 ± 1.03	1.94 ± 0.32	0.16 ± 0.03	0.01 ± 0.01
	Ghost-LiRA	0.788 ± 0.009	$\mathbf{49.43 \pm 1.19}$	$\mathbf{14.49 \pm 0.84}$	$\mathbf{2.98 \pm 0.47}$	$\mathbf{0.49 \pm 0.26}$
High vuln. (Top 30%)	Nasr (grad-norm)	0.721 ± 0.005	11.73 ± 0.92	0.38 ± 0.14	0.02 ± 0.02	0.00 ± 0.00
	Nasr (supervised)	0.783 ± 0.004	39.78 ± 0.94	10.60 ± 0.70	2.39 ± 0.38	0.39 ± 0.13
	Leino	0.871 ± 0.006	55.66 ± 2.13	12.99 ± 2.40	2.33 ± 1.06	0.37 ± 0.32
	Cohen (SIF)	0.802 ± 0.005	18.55 ± 1.62	0.53 ± 0.31	0.01 ± 0.02	0.00 ± 0.00
	Ghost-LiRA	$\mathbf{0.932 \pm 0.004}$	$\mathbf{78.59 \pm 1.43}$	$\mathbf{30.08 \pm 1.82}$	$\mathbf{7.02 \pm 1.12}$	$\mathbf{1.23 \pm 0.67}$
Low vuln. (Bot 30%)	Nasr (grad-norm)	0.785 ± 0.004	36.71 ± 0.79	3.92 ± 0.39	0.40 ± 0.08	0.02 ± 0.02
	Nasr (supervised)	0.753 ± 0.005	22.59 ± 1.12	1.85 ± 0.29	0.14 ± 0.05	0.03 ± 0.03
	Leino	0.768 ± 0.008	29.99 ± 1.19	4.10 ± 0.29	$\mathbf{0.50 \pm 0.08}$	$\mathbf{0.06 \pm 0.03}$
	Cohen (SIF)	$\mathbf{0.861 \pm 0.003}$	$\mathbf{42.42 \pm 1.03}$	$\mathbf{4.14 \pm 0.34}$	0.40 ± 0.07	0.04 ± 0.03
	Ghost-LiRA	0.590 ± 0.020	19.85 ± 1.47	3.19 ± 0.36	0.45 ± 0.11	$\mathbf{0.06 \pm 0.03}$

I Comparison with white-box signals

To examine whether white-box signals provide additional benefits to Ghost-LiRA, we consider two settings. Appendix I.1 compares Ghost-LiRA with existing white-box attacks, while Appendix I.2 evaluates Ghost-LiRA with white-box signals substituted for its black-box signals.

I.1 Comparison with white-box attacks

To compare Ghost-LiRA with white-box attacks, we use the attack of Nasr et al. (2019) in both its gradient-norm and supervised variants, the attack of Leino & Fredrikson (2020), and the self-influence function (SIF) attack of Cohen & Giryes (2024). Each attack is applied to the final target model as defined by its authors, on CIFAR-100 (ResNet-34) over 10 seeds, under the vulnerability-stratified protocol of Section 6. The gradient-norm and SIF attacks use no reference model, whereas the supervised variant and the attack of Leino & Fredrikson (2020) train an additional model. As reported in Table 7, the gradient-norm and SIF attacks score higher on the Bottom-30% subset than on the Top-30% subset at low FPR, while the other two do not show this pattern. On the Top-30% subset, Ghost-LiRA exceeds all four attacks at every FPR threshold.

I.2 Substituting white-box signals into Ghost-LiRA

We evaluate whether white-box signals provide additional information to Ghost-LiRA by keeping its scoring pipeline unchanged and replacing only the per-checkpoint signal φ_t in Eq. 7 with ψ_t, so that the score becomes $\psi_T(x,y) - \frac{1}{|\mathcal{W}_{\text{out}}|}\sum_{t\in\mathcal{W}_{\text{out}}}\psi_t(x,y)$. We consider three signals, each computed from the weights of a single checkpoint and the query sample, without any auxiliary model: (i) the last-layer gradient norm (Nasr et al., 2019), $\psi_t = \|p - \text{onehot}(y)\|\sqrt{\|h\|^2+1}$, where p is the softmax output and h the penultimate feature; (ii) parameter sensitivity, $\psi_t = \|\nabla_\theta \ell(\theta_t;(x,y))\|_2$ over all parameters; and (iii) loss-landscape curvature, a single-probe Hutchinson estimate $\psi_t = v^\top \nabla^2_\theta \ell(\theta_t;(x,y))\, v$ with a Rademacher vector v.

The results are reported in Table 8. Across all FPR thresholds, none of the three white-box signals achieves a higher TPR than the logit-based signal, on either the full pool or the Top-30% subset. As shown in Table 9, these signals also incur substantial computational overhead, requiring $2.0\times$, $20.1\times$, and $109.1\times$ the computation time of Ghost-LiRA for gradient norm, parameter sensitivity, and curvature, respectively.

Table 8: **Substituting white-box signals into Ghost-LiRA on CIFAR-100 (ResNet-34).** All variants share Ghost-LiRA's scoring pipeline, OUT window $\mathcal{W}_{\text{out}} = \{50, 60, 70, 80\}$, and EMA weights; only the per-checkpoint signal differs. No variant trains an auxiliary model. *Overall* aggregates the 50k evaluation pool (25k members, 25k non-members); *High vuln. (Top 30%)* and *Low vuln. (Bot 30%)* stratify by vulnerability. Values are mean $\pm$ std over 10 seeds. TPR in %. Best per (subset, metric) in **bold**.

Subset	Signal	AUC	TPR@10%	TPR@1%	TPR@0.1%	TPR@0.01%
Overall	Logit (Ghost-LiRA)	$\mathbf{0.788 \pm 0.009}$	$\mathbf{49.43 \pm 1.19}$	$\mathbf{14.49 \pm 0.84}$	$\mathbf{2.98 \pm 0.47}$	$\mathbf{0.49 \pm 0.26}$
	Gradient norm	0.686 ± 0.009	20.12 ± 0.54	2.45 ± 0.24	0.23 ± 0.04	0.03 ± 0.02
	Parameter sensitivity	0.775 ± 0.005	26.40 ± 1.02	2.17 ± 0.23	0.11 ± 0.04	0.00 ± 0.00
	Curvature (k=1)	0.708 ± 0.006	6.08 ± 0.83	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
High vuln. (Top 30%)	Logit (Ghost-LiRA)	$\mathbf{0.932 \pm 0.004}$	$\mathbf{78.59 \pm 1.43}$	$\mathbf{30.08 \pm 1.82}$	$\mathbf{7.02 \pm 1.12}$	$\mathbf{1.23 \pm 0.67}$
	Gradient norm	0.862 ± 0.005	49.83 ± 1.26	6.98 ± 0.73	0.68 ± 0.12	0.09 ± 0.05
	Parameter sensitivity	0.900 ± 0.004	60.71 ± 2.33	5.92 ± 0.72	0.27 ± 0.09	0.01 ± 0.01
	Curvature (k=1)	0.775 ± 0.009	13.74 ± 1.98	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
Low vuln. (Bot 30%)	Logit (Ghost-LiRA)	0.590 ± 0.020	$\mathbf{19.85 \pm 1.47}$	$\mathbf{3.19 \pm 0.36}$	$\mathbf{0.45 \pm 0.11}$	$\mathbf{0.06 \pm 0.03}$
	Gradient norm	0.504 ± 0.012	1.54 ± 0.24	0.06 ± 0.03	0.01 ± 0.01	0.00 ± 0.00
	Parameter sensitivity	$\mathbf{0.640 \pm 0.007}$	3.24 ± 0.43	0.20 ± 0.11	0.02 ± 0.02	0.00 ± 0.01
	Curvature (k=1)	0.631 ± 0.010	0.86 ± 0.25	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00

Table 9: **Computation time of each per-checkpoint signal on CIFAR-100 (ResNet-34).** Time per target model to obtain the signal at all five checkpoints ($\mathcal{W}_{\text{out}} \cup \{T\}$), including checkpoint loading. The logit row is the end-to-end cost of Ghost-LiRA from Table 5.

Signal	Time per model (s)	Relative to logit
Logit (Ghost-LiRA)	19.3	1.0×
Gradient norm	39.32	2.0×
Parameter sensitivity	387.98	20.1×
Curvature (k=1)	2105.16	109.1×

J Robustness and design space of the OUT proxy

J.1 Quantifying the OUT-proxy approximation quality

Section 4.2 and Figure 2b provided a visual argument that mid-stage checkpoints approximate the true OUT distribution on vulnerable samples. Here we quantify this approximation with two complementary metrics, evaluated against the true OUT mean μ_{out} estimated from N=256 shadow models.

Metrics. For each member sample (x, y) we measure how well the proxy approximates μ_{out} along two axes. *Mean bias* is the average of (proxy $-$ μ_{out}) over samples in a group, quantifying the systematic offset from the true OUT distribution. A positive value indicates overestimation of μ_{out}, which reflects partial memorization at the proxy's checkpoint stage. *Pearson* ρ captures per-sample ranking agreement between the proxy and μ_{out}. Because Ghost-LiRA's performance depends on the relative ordering of $\varphi_T - \mu_{\text{Ghost}}$ across samples, ranking preservation is the property most directly tied to attack performance.

Results. Table 10 reports both metrics across all four benchmarks, stratified by vulnerability group. Two patterns are consistent across all datasets.

First, $\text{Bias}_{\text{Ghost}}$ is smaller than $\text{Bias}_{\text{Final}}$ across all groups and datasets, with a single exception on the low-vulnerability subgroup of CIFAR-10, where both biases are negligibly small (< 0.5) and the comparison is noise-dominated. Second, ρ_{Ghost} exceeds ρ_{Final} with the same exception. CIFAR-10 is the easiest benchmark in our evaluation, and its low-vulnerability samples are fit well before the OUT window, leaving both proxies similarly close to μ_{out} with little room for differentiation.

Table 10: Quantitative comparison of the Ghost proxy (μ_{Ghost}) and the final checkpoint (ϕ_T) against the true OUT mean (μ_{out}, N=256 shadow models), stratified by vulnerability group. **Bias**: mean of (proxy $-$ μ_{out}); positive values indicate overestimation due to partial memorization. $\boldsymbol{\rho}$: Pearson correlation with μ_{out}; higher indicates better per-sample ranking agreement. Within-group ρ values are typically lower than the overall ρ due to range restriction: stratification narrows the spread of μ_{out} within each group, reducing the signal available for correlation. The Ghost-vs-Final gap in both metrics is largest on the high-vulnerability subgroup, where it matters most for Ghost-LiRA's attack performance. All values: mean $\pm$ std over 10 seeds.

Dataset	Group	$\text{Bias}_{\text{Ghost}}$	$\text{Bias}_{\text{Final}}$	ρ_{Ghost}	ρ_{Final}
CIFAR-10	Overall	-0.11 ± 0.16	$+2.07 \pm 0.19$	0.68 ± 0.01	0.39 ± 0.02
	High vuln. (Top 30%)	-0.17 ± 0.13	$+5.15 \pm 0.18$	0.58 ± 0.03	0.23 ± 0.02
	Mid vuln. (40%)	-0.50 ± 0.18	$+0.98 \pm 0.18$	0.37 ± 0.02	0.32 ± 0.02
	Low vuln. (Bot 30%)	$+0.49 \pm 0.19$	$+0.46 \pm 0.21$	0.25 ± 0.01	0.34 ± 0.02
CIFAR-100	Overall	$+1.47 \pm 0.16$	$+6.76 \pm 0.23$	0.77 ± 0.01	0.25 ± 0.02
	High vuln. (Top 30%)	$+2.94 \pm 0.13$	$+10.82 \pm 0.18$	0.60 ± 0.01	0.12 ± 0.01
	Mid vuln. (40%)	$+0.90 \pm 0.15$	$+6.49 \pm 0.21$	0.51 ± 0.02	0.13 ± 0.02
	Low vuln. (Bot 30%)	$+0.76 \pm 0.21$	$+3.07 \pm 0.32$	0.49 ± 0.03	0.21 ± 0.02
Tiny ImageNet	Overall	$+1.40 \pm 0.05$	$+4.43 \pm 0.03$	0.92 ± 0.00	0.22 ± 0.02
	High vuln. (Top 30%)	$+1.99 \pm 0.05$	$+6.88 \pm 0.05$	0.80 ± 0.00	0.32 ± 0.01
	Mid vuln. (40%)	$+1.37 \pm 0.05$	$+4.57 \pm 0.04$	0.83 ± 0.00	0.40 ± 0.01
	Low vuln. (Bot 30%)	$+0.87 \pm 0.05$	$+1.79 \pm 0.03$	0.86 ± 0.00	0.35 ± 0.01
WikiText-103	Overall	$+0.66 \pm 0.02$	$+2.84 \pm 0.03$	0.80 ± 0.01	0.59 ± 0.01
	High vuln. (Top 30%)	$+1.59 \pm 0.02$	$+4.18 \pm 0.03$	0.69 ± 0.01	0.51 ± 0.01
	Mid vuln. (40%)	$+0.54 \pm 0.02$	$+2.61 \pm 0.03$	0.82 ± 0.01	0.78 ± 0.00
	Low vuln. (Bot 30%)	-0.10 ± 0.03	$+1.79 \pm 0.03$	0.89 ± 0.01	0.88 ± 0.01

The Ghost-vs-Final gap is most informative on the high-vulnerability subgroup. On CIFAR-10, CIFAR-100, and WikiText-103, this subgroup exhibits the largest bias gap between Ghost and Final (e.g., on CIFAR-100, $\text{Bias}_{\text{Ghost}} = 2.94$ versus $\text{Bias}_{\text{Final}} = 10.82$, and $\rho_{\text{Ghost}} = 0.60$ versus $\rho_{\text{Final}} = 0.12$). Tiny ImageNet is the most challenging benchmark in our evaluation, yet it achieves the highest overall proxy quality ($\rho_{\text{Ghost}} = 0.92$), with strong ranking agreement across all vulnerability groups.

The absolute value of ρ_{Ghost} varies across datasets and groups, reflecting differences in training dynamics rather than proxy design. On WikiText-103, memorization begins earlier due to fine-tuning (Figure 5c), which compresses the temporal gap between mid-stage and final checkpoints for mid- and low-vulnerability samples, yielding high ρ for both proxies in those groups. On Tiny ImageNet, the longer training schedule and cosine annealing provide a wide separation between the OUT window and the memorization phase, resulting in uniformly high ρ_{Ghost} across all groups. In all cases, the gap $\rho_{\text{Ghost}} - \rho_{\text{Final}}$ confirms that mid-stage checkpoints preserve per-sample ranking substantially better than the final checkpoint.

The difference $\varphi_T - \mu_{\text{Ghost}}$ retains a meaningful memorization signal because μ_{Ghost} has not drifted as far as φ_T from μ_{out}. On low-vulnerability samples, IN and OUT distributions largely overlap (Figures 2b and 6), and no per-sample MIA, including Ghost-LiRA, can reliably attack these samples (Figures 2a and 5, bottom-30% curve).

J.2 Sensitivity to the OUT-window placement

OUT-window placement is robust under a sparse checkpoint cadence. Figure 10 sweeps Ghost-LiRA's OUT-window bounds $(\alpha_{\text{lo}}, \alpha_{\text{hi}})$ on the sparse checkpoint pool described in Appendix F.3 (every 10 epochs for CIFAR-10/100 and Tiny ImageNet, every epoch for WikiText-103). Public model suites such as Pythia (Biderman et al., 2023) release 154 checkpoints per model over 143,000 training steps, with regular saves every 1,000 steps (roughly 0.7% of training). Our sparse setup retains checkpoints only every 5% of training (10% for WikiText-103), an order of magnitude sparser.

Table 11: **Single-checkpoint Ghost-LiRA on CIFAR-100 (sparse pool).** The model owner is assumed to save checkpoints every 10 epochs; the OUT proxy is restricted to this set $\mathcal{C}_{\text{sparse}}$. Each row selects one checkpoint at the first epoch satisfying $\text{acc}_t^{\text{val}} \geq \alpha \cdot \text{acc}_{\max}^{\text{val}}$. The sparse baseline averages over the 4 saved checkpoints within $(\alpha_{\text{lo}}, \alpha_{\text{hi}}) = (0.90, 0.97)$, namely epochs $\{50, 60, 70, 80\}$. Values are mean $\pm$ std over 10 seeds. TPR in %.

α	t^*	AUC	TPR@10%	TPR@1%	TPR@0.1%	TPR@0.01%
0.70/0.75	30	$0.823_{\pm 0.006}$	$47.77_{\pm 1.99}$	$9.04_{\pm 2.02}$	$1.16_{\pm 0.56}$	$0.14_{\pm 0.11}$
0.80/0.85	40	$0.806_{\pm 0.007}$	$47.98_{\pm 0.73}$	$10.99_{\pm 0.96}$	$1.71_{\pm 0.33}$	$0.19_{\pm 0.14}$
0.90	50	$0.790_{\pm 0.010}$	$46.92_{\pm 0.75}$	$11.33_{\pm 0.73}$	$2.03_{\pm 0.26}$	$0.28_{\pm 0.14}$
0.95	70	$0.765_{\pm 0.013}$	$45.34_{\pm 1.90}$	$11.71_{\pm 1.33}$	$2.18_{\pm 0.53}$	$0.34_{\pm 0.25}$
0.97	90	$0.753_{\pm 0.010}$	$43.82_{\pm 1.24}$	$11.24_{\pm 0.99}$	$2.09_{\pm 0.46}$	$0.37_{\pm 0.29}$
1.00	180	$0.499_{\pm 0.023}$	$9.47_{\pm 1.25}$	$0.84_{\pm 0.17}$	$0.07_{\pm 0.03}$	$0.01_{\pm 0.01}$
Sparse	$\{50, 60, 70, 80\}$	$0.788_{\pm 0.009}$	$49.43_{\pm 1.19}$	$14.49_{\pm 0.84}$	$2.98_{\pm 0.47}$	$0.49_{\pm 0.26}$

In some narrow cells (e.g., low α_{lo} paired with $\alpha_{\text{hi}} = 0.85$ on WikiText-103), no sparse checkpoint falls in the range and the cell is marked "—" in the heatmap.

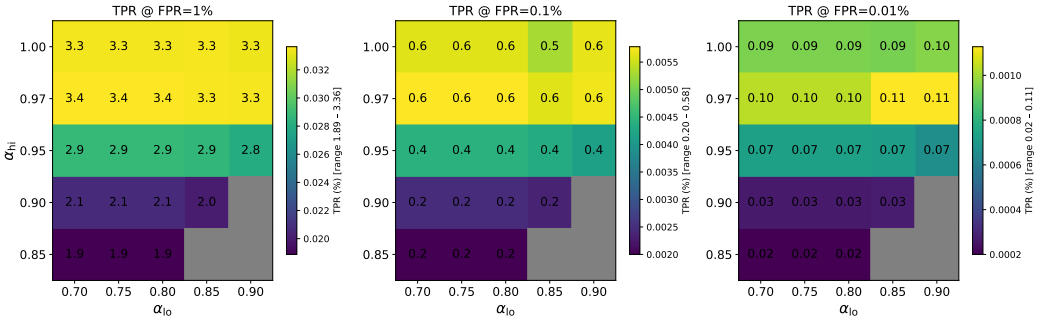
A clear plateau in performance forms at $\alpha_{\text{hi}} \in [0.95, 0.97]$ across all four benchmarks. Within each α_{hi} level, performance is essentially flat as α_{lo} varies. Across α_{hi}, performance degrades in two interpretable tail regimes: at $\alpha_{\text{hi}} \leq 0.90$, $\mathcal{W}_{\text{out}}$ is restricted to checkpoints whose underlying model has not yet generalized fully, so φ_t reflects under-fit behavior rather than the OUT distribution we intend to approximate; at $\alpha_{\text{hi}} = 1.00$, the window includes the late-memorization phase, where the OUT proxy drifts toward the IN model. The latter is sharpest on Tiny ImageNet, where the cosine schedule keeps the model near final accuracy for a wide late-training span. The setting $(\alpha_{\text{lo}}, \alpha_{\text{hi}}) = (0.90, 0.97)$ used throughout the main experiments (Appendix F.3) sits at the center of this plateau on every benchmark.

Sparse vs. dense checkpoint pools. Figure 11 reports the analogous heatmaps with $\mathcal{W}_{\text{out}}$ drawn from *every* per-epoch checkpoint ($\sim$10$\times$ more checkpoints, $\sim$50–75 on average). The plateau structure is qualitatively identical, and the dense pool serves as an upper bound on the signal accessible under finer temporal resolution. Sparse-pool TPR recovers $\sim$87% to $\sim$100% of dense-pool TPR, with sparse closest to dense on Tiny ImageNet and furthest at TPR@0.01% on CIFAR-100. We omit WikiText-103 from this analysis: its 10-epoch fine-tuning yields too few checkpoints to form a pool denser than the sparse one. Overall, the sparse pool used in our main experiments recovers most of the dense signal at $\sim$10$\times$ less storage cost.

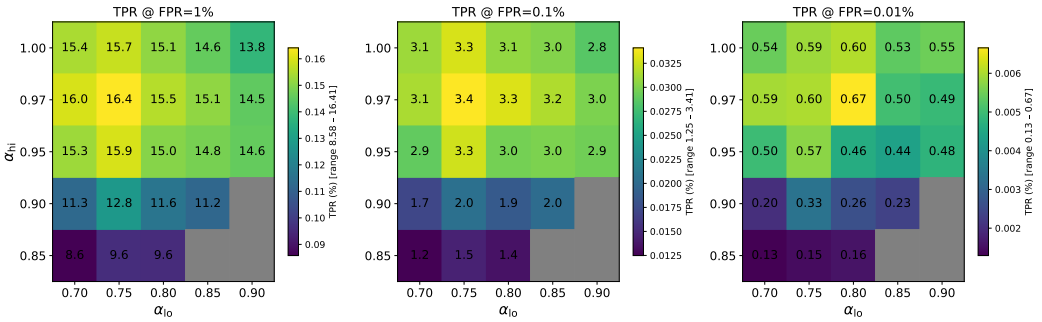
J.3 Single-checkpoint OUT proxies

A natural question following the OUT-window analysis is whether temporal averaging across multiple checkpoints is essential, or whether a *single* well-placed checkpoint already suffices. We test this by replacing μ_{Ghost} with the score of one checkpoint $t^*(\alpha) := \min\{t \in \mathcal{C}_{\text{sparse}} : \text{acc}_t^{\text{val}} \geq \alpha \cdot \text{acc}_{\max}^{\text{val}}\}$, where $\mathcal{C}_{\text{sparse}} = \{10, 20, \ldots, 200\}$ is the realistic every-10-epoch save cadence. Sweeping $\alpha \in \{0.70, 0.75, 0.80, 0.85, 0.90, 0.95, 0.97, 1.00\}$ yields six distinct epochs on the sparse pool (Table 11); the under-fitted endpoints ($\alpha \leq 0.75$) and the degenerate endpoint ($\alpha = 1.00$) are included to characterize the boundary regimes.

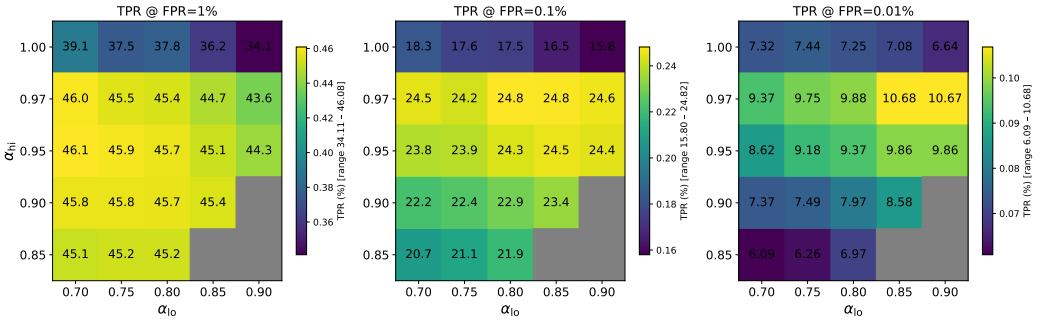
A single checkpoint suffices across a wide range of training stages. The best single-checkpoint setting ($\alpha = 0.95$, epoch 70) achieves TPR@0.1% FPR of 2.18%, recovering roughly 73% of the sparse 4-checkpoint baseline (2.98%). Temporal averaging therefore yields a modest but real gain over a well-placed single checkpoint. More notably, the single-checkpoint attack remains effective across a broad span of training. At $\alpha = 0.80$ (epoch 40, just past the onset of generalization), TPR@0.1% reaches 1.71%; even at $\alpha = 0.70$ (epoch 30, before the generalization phase has fully completed), it reaches 1.16%, recovering roughly 39% of the sparse baseline. Checkpoints past the onset of generalization recover a substantial fraction of the sparse baseline performance when used as a single OUT proxy.



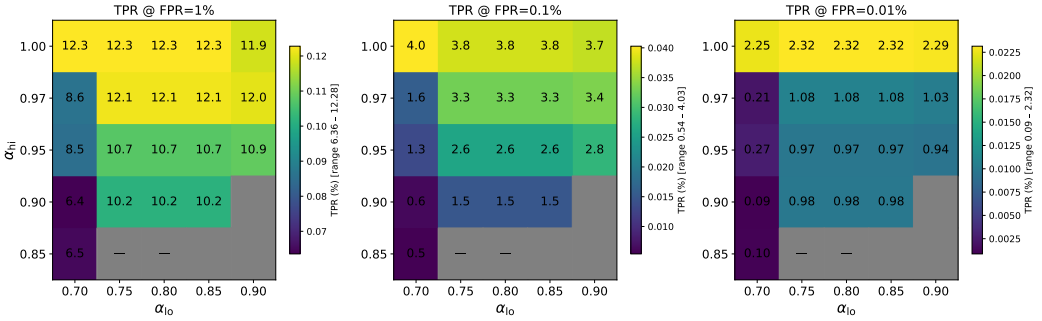
(a) CIFAR-10 (ResNet-18).



(b) CIFAR-100 (ResNet-34).

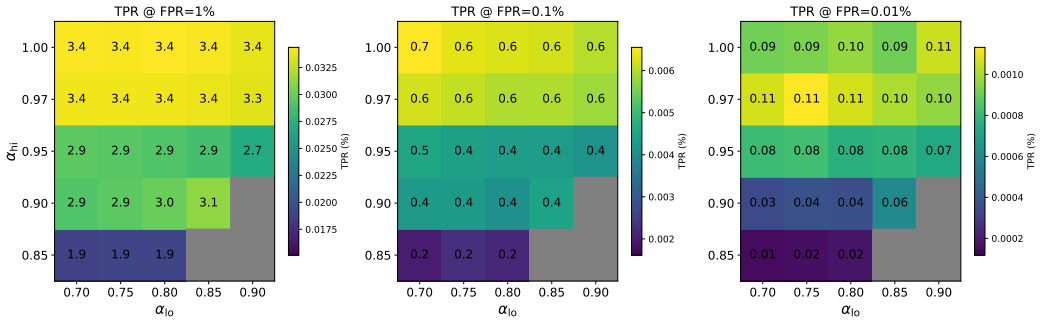


(c) Tiny ImageNet (Swin-T).

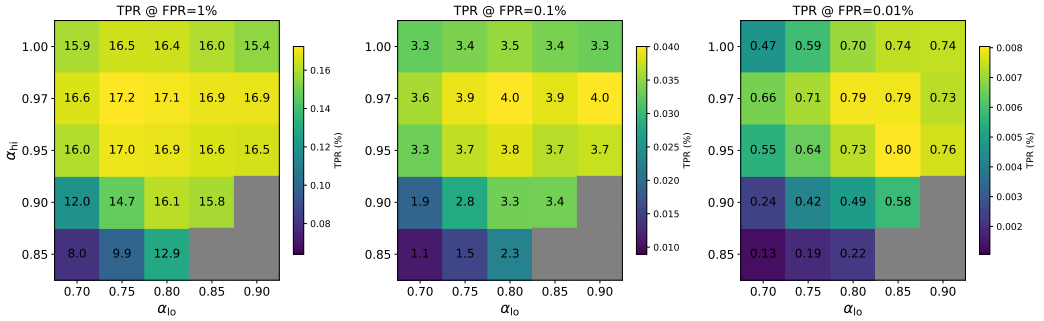


(d) WikiText-103 (GPT-2-Large).

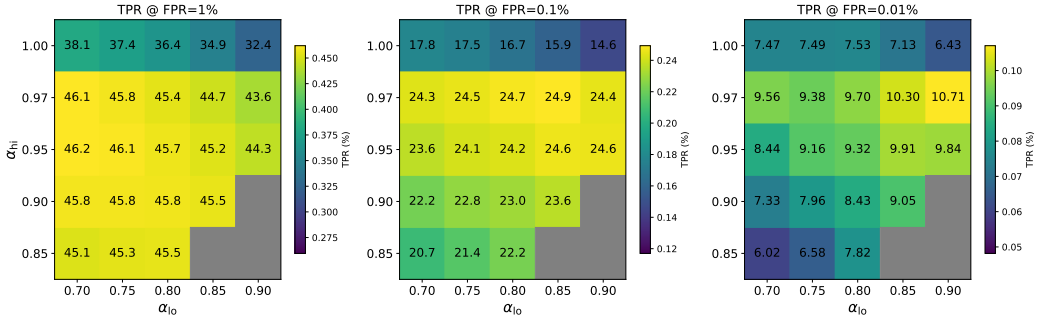
Figure 10: **Ghost-LiRA is robust to the OUT-window placement under a realistic checkpoint-saving cadence.** TPR @ 1%, 0.1%, and 0.01% FPR (left to right) on (a) CIFAR-10 (ResNet-18), (b) CIFAR-100 (ResNet-34), (c) Tiny ImageNet (Swin-T), and (d) WikiText-103 (GPT-2-Large), as a function of the window bounds $(\alpha_{\mathrm{lo}}, \alpha_{\mathrm{hi}})$. The OUT window $\mathcal{W}_{\mathrm{out}}$ contains every saved checkpoint within the alpha range, where checkpoints are assumed saved at the standard cadence used by modern training pipelines—every 10 epochs for the three vision benchmarks and every epoch for WikiText-103 (which trains for only 10 epochs). Gray cells correspond to invalid configurations where $\alpha_{\mathrm{lo}} \geq \alpha_{\mathrm{hi}}$; cells marked "—" indicate that no saved checkpoint falls in the alpha range for that cell. A clear plateau in performance forms at $\alpha_{\mathrm{hi}} \in [0.95, 0.97]$ across all four benchmarks.



(a) CIFAR-10 (ResNet-18).



(b) CIFAR-100 (ResNet-34).



(c) Tiny ImageNet (Swin-T).

Figure 11: **Dense per-epoch checkpoints provide a moderate upper bound on Ghost-LiRA's signal.** Companion to Figure 10 (sparse pool). TPR @ 1%, 0.1%, and 0.01% FPR (left to right) on the three image benchmarks (WikiText-103 is omitted due to its limited number of checkpoints), but with $\mathcal{W}_{\text{out}}$ allowed to draw from *every* per-epoch checkpoint within the alpha range. The resulting OUT window is roughly $10\times$ larger than the sparse pool ($\sim$50–75 checkpoints on average, vs. $\sim$5–8). Gray cells correspond to invalid configurations where $\alpha_{\text{lo}} \geq \alpha_{\text{hi}}$. The plateau structure is qualitatively identical to the sparse-pool counterpart—the same plateau shape, the same two tail regimes where signal degrades—but absolute TPR values are roughly 10–15$\%$ higher on average. The sparse pool used in the main results thus recovers most of the dense signal at roughly $10\times$ less storage cost.

Boundary regimes. Performance degrades smoothly outside the $\alpha \in [0.90, 0.97]$ plateau, in two interpretable directions. At $\alpha \leq 0.75$, the selected checkpoint ($t^* = 30$) lies before the generalization phase, and attack performance is correspondingly lower than at post-generalization checkpoints. At $\alpha = 1.00$, the rule selects $t^* = 180$, which sits deep within the memorization phase and is essentially indistinguishable from the final checkpoint T; the score $\varphi_T - \varphi_{t^*}$ collapses (TPR@0.1% FPR drops to 0.07%) and the attack reverts to random guessing. These two boundaries align with our central hypothesis. Vulnerable samples are best detected at checkpoints that have stabilized into generalization but have not yet entered the memorization phase.

Implications for the threat model. Access to an intermediate checkpoint of the target model is a restrictive assumption, as such checkpoints are rarely published intentionally. Nonetheless, the single-checkpoint results in Table 11 show that, if an adversary does obtain even one checkpoint, Ghost-LiRA can mount a meaningful attack. Meaningful performance is maintained across checkpoints spanning from shortly after the onset of generalization ($\alpha = 0.70$) through the late pre-memorization phase ($\alpha = 0.97$). In practice, this means that an adversary does not need to identify the optimal checkpoint or know the training schedule: any checkpoint that postdates early training and predates final convergence can serve as an OUT proxy, without knowledge of the target's training distribution, architecture, or hyperparameters.

J.4 OUT PROXIES FROM AN ADVERSARY-TRAINED MODEL

An external adversary may face limited access to the intermediate checkpoints of the target model. In such a setting, the adversary can obtain checkpoints by training a single model itself. This setting follows the threat model of per-sample MIAs such as LiRA and RMIA (Section 2): the adversary knows the target model's training recipe and has access to data drawn from the same distribution as the target model's training data. Since the adversary trains this model itself, it has access to the model's training and validation accuracies, and can therefore select the OUT window of Eq. 9 in the same way a model owner would. Ghost-LiRA then applies at the cost of training a single model.

This setting still differs from shadow-model calibration in per-sample MIAs in two respects. First, per-sample MIAs query the final checkpoint of each reference model, whereas Ghost-LiRA takes the final checkpoint only from the target model and draws its reference from the intermediate checkpoints of the trained model. Second, the trained model serves solely as an OUT reference. Because its training set is drawn from the same distribution, it may share samples with the training set of the target model. LiRA and RMIA handle such overlap by recording which samples each reference model was trained on and assigning each model to the IN or OUT side of every query. Ghost-LiRA does not track this overlap. It treats the selected mid-stage checkpoints as OUT for every query, including samples on which the trained model was in fact trained. This relies on the observation in Section 4 that vulnerable samples are memorized only after the OUT window, so that even a checkpoint whose training set contains the query behaves close to an OUT model on it. The membership score is computed as in Eq. 11, with φ_T taken from the target model and μ_{Ghost} from the checkpoints of the adversary's model.

Setup. We evaluate this setting on CIFAR-100 (ResNet-34) over 10 seeds. For each target model, the adversary's model is one of the other nine models in our pool: it is trained with the same recipe on a different random 25k subset of the same 50k pool and with a different seed. The adversary selects the OUT window from its own validation accuracy with $(\alpha_{\text{lo}}, \alpha_{\text{hi}}) = (0.90, 0.97)$ on the every-10-epoch checkpoint pool, which yields 4.9 checkpoints on average.

Results. Table 12 reports the results. The separation between vulnerable and safe samples is preserved: TPR@0.1%FPR is 18.12% on the top-30% subset against 0.77% on the bottom-30% subset. Ghost-LiRA therefore does not depend on properties specific to the target model's own training trajectory, and extends to external adversaries under the standard assumptions of per-sample MIAs at the cost of training one model. Because the adversary's model is trained on a different subset of the data, it has not seen a large share of the target model's members, and its checkpoints are true OUT models for these samples. The target model's own checkpoints, by contrast, come from a trajectory that has seen every member. This difference in training data is reflected in the higher TPR of the adversary-trained variant, 7.48% against 2.98% at 0.1% FPR on the full pool. The

Table 12: **Ghost-LiRA with OUT proxies from an adversary-trained model on CIFAR-100 (ResNet-34).** *Target ckpts* uses the target model's own checkpoints (model-owner setting, trains no model). *Adversary-trained* uses the checkpoints of one model trained by the adversary on a different 25k subset of the same 50k pool, with a different seed and the same recipe; its OUT window is selected from its own validation accuracy with $(\alpha_{\text{lo}}, \alpha_{\text{hi}}) = (0.90, 0.97)$ on the every-10-epoch pool. Both variants take φ_T from the final target model. *Overall* aggregates the 50k evaluation pool (25k members, 25k non-members); *High vuln. (Top 30%)* and *Low vuln. (Bot 30%)* stratify by vulnerability. Values are mean $\pm$ std over 10 seeds. TPR in %.

Subset	OUT proxy	Models trained	AUC	TPR@10%	TPR@1%	TPR@0.1%	TPR@0.01%
Overall	Target ckpts	0	0.788 ± 0.009	49.43 ± 1.19	14.49 ± 0.84	2.98 ± 0.47	0.49 ± 0.26
	Adversary-trained	1	0.837 ± 0.004	59.46 ± 0.65	23.21 ± 0.91	7.48 ± 0.82	1.98 ± 0.80
High vuln. (Top 30%)	Target ckpts	0	0.932 ± 0.004	78.59 ± 1.43	30.08 ± 1.82	7.02 ± 1.12	1.23 ± 0.67
	Adversary-trained	1	0.961 ± 0.003	88.88 ± 0.92	47.73 ± 2.38	18.12 ± 2.07	5.32 ± 2.13
Low vuln. (Bot 30%)	Target ckpts	0	0.590 ± 0.020	19.85 ± 1.47	3.19 ± 0.36	0.45 ± 0.11	0.06 ± 0.03
	Adversary-trained	1	0.652 ± 0.014	25.14 ± 1.42	4.39 ± 0.46	0.77 ± 0.10	0.16 ± 0.08

Table 13: **EMA ablation on CIFAR-100 (ResNet-34).** Ghost-LiRA performance with and without EMA smoothing (decay 0.999), using the same sparse checkpoint pool (every 10 epochs, $\mathcal{W}_{\text{out}} = \{50, 60, 70, 80\}$). Values are mean $\pm$ std over 10 seeds. TPR in %. Best per metric in **bold**.

Variant	AUC	TPR@10%	TPR@1%	TPR@0.1%	TPR@0.01%	TPR@0.001%
EMA	0.788 ± 0.009	49.43 ± 1.19	$\mathbf{14.49} \pm 0.84$	$\mathbf{2.98} \pm 0.47$	$\mathbf{0.49} \pm 0.26$	0.22 ± 0.14
No-EMA	$\mathbf{0.818} \pm 0.012$	$\mathbf{52.10} \pm 1.00$	13.85 ± 1.10	2.65 ± 0.42	0.46 ± 0.21	$\mathbf{0.23} \pm 0.11$

improvement, however, is not free. The adversary must train one model and hold data and a training recipe matching those of the target, whereas the model-owner setting requires no training at all.

J.5 Sensitivity to EMA smoothing

Our main experiments on the image benchmarks maintain an EMA of model weights (decay 0.999). To verify that Ghost-LiRA does not depend on this choice, we repeat the CIFAR-100 (ResNet-34) evaluation using checkpoints from the active SGD weights directly, with all other settings held fixed.

Table 13 reports the results. The two variants are comparable across the privacy-critical low-FPR regime: TPR@0.1%FPR is 2.98% with EMA versus 2.65% without, and the differences fall well within the combined standard deviation at all thresholds below 1%FPR. The non-EMA variant achieves a higher AUC (0.818 vs. 0.788), but as discussed in Section 1, AUC integrates over the high-FPR range that is less informative for worst-case privacy risk.

This result is expected. Ghost-LiRA's signal relies on vulnerable samples being memorized late in training (Section 4), a property of SGD dynamics that has been observed independently of EMA (Arpit et al., 2017; Mangalam & Prabhu, 2019; Jiang et al., 2021). Consistent with this, WikiText-103, which uses no EMA, also yields competitive Ghost-LiRA performance in the main experiments (Section 6).

K Hidden leakage across unlearning algorithms

This appendix extends the unlearning evaluation of Section 7 to two additional algorithms: Neg-Grad+ (Kurmanji et al., 2023) and SalUn (Fan et al., 2024). SCRUB and SalUn are currently regarded as state-of-the-art unlearning methods. Results for all three algorithms (including SCRUB from the main text) are reported together in Table 15.

Ghost-LiRA configuration for unlearning evaluation. The OUT window $\mathcal{W}_{\text{out}}$ and its associated checkpoints are taken from the original training trajectory, identical to the configuration described in Appendix F.3. Only the final-model confidence $\varphi_T(x, y)$ is replaced with the output of the unlearned

model. This reflects the intended use case: the model owner evaluates whether unlearning has successfully removed the membership signal that was present at the end of original training.

Unlearning hyperparameters. Table 14 reports the hyperparameters used for each unlearning algorithm. All hyperparameters were selected via grid search on a single held-out seed and then fixed across the remaining 9 seeds.

Test accuracy confirms sufficient unlearning pressure. Both SCRUB (91.52%) and SalUn (90.96%) achieve test accuracy well below the retrained reference (92.11%), indicating that the residual leakage is not due to insufficient unlearning effort. NegGrad+ (92.04 $\pm$ 0.41%) has a mean only 0.07 percentage points below the retrained reference (92.11 $\pm$ 0.18%), and the two distributions overlap considerably. Nevertheless, the substantial residual leakage detected by Ghost-LiRA (TPR@1%FPR 22.24% vs. 0.06% retrained) cannot be attributed to the model retaining too much of its original fit. SalUn, in fact, over-forgets by more than a full percentage point yet still leaves detectable signal on the vulnerable subgroup (Ghost-LiRA TPR@1%FPR 3.77%), degrading utility without eliminating leakage where it matters most.

Implications for unlearning evaluation and design. The pattern in Table 15 mirrors the main claim of Section 3[9]. All three algorithms drive aggregated Loss attack AUC below 0.5, which would conventionally indicate successful forgetting. However, per-sample MIAs expose substantial residual leakage on the vulnerable subgroup, with TPR@1%FPR an order of magnitude above the retrained reference. Prior unlearning evaluations relying on aggregated population-based MIAs may therefore have systematically overestimated the degree of forgetting, especially on the privacy-sensitive samples where reliable evaluation matters most. The implication extends to algorithm design. Unlearning hyperparameters tuned against a population-based MIA signal optimize for a metric that can be uninformative, or even inverted, on the vulnerable subgroup. Such configurations may appear optimal in aggregate while leaving substantial leakage on the samples that most need to be forgotten.

Table 14: Hyperparameters for unlearning algorithms (CIFAR-10, ResNet-18).

Algorithm	Key hyperparameters
NegGrad+	lr=0.01, epochs=5, α=0.999
SCRUB	lr=0.001, epochs=4, msteps=5, T=4.0, γ=0.99, α=0.001
SalUn	lr=0.01, epochs=10, mask_ratio=0.5

[9] As shown in Appendix D.2, all final-output scores (e.g., modified entropy, max softmax confidence) are equally indistinguishable in the region where inversion occurs, so the choice among them does not affect our conclusions.

[10] Throughout this table, forget-set samples are labeled as positives under all three model states. For the retrained model, these samples are true non-members, so the reported TPR functions as a null baseline. Any excess of the unlearned model's TPR over this baseline reflects residual membership traces that the unlearning algorithm failed to remove.

Table 15: **Aggregated evaluation conceals residual leakage on vulnerable samples across multiple unlearning algorithms.** MIA performance before and after machine unlearning on CIFAR-10 (ResNet-18, 10 seeds, mean ± std). *Original/Retrained*: target model before unlearning, and the from-scratch reference without the forget set. *Aggregated*: population-level evaluation as in prior unlearning work. *High-Vulnerable*: top-5% most vulnerable samples. Red: signal inversion—Loss on the original model already underperforms a random guess on the vulnerable subgroup (Section 3). Underline (Aggregated): residual leakage that survives unlearning even under aggregated evaluation, consistent with the observation of Hayes et al. (2025) that population-based MIAs may yield a false sense of privacy in unlearning evaluation. **Bold** (High-Vulnerable): our vulnerability-stratified protocol extends this observation to the per-sample regime, exposing substantially larger residual leakage concentrated on the vulnerable subgroup, which aggregated evaluation dilutes by averaging with typical samples. Across NegGrad+, SCRUB, and SalUn, Loss AUC drops below 0.5, falsely signaling success. Yet per-sample MIAs (LiRA, Ghost-LiRA) expose persistent leakage on vulnerable samples.

			Aggregated (prior work)			High-Vulnerable (top 5%)		
State	Test Acc. (%)	MIA Attack	AUC	TPR@1%	TPR@0.1%	AUC	TPR@1%	TPR@0.1%
Original	92.66 ± 0.27	Loss	0.536 ± 0.006	1.14 ± 0.22	0.10 ± 0.04	0.441 ± 0.010	0.65 ± 0.39	0.03 ± 0.05
		LiRA online	0.788 ± 0.002	46.09 ± 0.35	15.24 ± 2.13	0.987 ± 0.003	91.79 ± 0.66	30.36 ± 4.25
		Ghost-LiRA (Ours)	0.730 ± 0.004	15.43 ± 1.72	3.58 ± 0.95	0.936 ± 0.006	30.32 ± 3.54	7.15 ± 1.89
Retrained[10]	92.11 ± 0.18	Loss	0.323 ± 0.003	0.63 ± 0.11	0.09 ± 0.07	0.051 ± 0.001	0.00 ± 0.00	0.00 ± 0.00
		LiRA online	0.338 ± 0.005	0.52 ± 0.27	0.08 ± 0.05	0.113 ± 0.009	0.65 ± 0.26	0.09 ± 0.08
		Ghost-LiRA (Ours)	0.348 ± 0.004	0.15 ± 0.07	0.01 ± 0.03	0.198 ± 0.007	0.06 ± 0.05	0.00 ± 0.00
After NegGrad+	92.04 ± 0.41	Loss	0.484 ± 0.009	0.80 ± 0.20	0.10 ± 0.06	0.345 ± 0.014	0.18 ± 0.11	0.00 ± 0.00
		LiRA online	0.755 ± 0.014	39.44 ± 1.95	11.19 ± 1.02	**0.924** ± 0.027	**78.47** ± 3.99	**22.29** ± 2.07
		Ghost-LiRA (Ours)	0.712 ± 0.010	11.29 ± 1.65	2.71 ± 0.76	**0.902** ± 0.018	**22.24** ± 3.36	**5.41** ± 1.51
After SCRUB	91.52 ± 0.36	Loss	0.417 ± 0.007	0.42 ± 0.14	0.07 ± 0.04	0.227 ± 0.013	0.00 ± 0.00	0.00 ± 0.00
		LiRA online	0.589 ± 0.016	6.56 ± 1.89	0.67 ± 0.18	**0.610** ± 0.032	**12.29** ± 3.75	**1.22** ± 0.38
		Ghost-LiRA (Ours)	0.597 ± 0.018	3.76 ± 0.88	0.82 ± 0.29	**0.676** ± 0.037	**6.67** ± 1.90	**1.59** ± 0.58
After SalUn	90.96 ± 0.31	Loss	0.336 ± 0.004	0.43 ± 0.09	0.07 ± 0.06	0.144 ± 0.007	0.01 ± 0.02	0.00 ± 0.00
		LiRA online	0.441 ± 0.008	3.42 ± 0.64	1.03 ± 0.29	0.364 ± 0.016	**5.66** ± 1.21	**1.98** ± 0.56
		Ghost-LiRA (Ours)	0.564 ± 0.008	1.93 ± 0.42	0.23 ± 0.11	0.719 ± 0.019	**3.77** ± 0.80	**0.46** ± 0.21