SHARPMIA: CRITICAL SHARPNESS BEYOND THE INPUT HESSIAN FOR MEMBERSHIP INFERENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Membership inference seeks to answer if a given *example* is in a model's training set. It is both an instrument to measure a model's data retention and an attack vector to leak private data. Nearly every attack against vision models reads the model at the example itself and differs only in what it compares that reading against; the one line of work that moves the reading into the loss geometry takes the trace of the input Hessian to extract curvature, still at the example. We find that the loss carries a membership signal not only at the example but also in the local geometry of its neighborhood. To characterize the geometry and derive a signal, we consider the changes in loss over a finite distance across the example to extract *critical sharpness*, λ_c—this is effectively how far one must walk from the example before the loss returns to its starting value. A walk needs a direction. Consequently, computing λ_c needs the target model's gradient. But, we show that it is not necessary. So the signal poses a problem by facilitating a black-box attack under Carlini et al. (2022)'s offline, black-box threat model. Effectively, a walking direction can be averaged from the adversary's access to reference models without the exampleto approximate λ_c. Interestingly, we find the critical sharpness and prior work's curvature evidence are strong on different examples. So, we fuse both signals together with a difficulty prior we introduce to aid signal fusion. Reproducing 11 prior attacks, and evaluating our method, SHARPMIA, with standard models and benchmarks, we show that: *i) sharpness leaks membership information; and ii) combining it leads to the strongest attack*.

1 INTRODUCTION

Understanding what information a machine learning model can reveal about its training data is a critical concern for models trained on private, sensitive, and copyrighted data (Hu et al., 2022). Membership inference (Shokri et al., 2017) seek to ascertain the exposure of a chosen example datum. Given a target model and a candidate example, the objective is to probe the target to establish whether the example was in the model's training set. Its use is not limited to attacking models. Extraction pipelines use a membership test to isolate the examples memorized by a model (Carlini et al., 2023), it is used as an instrument for auditing machine unlearning (Chen et al., 2021), and it can establish if a model was trained on data it had no right to use (Murakonda & Shokri, 2020).

Our work investigates a new signal source to infer membership—*the critical sharpness of the curvature of the loss with respect to the input*. Critical sharpness is simply a walk over the curvature along a negative gradient to understand the loss geometry away from a test example. Interestingly, we show a method exists to read critical sharpness in a blackbox threat model setting.

In general, black-box attacks exploit the gap between the model's behavior on members (training examples) and on non-members. The simplest attacks in black box threat models threshold the loss on an example (Yeom et al., 2018) or a modified entropy of the output (Song & Mittal, 2021). Stronger attacks train *shadow* models or *reference models* (Shokri et al., 2017) on data from a similar distribution to training data. The scores from the reference models that did not train on the example are used to ask whether the targeted model's score on the example fits (Watson et al., 2022; Ye et al., 2022; Carlini et al., 2022; Zarifzadeh et al., 2024). Recent attacks feed several scores to train a classifier (He et al., 2024; Shi et al., 2024) or train the references to imitate the target (Du et al.,

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.

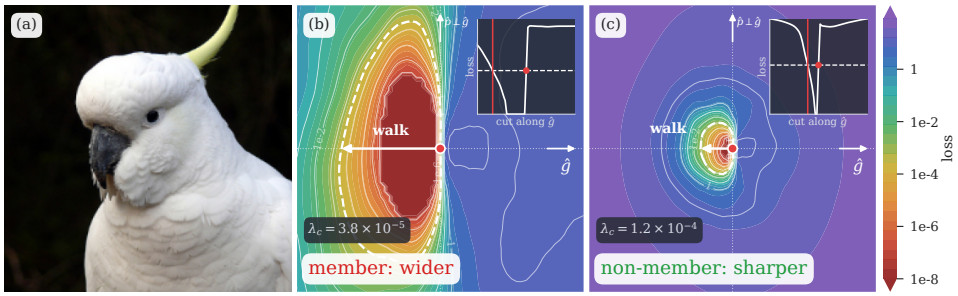

Figure 1: The contour plots in (b) and (c) shows the loss in the *input* space around the example x of a Cockatoo in (a) for two ResNet-50 models, one trained with it (member) and one without it (non-member) on the ImageNet task. The two contour plots share the same scale for ease of comparisons. $\hat{g}$ is the input gradient direction at x, $\hat{p}$ is a direction orthogonal to it, and each point on the plots in (b) & (c) is a perturbed image, colored by its loss. The inset at the top right of each panel cuts the plane along $\hat{g}$—the loss along the **walk**—and shows the starting loss $\ell(x)$ (dashed) and the point η_c where the loss returns to it (red vertical line). Walking from x (●) against the gradient $\hat{g}$, the loss first falls and then climbs to $\ell(x)$ (dashed line in the inset). The length of the **walk**, measured in steps of the gradient, is η_c and the critical sharpness is $\lambda_c = 2/\eta_c$. We can clearly observe the sharpness (the walk) is wider for a member and shorter or sharper for a non-member. Later, Figure 3 show the same sharpness relationships between member and non-members exist on all *benchmarks*.

2026a). *Notably, all the methods examine a signal from a point estimate—a model's output at the example's coordinates or at a fixed set of its augmentations—to assess membership.*

A separate line of inquiry investigates the curvature of the loss with respect to the input because of its links to memorization (Garg et al., 2024; Ravikumar et al., 2024a). (Ravikumar et al., 2024b) turned this link into an attack that estimates the trace of the input Hessian or curvature from target model outputs. The trace is still a point estimate—a description of how the loss *bends at the example*.

In contrast, we hypothesize that the local geometry of the loss landscape surrounding an example provides further evidence of membership. The hypothesis is motivated by the observation that training on an example can reduce the loss not only at the example itself, but also in its local neighborhood. Consequently, the geometry of the example's neighborhood loss landscape may capture stronger evidence to assert membership.

We consider assessing the geometry of the neighborhood using the *critical sharpness of the loss with respect to the input*. As we illustrate with a "Cockatoo" example in Figure 1, we take a walk from the example along the negative gradient, record the step η_c at which the loss returns to its starting value, and read $\lambda_c = 2/\eta_c$. Then, the distance of the walk effectively provides a concise description of the geometry surrounding the example to yield a discriminative signal capable of separating members from non-members.

Surprisingly, the *walk* does not need the target model's gradient to follow. A natural direction to walk is indeed the target model's own negative gradient. Obtaining the gradient is a simple task for a white-box adversary but a difficult one for black-box adversary. Interestingly, we find the walk *is* possible with only model outputs and without accessing model parameters in the settings of a black-box threat model in prior work. We observe, the input Hessian is *effectively rank one* where the target model is confident of the presented test example. So, near the example, the loss bends along any direction as it does along the gradient but scaled by a single cosine factor and in a direction that can be averaged from the adversary's reference models. In fact, the ordering of the gradient readings from the target model compared to the reference model approach yields a Spearman correlation coefficient of 0.98 as we show in Appendix A.

Consequently, the critical sharpness signal can be read under the black-box threat model in prior work (Shokri et al., 2017; Carlini et al., 2022; Watson et al., 2022; Zarifzadeh et al., 2024)—simply reading the target model outputs and consulting offline with reference models.

Fusing critical sharpness and curvature yields stronger evidence. Our work suggests the critical sharpness and the curvature of Ravikumar et al. (2024b) are complementary as neither leads on every

example and the strength of membership evidence depends on the *difficulty* of the example. For instance, with CIFAR-10, the curvature or trace information is more informative on the low $\mathcal{D}(z)$ examples and critical sharpness on the rest—see Table 2. Consequently we propose fusing the two signals and providing a measure of difficulty, offline, using reference models never trained on the example. We refer to the prior as $\mathcal{D}(z)$. Importantly, $\mathcal{D}(z)$ reads nothing from the target model and is derived only from the adversary's reference models. Following the signal fusion strategies of He et al. (2024) and Shi et al. (2024), we propose learning to fuse readings. Under the black-box threat model, the fused attack, SHARPMIA, using curvature and sharpness information conditioned on the difficulty prior perform at or above state-of-the-art methods.

Contributions. We summarize the contributions from our study as:

- **Critical sharpness leaks membership.** We formalize the critical sharpness of the loss landscape with respect to inputs and demonstrate it leaks membership information (§3.2, Figure 3).
- **Sharpness is accessible in a black-box threat model.** We show sharpness can be computed without requiring the target model's gradient. A direction averaged from the adversary's own reference models can be walked instead with little effect on success (within 0.05 AUROC difference shown in Appendix A with §3.3 explaining the reason for the substitution's effectiveness).
- **Fusing complementary evidence of sharpness and curvature results in the strongest attack.** We find sharpness and curvature are complementary. Using critical sharpness with the complementary signals from the curvature, conditioned by the difficulty prior we introduce, yields the strongest black-box attack—the proposed SHARPMIA attack (§3).

Scope and findings. To our knowledge, ours is a first demonstration of critical sharpness leaking membership information and the combination of signals under an offline *difficulty* prior. All of the attacks are evaluated under the black-box, offline threat model of prior work detailed in §3.1 for a fair comparison. On CIFAR-10 and CIFAR-100, SHARPMIA has the highest AUROC and the highest TPR at 1% and 0.1% FPR of the attacks in Table 1, this includes learned-fusion attacks of He et al. (2024) and Shi et al. (2024). And on ImageNet it is on par or better than the strongest of the existing method for inference. Table 1 summarises the full comparison.

2 RELATED WORK

We examine prior methods and their information sources below.

Signals & readouts at the example. Under widely adopted black box settings, Shokri et al. (2017) introduced shadow models or *reference models*—the setting we also use. Yeom et al. (2018) threshold the loss and Song & Mittal (2021) use a modified entropy. Most later attacks compare that reading against reference models, which are split into those trained on an example and those not. Watson et al. (2022) subtract the mean of the latter group and Ye et al. (2022) take a percentile, Sablayrolles et al. (2019) read both groups and set the per-example loss threshold that best separates them, Carlini et al. (2022) read both groups and form a per-example likelihood ratio, Zarifzadeh et al. (2024) compare the example against a population of held-out points, and Bertran et al. (2023) replace the reference models by a quantile regressor. Then, in label-only attacks, the predicted class alone is used to extract information (Choquette-Choo et al., 2021; Li & Zhang, 2021). In each, the comparison serves to normalize the targeted model's own reading. Notably, in addition, white-box attacks also read the target's gradients and activations (Nasr et al., 2019) for membership inference.

Learned signal fusion. He et al. (2024) give a raw score and the calibrated score to a four-layer MLP. Shi et al. (2024) give a target score, a calibrated score and the class label to a small network, and weight the calibrated score by a neighbourhood factor computed from cosine similarity in an embedding space. Benefiting from prior work, we also adopt a combiner (§3.4). Appendix C describes how our reproduction departs from prior work.

Methods altering the reference models. Recently, approaches investigate changes to the reference models instead of pursuing new signal sources. Du et al. (2026b) retrain the reference models conditioned on the inferred membership of other examples, and Du et al. (2026a) train them to imitate the target directly. This axis of investigations is orthogonal to ours, since it changes the reference pool and not the readout. Du et al. (2026a) trains models against the target, and at its own 18 augmentations. The method has the highest TPR at low FPR on CIFAR-10. Table 1 includes it, and Appendix D compares with it at its own evaluation setting.

Curvature and sharpness. Garg et al. (2024) link the curvature of the loss with respect to the input to memorization, and Ravikumar et al. (2024b) built the direct predecessor to our attack by reading the trace of the input Hessian with a zero-order stencil for a black-box setting. Although sharpness is not investigated for membership inference, it exists in prior work but as the sharpness of the loss with respect to the *parameters* as opposed to the input space investigated in our work. Notably, Moosavi-Dezfooli et al. (2019) measure curvature with a finite step rather than in the limit, a published precedent for measuring curvature over a finite distance, as λ_c does in our work. Then, sharpness of the loss in parameter space is a long-standing generalization signal (Keskar et al., 2017; Foret et al., 2021), and Cohen et al. (2021) show that gradient descent settles at a sharpness of $2/\eta$, the identity λ_c inverts. Then, the concurrent study Kalra et al. (2026) turn that identity into a reading by walking along the optimizer's update at a training checkpoint, and λ_c is their construction moved to input space, where the model is fixed and the example moves.

Memorization and difficulty. Feldman (2020); Feldman & Zhang (2020) define memorization as the change in a model's prediction on an example when it is removed from training. Example difficulty measured this way is used to calibrate attacks (Watson et al., 2022), but we are not aware of a prior attack using difficulty measures from reference models to support signal fusion.

3 METHOD

3.1 NOTATION AND THREAT MODEL

We work in input space throughout. A model $f_{\boldsymbol{\theta}}$ maps an image $\boldsymbol{x} \in \mathbb{R}^d$ to logits, and for an example $z = (\boldsymbol{x}, y)$ we write $\ell(\boldsymbol{x}) = \ell(f_{\boldsymbol{\theta}}(\boldsymbol{x}), y)$ for its cross-entropy loss, $\boldsymbol{g} = \nabla_{\boldsymbol{x}}\ell(\boldsymbol{x})$ for the input gradient and $\boldsymbol{H} = \nabla^2_{\boldsymbol{x}}\ell(\boldsymbol{x})$ for the input Hessian, with $d = 3{\times}32{\times}32$ on CIFAR and $d = 3{\times}224{\times}224$ on ImageNet. Every quantity we read varies the input $\boldsymbol{x}$ with the weights $\boldsymbol{\theta}$ held fixed, unlike the parameter-space sharpness of Kalra et al. (2026), which varies $\boldsymbol{\theta}$. The adversary trains a pool of reference models $\mathcal{R} = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_N\}$ with training sets $D_1, \ldots, D_N$ that it chose itself, and attacks a target $\boldsymbol{\theta}_\star$ whose training set $D_\star$ it does not know, except that it is drawn from the same distribution as the adversary's own data. For an example z, the reference models not trained on it are

$$\mathcal{O}(z) \;=\; \{\, m \;:\; z \notin D_m \,\}. \tag{1}$$

We write $R(\boldsymbol{g}) = \boldsymbol{g}^\top \boldsymbol{H}\boldsymbol{g}/\|\boldsymbol{g}\|^2$ for the curvature at the example along $\boldsymbol{g}$, λ_c for the critical sharpness of §3.2, $\lambda_{\boldsymbol{v}}$ for the critical sharpness taken along a probe direction $\boldsymbol{v}$ in §3.3, and $\mathcal{D}(z)$ for the difficulty prior built in §3.4.

Membership inference threat model. Given a target model $f_{\boldsymbol{\theta}_\star}$ and an example z, the adversary decides whether $z \in D_\star$. They submit inputs and observes the output probabilities, with no access to the target's parameters or gradients (Shokri et al., 2017; Carlini et al., 2022). As in prior work (Carlini et al., 2022; He et al., 2024; Zarifzadeh et al., 2024; Du et al., 2026a), the adversary can sample data from the training distribution and knows the target's architecture, so trains the N reference models and knows both their parameters and their training sets. The attack is offline (Watson et al., 2022; Carlini et al., 2022): the reference models are trained once, and an example is scored only against $\mathcal{O}(z)$, so every new sample needs no further training. We do not consider white-box (Nasr et al., 2019) or label-only (Choquette-Choo et al., 2021; Li & Zhang, 2021) settings.

3.2 CRITICAL SHARPNESS

We want to know how sharp the basin of the loss around an example is. Let $\psi(\eta) = \ell(\boldsymbol{x} - \eta\boldsymbol{g})$ be the loss along the negative *unnormalized* gradient. To second order, it is a parabola,

$$\psi(\eta) \;\approx\; \ell(\boldsymbol{x}) - \eta\,\|\boldsymbol{g}\|^2 + \tfrac{1}{2}\,\lambda\,\eta^2\,\|\boldsymbol{g}\|^2, \tag{2}$$

whose value $\ell(\boldsymbol{x})$ and slope $-\|\boldsymbol{g}\|^2$ at $\eta = 0$ are fixed by the example. The curvature λ along $\boldsymbol{g}$ sets the sharpness of the basin. Setting $\psi(\eta) = \psi(0)$ in (2), the parabola returns to its starting value, besides $\eta = 0$, at

$$\eta \;=\; 2/\lambda, \tag{3}$$

so a sharper basin returns sooner. At the example the curvature is $\lambda = R(\boldsymbol{g})$, and the parabola predicts a return at $2/R(\boldsymbol{g})$ (Figure 2*(a)*, ❶).

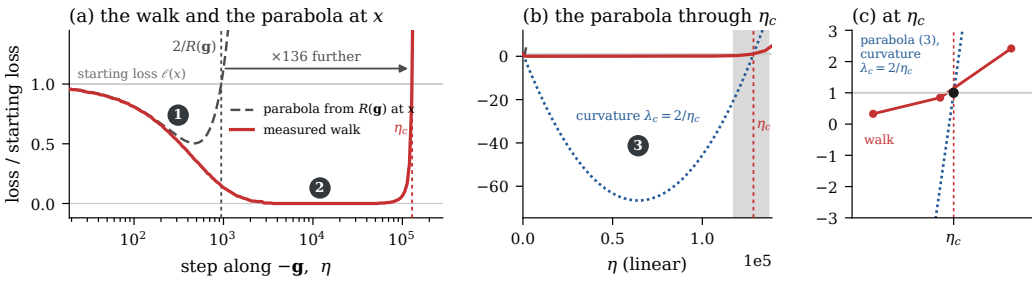

Figure 2: What does critical sharpness reads? Results for a typical member of a CIFAR-10 target model walking along the negative gradient and loss is relative to where it started. *(a)* ❶ The parabola with the value, slope and curvature $R(\boldsymbol{g})$ of the loss at the example returns to the starting loss at $2/R(\boldsymbol{g})$. ❷ Walking on, the measured loss falls to zero, stays on that floor, and returns only 136 steps later, at η_c. *(b)* The same walk on a linear step axis. ❸ The parabola with the same value and slope that returns at η_c has curvature $\lambda_c = 2/\eta_c$, the critical sharpness. *(c)* The return magnified, with the sampled steps of the walk as dots: the walk crosses the starting loss at η_c, where parabola ❸ returns too.

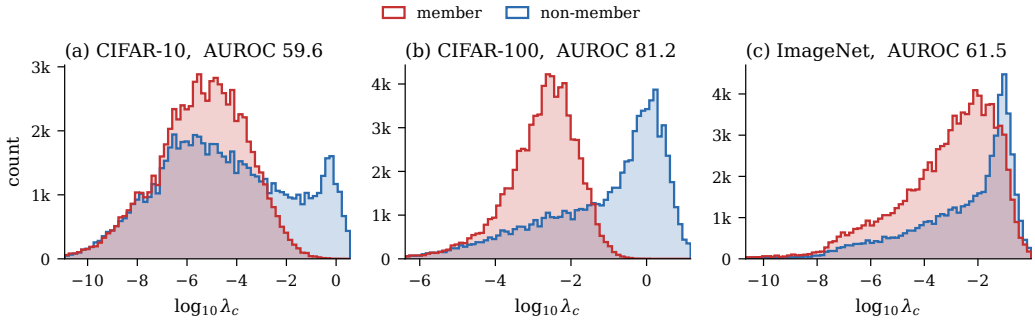

Figure 3: Members have lower critical sharpness than non-members on all three benchmarks. Each panel shows the distribution of $\log_{10} \lambda_c$ over every pair of an image and a target model, members in red and non-members in blue. AUROC shows how well the two are separated (50 is chance).

We walk away from the example to see where the loss actually returns. The critical step is the infimum

$$\eta_c \;=\; \inf\{\eta > 0 : \psi(\eta) \geq \psi(0)\}, \tag{4}$$

the smallest step at which the loss is back at its starting value. The measured walk returns much later than the parabola predicts, by a median factor of 59 to 65 on CIFAR-10 and 25 to 41 on CIFAR-100. The basin is not the parabola at the example: the loss falls, often all the way to zero, stays on that floor, and then climbs into a wall (Figure 2*(a)*, ❷).

To read the curvature of the actual basin, we keep the value and slope of Equation 2 and choose the curvature so that the parabola returns where the walk does. By Equation 3 this is the critical sharpness:

$$\lambda_c \;=\; 2/\eta_c \tag{5}$$

(Figure 2*(b)*, ❸). It equals $R(\boldsymbol{g})$ only when the loss along the walk is itself a parabola, and otherwise measures how far the floor extends, a property of the loss away from the example rather than at it. A walk that never returns gives $\eta_c = \infty$ and $\lambda_c = 0$, the flattest reading. Counting the walk in multiples of $\boldsymbol{g}$ divides out the slope at $\boldsymbol{x}$, so examples with different slopes are compared on the same footing. We migrate this definition and its solver, an exponential search followed by bisection, from the parameter space of Kalra et al. (2026) to input space (Appendix F). Figure 3 shows the consequence for membership: members sit at lower critical sharpness than non-members on all three benchmarks.

3.3 READING THE WALK BLACK-BOX

Nothing in Equation 4 requires the walk to follow $\boldsymbol{g}$, which needs white-box access. Along a direction $\boldsymbol{v}$ the walk is $\psi_{\boldsymbol{v}}(\eta) = \ell(\boldsymbol{x} - \eta\boldsymbol{v})$, and its quadratic model returns to the starting loss at $2/\tilde{\lambda}_{\boldsymbol{v}}$ with

$$\tilde{\lambda}_{\boldsymbol{v}} \;=\; \frac{\boldsymbol{v}^\top \boldsymbol{H}\boldsymbol{v}}{\boldsymbol{v}^\top \boldsymbol{g}}, \tag{6}$$

which is $R(\boldsymbol{g})$ at $\boldsymbol{v} = \boldsymbol{g}$. Where the model is confident, $\ell \approx e^{-m}$ with m the logit margin, so $\boldsymbol{H} = \ell(\nabla m\, \nabla m^\top - \nabla^2 m)$. The first term is rank one, and for piecewise linear activations the second vanishes almost everywhere. With $\boldsymbol{v}$ rescaled to $\|\boldsymbol{v}\| = \|\boldsymbol{g}\|$ and c its cosine with $\boldsymbol{g}$, Equation 6 becomes $\tilde{\lambda}_{\boldsymbol{v}} = c\, R(\boldsymbol{g})$, so the direction enters only through one scalar. Appendix A verifies the rank-one step and measures how little the probe changes the ordering of examples.

Two quantities of the target are therefore enough: a direction and a length. For the direction we use the mean unit gradient of the reference models that never saw the example,

$$\boldsymbol{v} \;\propto\; \sum_{m \in \mathcal{O}(z)} \frac{\boldsymbol{g}_m}{\|\boldsymbol{g}_m\|}, \tag{7}$$

which uses unit gradients because $\|\boldsymbol{g}_m\|$ varies over orders of magnitude between reference models, and which reads only $\mathcal{O}(z)$, so the construction stays offline. The same direction is walked on every model. Its length is the model's gradient norm, estimated in the same way on the reference models and on the target. We estimate $\|\boldsymbol{g}\|$ by zero-order finite-difference. With $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_k$ Rademacher vectors normalized to unit length and a step h, the central difference $D_i = [\ell(\boldsymbol{x} + h\boldsymbol{u}_i) - \ell(\boldsymbol{x} - h\boldsymbol{u}_i)]/2h$ estimates $\boldsymbol{u}_i^\top \boldsymbol{g}$, whose second moment is $\|\boldsymbol{g}\|^2/d$, so

$$\widehat{\|\boldsymbol{g}\|} \;=\; \sqrt{\frac{d}{k} \sum_{i=1}^{k} D_i^2}. \tag{8}$$

This costs $2k$ forward queries and no backward pass through the target.

As along $\boldsymbol{g}$ (§3.2), the walk along $\boldsymbol{v}$ does not follow its parabola, so we do not use $\tilde{\lambda}_{\boldsymbol{v}}$. We run the same bisection along $\boldsymbol{v}$ on the target and on every reference model, and write $\lambda_{\boldsymbol{v}} = 2/\eta_c$ for the result, with η_c the return point of $\psi_{\boldsymbol{v}}$ as in Equation 4. Walking $\boldsymbol{v}$ instead of $\boldsymbol{g}$ costs the attack little: $\lambda_{\boldsymbol{v}}$ has Spearman 0.98 with λ_c on CIFAR, and SHARPMIA scores within 0.05 AUROC of its white-box version on every benchmark (Appendix A, Table 4).

3.4 THE ATTACK

SHARPMIA reads three quantities of a model at an example, the critical sharpness λ of §3.3, the input loss curvature trace tr of Ravikumar et al. (2024b) and the loss ℓ itself. Write $s_m(z)$ for reading s taken on model $\boldsymbol{\theta}_m$ at example z, put every reading on the same scale with

$$u_m^s(z) \;=\; \frac{1}{|A|} \sum_{a \in A} -\log\big(s_m(z^{(a)}) + \epsilon\big), \tag{9}$$

averaged over the two augmentation views A, with $\epsilon = 10^{-22}$ guarding the logarithm, and form three functions of the example z, the reading s and the model v under attack,

$$\mathrm{raw}(z, s, v) \;=\; u_v^s(z), \tag{10}$$

$$\mathrm{out}(z, s, v) \;=\; \frac{1}{|\mathcal{O}(z) \setminus \{v\}|} \sum_{m \in \mathcal{O}(z) \setminus \{v\}} u_m^s(z), \tag{11}$$

$$\mathrm{cal}(z, s, v) \;=\; \mathrm{raw}(z, s, v) - \mathrm{out}(z, s, v). \tag{12}$$

raw is what the model under attack says, out is what models that never saw the example say, and cal is how far the first departs from the second, which is where membership shows. Removing v from the average is a leave-one-out when v is a reference model and does nothing when $v = \boldsymbol{\theta}_\star$, so training and deployment share one formula. For the two curvature readings, $\mathrm{out}(z, s, v)$ is the difficulty calibrator: the level that models which never saw the example assign to the reading, which

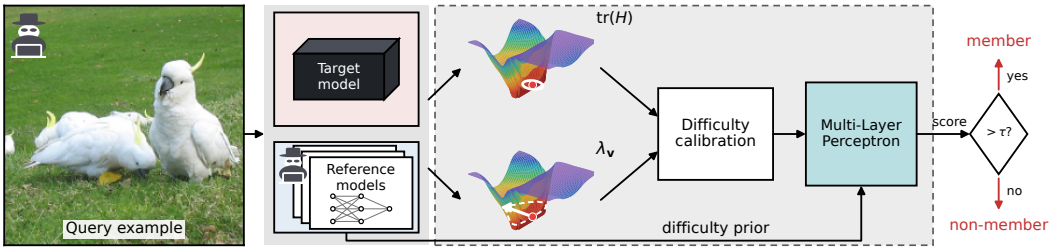

Figure 4: ***The* SHARPMIA *pipeline***. The grey frame illustrates our attack. The target model is a black box and is only queried. From one query example we can read two signals from the target: the input loss curvature of Ravikumar et al. (2024b) at the example (top), and the critical sharpness λ_v, read along a walk until the loss returns (bottom, §3.2 and §3.3). Both are calibrated against reference models the adversary trained and these also provide the difficulty prior $\mathcal{D}(z)$, informing the combiner the hardness of the example (§3.4). Neither reading needs a gradient of the target: the walk takes its direction from the reference models and its length from forward queries (Appendix A).

the model under attack is compared against in Equation 12, as in the difficulty calibration of Watson et al. (2022) that He et al. (2024) also use. For the loss it is the difficulty prior,

$$\mathcal{D}(z) \;=\; \mathrm{out}(z, \ell, v), \tag{13}$$

which enters the combiner on its own rather than as a reference for another reading.

The example is scored from five of these numbers,

$$x(z, v) = \big[\,\mathrm{raw}(z, \lambda, v),\ \mathrm{cal}(z, \lambda, v),\ \mathrm{raw}(z, \mathrm{tr}, v),\ \mathrm{cal}(z, \mathrm{tr}, v),\ \mathrm{out}(z, \ell, v)\,\big]. \tag{14}$$

The fifth is the difficulty prior of Equation 13. It reads no model under attack and costs no query beyond those the calibration already makes. On its own it is not a membership signal, scoring 49.88, 49.98 and 50.08 AUROC on the three benchmarks. What it carries is how hard the example is, which is the coordinate along which the two curvature readings trade places (Table 2). The trace leads on the low $\mathcal{D}(z)$ examples of every benchmark. The critical sharpness leads in all three metrics on the middle and high $\mathcal{D}(z)$ thirds of CIFAR-10, with 64.44 against 64.09 AUROC and 3.25% against 2.85% of members caught at 1% FPR on the middle third, and on the high $\mathcal{D}(z)$ third of ImageNet, with 59.46 against 58.95 AUROC.

The combiner. The five numbers are on incomparable scales, so each is replaced by its rank among the reference models and mapped through Φ^{-1}, the inverse standard normal CDF. A small multilayer perceptron with hidden layers of 16 and 8 units then maps them to a membership score, as in the learned combiner of He et al. (2024). It is trained with L_2 penalty 10^{-2} and early stopping, and five random seeds are averaged.

Training the combiner without the target. The labels come from the reference pool. The adversary draws auxiliary examples Z_{aux} from the pool its reference models were split from. Each reference model plays the model under attack in turn, and its label is free because the adversary chose the splits,

$$X_{\mathrm{train}} = \Big\{\, \big(x(z, \boldsymbol{\theta}_r), \mathbb{1}[z \in D_r]\big) \;:\; z \in Z_{\mathrm{aux}},\ \boldsymbol{\theta}_r \in \mathcal{R} \,\Big\}, \tag{15}$$

which gives N rows per auxiliary example. The transform and the network are fitted on X_{train} once and then frozen, and the target is scored with the same formula, $\mathrm{score}(z) = f\big(x(z, \boldsymbol{\theta}_\star)\big)$. Figure 4 summarizes the pipeline.

4 EXPERIMENTS

We describe our experimental settings and investigation in the following sections.

Datasets and models. We use CIFAR-10, CIFAR-100 (Krizhevsky, 2009) and ImageNet (Russakovsky et al., 2015). Regarding models, for CIFAR-10 and CIFAR-100 we use the 67 ResNet-18 (He et al., 2016) models released by Ravikumar et al. (2024b); for ImageNet, 67 ResNet-50 (He

Table 1: Method & signals comparisons. Each entry is the mean over the three target models, with standard deviations in Appendix C. Every attack draws on the same reference model pool, reads two views of each image, the image and its horizontal flip (§4)—Appendix E reports results for one view. **Bold** marks the best and underline the second in each column. C10, C100 and ImNet are CIFAR-10, CIFAR-100 and ImageNet, respectively. Combining sharpness leads to a stronger attack.

Method	AUROC			TPR@1%FPR			TPR@0.1%FPR		
	C10	C100	IN	C10	C100	IN	C10	C100	IN
SHARPMIA (ours)	**74.94**	**95.38**	**79.72**	**18.09**	**56.24**	**20.75**	**9.89**	**38.48**	9.35
λ_v alone (ours)	66.73	85.94	71.69	8.08	43.33	13.84	0.69	23.57	4.05
Trace (Ravikumar et al., 2024b)	67.83	86.25	75.46	12.45	44.32	17.27	3.16	28.93	0.49
RAPID (He et al., 2024)	73.64	95.09	79.67	15.60	52.38	20.01	7.63	31.01	**9.56**
LDC-MIA (Shi et al., 2024)	70.73	94.64	78.99	13.34	49.89	19.01	6.00	28.83	8.39
IMIA (Du et al., 2026a)	72.73	94.32	75.51	16.61	51.86	17.73	8.46	32.09	7.51
RMIA (Zarifzadeh et al., 2024)	65.97	87.83	77.44	9.15	34.30	18.63	4.52	22.95	8.71
LiRA (Carlini et al., 2022)	67.86	86.02	76.14	12.87	47.58	17.78	5.96	31.80	6.67
LOSS (Yeom et al., 2018)	60.44	82.11	63.36	1.08	0.94	1.09	0.11	0.07	0.11
Sablayrolles et al. (2019)	61.50	81.11	76.35	5.69	21.04	16.99	2.05	10.49	7.86
Calibration (Watson et al., 2022)	58.58	71.66	69.19	4.40	14.86	11.71	1.53	6.62	5.35
Attack-R (Ye et al., 2022)	67.30	90.88	75.52	13.77	34.71	15.75	2.59	6.63	5.70
Mentr (Song & Mittal, 2021)	60.39	82.28	63.42	1.04	0.91	1.04	0.08	0.06	0.11

et al., 2016) models from the pool released by Feldman & Zhang (2020). Each CIFAR model is trained on a random half of the training set and each ImageNet model on a random 70% of it, with the split recorded, so every example is a member of some models and a non-member of others.

Protocol. The model pools, split sizes and two views of each example, the image and its horizontal flip, follow Ravikumar et al. (2024b), and Appendix E repeats Table 1 with one view. Per dataset, 64 models are reference models and 3 are target models, the same for every attack. Each attack is evaluated on the $50{,}000$ images of the CIFAR training split, and on a fixed pool of $50{,}000$ ImageNet training images. For each example, SHARPMIA reads only the reference models not trained on it, a median of 32 of the 64 on CIFAR (27 to 37 from the 10th to the 90th percentile) and 19 on ImageNet (15 to 24). Attacks that calibrate against reference models use those models' mean, and the likelihood-ratio attacks use the one-sided offline test of Carlini et al. (2022). Learned combiners, ours and the baselines', are trained on the reference models only, each in turn playing the target, on Z_{aux}, a subsample of $8{,}000$ images, and are then applied to the full pool of every target.

Metrics. Sweeping a threshold over an attack's scores and plotting the true positive rate (TPR) against the false positive rate (FPR) gives its receiver operating characteristic (ROC) curve. We report the area under that curve (AUROC), which is 100% for an attack that separates members from non-members perfectly and 50% for one that guesses. An attack is useful in practice only if it flags few non-members, so we also report the TPR at low FPR.

4.1 MAIN RESULTS

How does SHARPMIA and sharpness signals compare with state-of-the-art?

Table 1 compares SHARPMIA against prior attacks, reproduced under the protocol of §4, on CIFAR-10, CIFAR-100 and ImageNet. Appendix C gives reproduction details. Appendix D compares with IMIA at its own evaluation setting.

On CIFAR-10 and CIFAR-100, SHARPMIA has the highest AUROC and the highest TPR at both false positive rates. Neither curvature reading is strong on its own. Scored alone with the offline one-sided test (the LR head, compared with the NLL head in Appendix B.2), the black-box critical sharpness reaches 66.73 and 85.94 AUROC, slightly below the input loss curvature attack at 67.83 and 86.25. On ImageNet SHARPMIA has the highest AUROC and TPR at 1% FPR, and RAPID the highest TPR at 0.1% FPR (9.56 against 9.35).

Table 2: The sharpness and curvature are informative on different examples. The examples are ranked by the difficulty prior $\mathcal{D}(z)$ in Equation 13, computed from the reference models alone, and binned into equal thirds: low $\mathcal{D}(z)$, mid and high $\mathcal{D}(z)$. Each entry scores one reading, its raw and calibrated values entered together into the combiner of SHARPMIA, with the two augmented views of each example, averaged over the three target models. Bold marks the better of the two readings. The last block is SHARPMIA. The critical sharpness leads on the mid and high $\mathcal{D}(z)$ thirds of CIFAR-10 and on the high $\mathcal{D}(z)$ third of ImageNet.

		CIFAR-10			CIFAR-100			ImageNet		
Reading	Metric	low	mid	high	low	mid	high	low	mid	high
$\lambda_{\boldsymbol{v}}$	AUROC	92.09	**64.44**	**51.82**	99.34	96.76	72.84	80.84	76.31	**59.46**
(raw + cal)	TPR@1%	31.86	**3.25**	**1.27**	93.55	43.11	5.16	23.57	**8.91**	**2.70**
	TPR@0.1%	13.66	**0.48**	**0.18**	66.14	11.26	0.78	4.69	**1.61**	**0.31**
trace	AUROC	**93.72**	64.09	51.49	**99.96**	**97.62**	**73.89**	**91.54**	**76.95**	58.95
(raw + cal)	TPR@1%	**35.76**	2.85	1.11	**99.96**	**49.88**	**5.50**	**34.16**	8.09	2.19
	TPR@0.1%	**17.22**	0.34	0.06	**90.33**	**16.54**	**0.79**	**6.92**	1.11	0.18
SHARPMIA	AUROC	93.99	64.62	52.00	99.97	97.64	74.26	92.72	77.29	59.52
	TPR@1%	40.31	3.37	1.38	99.98	49.17	5.90	42.44	9.12	2.72
	TPR@0.1%	22.79	0.52	0.20	93.27	16.12	1.02	21.31	2.15	0.36

Do signals from curvature & sharpness catch the same member examples?

Table 2 shows that they do not. The trace is stronger on low $\mathcal{D}(z)$ examples, which every reading attacks well, while the critical sharpness is stronger where membership is hardest to infer, on high $\mathcal{D}(z)$ examples. Since $\mathcal{D}(z)$ tells the combiner which kind of example it faces, SHARPMIA fuses both readings with $\mathcal{D}(z)$. On CIFAR-100 both readings are already high and the trace leads on every third, yet SHARPMIA is above the trace in AUROC on every third. In addition, Table 3 further highlights the evidence for memorization from curvature (trace) and sharpness, while related, are complementary where their fusion in SHARPMIA leads to improved evidence and higher scores.

Table 3: Signal source contributions. Notably SHARPMIA fuses calibrated values of both signals conditioned by $\mathcal{D}(z)$. Results are averaged over the three target models. The SHARPMIA row is from Table 1. The last row is the gain over the better signal (trace/$\lambda_{\boldsymbol{v}}$) (AUC: AUROC, @ is TPR@).

	CIFAR-10			CIFAR-100			ImageNet		
Reading	AUC	@1%	@0.1%	AUC	@1%	@0.1%	AUC	@1%	@0.1%
trace (raw)	66.44	1.84	0.17	88.11	5.00	0.43	65.84	1.62	0.16
$\lambda_{\boldsymbol{v}}$ (raw)	65.73	1.75	0.15	84.34	3.75	0.41	63.22	1.73	0.19
SHARPMIA	74.94	18.09	9.89	95.38	56.24	38.48	79.72	20.75	9.35
gain	+8.50	+16.25	+9.72	+7.27	+51.24	+38.05	+13.88	+19.02	+9.16

Additional investigations and results

Further, we provide: i) additional method details, its agreement with the white-box λ_c, the rank-one check in Appendix A; ii) ablations of the combiner in Appendix B; iii) the standard deviations of Table 1 and how our combiner reproduction depart from prior work in Appendix C; iv) a comparison with the imitative reference (IMIA) in Appendix D; and v) One view example results in Appendix E.

5 CONCLUSION

We investigate the loss geometry in the locality of an example in the *input space* as a signal source for membership inference. We formalize the critical sharpness of the loss with respect to the input as a measure of the local geometry, and show that it leaks membership information. Problematically, using the method we devise, the critical sharpness can also be extracted offline and under the black-box threat model. Because critical sharpness and the input loss curvature evidence are found to be strong on different examples, we fuse the two through a difficulty prior formed from reference models alone. The fused attack, SHARPMIA, is stronger. The strength of the signal is that it complements the curvature at the example rather than replacing it. Its weakness is cost: it takes more queries than a curvature-based attack, a walk per example. We believe our findings advance our understanding of the loss geometry in the context of privacy and memorization while contributing to methods to audit privacy risks of deep learning models.

AI USE STATEMENT

We used generative AI (GenAI) in four ways. **To aid or polish writing**: GenAI checked grammar, suggested word choices, typeset the mathematical expressions in LaTeX, and transcribed tables of results from CSV and JSON files into LaTeX. **For retrieval and discovery**: GenAI helped locate related work and the passages of those papers that describe their methods, and the authors read the papers and chose what to cite. **For research execution**: GenAI assisted in implementing the code that computes the readings, trains the combiner and evaluates every attack. **To draft sections of the paper**: GenAI drafted some sentences and paragraphs, which the authors edited.

GenAI was not used for research ideation: the research ideas are the authors' own, and GenAI did not propose hypotheses or choose which experiments and analyses to run. GenAI was not used to generate data sets or to formulate or prove mathematical claims. Every number in the paper is read from a results file written by the script that produced it, and the authors checked the code, the scripts and the transcribed tables against those files. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies membership inference with dual use predicaments, an attack on the privacy of training data or an auditing tool to evaluate the privacy risks of models. However, our intentions are to support the community to better understand the privacy risks. Notably, all experiments use publicly released models and public datasets. Apart from the imitative models of Appendix D, which we train with the released code of Du et al. (2026a), no model was trained by us, and no private data was accessed. This strategy supports the purpose of the work, to measure membership leakage possible with the signal source we explore, so that privacy audits based on membership inference do not underestimate what a model reveals about its training data.

REPRODUCIBILITY STATEMENT

All models are publicly released by prior work (§4), and the reference and target indices, the cached signals and the scripts that produce every table will be in the released code. We train no models other than the imitative models of Appendix D. Scoring the pools is the only step that needs a GPU: every reading in the paper is computed once per (model, example, view) and cached as a per-example array, and every table, figure and ablation is then computed from those arrays. We will release the scoring and analysis code, the cached score arrays themselves, and the split indices needed to reproduce every number without repeating the GPU work. The attack, its preparation on the reference models, and the two operators it calls, the critical sharpness solver and the zero-order trace, are given as pseudocode in Appendix F.

REFERENCES

Martin Bertran, Shuai Tang, Aaron Roth, Michael Kearns, Jamie Morgenstern, and Zhiwei Steven Wu. Scalable membership inference attacks via quantile regression. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramèr. Membership inference attacks from first principles. In *IEEE Symposium on Security and Privacy (S&P)*, 2022.

Nicholas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramèr, Borja Balle, Daphne Ippolito, and Eric Wallace. Extracting training data from diffusion models. In *USENIX Security Symposium*, 2023.

Min Chen, Zhikun Zhang, Tianhao Wang, Michael Backes, Mathias Humbert, and Yang Zhang. When machine unlearning jeopardizes privacy. In *ACM Conference on Computer and Communications Security (CCS)*, 2021.

Christopher A Choquette-Choo, Florian Tramer, Nicholas Carlini, and Nicolas Papernot. Label-only membership inference attacks. In *International Conference on Machine Learning*, pp. 1964–1974. PMLR, 2021.

Jeremy M. Cohen, Simran Kaur, Yuanzhi Li, J. Zico Kolter, and Ameet Talwalkar. Gradient descent on neural networks typically occurs at the edge of stability. In *International Conference on Learning Representations (ICLR)*, 2021.

Yuntao Du, Yuetian Chen, Hanshen Xiao, Bruno Ribeiro, and Ninghui Li. Imitative membership inference attack. In *USENIX Security Symposium*, 2026a. 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 *Network and Distributed System Security Symposium (NDSS)*, 2026b. arXiv:2507.21412.

Vitaly Feldman. Does learning require memorization? a short tale about a long tail. In *ACM Symposium on Theory of Computing (STOC)*, 2020.

Vitaly Feldman and Chiyuan Zhang. What neural networks memorize and why: Discovering the long tail via influence estimation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations (ICLR)*, 2021.

Isha Garg, Deepak Ravikumar, and Kaushik Roy. Memorization through the lens of curvature of loss function around samples. In *International Conference on Machine Learning (ICML)*, 2024.

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

Yu He, Boheng Li, Yao Liu, Zhuotao Liu, Ruihao Yang, Tianhang Zhang, and Kui Ren. Is difficulty calibration all we need? towards more practical membership inference attacks. In *ACM Conference on Computer and Communications Security (CCS)*, 2024.

Hongsheng Hu, Zoran Salcic, Lichao Sun, Gillian Dobbie, Philip S. Yu, and Xuyun Zhang. Membership inference attacks on machine learning: A survey. *ACM Computing Surveys*, 54(11s):1–37, 2022.

Dayal Singh Kalra, Jean-Christophe Gagnon-Audet, Andrey Gromov, Ishita Mediratta, Kelvin Niu, Alexander H. Miller, and Michael Shvartsman. A scalable measure of loss landscape curvature for analyzing the training dynamics of llms. *arXiv preprint arXiv:2601.16979*, 2026.

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. In *International Conference on Learning Representations (ICLR)*, 2017.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Zheng Li and Yang Zhang. Membership leakage in label-only exposures. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pp. 880–895, 2021.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Jonathan Uesato, and Pascal Frossard. Robustness via curvature regularization, and vice versa. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. arXiv:1811.09716.

Sasi Kumar Murakonda and Reza Shokri. ML privacy meter: Aiding regulatory compliance by quantifying the privacy risks of machine learning. *arXiv preprint arXiv:2007.09339*, 2020.

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.

Deepak Ravikumar, Efstathia Soufleri, Abolfazl Hashemi, and Kaushik Roy. Unveiling privacy, memorization, and input curvature links. In *International Conference on Machine Learning (ICML)*, 2024a.

Deepak Ravikumar, Efstathia Soufleri, and Kaushik Roy. Curvature clues: Decoding deep learning privacy with input loss curvature. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024b.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115 (3):211–252, 2015.

Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, Yann Ollivier, and Hervé Jégou. White-box vs black-box: Bayes optimal strategies for membership inference. In *International Conference on Machine Learning (ICML)*, 2019.

Haonan Shi, Tu Ouyang, and An Wang. Learning-based difficulty calibration for enhanced membership inference attacks. In *IEEE European Symposium on Security and Privacy (EuroS&P)*, 2024.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *IEEE Symposium on Security and Privacy (S&P)*, 2017.

Liwei Song and Prateek Mittal. Systematic evaluation of privacy risks of machine learning models. In *USENIX Security Symposium*, 2021.

Lauren Watson, Chuan Guo, Graham Cormode, and Alexandre Sablayrolles. On the importance of difficulty calibration in membership inference attacks. In *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 *ACM Conference on Computer and Communications Security (CCS)*, 2022.

Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy risk in machine learning: Analyzing the connection to overfitting. In *IEEE Computer Security Foundations Symposium (CSF)*, 2018.

Sajjad Zarifzadeh, Philippe Liu, and Reza Shokri. Low-cost high-power membership inference attacks. In *International Conference on Machine Learning (ICML)*, 2024.

A THE BLACK-BOX READING

§3.3 gives the construction in outline. This section states it in full and accounts for what it costs.

Rank one in practice. §3.3 argues that where the model is confident the input Hessian is effectively rank one, so a walk along $\boldsymbol{v}$ reads the same curvature scaled by a single cosine. The ratio $R(\boldsymbol{v})/(c^2 R(\boldsymbol{g}))$, which rank one puts at 1, has median 1.00 to 1.02 on six models each of CIFAR-10 and CIFAR-100, with an upper quartile of 1.05 to 1.14, and the participation ratio of the input Hessian is 1.084. Across images $R(\boldsymbol{g})$ spans about six orders of magnitude and c about 0.3 of one, so the ordering of examples is set by the curvature and not by the probe. Along the translation tangent of the training augmentation, which is almost orthogonal to $\boldsymbol{g}$ ($|c|$ median 0.006), the reading still has Spearman 0.96 with λ_c, and along the direction of Equation 7 (c median 0.48 to 0.52 on CIFAR-10 and 0.46 to 0.47 on CIFAR-100) it has 0.98 to 0.99.

White box against black box. Walking $\boldsymbol{v}$ at the estimated length instead of the target's own gradient changes the reading little. Over the $50{,}000$ examples, $\lambda_{\boldsymbol{v}}$ has Spearman 0.978 to 0.990 with the white-box λ_c on the three target models and two views (median 0.983 on CIFAR-10 and 0.979 on CIFAR-100). Table 4 scores both readings, alone under the one-sided test and inside SHARPMIA. Alone, the black-box reading is 0.28, 0.48 and 1.06 AUROC below the white-box one. Inside SHARPMIA the gap closes: the black-box attack is 0.05 above the white-box one on CIFAR-10 and ImageNet and 0.04 below on CIFAR-100, and within 0.11 at both false positive rates.

Table 4: White-box λ_c against black-box $\lambda_{\boldsymbol{v}}$, alone under the offline one-sided test and inside SHARPMIA. Mean over the three target models. The black-box rows are those of Table 1, with two views.

		CIFAR-10		CIFAR-100		ImageNet	
	Metric	λ_c	$\lambda_{\boldsymbol{v}}$	λ_c	$\lambda_{\boldsymbol{v}}$	λ_c	$\lambda_{\boldsymbol{v}}$
Alone	AUROC	67.01	66.73	86.42	85.94	72.75	71.69
	TPR@1%	8.89	8.08	44.97	43.33	16.28	13.84
	TPR@0.1%	1.82	0.69	27.02	23.57	6.25	4.05
SHARPMIA	AUROC	74.89	74.94	95.42	95.38	79.67	79.72
	TPR@1%	18.07	18.09	56.27	56.24	20.70	20.75
	TPR@0.1%	9.93	9.89	38.49	38.48	9.46	9.35

The length. Equation 8 uses $h = 0.1$ and $k = 32$, which is 64 forward queries per image. The estimate has Spearman 0.999 with the exact norm, and the attack is insensitive to its per-image error: with k at 4, 8, 16 and 32, SHARPMIA scores 74.91, 74.89, 74.95 and 74.94 AUROC on CIFAR-10 and 95.38, 95.35, 95.37 and 95.38 on CIFAR-100.

B ABLATIONS

Each ablation below changes one part of SHARPMIA and keeps everything else as in §3.4. They are on CIFAR-10 and CIFAR-100.

B.1 COMBINER ARCHITECTURE

Table 5 replaces the MLP$(16, 8)$ by other combiners, with the same five features. A linear combiner loses about 5 AUROC points and most of the true positives at low false positive rates, so the five features need a non-linear combiner. Beyond that the size of the network matters little: every MLP in the table is within 0.20 AUROC of the others on both benchmarks, and we use MLP$(16, 8)$, a small network on that plateau.

Table 5: The combiner. Mean over the three target models and five seeds, offline black-box pipeline, two views of each example.

Variant	CIFAR-10			CIFAR-100		
	AUROC	TPR@1%	TPR@0.1%	AUROC	TPR@1%	TPR@0.1%
Logistic regression	69.83	2.92	0.00	89.54	15.95	0.04
MLP(8)	74.81	17.70	9.57	95.31	55.81	37.00
MLP$(16, 8)$, used	74.94	18.09	9.89	95.38	56.24	38.48
MLP$(32, 16)$	74.95	18.06	9.97	95.41	56.32	38.61
MLP$(64, 32, 16)$	75.01	18.09	9.97	95.42	56.35	38.66

B.2 SINGLE READINGS UNDER THE TWO HEADS

Ravikumar et al. (2024b) score a single reading with two heads. The negative log-likelihood (NLL) head fits one Gaussian to the reading over the reference models trained on the example and one over those not trained on it, and compares the target model's reading under the two, so it is online. The likelihood-ratio (LR) head is the offline one-sided test of Carlini et al. (2022): it fits only the second Gaussian and scores how far the target model's reading falls on the member side of it. Table 6 scores the critical sharpness and the trace alone under both heads. Neither head makes a single reading strong: the best AUROC is 72.45, 93.59 and 77.89 on the three benchmarks, against 74.94, 95.38 and 79.72 for SHARPMIA with the same two views.

Table 6: The critical sharpness and the trace alone under the NLL head and the LR head. Mean over the three target models, two views of each example. 1% and 0.1% are the TPR at that FPR.

	CIFAR-10			CIFAR-100			ImageNet		
Head, reading	AUROC	1%	0.1%	AUROC	1%	0.1%	AUROC	1%	0.1%
NLL, λ_v	72.45	8.05	0.04	93.21	36.38	3.91	74.90	8.57	0.58
NLL, trace	69.18	1.08	0.13	93.59	43.03	9.99	77.89	10.32	0.14
LR, λ_v	66.73	8.08	0.69	85.94	43.33	23.57	71.69	13.84	4.05
LR, trace	67.83	12.45	3.16	86.25	44.32	28.93	75.46	17.27	0.49

C Reproduction Notes

Table 7: AUROC of Table 1, mean $\pm$ standard deviation over the three target models.

Method	CIFAR-10	CIFAR-100	ImageNet
SharpMia (ours)	**74.94** $\pm$1.46	**95.38** $\pm$0.23	**79.72** $\pm$0.04
λ_v alone, (ours)	66.73 $\pm$1.14	85.94 $\pm$0.53	71.69 $\pm$0.27
Trace (Ravikumar et al., 2024b)	67.83 $\pm$0.55	86.25 $\pm$0.39	75.46 $\pm$0.31
RAPID (He et al., 2024)	73.64 $\pm$0.80	95.09 $\pm$0.22	79.67 $\pm$0.04
LDC-MIA (Shi et al., 2024)	70.73 $\pm$0.47	94.64 $\pm$0.25	78.99 $\pm$0.15
RMIA (Zarifzadeh et al., 2024)	65.97 $\pm$1.20	87.83 $\pm$0.36	77.44 $\pm$0.18
LiRA (Carlini et al., 2022)	67.86 $\pm$0.83	86.02 $\pm$0.69	76.14 $\pm$0.10
LOSS (Yeom et al., 2018)	60.44 $\pm$0.92	82.11 $\pm$0.38	63.36 $\pm$0.06
Sablayrolles et al. (2019)	61.50 $\pm$0.97	81.11 $\pm$0.48	76.35 $\pm$0.16
Calibration (Watson et al., 2022)	58.58 $\pm$1.05	71.66 $\pm$0.61	69.19 $\pm$0.28
Attack-R (Ye et al., 2022)	67.30 $\pm$1.54	90.88 $\pm$0.23	75.52 $\pm$0.12
Mentr (Song & Mittal, 2021)	60.39 $\pm$1.06	82.28 $\pm$0.33	63.42 $\pm$0.06

Table 8: TPR at 1% FPR of Table 1, mean $\pm$ standard deviation over the three target models.

Method	CIFAR-10	CIFAR-100	ImageNet
SharpMia (ours)	**18.09** $\pm$1.79	**56.24** $\pm$0.64	**20.75** $\pm$0.33
λ_v alone (ours)	8.08 $\pm$3.10	43.33 $\pm$1.42	13.84 $\pm$0.16
Trace (Ravikumar et al., 2024b)	12.45 $\pm$1.04	44.32 $\pm$0.40	17.27 $\pm$0.56
RAPID (He et al., 2024)	15.60 $\pm$0.42	52.38 $\pm$0.51	20.01 $\pm$0.17
LDC-MIA (Shi et al., 2024)	13.34 $\pm$0.56	49.89 $\pm$0.84	19.01 $\pm$0.30
RMIA (Zarifzadeh et al., 2024)	9.15 $\pm$0.64	34.30 $\pm$0.72	18.63 $\pm$0.17
LiRA (Carlini et al., 2022)	12.87 $\pm$3.15	47.58 $\pm$0.69	17.78 $\pm$0.38
LOSS (Yeom et al., 2018)	1.08 $\pm$0.03	0.94 $\pm$0.07	1.09 $\pm$0.04
Sablayrolles et al. (2019)	5.69 $\pm$0.21	21.04 $\pm$0.47	16.99 $\pm$0.55
Calibration (Watson et al., 2022)	4.40 $\pm$0.06	14.86 $\pm$0.16	11.71 $\pm$0.35
Attack-R (Ye et al., 2022)	13.77 $\pm$1.42	34.71 $\pm$2.29	15.75 $\pm$0.65
Mentr (Song & Mittal, 2021)	1.04 $\pm$0.10	0.91 $\pm$0.15	1.04 $\pm$0.12

Every baseline in Table 1 is reproduced by us under the protocol of §4 rather than quoted, so that the reference models, the views, the target models and the evaluation pool are the same for all of them. For Shi et al. (2024) we follow their released code: the neighbourhood factor with $\theta = 0$, the mean logits over the reference models not trained on x as the embedding, Z_{aux} as the auxiliary set, and their training recipe (SGD, learning rate 0.004, 35 epochs). As a check, Shi et al. (2024) report LDC-MIA 3.3 AUROC above LiRA on CIFAR-10 with 64 reference models, and in Table 1 it is 2.9 above (70.73 against 67.86).

Table 9: TPR at 0.1% FPR of Table 1, mean $\pm$ standard deviation over the three target models.

Method	CIFAR-10	CIFAR-100	ImageNet
SHARPMIA (ours)	**9.89** $\pm$1.32	**38.48** $\pm$1.53	9.35 $\pm$0.74
λ_v alone (ours)	0.69 $\pm$1.18	23.57 $\pm$6.23	4.05 $\pm$0.19
Trace (Ravikumar et al., 2024b)	3.16 $\pm$0.09	28.93 $\pm$2.51	0.49 $\pm$0.11
RAPID (He et al., 2024)	7.63 $\pm$0.53	31.01 $\pm$1.32	**9.56** $\pm$0.70
LDC-MIA (Shi et al., 2024)	6.00 $\pm$0.71	28.83 $\pm$1.74	8.39 $\pm$0.70
RMIA (Zarifzadeh et al., 2024)	4.52 $\pm$0.37	22.95 $\pm$1.15	8.71 $\pm$0.72
LiRA (Carlini et al., 2022)	5.96 $\pm$2.36	31.80 $\pm$1.09	6.67 $\pm$0.55
LOSS (Yeom et al., 2018)	0.11 $\pm$0.00	0.07 $\pm$0.01	0.11 $\pm$0.00
Sablayrolles et al. (2019)	2.05 $\pm$0.26	10.49 $\pm$0.73	7.86 $\pm$0.47
Calibration (Watson et al., 2022)	1.53 $\pm$0.24	6.62 $\pm$0.28	5.35 $\pm$0.88
Attack-R (Ye et al., 2022)	2.59 $\pm$0.51	6.63 $\pm$0.64	5.70 $\pm$0.53
Mentr (Song & Mittal, 2021)	0.08 $\pm$0.03	0.06 $\pm$0.02	0.11 $\pm$0.00

D IMITATIVE REFERENCE MODELS

IMIA (Du et al., 2026a) trains its reference models to imitate the target. We run it offline, in the non-adaptive form of its released code: the reference readings it compares against are never taken from a model trained on the example. Table 10 places every attack on one evaluation set, the 47 reference models and $8{,}000$ images of the setting of IMIA, and reads each example at the first k of its 18 random augmentations. Within a block the reference models change and the augmentations do not. Across the two blocks the augmentations change and everything else does not.

Table 10: SHARPMIA against IMIA, offline, at $k = 2$ and $k = 18$ random augmentations of each image, on the same 47 reference models and $8{,}000$ images. *Reference models* are ordinary reference models, as SHARPMIA uses. *Imitative models* are trained against each target. Mean $\pm$ standard deviation over the three target models, and bold marks the best in each column within a block. With the same reference models SHARPMIA is ahead of IMIA on every metric at both k. IMIA leads only at low false positive rates on CIFAR-10, and only with imitative models. More augmentations raise every attack and keep SHARPMIA ahead on AUROC and on CIFAR-100.

Method	CIFAR-10			CIFAR-100		
	AUROC	TPR@1%	TPR@0.1%	AUROC	TPR@1%	TPR@0.1%
$k = 2$ *augmentations*						
SHARPMIA, reference	**74.79** $\pm$1.14	**17.43** $\pm$0.52	8.72 $\pm$0.78	**95.69** $\pm$0.19	**57.46** $\pm$1.02	**39.31** $\pm$3.99
IMIA, reference	73.13 $\pm$1.17	15.60 $\pm$1.06	8.03 $\pm$0.71	94.85 $\pm$0.21	53.38 $\pm$0.87	31.85 $\pm$5.64
IMIA, imitative	72.47 $\pm$1.36	16.44 $\pm$0.93	**9.29** $\pm$1.68	95.24 $\pm$0.16	50.65 $\pm$0.94	30.62 $\pm$3.22
$k = 18$ *augmentations*						
SHARPMIA, reference	**77.12** $\pm$1.25	20.30 $\pm$2.11	11.62 $\pm$2.08	**96.60** $\pm$0.15	**61.69** $\pm$0.91	**45.76** $\pm$1.21
IMIA, reference	75.43 $\pm$1.18	18.30 $\pm$1.51	10.99 $\pm$3.03	95.69 $\pm$0.07	56.97 $\pm$1.97	35.38 $\pm$4.04
IMIA, imitative	75.12 $\pm$1.57	**21.46** $\pm$1.79	**13.35** $\pm$1.31	96.20 $\pm$0.09	56.72 $\pm$0.91	37.26 $\pm$1.58

SHARPMIA is ahead on AUROC in both settings and on every metric on CIFAR-100. On CIFAR-10, IMIA with imitative models is ahead at 0.1% FPR at $k = 2$ and at both low false positive rates at $k = 18$. With our ordinary reference models in place of its imitative ones, IMIA is below SHARPMIA on every metric in both settings. The imitative models are trained against each target, a cost and an access assumption our threat model (§3.1) does not make.

E ONE VIEW, NOT AUGMENTED

Table 1 reads two views of each example, the image and its horizontal flip. Table 11 repeats the comparison with one view, the image itself. The second view raises SHARPMIA by 1.83, 0.48 and 1.65 AUROC on the three benchmarks, and RAPID by 1.79, 0.52 and 1.02.

Table 11: Table 1 with **one** view of each image, the image itself. Mean over the three target models. Bold marks the best in each column and underline the second.

Method	AUROC			TPR@1%FPR			TPR@0.1%FPR		
	C10	C100	IN	C10	C100	IN	C10	C100	IN
SHARPMIA (ours)	**73.11**	**94.90**	<u>78.07</u>	**15.90**	**54.09**	**18.95**	**7.92**	**35.82**	<u>8.67</u>
$\lambda_{\boldsymbol{v}}$ alone (ours)	65.71	85.36	70.24	7.32	40.88	12.64	1.15	21.63	3.39
Trace (Ravikumar et al., 2024b)	66.17	86.00	74.16	11.46	42.99	15.87	2.65	27.04	0.37
RAPID (He et al., 2024)	<u>71.85</u>	<u>94.57</u>	**78.65**	13.68	<u>50.34</u>	<u>18.57</u>	<u>5.98</u>	<u>30.64</u>	**8.96**
LDC-MIA (Shi et al., 2024)	68.28	94.06	77.91	11.18	47.61	18.02	4.58	27.36	7.75
RMIA (Zarifzadeh et al., 2024)	64.16	87.03	76.31	7.65	32.56	17.14	3.68	21.21	7.95
LiRA (Carlini et al., 2022)	66.98	85.69	75.25	11.06	45.32	15.93	4.82	29.80	4.77
LOSS (Yeom et al., 2018)	60.40	82.14	63.36	1.07	0.93	1.09	0.11	0.08	0.11
Sablayrolles et al. (2019)	61.59	81.06	76.35	5.66	21.16	16.99	2.05	10.29	7.86
Calibration (Watson et al., 2022)	58.54	71.64	69.19	4.56	14.73	11.71	1.53	6.59	5.35
Attack-R (Ye et al., 2022)	67.33	90.88	75.52	<u>13.87</u>	33.70	15.75	2.49	6.50	5.70
Mentr (Song & Mittal, 2021)	60.39	82.28	63.42	1.04	0.91	1.04	0.08	0.06	0.11

F SHARPMIA'S ALGORITHM IN PSEUDOCODE

Algorithm 1 scores an example with a combiner that Algorithm 2 fits once on the reference models. Both call two operators on a model at an example: the critical sharpness along a given direction (Algorithm 3) and the zero-order trace of the input Hessian of Ravikumar et al. (2024b) (Algorithm 4).

Algorithm 1 SHARPMIA: scoring an example on the target

Require: frozen transform and combiner f from Algorithm 2, reference models $\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_N$ with training sets $D_1, \dots, D_N$, target $\boldsymbol{\theta}_\star$ (queries only), example $z = (\boldsymbol{x}, y)$, views A, threshold τ

1: **for** each view $a \in A$ **do**
2: $\quad \boldsymbol{v} \leftarrow \sum_{m \in \mathcal{O}(z)} \boldsymbol{g}_m / \|\boldsymbol{g}_m\|$, normalized {Equation 7, reference gradients at $\boldsymbol{x}^{(a)}$}
3: $\quad \hat{\boldsymbol{g}}_\star \leftarrow \widehat{\|\boldsymbol{g}\|}\, \boldsymbol{v}$ {predicted target gradient, Equation 8, $2k$ queries}
4: $\quad \lambda_\star \leftarrow \textsc{Sharp}(\boldsymbol{\theta}_\star, \boldsymbol{x}^{(a)}, y, \hat{\boldsymbol{g}}_\star)$ {Algorithm 3}
5: $\quad \mathrm{tr}_\star \leftarrow \textsc{Trace}(\boldsymbol{\theta}_\star, \boldsymbol{x}^{(a)}, y)$ {Algorithm 4}
6: **end for**
7: $u^s_\star(z) \leftarrow \frac{1}{|A|} \sum_{a \in A} -\log\left(s_\star(z^{(a)}) + \epsilon\right)$ for $s \in \{\lambda, \mathrm{tr}\}$ {Equation 9}
8: $u^s_m(z)$ for every $m \in \mathcal{O}(z)$, as in lines 1 to 8 of Algorithm 2
9: $\mathrm{out}(z, s, \star) \leftarrow \frac{1}{|\mathcal{O}(z)|} \sum_{m \in \mathcal{O}(z)} u^s_m(z)$ {Equation 11}
10: $\mathrm{cal}(z, s, \star) \leftarrow u^s_\star(z) - \mathrm{out}(z, s, \star)$ {Equation 12}
11: $x \leftarrow [u^\lambda_\star, \mathrm{cal}(z, \lambda, \star), u^{\mathrm{tr}}_\star, \mathrm{cal}(z, \mathrm{tr}, \star), \mathrm{out}(z, \ell, \star)]$ {Equation 14}
12:
13: **return** $\mathrm{score}(z) = f(x)$, and z is a member if $\mathrm{score}(z) > \tau$

Algorithm 3 is a direct, input-space generalization of Algorithms 1 and 2 of Kalra et al. (2026): their parameter-space update $\boldsymbol{\theta} - \eta\Delta\boldsymbol{\theta}$ (a single scalar search per training checkpoint) becomes $\boldsymbol{x} - \eta\boldsymbol{d}$ (a single scalar search per image), with $\boldsymbol{d}$ the input gradient unless a direction is given, and we do not claim this two-phase exponential-search-then-bisection procedure as our contribution. Every search in this paper starts from $\eta_0 = 1$.

Algorithm 2 Preparation: fitting the combiner on the reference models, once

Require: reference models $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_N$ with training sets $D_1, \ldots, D_N$, auxiliary examples Z_{aux}, views A

for each $z' = (\boldsymbol{x}, y) \in Z_{\text{aux}}$, reference model m and view $a \in A$ **do**
  $\boldsymbol{g}_m \leftarrow \nabla_{\boldsymbol{x}} \ell(f_{\boldsymbol{\theta}_m}(\boldsymbol{x}^{(a)}), y)$ {white-box, the adversary owns $\boldsymbol{\theta}_m$}
end for
for each $z' \in Z_{\text{aux}}$, reference model m and view $a \in A$ **do**
  $\boldsymbol{v} \leftarrow \sum_{m' \in \mathcal{O}(z') \setminus \{m\}} \boldsymbol{g}_{m'} / \|\boldsymbol{g}_{m'}\|$, normalized {leave-one-out}
  $\lambda_m \leftarrow \textsc{Sharp}(\boldsymbol{\theta}_m, \boldsymbol{x}^{(a)}, y, \widehat{\|\boldsymbol{g}_m\|}\, \boldsymbol{v})$ {Equation 8}, $\quad \text{tr}_m \leftarrow \textsc{Trace}(\boldsymbol{\theta}_m, \boldsymbol{x}^{(a)}, y)$, $\quad \ell_m \leftarrow \ell(f_{\boldsymbol{\theta}_m}(\boldsymbol{x}^{(a)}), y)$
end for
$u_m^s(z')$ for every m, z' and $s \in \{\lambda, \text{tr}, \ell\}$ {Equation 9}
for each $z' \in Z_{\text{aux}}$ and reference model r **do**
  $\text{out}(z', s, r) \leftarrow$ mean of $u_m^s(z')$ over $m \in \mathcal{O}(z') \setminus \{r\}$ {leave-one-out, Equation 11}
  $\text{cal}(z', s, r) \leftarrow u_r^s(z') - \text{out}(z', s, r)$
  $x(z', r) \leftarrow [u_r^\lambda, \text{cal}(z', \lambda, r), u_r^{\text{tr}}, \text{cal}(z', \text{tr}, r), \text{out}(z', \ell, r)]$, $\quad$ label $\mathbb{1}[z' \in D_r]$
end for
fit the rank and Φ^{-1} transform and the MLP$(16, 8)$ f on the $N\,|Z_{\text{aux}}|$ rows {Equation 15}

return f, frozen

Algorithm 3 Exponential search + bisection for a single image (generalizes Algorithms 1 and 2 of Kalra et al. (2026) to input space)

Require: model $f_{\boldsymbol{\theta}}$, example $(\boldsymbol{x}, y)$, direction $\boldsymbol{d}$ (optional, default $\nabla_{\boldsymbol{x}} \ell(\boldsymbol{x})$), starting guess η_0, tolerance level tol, max iterations T

$\ell_0 \leftarrow \ell(\boldsymbol{x})$; $\quad \eta \leftarrow \eta_0$
if $\ell(\boldsymbol{x} - \eta \boldsymbol{d}) < \ell_0$ **then**
  dir $\leftarrow +1$
else
  dir $\leftarrow -1$
end if
for $t = 1$ **to** T **do**
  $\eta \leftarrow \eta \cdot 2^{\text{dir}}$
  if dir$=+1$ **and** $\ell(\boldsymbol{x} - \eta \boldsymbol{d}) > \ell_0$ **then**
    $[\eta_{\text{lo}}, \eta_{\text{hi}}] \leftarrow [\eta/2,\ \eta]$; **break**
  else if dir$=-1$ **and** $\ell(\boldsymbol{x} - \eta \boldsymbol{d}) < \ell_0$ **then**
    $[\eta_{\text{lo}}, \eta_{\text{hi}}] \leftarrow [\eta,\ 2\eta]$; **break**
  end if
end for
while $1 - \eta_{\text{lo}}/\eta_{\text{hi}} \geq 2^{-\text{tol}}$ **do**
  $\eta_{\text{mid}} \leftarrow (\eta_{\text{lo}} + \eta_{\text{hi}})/2$
  if $\ell(\boldsymbol{x} - \eta_{\text{mid}} \boldsymbol{d}) > \ell_0$ **then** $\eta_{\text{hi}} \leftarrow \eta_{\text{mid}}$ **else** $\eta_{\text{lo}} \leftarrow \eta_{\text{mid}}$
end while
$\eta_c \leftarrow (\eta_{\text{lo}} + \eta_{\text{hi}})/2$;
return $\lambda_c = 2/\eta_c$

Algorithm 4 TRACE: zero-order estimate of the input Hessian trace, as released by Ravikumar et al. (2024b)

Require: model $f_{\boldsymbol{\theta}}$, example $(\boldsymbol{x}, y)$, step $h = 10^{-3}$, probes $n = 10$

$c \leftarrow 0$
for $i = 1$ **to** n **do**
  draw $\boldsymbol{u}, \boldsymbol{w} \in \{-h, +h\}^d$ with independent random signs
  $c \leftarrow c + \big|\, \ell(\boldsymbol{x} + \boldsymbol{u} + \boldsymbol{w}) - \ell(\boldsymbol{x} - \boldsymbol{u} + \boldsymbol{w}) - \ell(\boldsymbol{x} + \boldsymbol{u} - \boldsymbol{w}) + \ell(\boldsymbol{x} - \boldsymbol{u} - \boldsymbol{w}) \,\big| / 4h$
end for

return $\text{tr} = c/n$ {$4n$ forward passes}