Mitigating Privacy Risk in Membership Inference by Convex-Concave Loss

Zhenlong Liu [1] **Lei Feng** [2] **Huiping Zhuang** [3] **Xiaofeng Cao** [4] **Hongxin Wei** [1]

Abstract

Machine learning models are susceptible to membership inference attacks (MIAs), which aim to infer whether a sample is in the training set. Existing work utilizes gradient ascent to enlarge the loss variance of training data, alleviating the privacy risk. However, optimizing toward a reverse direction may cause the model parameters to oscillate near local minima, leading to instability and suboptimal performance. In this work, we propose a novel method – Convex-Concave Loss (**CCL**), which enables a high variance of training loss distribution by gradient descent. Our method is motivated by the theoretical analysis that convex losses tend to decrease the loss variance during training. Thus, our key idea behind CCL is to reduce the convexity of loss functions with a concave term. Trained with CCL, neural networks produce losses with high variance for training data, reinforcing the defense against MIAs. Extensive experiments demonstrate the superiority of CCL, achieving a state-of-the-art balance in the privacy-utility trade-off.

1. Introduction

Deep Neural Networks (DNNs) have achieved tremendous performance for various learning tasks, with sufficient capacity (Zhang et al., 2021). The powerful capability enables models to memorize information of training data (Zhang et al., 2017), therefore being highly susceptible to membership inference attacks (MIAs) (Shokri et al., 2017). In particular, membership inference attacks are designed to infer whether a sample is included in the training set of a target model. Such attacks can increase the risks of violating privacy regulations, making it challenging to apply machine learning techniques in sensitive applications, like health care (Paul et al., 2021), financial service (Mahalle et al., 2018), and DNA sequence analysis (Arshad et al., 2021). This gives rise to the importance of designing robust algorithms to secure DNNs from MIAs.

In the literature, sample loss has been a representative metric of membership inference attacks. In particular, a large gap in the expected loss values on the member (training) and non-member data is proved to be sufficient for attacks (Yeom et al., 2018). Consequently, defenders can mitigate the privacy risk by reducing distinguishability between the member and non-member loss distributions. Recently, RelaxLoss (Chen et al., 2022) utilizes gradient ascent to promote a high variance of training loss distribution, which is shown to strengthen the defense against MIAs. However, optimizing toward a reverse direction may cause the model parameters to oscillate near local minima, leading to instability and suboptimal performance. This motivates our method, which enables us to enlarge the training loss variance via gradient descent.

In this work, we propose a novel and generalized method – Convex-Concave Loss (**CCL**), by integrating a concave term into convex losses. Our method is motivated by a theoretical analysis of the connection between the convexity of loss functions and the resulting loss variance. We demonstrate that convex loss functions are optimized to encourage a small loss variance (see Theorem 3.1), being vulnerable to membership inference attacks. On the contrary, concave functions can increase the loss variance during gradient descent, which are expected to mitigate privacy risk.

Thus, our key idea behind CCL is to decrease the convexity of loss functions for a large loss variance. This can be achieved by incorporating a concave term into the original convex losses, e.g., cross-entropy loss. In effect, the resulting loss weakens the convexity of the original convex loss at the late stage of training and can converge to the optimum of the convex loss. Trained with CCL, the network tends to produce losses with high variance for training data, reducing the differentiability of sample losses between the member and non-member data.

To verify the effectiveness of our method, we conduct extensive evaluations on five datasets, including Texas100 (Texas Department of State Health Services, 2006), Purchase100 (Kaggle, 2014), CIFAR-10/100 (Krizhevsky et al.,

[1]Southern University of Science and Technology [2]Singapore University of Technology and Design [3]South China University of Technology [4]Jilin University. Correspondence to: Hongxin Wei <weihx@sustech.edu.cn>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

2009), and ImageNet (Russakovsky et al., 2015) datasets. The results demonstrate our methods can improve utility-privacy trade-offs across a variety of attacks based on neural network, metric, and data augmentation. For example, our method formulated using a concave quadratic function, significantly diminishes the attack advantage in loss-metric-based from 29.67% to 18.40% - a relative reduction of 62.01% in privacy risk, whilst preserving the test accuracy not worse than the vanilla model. Our code is available at https://github.com/ml-stat-Sustech/ConvexConcaveLoss.

Our contributions are summarized as follows:

1. We introduce the concept of Convex-Concave Loss (CCL), a generalized loss function that incorporates a concave term into the original convex loss, i.e., Cross-Entropy (CE) loss. This approach stands as a novel and effective countermeasure against MIAs.
2. We provide rigorous theoretical analyses to establish a key insight: convex loss functions tend to decrease the loss variance. In contrast, concave functions can enlarge the variance of the training loss distribution.
3. We establish that CCL offers a state-of-the-art balance in the privacy-utility trade-off, with extensive experiments on Texas100, Purchase100, CIFAR-10/100, and ImageNet datasets with diverse model architectures.

2. Background

Setup. In this paper, we study the problem of membership inference attacks in K-class classification tasks. Let the feature space be $\mathcal{X} \subset \mathbb{R}^d$ and the label space be $\mathcal{Y} = \{1, \ldots, K\}$. Let us denote by $(\boldsymbol{x}, y) \in (\mathcal{X} \times \mathcal{Y})$ an example containing an instance $\boldsymbol{x}$ and a real-valued label y. Given a training dataset $\mathcal{S} = \{(\boldsymbol{x_n}, y_n)\}_{i=1}^{N}$ *i.i.d.* sampled from the data distribution $\mathcal{D}$, our goal is to learn a model $h_S \in \mathcal{H}$, that minimizes the following expected risk:

$$R(\mathcal{L}) = \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}[\mathcal{L}(h(\boldsymbol{x}), y)] \quad (1)$$

where $\mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}$ denotes the expectation over the data distribution $\mathcal{D}$ and $\mathcal{L}$ is a conventional loss function (such as cross-entropy loss) for classification.

Membership Inference Attacks. Given a data point $(\boldsymbol{x}, y)$ and a trained target model $h_{\mathcal{S}}$, attackers aim to identify if $(\boldsymbol{x}, y)$ is one of the members in the training set $\mathcal{S}$, which is called membership inference attacks (MIAs) (Shokri et al., 2017; Yeom et al., 2018; Salem et al., 2019). In MIAs, it is generally assumed that attackers can query the model predictions $h_S(\boldsymbol{x})$ for any instance $\boldsymbol{x}$. Here, we focus on standard black-box attacks (Irolla & Châtel, 2019), where attackers can access the knowledge of model architecture and the data distribution $\mathcal{D}$.

In the process of attack, the attacker has access to a query set $\mathcal{Q} = \{(\boldsymbol{z}_i, m_i)\}_{i=1}^{J}$, where $\boldsymbol{z}_i$ denotes the ith data point $(\boldsymbol{x}_i, y_i)$ and m is the membership attribute of the given data point $(\boldsymbol{x}_i, y_i)$ in the training dataset $\mathcal{S}$, i.e., $m_i = \mathbb{I}[(\boldsymbol{x}_i, y_i) \in \mathcal{S}]$. In particular, the query set $\mathcal{Q}$ contains both member (training) and non-member samples, drawn from the data distribution $\mathcal{D}$. Then, the attacker $\mathcal{A}$ can be formulated as a binary classifier, which predicts $m_i \in \{0, 1\}$ for a given example $(\boldsymbol{x}_i, y_i)$ and a target model $h_{\mathcal{S}}$. To quantify the performance of the attack model $\mathcal{A}$, we use the *membership advantage* (Yeom et al., 2018) :

$$\begin{aligned} Adv(\mathcal{A}) &:= \Pr(\mathcal{A}(h_{\mathcal{S}}(\boldsymbol{x}), y) = 1 | m = 1) \\ &\quad - \Pr(\mathcal{A}(h_{\mathcal{S}}(\boldsymbol{x}), y) = 1 | m = 0) \\ &= 2\Pr(\mathcal{A}(h_{\mathcal{S}}(\boldsymbol{x}), y) = m) - 1 \end{aligned} \quad (2)$$

Equivalently, $Adv(\mathcal{A})$ can be seen as the difference between $\mathcal{A}$'s true and false positive rates.

Loss variance. In the literature, Theorem 3 in (Yeom et al., 2018) shows that, for regression tasks, the *membership advantage* can be approximated as $\mathrm{erf}(1/\sqrt{2}) - \mathrm{erf}(\sigma_{\mathcal{S}}/\sqrt{2}\sigma_{\mathcal{D}})$, where $\mathrm{erf}(x) = \frac{1}{\sqrt{\pi}} \int_{-z}^{x} \exp(-t^2)dt$, $\sigma_{\mathcal{S}}$ and $\sigma_{\mathcal{D}}$ denote the loss variances of member and non-member data over examples, respectively. This suggests that increasing the loss variance of training data $\sigma_{\mathcal{S}}$ can help decrease the membership advantage, i.e., enhance the defense against MIAs. We formally discuss the effect of loss variance on the attack advantage in Section 6.

Recently, RelaxLoss (Chen et al., 2022) applies gradient ascent to increase the loss variance of the training data, thereby alleviating the privacy risk. However, optimizing toward a reverse direction may cause the model parameters to oscillate near local minima, preventing convergence to the global optimum. Consequently, the inconsistency of optimizing directions over iterations will result in the sub-optimal performance of the trained model in utility (see Figure 2). This motivates us to design a loss function that increases σ_S during gradient descent.

3. Theoretical Motivation

In this section, we begin with a formal analysis to show that cross-entropy loss tends to reduce $\sigma_{\mathcal{S}}$ due to its convexity. Based on this, we propose concave functions, which are theoretically shown to increase $\sigma_{\mathcal{S}}$.

For a sample $\boldsymbol{x} \in \mathcal{X}$, we denote the distribution over different labels by $q(k|\boldsymbol{x})$, the output probability of $h_S(\boldsymbol{x})$ by $p(k|\boldsymbol{x})$. For simplicity, we denote p_k, q_k as abbreviations for $p(k|\boldsymbol{x})$ and $q(k|\boldsymbol{x})$, respectively. In particular, the confidence in the true label $p(y|\boldsymbol{x})$ is abbreviated as p_y.

3.1. Convex function decreases the loss variance

Here, we provide a formal analysis to show how the loss function influences the loss variance of training data. Given a certain model, we can view $p_y = p(y|\boldsymbol{x})$ as a random variable, which depends on the pair of random variables $(\boldsymbol{x}, y) \sim \mathcal{D}$. Then, the training objective (1) with cross-entropy loss ℓ_{ce} can be rewritten as:

$$\min_{h} \quad \mathbb{E}_{\mathcal{D}}[-\log p_y]$$

where $\mathbb{E}_{\mathcal{D}}$ denotes the expectation over the data distribution $\mathcal{D}$. Let $1-\epsilon$ and σ^2 be the mean and variance of p_y over the data distribution $\mathcal{D}$, where $0 < \epsilon < 1$. By Taylor expansion, we have

$$\begin{aligned}\mathbb{E}_{\mathcal{D}}(-\log p_y) &\geqslant \mathbb{E}_{\mathcal{D}}[(1-p_y) + \frac{1}{2}(1-p_y)^2] \\ &= \mathbb{E}_{\mathcal{D}}[(1-p_y)] + \frac{1}{2}\mathbb{E}_{\mathcal{D}}[(1-p_y)^2] \\ &= \epsilon + \frac{1}{2}(\sigma^2 + \epsilon^2)\end{aligned}$$

Thus, we obtain a lower bound for the expected value of training loss, which depends on ϵ and σ^2. It implies that the training loss can be optimized toward a smaller value of variance σ^2, corresponding to a smaller loss variance [1].

Note that the above property of cross-entropy loss is dependent on the positive coefficient of σ^2, which can be calculated from the second-order derivation of cross-entropy loss. In other words, the relationship between cross-entropy loss and loss variance stems from its convexity. This insight can be formalized as follows:

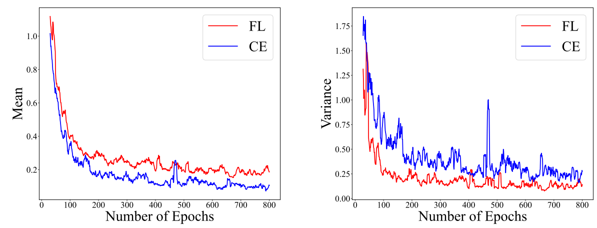
Theorem 3.1. *Given a twice continuously differentiable function $\ell \in C^2(0,1]$ such that $\ell(1) = 0$ and $\ell'(x) < 0, \forall x \in (0,1]$. If ℓ is strictly **convex**, then*

$$\mathbb{E}_{\mathcal{D}}[\ell(p_y)] \geqslant A\epsilon + \frac{B}{2}(\epsilon^2 + \sigma^2)$$

where $A = -\ell'(1) > 0$, $B \geqslant 0$ is a non-negative lower bound of $\ell''(x)$.

The detailed proof is presented in the Appendix A. By Theorem 3.1, we show that the coefficient of σ^2 is positive if the loss function is strictly convex with respect to p_y. Similar to cross-entropy loss, such convex loss functions will be optimized to encourage a smaller loss variance, being vulnerable to membership inference attacks.

To provide a straightforward view, we empirically verify the connection between convexity and loss variance. In particular, we introduce Focal Loss (Lin et al., 2017): $\ell_{\text{fl}} = -(1-p_y)^{\gamma}\log(1-p_y)$ with $\gamma = 2$, which exhibits stronger convexity than cross-entropy loss, i.e., $\ell''_{\text{fl}} \geqslant \ell''_{\text{ce}}, \forall x \in (0,1]$. In Figure 1, we plot the dynamics of mean and variance of losses for the training data during the training. Indeed, Focal loss results in a smaller loss variance than cross-entropy loss, with a higher mean of losses. In this way, we verified that the convexity of loss functions contributes to the decrease of loss variance, which may exacerbate the privacy risk of neural networks. We proceed by exploring concave functions, targeting this problem.



(a) Mean of Loss (b) Variance of Loss

Figure 1. The mean and variance of loss under different epochs. Models are trained on CIFAR-10 with Resnet-34 using Cross-entropy loss (CE) and Focal loss (FL).

3.2. Can concave functions increase the loss variance?

In our previous analysis, we show that convex loss can enlarge the privacy risk with a small loss variance. Conversely, we explore the properties of concave functions in the same setting, as shown below.

Theorem 3.2. *Given a twice continuously differentiable function $\ell \in C^2[0,1]$ such that $\ell(1) = 0$ and $\ell'(x) < 0, \forall x \in [0,1]$. If ℓ is strictly **concave**, there must exist a **negative** constant $B \leqslant 0$ such that*

$$\mathbb{E}_{\mathcal{D}}[\ell(p_y)] = A\epsilon + B(\sigma^2 + \epsilon^2) \tag{3}$$

where $A = -\ell'(1) > 0$.

The detailed proof is presented in the Appendix B. Given the above theorem, we find that the coefficient of σ^2 is **negative** if the loss function ℓ is strictly concave. In other words, a concave loss function can increase the loss variance during gradient descent, which is the expected property for mitigating privacy risk. In what follows, we propose a general framework that endows the robustness to cross-entropy loss against membership inference attacks.

4. Our Proposed Method

Theorem 3.2 indicates that concave functions can be leveraged to design loss functions. To design our loss function, we first introduce a formal definition of the concave term.

Definition 4.1 (Concave Term)**.** We define a concave func-

[1]The monotonic relationship between loss variance and σ^2 is proved in Appendix C.

tion set as:

$$\mathcal{F} = \{f \in C^2[0,1] \mid f'(x) < 0, f''(x) < 0, \forall x \in [0,1]\}$$

In this definition, (1)$f'(x) < 0$ ensures that the smaller the objective loss, the larger the confidence of true label p_y, which can preserve utility, (2) $f''(x) < 0$ ensures that this a concave function that can help increase the variance of p_y.

Convex-concave loss. We propose to add a concave term into the original loss function (e.g., cross-entropy loss), which is called *Convex-Concave Loss* (CCL):

$$\ell_{\text{ccl}} = \alpha\hat{\ell} + (1-\alpha)\tilde{\ell} \tag{4}$$

where $\hat{\ell}$ is the origin convex function, $\tilde{\ell} \in \mathcal{F}$ is a concave term, and $\alpha \in [0,1]$ denotes a hyperparameter to adjust the trade-off between privacy and utility flexibly. In this paper, we just consider cross-entropy loss as the original loss function. The experiments with other convex loss functions are provided in Appendix E.

Theorem 3.2 indicates that concave functions can be leveraged to design loss functions. However, in the early stages of training epochs, this approach tends to result in a smaller step size in the gradient ascent process. Consequently, we employ CE loss to facilitate better convergence (Zhang & Sabuncu, 2018) and more effective learning.

For specific concave functions, it is possible to select commonly used monotonic non-linear functions for their design. For instance, we can take the negative of the exponential function as Concave Exponential Loss (CEL):

$$\tilde{\ell}_{\text{exp}} = -\exp(p_y) \tag{5}$$

Alternatively, we can just employ a quadratic polynomial function as Concave Quadratic Loss (CQL):

$$\tilde{\ell}_{\text{qua}} = -p_y - \frac{1}{2}p_y^2 \tag{6}$$

These two concave terms are all in $\mathcal{F}$. We denote these two concave functions integrated with ℓ_{ce} as CCEL and CCQL.

Gradient analysis. There we derive the gradients of CCL. Consider the case of a single true label, we obtain the gradient of the concave term $\tilde{l} \in \mathcal{F}$ w.r.t the logits z_j as follows:

$$\frac{\partial\tilde{\ell}}{\partial z_j} = \frac{\partial p_y}{\partial z_j}\tilde{\ell}'(p_y) \tag{7}$$

$$= \begin{cases} p_y(1-p_y)\cdot\tilde{\ell}'(p_y) \leqslant 0, & j = y \\ -p_j p_y\cdot\tilde{\ell}'(p_y) \geqslant 0, & \text{otherwise} \end{cases} \tag{8}$$

As for CE loss ℓ_{ce}, the gradient is

$$\frac{\partial\ell_{\text{ce}}}{\partial z_j} = \begin{cases} p_y - 1 \leqslant 0, & j = y \\ p_j \geqslant 0, & \text{otherwise} \end{cases} \tag{9}$$

It is clear that the gradient of the concave term in $\mathcal{F}$ has the same sign with CE loss for each j, so does the complete loss function ℓ_{ccl}:

$$\frac{\partial\ell_{\text{ccl}}}{\partial z_j} = \begin{cases} (p_y-1)[\alpha-(1-\alpha)p_y\tilde{\ell}'(p_y)] \leqslant 0, & j = y \\ p_j[\alpha-(1-\alpha)p_y\tilde{\ell}'(p_y)] \geqslant 0, & \text{otherwise} \end{cases} \tag{10}$$

$$= [\alpha-(1-\alpha)p_y\tilde{\ell}'(p_y)]\frac{\partial\ell_{\text{ce}}}{\partial z_j} \tag{11}$$

where $\alpha-(1-\alpha)p_y\tilde{\ell}'(p_y) \geqslant 0$. Considering the concavity of $\tilde{\ell}$, it follows that a greater p_y will result in a larger magnitude of $|p_y\tilde{\ell}'(p_y)|$. This can be interpreted such that $\alpha-(1-\alpha)p_y\tilde{\ell}'(p_y)$ acts as an acceleration coefficient. That is, for these samples with higher p_y, this coefficient serves to increase p_y more rapidly. Consequently, this leads to a broadening in the range of confidence distributions.

Furthermore, we provide bounds of the gradient of ℓ_{ccl}.

Proposition 4.2. *For any input $\boldsymbol{x}$ and any $\alpha > 0$, the gradient of ℓ_{ccl} w.r.t logits z_j is bounded above and below as follows:*

$$\alpha\frac{\partial\ell_{\text{ce}}}{\partial z_j} \leqslant \frac{\partial\ell_{\text{ccl}}}{\partial z_j} \leqslant [\alpha + A(1-\alpha)]\frac{\partial\ell_{\text{ce}}}{\partial z_j} \tag{12}$$

where $A = -\tilde{\ell}'(1) > 0$.

From Proposition 4.2, the gradient of ℓ_{ccl} is bounded by the scaled gradient of ℓ_{ce} in each dimension. Therefore, the gradient of the proposed loss $\frac{\partial\ell_{\text{ccl}}}{\partial z_j}$ will approach zero when CE loss achieves its optimum ($\frac{\partial\ell_{\text{ce}}}{\partial z_j} = 0$). This indicates that our approach is capable of converging in the same region where CE achieves its optimum.

5. Experiments

In this section, we validate the effectiveness of our CCL across a wide range of datasets with diverse models, various attack models, and multiple defense baselines.

5.1. Setups

Datasets. In our evaluation, we employ five datasets: Texas100 (Texas Department of State Health Services, 2006), Purchase100 (Kaggle, 2014), CIFAR-10, CIFAR-100 (Krizhevsky et al., 2009), and ImageNet (Russakovsky et al., 2015). For standard training methods, we split each dataset into four subsets, with each subset serving alternately as the training or testing set for the target and shadow models. As for adversarial training algorithms that incorporate adversary loss—such as Mixup+MMD (Li et al., 2021) and adversarial regularization (Nasr et al., 2018)—we divide the datasets into five subsets. The additional subset is specifically utilized to generate adversary loss.

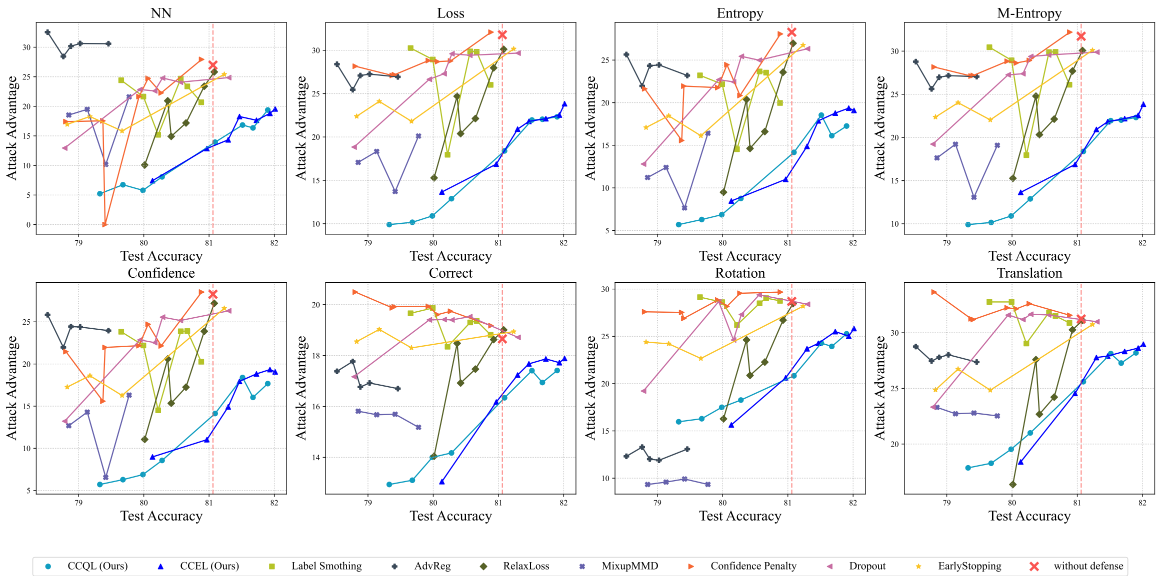


Figure 2. Comparisons of seven defense mechanisms on CIFAR-10 dataset utilizing Resnet34 architecture. Each subplot is allocated to a distinct attack method, wherein individual curves represent the performance of a defense mechanism under different hyperparameter settings. The horizontal axis represents the target models' test accuracy (the higher the better), and the vertical axis represents the corresponding attack advantage (defined in Definition 2, the lower the better). To underscore the disparity between the defense methods and the vanilla (undefended model), we plot the dotted line originating from the vanilla results.

Training details. We train the models using SGD with a momentum of 0.9, a weight decay of 0.0005, and a batch size of 128. We set the initial learning rate as 0.1 and drop it by a factor of 10 at each decay epoch. For CIFAR-10 and CIFAR-100, we conduct training using a 34-layer ResNet (He et al., 2016) and a 121-layer DenseNet (Huang et al., 2017), with decay milestones set at 150, 225 over a total of 300 epochs. In the case of Imagenet, we employ a 121-layer DenseNet with decay milestones at $\{30, 60\}$, spanning a total of 90 epochs. For Texas100 and Purchase100, training is performed using MLPs as described in previous studies (Nasr et al., 2018; Jia et al., 2019), with decay milestones at $\{50, 100\}$ across 120 total epochs.

5.2. Hyperparameter Tuning

In our approach to hyperparameter tuning, we align with the protocols established by previous work (Chen et al., 2022). In particular, we employ hyperparameter tuning focused on a single hyperparameter, α defined in 4. Through a detailed grid search on a validation set, we adjust α to achieve an optimal balance. This process involves evaluating the privacy-utility implications at various levels of α and then selecting the value that aligns with our specific privacy/utility objectives, thereby enabling precise management of the model's privacy and utility. Following focal loss (Lin et al., 2017), we set a scalar factor on our loss functions. Specifically, for CIFAR-10, the scale factor is 0.01; for CIFAR-100, it is 0.05. For other datasets, we set it as 1. As for our hyperparameter α, we vary it across $\{0.1, \ldots, 0.9\}$.

Attack methods. In our study, we experiment with three classes of MIA: (1) Neural Network-based Attack (NN) (Shokri et al., 2017; Hu et al., 2022a), which leverages the full logits prediction as input for attacking the neural network model. (2) Metric-based Attack, employing specific metrics computation followed by a comparison with a preset threshold to ascertain the data record's membership status. The metrics we chose for our experiments include Correctness, Loss (Yeom et al., 2018), Confidence, Entropy (Salem et al., 2019), and Modified-Entropy (M-entropy) (Song & Mittal, 2021). (3) Augmentation-based Attack (Choquette-Choo et al., 2021), utilizing prediction data derived through data augmentation techniques as inputs for a binary classifier model. In this category, we specifically implemented rotation and translation augmentations.

For the details of the attack, we assume the most powerful black-box adaptive attack scenario: the adversary has complete knowledge of our defense mechanism and selected hyperparameters. To implement this, we train shadow models with the same settings used for our target models.

Defense baselines. We compare CCL with seven defense methods: RelaxLoss (Chen et al., 2022), Mixup+MMD (Li et al., 2021), Adversary regularization (Adv-Reg) (Li & Zhang, 2021), Dropout (Srivastava et al., 2014), Label

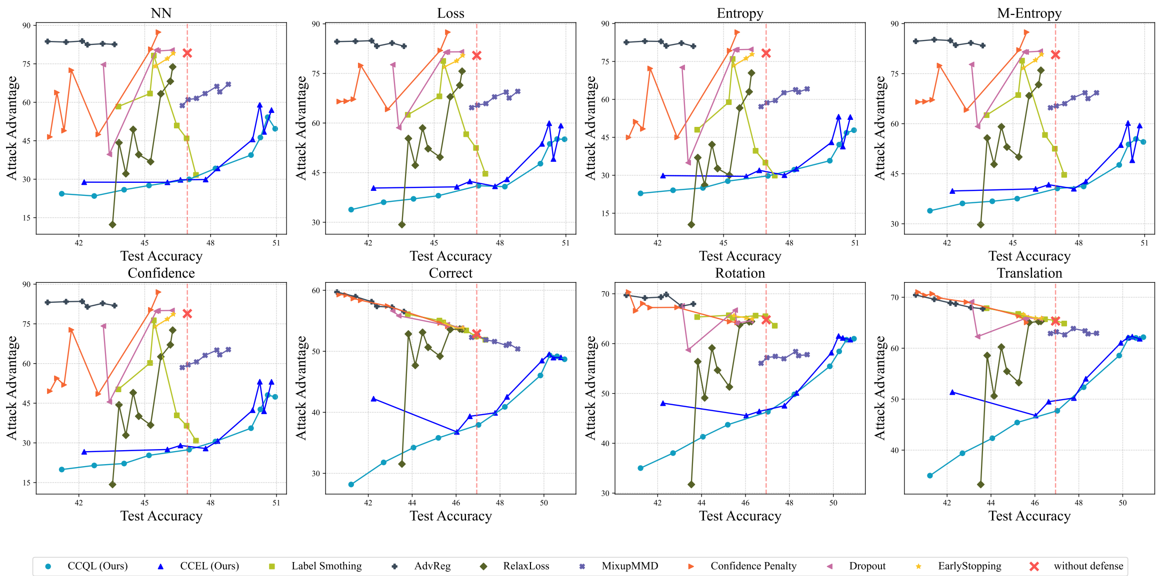


Figure 3. Comparisons of seven defense mechanisms on CIFAR-100 dataset utilizing Densenet121 architecture. Each subplot is allocated to a distinct attack method, wherein individual curves represent the performance of a defense mechanism under different hyperparameter settings. The horizontal axis represents the target models' test accuracy (the higher the better), and the vertical axis represents the corresponding attack advantage (defined in Definition 2, the lower the better). To underscore the disparity between the defense methods and the vanilla (undefended model), we plot the dotted line originating from the vanilla results.

Smoothing (Guo et al., 2017), Confidence Penalty (Pereyra et al., 2017), and Early Stopping (Yao et al., 2007).

Evaluation Metrics. To comprehensively assess our method's impact on privacy and utility, we employ three evaluation metrics that encapsulate utility, privacy, and the balance between the two. Utility is gauged by the test accuracy of the target model. Privacy is measured through the attack advantage, as defined in Equation 2. To assess the trade-off between utility and privacy, we utilize the P1 score (Paul et al., 2021), which is defined as:

$$P1 = 2 * \frac{Acc * (1 - Adv)}{(Acc) + (1 - Adv)} \tag{13}$$

where Acc denotes the test accuracy and Adv denotes the attack advantage of the attacker on the target model.

5.3. Results

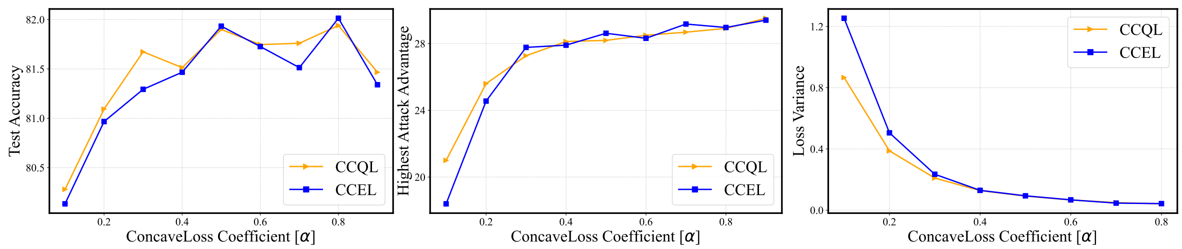
Can CCL improve privacy-utility trade-off ? In Picture 2 and Picture 3, we plot privacy-utility curves to show the privacy-utility trade-off. The horizontal axis represents the performance of the target model, and the vertical axis represents the attack advantage defined in 2. A salient observation is that both of our methods drastically improve the privacy-utility trade-off. In particular, for these points that perform better than vanilla for utility (the area to the right of the dotted line), the privacy-utility curves of our methods are always below those of others. This means we can always obtain the highest privacy for any utility requirement higher than the undefended model. For example, on the CIFAR10 dataset, we focus on the hyperparameter α corresponding to the model with the lowest attack advantage with the constrain condition that test accuracy is better than vanilla, then our method with quadratic function can decrease the attack advantage of loss-metric-based from 29.67% to 18.40% compared with Dropout (the most powerful defend method under our condition above).

Is CCL effective with different datasets? To ascertain the efficacy of our proposed method across heterogeneous data, we have executed a series of experiments on a diverse array of datasets, encompassing tabular and image datasets. For the experimental results shown in Table 1, we have set the adjustment coefficient of CCL to a constant value, specifically $\alpha = 0.5$. To assess the privacy-utility balanced performance, we use the highest attack advantage of all attack methods to calculate the P1 score From the results, we observe that both of our methods yield a consistent improvement in the P1 score.

How does α affect utility and privacy? In Figure 4a, Figure 4b and Figure 4c, we conduct an ablation study to examine the impact of the coefficient α in our method on both utility, privacy, and loss variance. The analysis is based on CIFAR-10. As is shown in Figure 4c, our findings are in alignment with the insights provided in Theorem

Dataset	Texas	Purchase	ImageNet	CIFAR-10	CIFAR-100
CCQL	0.607	**0.868**	**0.610**	0.769	**0.487**
CCEL	**0.608**	0.864	0.609	**0.797**	0.480
w/o	0.551	0.858	0.598	0.741	0.273

Table 1. P1 score (defined in Equation 13) evaluated on target models trained on different datasets. The bold indicates the best results. Here, ”w/o” denotes undefended models.



(a) Utility (b) Privacy (c) Variance

Figure 4. The effect of α on utility (test accuracy), privacy (highest attack advantage), and loss variance.

3.2 and Theorem 3.1. As the α decreases, the effect of the concave term becomes more significant, leading to a gradual increase in variance. On the other hand, a larger α value brings our loss function closer to the cross-entropy loss, thereby increasing the privacy risk. Conversely, a smaller α value leads to a smaller gradient effect, culminating in underfitting, which consequently diminishes accuracy.

Convergence analysis. As demonstrated in Equation 8, our CCL induces a larger gradient compared to CE loss when the true label confidence p_y approaches 1. Conversely, during the initial epochs when p_y is smaller, our approach tends to result in smaller gradient steps. Although a detailed gradient analysis is presented in Section 4, it may also ask: Can a model utilizing CCL achieve a stable state? Furthermore, does the implementation of CCL lead to a slower convergence rate? There, we conduct an experiment on CIFAR-10 datasets with fixed $\alpha = 0.5$ and plot the training loss curve and the test accuracy curve. As Figure 5 shows, both of our proposed CCL functions can converge properly, and there is no significantly longer training time than that of CE to reach convergence.

6. Discussion

How does loss variance affect the attack advantage? Normally, the target model is trained on its training members with the objective of minimizing the error in its predictions. Consequently, the prediction error for a sample within the training dataset would be less than that for a sample within the testing dataset. In this way, the attack model $\mathcal{A}_{\text{loss}}$ (Yeom et al., 2018) is defined as:

$$\mathcal{A}_{\text{loss}} = \mathbb{I}(\mathcal{L}_{\text{CE}}(h_S(\boldsymbol{x}), y) \leqslant \tau) \tag{14}$$

That is, given a sample, we calculate its loss by the target model and then infer it as a member if its loss is smaller than a preset threshold τ[2].

Under the common Gaussian assumption of loss distribution (Yeom et al., 2018; Chen et al., 2022), we first introduce a theorem to show how loss variance affects the attack advantage of attack model $\mathcal{A}_{\text{loss}}$ (defined in Equation 14).

Theorem 6.1. *Suppose ϵ is a random variable denoting loss, such that $\epsilon \sim N(\mu_S, \sigma_S^2)$ when $m = 1$ and $\epsilon \sim N(\mu_D, \sigma_D^2)$ when $m = 0$. Then the membership advantage of $\mathcal{A}_{\text{loss}}$ is:*

$$\begin{aligned} Adv =& \Pr(\mathcal{A} = 1|m = 1) - \Pr(\mathcal{A} = 1|m = 0) & (15) \\ =& \Pr(\epsilon \leqslant \tau|m = 1) - \Pr(\epsilon \leqslant \tau|m = 0) & (16) \\ =& \Phi(\frac{\tau - \mu_S}{\sigma_S}) - \Phi(\frac{\tau - \mu_D}{\sigma_D}) & (17) \end{aligned}$$

where $\Phi(\cdot)$ is the cumulative distribution function of standard normal distribution.

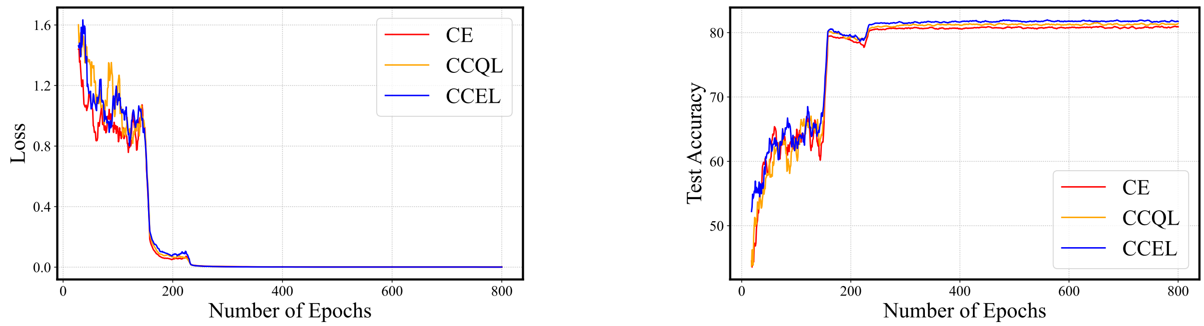
Note that $\Pr(\mathcal{A} = 1|m = 0)$ is false positive rates of the adversary, which is expected to be controlled at a small value (Leemann et al., 2023; Tan et al., 2022). Assume τ is chosen such that $\Phi(\frac{\tau - \mu_D}{\sigma_D}) = \alpha$, then we have:

$$Adv = \Phi\{\frac{\Phi^{-1}(\alpha)\sigma_D + \mu_D - \mu_S}{\sigma_S}\} - \alpha \tag{18}$$

This implies that increasing the variance of training loss distribution σ_S can help to decrease the attack advantage.

Can our method converge? In general, non-convex losses do not prevent the convergence of optimization,

[2] τ is determined by shadow models



(a) Loss

(b) Test Accuracy

Figure 5. Convergence analysis of CCL on CIFAR-10 dataset

which has been well-studied in the literature (Khaled & Richtárik, 2020; Zou et al., 2019; Garrigos & Gower, 2023; Défossez et al., 2022). For example, the work (Khaled & Richtárik, 2020) proves the convergence of SGD in the non-convex setting under expected smoothness assumption and yields the optimal convergence rate $\mathcal{O}\left(\varepsilon^{-4}\right)$ for finding a stationary point of non-convex smooth functions. Another work (Zou et al., 2019) introduces a sufficient condition to guarantee the global convergence of generic Adam/RMSProp optimizers in the non-convex setting.

Here we conclude two common sufficient conditions related to loss function as follows:

(C1) The minimum value of $\mathcal{L}$ is lower-bounded.

(C2) $\mathcal{L}$ is smooth, i.e., $\|\nabla\mathcal{L}(\boldsymbol{x}) - \nabla\mathcal{L}(\boldsymbol{y})\| < L_1\|\boldsymbol{x}-\boldsymbol{y}\|$, for all $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$ and for any $L_1 > 0$.

We proceed by proving that CCL satisfies the two conditions, supporting the convergence of CCL with popular optimizers, such as SGD, Adam, and RMSProp.

CCL satisfies the condition (C1). As $\mathcal{L}$ is just linear combination of ℓ_{ccl}, so we just prove ℓ_{ccl} is lower bounded. By definition, ℓ_{ccl} is a monotonic decreasing function and closed on the right side of the domain $(0, 1]$, so it is lower bounded.

CCL satisfies the condition (C2). Before we prove C2, we first introduce two lemmas.

Lemma 6.2. *Let $A(x) : D \subseteq \mathbb{R}^n \to \mathbb{R}^{m\times p}$ and $B(x) : D \subseteq \mathbb{R}^n \to \mathbb{R}^{p\times q}$ be two Lipschitz continuous matrix functions defined on the same domain D, with Lipschitz constants L_A and L_B, respectively. If $\|A(x)\|$ and $\|B(x)\|$ are bounded for all $x \in D$ by some constants M_A and M_B, then the product function $C(x) = A(x)B(x)$ is also Lipschitz continuous on D.*

Lemma 6.3. *If two functions $f : \mathbb{R}^n \to \mathbb{R}^m$ and $g : \mathbb{R}^m \to \mathbb{R}^p$ are Lipschitz continuous, with Lipschitz constants L_f and L_g respectively, then their composite function $h = g \circ f$ defined as $h(x) = g(f(x))$ is also Lipschitz continuous.*

The proofs of these two Lemmas 6.2 and 6.3 are provided in the Appendix D. By chain rule, we have $\nabla\mathcal{L} = (\nabla\boldsymbol{z})^\top \frac{\partial\mathcal{L}}{\partial\boldsymbol{z}}$. Our method differs from CE loss in $\frac{\partial\mathcal{L}}{\partial\boldsymbol{z}}$, we only need to prove that our loss function ℓ_{ccl} is smooth with respect to logits $\boldsymbol{z}$. Note that the confidence in true label p_y can be written as $\boldsymbol{y}^\top\boldsymbol{p}$, so we have $\frac{\partial\ell_{\text{ccl}}}{\partial\boldsymbol{z}} = [\alpha - (1-\alpha)\tilde{\ell}'(\boldsymbol{y}^\top\boldsymbol{p})\boldsymbol{y}^\top\boldsymbol{p}](\boldsymbol{p}-\boldsymbol{y})$, where $\boldsymbol{p}$ is probability vector and $\boldsymbol{y}$ is one-hot label vector. Given Lemma 6.2 and 6.3, we can simplify the proof to establishing that both $\tilde{\ell}'$ and $\boldsymbol{y}^\top\boldsymbol{p}$ are Lipschitz continuous.

Since $\tilde{\ell} \in C^2[0, 1]$, there exits an upper bound B such that $\tilde{\ell}'' \leqslant B$, which follows that $\|\tilde{\ell}'(x) - \tilde{\ell}'(y)\| \leqslant B\|x - y\|$. With the fact that Softmax function is Lipschitz continuous [5], we have $\|\boldsymbol{y}^\top\boldsymbol{p}_1 - \boldsymbol{y}^\top\boldsymbol{p}_2\| \leqslant \|\boldsymbol{y}\| \cdot \|\boldsymbol{p}_1 - \boldsymbol{p}_2\| = \|\boldsymbol{p}_1 - \boldsymbol{p}_2\| \leqslant L_2\|\boldsymbol{z_1} - \boldsymbol{z_1}\|$.

By Lemma 6.2 and Lemma 6.3, we conclude that $\frac{\partial\ell_{\text{ccl}}}{\partial\boldsymbol{z}}$ is Lipschitz continuous, followed by it is smoothness.

7. Related Work

Membership Inference Attacks (MIAs). Membership Inference Attacks (MIAs), first introduced by Shokri et al. (2017) for machine learning, utilize multiple shadow models and a neural network-based attack model to identify a target model's predictions on member versus non-member data, that is NN-based attack(Zhang et al., 2023; Nasr et al., 2019). Metric-based attack computes custom metrics such as Loss (Yeom et al., 2018), Confidence(Liu et al., 2019a), Entropy (Salem et al., 2019), Modified-Entropy (Song & Mittal, 2021), and gradient norm (Leemann et al., 2023; Nasr et al., 2019; Sablayrolles et al., 2019) to derive a threshold for distinction. As an extension to metric-based attacks, recent works (Lopez et al., 2023; Carlini et al., 2022; Ye et al., 2022; Watson et al., 2022) design per-example attacks. Boundary-based attacks (Li & Zhang, 2021; Choquette-

Choo et al., 2021) gauge the necessary perturbation magnitude for membership inference. Augmentation-based attacks (Choquette-Choo et al., 2021; Ko et al., 2023) leverage the resilience of training samples to data augmentation compared to testing samples.

Beyond supervised classification, MIAs have been extended to additional fields, such as graph embedding models (Wang & Wang, 2023; Duddu et al., 2020), graph neural network(Liu et al., 2022; Conti et al., 2022; Olatunji et al., 2021), generative models (Pang et al., 2023; Dubiński et al., 2024; van Breugel et al., 2023; Hu & Pang, 2021; Chen et al., 2020a; Liu et al., 2019b), contrastive learning (Liu et al., 2021; Ko et al., 2023), language models (Mattern et al., 2023; Mireshghallah et al., 2022).

Defend against MIAs. The existing defense technologies can be classified into four categories(Hu et al., 2023): regularization, transfer learning, information perturbation, and generative models-based.

(i) Regularization: Many papers (Leino & Fredrikson, 2020; Salem et al., 2019; Yeom et al., 2018; Shokri et al., 2017) have pointed out that overfitting is a major factor in the success of MIAs, hence regularization technology such as Label Smoothing (Guo et al., 2017), Confidence Penalty (Pereyra et al., 2017), and Early Stopping (Yao et al., 2007), Adversary regularization (Nasr et al., 2018), Dropout (Srivastava et al., 2014), Pruning(Wang et al., 2021), HAMP (Chen & Pattabiraman, 2024), and RelaxLoss (Chen et al., 2022) can certainly defend against MIAs.

(ii) Transfer Learning has been shown to effectively protect member privacy by using knowledge from similar but different data, reducing direct access to sensitive target data. In particular, Knowledge distillation (Mazzone et al., 2022; Hinton et al., 2015; Shejwalkar & Houmansadr, 2021; Zheng et al., 2021; Tang et al., 2022) uses a large teacher model to train a smaller student model, transferring the knowledge while retaining similar accuracy. Domain adaptation (Weiss et al., 2016; Huang, 2021; Huang et al., 2021) transfers knowledge from a source domain to a related but different target domain by extracting shared representations.

(iii) Information perturbation protects privacy information by adding customized noise to the data, training procession, and outputs. Specially, this methodology is typically classified into three methods: differential privacy, output perturbation, and data augmentation. Differential privacy (Tan et al., 2022; Dwork, 2008; Chen et al., 2020b; Jayaraman & Evans, 2019; Nasr et al., 2021; Rahman et al., 2018; Kim et al., 2021; Truex et al., 2019) adds noise perturbations to the real data, ensuring that the results of an algorithm on adjacent datasets (differing by one element) are statistically indistinguishable. Output perturbation (Xue et al., 2022; Jia et al., 2019) based on the intuition that black-box MIAs can only utilize this output information to make inferences, so lightly altering the results returned by the model weakens the performance of MIAs. Data perturbation (Chen et al., 2021b; Wang et al., 2019; Kandpal et al., 2022) add perturbations directly to the data, making it more challenging for attackers to infer whether specific data points were used in training the model. Techniques such as data augmentation, including rotation, clipping, and mix-up (Li et al., 2021; Kaya & Dumitras, 2021), can also fall into this category.

(iv) Generative models-based methods (Chen et al., 2021a; Hu et al., 2022b; Paul et al., 2021) generated substitute training data using generative models to reduce information leakage. Our approach is a regularization technique integrated into the loss function.

8. Conclusion

In this paper, we introduce Convex-Concave Loss, a simple method for formulating loss functions that can help defend against MIAs. Specifically, we propose to integrate a concave term with CE loss to magnify loss variance. As a result, our proposed method could mitigate privacy risks by reducing the gap between training and testing loss distribution. Moreover, we present theoretical analyses of how convex and concave loss functions affect loss variance during optimization, which is the key insight enlightening and certifying our method. Extensive experiments show that CCL can improve the privacy-utility trade-off.

A few open questions remain: Firstly, our method aims to increase the variance of output metrics, but its role in defending against label-only attacks (such as data augmentation attacks) remains unexplored. Moreover, our method cannot break the trade-off between utility and MIA defense, which might be a potential direction for future work.

Acknowledgements

This research is supported by the Shenzhen Fundamental Research Program (Grant No.JCYJ20230807091809020). Huiping Zhuang is supported by the National Natural Science Foundation of China (Grant No. 6230070401), Xiaofeng Cao is supported by the National Natural Science Foundation of China (Grant No.62206108). We gratefully acknowledge the support of the Center for Computational Science and Engineering at the Southern University of Science and Technology.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

References

Arshad, S., Arshad, J., Khan, M. M., and Parkinson, S. Analysis of security and privacy challenges for dna-genomics applications and databases. *Journal of Biomedical Informatics*, 119:103815, 2021.

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

Chen, D., Yu, N., Zhang, Y., and Fritz, M. Gan-leaks: A taxonomy of membership inference attacks against generative models. In *Proceedings of the 2020 ACM SIGSAC conference on computer and communications security*, pp. 343–362, 2020a.

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

Chen, J., Wang, W. H., and Shi, X. Differential privacy protection against membership inference attack on machine learning for genomic data. In *BIOCOMPUTING 2021: Proceedings of the Pacific Symposium*, pp. 26–37. World Scientific, 2020b.

Chen, J., Guo, Y., Zheng, Q., and Chen, H. Protect privacy of deep classification networks by exploiting their generative power. *Machine Learning*, 110:651–674, 2021a.

Chen, J., Wang, W. H., Gao, H., and Shi, X. Par-gan: improving the generalization of generative adversarial networks against membership inference attacks. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data mining*, pp. 127–137, 2021b.

Chen, Z. and Pattabiraman, K. Overconfidence is a dangerous thing: Mitigating membership inference attacks by enforcing less confident prediction. In *30th Annual Network and Distributed System Security Symposium*, 2024.

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

Conti, M., Li, J., Picek, S., and Xu, J. Label-only membership inference attack against node-level graph neural networks. In *Proceedings of the 15th ACM Workshop on Artificial Intelligence and Security*, pp. 1–12, 2022.

Défossez, A., Bottou, L., Bach, F. R., and Usunier, N. A simple convergence proof of adam and adagrad. *Trans. Mach. Learn. Res.*, 2022, 2022. URL https://openreview.net/forum?id=ZPQhzTSWA7.

Dubiński, J., Kowalczuk, A., Pawlak, S., Rokita, P., Trzciński, T., and Morawiecki, P. Towards more realistic membership inference attacks on large diffusion models. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 4860–4869, 2024.

Duddu, V., Boutet, A., and Shejwalkar, V. Quantifying privacy leakage in graph embedding. In *MobiQuitous 2020-17th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, pp. 76–85, 2020.

Dwork, C. Differential privacy: A survey of results. In *International Conference on Theory and Applications of Models of Computation*, pp. 1–19. Springer, 2008.

Garrigos, G. and Gower, R. M. Handbook of convergence theorems for (stochastic) gradient methods. *arXiv preprint arXiv:2301.11235*, 2023.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks. In *International Conference on Machine Learning*, pp. 1321–1330. PMLR, 2017.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Hu, H. and Pang, J. Membership inference attacks against gans by leveraging over-representation regions. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pp. 2387–2389, 2021.

Hu, H., Salcic, Z., Sun, L., Dobbie, G., Yu, P. S., and Zhang, X. Membership inference attacks on machine learning: A survey. *ACM Computing Surveys (CSUR)*, 54(11s):1–37, 2022a.

Hu, L., Li, J., Lin, G., Peng, S., Zhang, Z., Zhang, Y., and Dong, C. Defending against membership inference attacks with high utility by gan. *IEEE Transactions on Dependable and Secure Computing*, 2022b.

Hu, L., Yan, A., Yan, H., Li, J., Huang, T., Zhang, Y., Dong, C., and Yang, C. Defenses to membership inference attacks: A survey. *ACM Computing Surveys*, 56(4):1–34, 2023.

Huang, G., Liu, Z., Van Der Maaten, L., and Weinberger, K. Q. Densely connected convolutional networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 4700–4708, 2017.

Huang, H. Defense against membership inference attack applying domain adaptation with addictive noise. *Journal of Computer and Communications*, 9(05):92–108, 2021.

Huang, H., Luo, W., Zeng, G., Weng, J., Zhang, Y., and Yang, A. Damia: leveraging domain adaptation as a defense against membership inference attacks. *IEEE Transactions on Dependable and Secure Computing*, 19 (5):3183–3199, 2021.

Irolla, P. and Châtel, G. Demystifying the membership inference attack. In *2019 12th CMI Conference on Cybersecurity and Privacy (CMI)*, pp. 1–7, 2019. doi: 10.1109/CMI48017.2019.8962136.

Jayaraman, B. and Evans, D. Evaluating differentially private machine learning in practice. In *28th USENIX Security Symposium (USENIX Security 19)*, pp. 1895–1912, 2019.

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

Kaggle. Acquire valued shoppers challenge, 2014. URL https://www.kaggle.com/c/acquire-valued-shoppers-challenge/data.

Kandpal, N., Wallace, E., and Raffel, C. Deduplicating training data mitigates privacy risks in language models. In *International Conference on Machine Learning*, pp. 10697–10707. PMLR, 2022.

Kaya, Y. and Dumitras, T. When does data augmentation help with membership inference attacks? In *International Conference on Machine Learning*, pp. 5345–5355. PMLR, 2021.

Khaled, A. and Richtárik, P. Better theory for sgd in the nonconvex world. *arXiv preprint arXiv:2002.03329*, 2020.

Kim, M., Günlü, O., and Schaefer, R. F. Federated learning with local differential privacy: Trade-offs between privacy, utility, and communication. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2650–2654. IEEE, 2021.

Ko, M., Jin, M., Wang, C., and Jia, R. Practical membership inference attacks against large-scale multi-modal models: A pilot study. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4871–4881, 2023.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. *Master's thesis, Department of Computer Science, University of Toronto*, 2009.

Leemann, T., Pawelczyk, M., and Kasneci, G. Gaussian membership inference privacy. In *37th Conference on Neural Information Processing Systems (NeurIPS)*, 2023.

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

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

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

Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2980–2988, 2017.

Liu, G., Wang, C., Peng, K., Huang, H., Li, Y., and Cheng, W. Socinf: Membership inference attacks on social media health data with machine learning. *IEEE Transactions on Computational Social Systems*, 6(5):907–921, 2019a.

Liu, H., Jia, J., Qu, W., and Gong, N. Z. Encodermi: Membership inference against pre-trained encoders in contrastive learning. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pp. 2081–2095, 2021.

Liu, K. S., Xiao, C., Li, B., and Gao, J. Performing co-membership attacks against deep generative models. In *2019 IEEE International Conference on Data Mining (ICDM)*, pp. 459–467. IEEE, 2019b.

Liu, Z., Zhang, X., Chen, C., Lin, S., and Li, J. Membership inference attacks against robust graph neural network. In *International Symposium on Cyberspace Safety and Security*, pp. 259–273. Springer, 2022.

Lopez, M. B., Tang, S., Kearns, M., Morgenstern, J., Roth, A., and Wu, Z. S. Scalable membership inference attacks via quantile regression. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Mahalle, A., Yong, J., Tao, X., and Shen, J. Data privacy and system security for banking and financial services industry based on cloud computing infrastructure. In *2018 IEEE 22nd International Conference on Computer Supported Cooperative Work in Design ((CSCWD))*, pp. 407–413. IEEE, 2018.

Malinin, A. and Gales, M. Predictive uncertainty estimation via prior networks. *Advances in Neural Information Processing Systems*, 31, 2018.

Mattern, J., Mireshghallah, F., Jin, Z., Schölkopf, B., Sachan, M., and Berg-Kirkpatrick, T. Membership inference attacks against language models via neighbourhood comparison. In *ACL (Findings)*, pp. 11330–11343. Association for Computational Linguistics, 2023.

Mazzone, F., van den Heuvel, L., Huber, M., Verdecchia, C., Everts, M., Hahn, F., and Peter, A. Repeated knowledge distillation with confidence masking to mitigate membership inference attacks. In *Proceedings of the 15th ACM Workshop on Artificial Intelligence and Security*, pp. 13–24, 2022.

Mireshghallah, F., Goyal, K., Uniyal, A., Berg-Kirkpatrick, T., and Shokri, R. 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. Association for Computational Linguistics, 2022.

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

Nasr, M., Shokri, R., and Houmansadr, A. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *2019 IEEE Symposium on Security and Privacy (SP)*, pp. 739–753. IEEE, 2019.

Nasr, M., Songi, S., Thakurta, A., Papernot, N., and Carlin, N. Adversary instantiation: Lower bounds for differentially private machine learning. In *2021 IEEE Symposium on Security and Privacy (SP)*, pp. 866–882. IEEE, 2021.

Oehlert, G. W. A note on the delta method. *American Statistician*, pp. 27–29, 1992.

Olatunji, I. E., Nejdl, W., and Khosla, M. Membership inference attack on graph neural networks. In *2021 Third IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA)*, pp. 11–20. IEEE, 2021.

Pang, Y., Wang, T., Kang, X., Huai, M., and Zhang, Y. White-box membership inference attacks against diffusion models. *arXiv preprint arXiv:2308.06405*, 2023.

Paul, W., Cao, Y., Zhang, M., and Burlina, P. Defending medical image diagnostics against privacy attacks using generative methods: Application to retinal diagnostics. In *Clinical Image-Based Procedures, Distributed and Collaborative Learning, Artificial Intelligence for Combating COVID-19 and Secure and Privacy-Preserving Machine Learning: 10th Workshop, CLIP 2021, Second Workshop, DCL 2021, First Workshop, LL-COVID19 2021, and First Workshop and Tutorial, PPML 2021, Held in Conjunction with MICCAI 2021, Strasbourg, France, September 27 and October 1, 2021, Proceedings 2*, pp. 174–187. Springer, 2021.

Pereyra, G., Tucker, G., Chorowski, J., Kaiser, L., and Hinton, G. E. Regularizing neural networks by penalizing confident output distributions. In *International Conference on Learning Representations*, 2017.

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

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115: 211–252, 2015.

Sablayrolles, A., Douze, M., Schmid, C., Ollivier, Y., and Jégou, H. White-box vs black-box: Bayes optimal strategies for membership inference. In *International Conference on Machine Learning*, pp. 5558–5567. PMLR, 2019.

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

Shejwalkar, V. and Houmansadr, A. Membership privacy for machine learning models through knowledge transfer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 9549–9557, 2021.

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

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

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

Tan, J., Mason, B., Javadi, H., and Baraniuk, R. Parameters or privacy: A provable tradeoff between overparameterization and membership inference. *Advances in Neural Information Processing Systems*, 35:17488–17500, 2022.

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

Texas Department of State Health Services. Texas hospital inpatient discharge public use data file, 2006. URL https://www.dshs.texas.gov/thcic/hospitals/Inpatientpudf.shtm.

Truex, S., Liu, L., Gursoy, M. E., Wei, W., and Yu, L. Effects of differential privacy and data skewness on membership inference vulnerability. In *2019 First IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA)*, pp. 82–91. IEEE, 2019.

van Breugel, B., Sun, H., Qian, Z., and van der Schaar, M. Membership inference attacks against synthetic data through overfitting detection. In *AISTATS*, volume 206 of *Proceedings of Machine Learning Research*, pp. 3493–3514. PMLR, 2023.

Wang, C., Liu, G., Huang, H., Feng, W., Peng, K., and Wang, L. Miasec: Enabling data indistinguishability against membership inference attacks in mlaas. *IEEE Transactions on Sustainable Computing*, 5(3):365–376, 2019.

Wang, X. and Wang, W. H. Link membership inference attacks against unsupervised graph representation learning. In *Proceedings of the 39th Annual Computer Security Applications Conference*, pp. 477–491, 2023.

Wang, Y., Wang, C., Wang, Z., Zhou, S., Liu, H., Bi, J., Ding, C., and Rajasekaran, S. Against membership inference attack: Pruning is all you need. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pp. 3141–3147. ijcai.org, 2021.

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

Weiss, K., Khoshgoftaar, T. M., and Wang, D. A survey of transfer learning. *Journal of Big data*, 3(1):1–40, 2016.

Xue, M., Yuan, C., He, C., Wu, Y., Wu, Z., Zhang, Y., Liu, Z., and Liu, W. Use the spear as a shield: An adversarial example based privacy-preserving technique against membership inference attacks. *IEEE Transactions on Emerging Topics in Computing*, 11(1):153–169, 2022.

Yao, Y., Rosasco, L., and Caponnetto, A. On early stopping in gradient descent learning. *Constructive Approximation*, 26:289–315, 2007.

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

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

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. Understanding deep learning requires rethinking generalization. In *5th International Conference on Learning Representations*, 2017.

Zhang, X., Xiong, H., and Wu, D. Rethink the connections among generalization, memorization, and the spectral bias of dnns. In Zhou, Z.-H. (ed.), *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pp. 3392–3398. International Joint Conferences on Artificial Intelligence Organization, 8 2021. doi: 10.24963/ijcai.2021/467. URL https://doi.org/10.24963/ijcai.2021/467. Main Track.

Zhang, Y., Zhao, L., and Wang, Q. Mida: Membership inference attacks against domain adaptation. *ISA transactions*, 2023.

Zhang, Z. and Sabuncu, M. Generalized cross entropy loss for training deep neural networks with noisy labels. *Advances in Neural Information Processing Systems*, 31, 2018.

Zheng, J., Cao, Y., and Wang, H. Resisting membership inference attacks through knowledge distillation. *Neurocomputing*, 452:114–126, 2021.

Zou, F., Shen, L., Jie, Z., Zhang, W., and Liu, W. A sufficient condition for convergences of adam and rmsprop. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11127–11135, 2019.

A. Proof of Theorem 3.1

Proof. Since ℓ is strictly convex, $\ell'' > 0$, so there must exits infimum $B = \inf \ell''(x) \geqslant 0$.

$$\begin{aligned}\mathbb{E}_{\mathcal{D}}\ell(p_y) &= \mathbb{E}_{\mathcal{D}}[-\ell'(1)(1-p_y) + \frac{1}{2}\ell''(\xi(p_y))(1-p_y)^2] \\ &\geqslant -\ell'(1)\mathbb{E}_{\mathcal{D}}[(1-p_y)] + \frac{B}{2}\mathbb{E}_{\mathcal{D}}(1-p_y)^2 \\ &= A\epsilon + \frac{B}{2}(\epsilon^2 + \sigma^2)\end{aligned}$$

where $A = -\ell'(1) > 0$, $B \geqslant 0$ is a non-negative lower bound of $\ell''(x)$.

which concludes the proof. □ □

B. Proof of Theorem 3.2

Proof. Since ℓ is concave, so $\ell'(1) \leqslant \ell'(x) < 0$. Note that $\ell(1) = 0$, then by Mean Value Theorem, we have $\ell(0) = \ell(0) - \ell(1) = -\ell'(\xi_1) \leqslant -\ell'(1)$.

Hence, ℓ is bound and continues in $[0, 1]$, so ℓ'' is also continues in $[0, 1]$.

Suppose that the probability density function of p_y is $f(x)$

$$\begin{aligned}\mathbb{E}_{\mathcal{D}}\ell(p_y) &= \mathbb{E}_{\mathcal{D}}[-\ell'(1)(1-p_y) + \frac{1}{2}\ell''(\xi(p_y))(1-p_y)^2] \\ &= A\epsilon + \frac{1}{2}\mathbb{E}_{\mathcal{D}}[\ell''(\xi(p_y))(1-p_y)^2] \\ &= A\epsilon + \frac{1}{2}\int_0^1 \ell''(\xi(x))f(x)(1-x)^2 dx \\ &= A\epsilon + \frac{\ell''(\xi)}{2}\int_0^1 f(x)(1-x)^2 dx \quad \text{(By First Mean Value Theorem for Integration)} \\ &= A\epsilon + B(\epsilon^2 + \sigma^2)\end{aligned}$$

which concludes the proof.

□

C. Connection between the loss variance and σ^2

Notably, the metric commonly employed for MIA is typically a function of the probability vector $\mathbf{P} = (p(1|\boldsymbol{x}), p(2|\boldsymbol{x}), \dots, p(K|\boldsymbol{x}))^\top$. An example of such a metric is the entropy, given by $-\sum_{k=1}^{K} p(k|\boldsymbol{x}) \log p(k|\boldsymbol{x})$, which constitutes a mapping from $\mathbb{R}^K$ to $\mathbb{R}$. Consequently, we demonstrate that augmenting the variance $\mathrm{Var}(f(\boldsymbol{P}))$ is equivalent to amplifying $\mathrm{Var}(p_y)$ while maintaining a constant mean.

Consider $\mathbf{P} = (P_1, P_2, \dots, P_K)$ as a K-dimensional random vector, where P_k represents the random variable for $p(k|\boldsymbol{x})$. Below we provide an approximate of $\mathrm{Var}(f(\boldsymbol{P}))$ by delta method (Oehlert, 1992).

Lemma C.1. *Let* $\mathbf{P}$ *be a K-dimensional random vector in* $\mathcal{P}$ *and* $\boldsymbol{\mu}$ *be the mean vector of* $\mathbf{P}$. *Suppose* $f : \mathcal{P} \to \mathbb{R}$ *is a differentiable function and* $\mathbf{J}(\boldsymbol{\mu})$ *is the Jacobian matrix of* f *evaluated at* $\boldsymbol{\mu}$, *then the variance of* $f(\mathbf{P})$ *can be approximated by*

$$\mathrm{Var}(f(\mathbf{P})) \approx \mathbf{J}(\boldsymbol{\mu})^\top\, \boldsymbol{\Sigma}\, \mathbf{J}(\boldsymbol{\mu}) \tag{19}$$

where $\boldsymbol{\Sigma} = \mathrm{Cov}(\mathbf{P})$ *is covariance matrix of* $\mathbf{P}$ *and the elements of* $\mathbf{J}$ *are given by* $J_i = \frac{\partial f}{\partial P_i}(\boldsymbol{\mu})$ *for* $i = 1, \dots, K$.

Based on Lemma C.1, we can establish a relationship in variance between the confidence of true label p_y and metrics used for MIA with respect to prediction probability. Assuming that the output of a neural network after a softmax layer follows

Dirichlet distribution (Malinin & Gales, 2018), then we explore the connection between the loss variance and σ^2, as shown below.

Proposition C.2. *Let* $\mathbf{P}$ *follows Dirichlet distribution in the simplex* Δ^{K-1}, *where* $\Delta^{K-1} = \left\{(x_1, x_2, \ldots, x_K) \in \mathbb{R}^K \mid x_i \geq 0, \sum_{i=1}^K x_i = 1\right\}$. *With a fixed* $\mathbb{E}\mathbf{P} = \boldsymbol{\mu}$, *for any* $i \in \mathcal{Y}$, *an increase in* $\mathrm{Var}(P_i)$ *implies a corresponding increase in* $\mathrm{Var}(f(\mathbf{P}))$

Proof. Suppose $\mathbf{P_1} \sim \mathrm{Dirichlet}(\alpha_1, \alpha_2, \ldots, \alpha_K)$ and $\mathbf{P_2} \sim \mathrm{Dirichlet}(\beta_1, \beta_2, \ldots, \beta_K)$ such that $\mathbb{E}\mathbf{P_1} = \mathbb{E}\mathbf{P_2} = \boldsymbol{\mu}$ and $\mathrm{Var}(P_{2,t}) > \mathrm{Var}(P_{1,t})$, where $P_{i,j}$ is the j-th element of $\mathbf{P_i}$ and $t \in \mathcal{Y}$.

By $\mathbb{E}\mathbf{P_1} = \mathbb{E}\mathbf{P_2}$, we have

$$\mathbb{E}[P_{1,j}] = \frac{\alpha_j}{\alpha_0} = \frac{\beta_j}{\beta_0} = \mathbb{E}[P_{2,j}], \quad \text{for any } j = 1, 2, \ldots, K$$

where $\alpha_0 = \sum_{j=1}^K \alpha_j$ and $\beta_0 = \sum_{j=1}^K \beta_j$.

By the condition $\mathrm{Var}(P_{2,t}) > \mathrm{Var}(P_{1,t})$, we have

$$\mathrm{Var}(P_{2,t}) > \mathrm{Var}(P_{1,t}) \Longrightarrow \frac{(\tilde{\alpha}_t)(1-\tilde{\alpha}_t)}{\alpha_0 + 1} > \frac{(\tilde{\beta}_t)(1-\tilde{\beta}_t)}{\beta_0 + 1} \Longrightarrow \beta_0 > \alpha_0$$

where $\tilde{\alpha}_t = \frac{\alpha_t}{\alpha_0} = \frac{\beta_t}{\beta_0} = \tilde{\beta}_t$.

It follows that

$$\mathbf{\Sigma}_{ij}^{(1)} = \mathrm{Cov}(P_{1,i}, P_{1,j}) = \frac{-\tilde{\alpha}_i\tilde{\alpha}_j}{\alpha_0 + 1} > \frac{-\tilde{\beta}_i\tilde{\beta}_j}{\beta_0 + 1} = \mathrm{Cov}(P_{2,i}, P_{2,j}) = \mathbf{\Sigma}_{ij}^{(2)}, \quad \text{for any } i \neq j$$

$$\mathrm{Var}(P_{2,j}) > \mathrm{Var}(P_{1,j}), \quad \text{for any } j = 1, 2, \ldots, K$$

This implies

$$\begin{aligned}\mathbf{J}(\boldsymbol{\mu})^\top \mathbf{\Sigma}^{(1)} \mathbf{J}(\boldsymbol{\mu}) &= \mathbf{J}(\boldsymbol{\mu})^\top [\frac{\beta_0 + 1}{\alpha_0 + 1}\mathbf{\Sigma}^{(2)}] \mathbf{J}(\boldsymbol{\mu}) \\ &> \mathbf{J}(\boldsymbol{\mu})^\top \mathbf{\Sigma}^{(2)} \mathbf{J}(\boldsymbol{\mu})\end{aligned}$$

By Lemma C.1, we have $\mathrm{Var}(f(\mathbf{P})) \approx \mathbf{J}(\boldsymbol{\mu})^\top \mathbf{\Sigma} \mathbf{J}(\boldsymbol{\mu})$, so we can conclude the proof. □

□

According to Proposition C.2, any metric related to $f(\mathbf{P})$—including entropy and modified-entropy (Song & Mittal, 2021)—will exhibit increased variance in response to a rise in the variance of p_y. Building upon this insight and in conjunction with 6.1, it can be deduced that amplifying the variance of p_y during the training phase can enhance the model's resilience against metric-based MIA.

D. Proof of Lemma 6.2 and Lemma 6.3

D.1. Proof of Lemma 6.2

Proof. Given that $A(x)$ and $B(x)$ are Lipschitz continuous, for all $x, y \in D$, we have

$$\|A(x) - A(y)\| \leq L_A \|x - y\| \|B(x) - B(y)\| \qquad \leq L_B \|x - y\| \tag{20}$$

Consider the product $C(x) = A(x)B(x)$. For any $x, y \in D$,

$$\begin{aligned}
\|C(x) - C(y)\| &= \|A(x)B(x) - A(y)B(y)\| \\
&= \|A(x)B(x) - A(x)B(y) + A(x)B(y) - A(y)B(y)\| \\
&= \|A(x)(B(x) - B(y)) + (A(x) - A(y))B(y)\| \\
&\leqslant \|A(x)\| \cdot \|B(x) - B(y)\| + \|A(x) - A(y)\| \cdot \|B(y)\| \\
&\leqslant M_A L_B \|x - y\| + L_A M_B \|x - y\| \\
&= (M_A L_B + L_A M_B)\|x - y\|
\end{aligned}$$

Therefore, proving that $C(x) = A(x)B(x)$ is Lipschitz continuous with a Lipschitz constant $L_C = M_A L_B + L_A M_B$, under the given conditions of boundedness and Lipschitz continuity of $A(x)$ and $B(x)$. □

D.2. Proof of Lemma 6.3

Since f and g are Lipschitz continuous, we have for all $x, y \in \mathbb{R}^n$,

$$\|f(x) - f(y)\| \leq L_f \|x - y\|,$$

and for all $u, v \in \mathbb{R}^m$,

$$\|g(u) - g(v)\| \leq L_g \|u - v\|.$$

Consider two points $x, y \in \mathbb{R}^n$. We want to show that h is Lipschitz continuous, i.e., there exists a constant L_h such that

$$\|h(x) - h(y)\| = \|g(f(x)) - g(f(y))\| \leq L_h \|x - y\|.$$

Using the Lipschitz continuity of g and then f, we have

$$\|g(f(x)) - g(f(y))\| \leq L_g \|f(x) - f(y)\| \leq L_g L_f \|x - y\|.$$

Setting $L_h = L_g L_f$, we see that

$$\|h(x) - h(y)\| \leq L_h \|x - y\|,$$

proving that the composition $h = g \circ f$ is Lipschitz continuous with Lipschitz constant $L_h = L_g L_f$.

E. Can CCL improve other convex functions?

Our method defined in Equation 4 just integrates CE as the convex loss function. There we show that our method can also improve other convex loss functions such as Focal Loss (Lin et al., 2017).

$$\ell = \alpha \ell_{\mathrm{FL}} + (1 - \alpha)\tilde{\ell}$$

where ℓ_{FL} is Focal Loss and $\tilde{\ell} \in \mathcal{F}$.

In the experiments, we use Resnet-34 trained on CIFAR-10. For focal loss, we fixed $\gamma = 2$. In particular, we use CCLE and select the best α in $\{0.2, 0.4, 0.6, 0.8\}$ with restricted condition that performs better utility. Our results in Figure 6 show that focal loss equipped with a concave term helps defend MIA across eight attack methods.

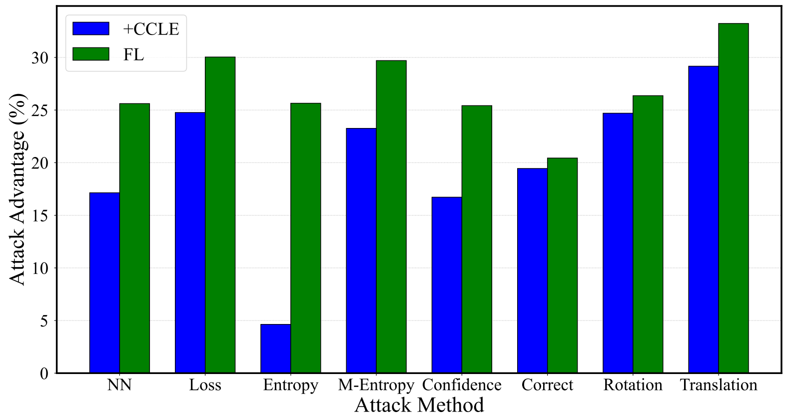


Figure 6. Attack advantage comparison among focal loss (FL) and focal loss equipped with CLE (+CCLE) on CIFAR-10.

F. Defense Methods with Hyperparameter

F.1. Other Defense Methods

RelaxLoss. RelexLoss (Chen et al., 2022) reduces the gap between the member and non-member loss distribution by applying gradient ascent as long as the average loss of the current batch is smaller than α. We vary the α over {0.01, 0.04, 0.1, 0.2, 0.4, 0.8, 1.6, 3.2 }.

Mixup+MMD. Mixup+MMD integrates the Maximum Mean Discrepancy (MMD) approach with mix-up training techniques. Specifically, MMD serves as a metric for quantifying the divergence between two empirical data distributions, functioning as follows

$$\text{Distance}(X, Y) = \left\| \frac{1}{n} \sum_{i=1}^{n} \phi(x_i) - \frac{1}{m} \sum_{j=1}^{m} \phi(y_j) \right\|_H$$

where $\phi(\dot{})$ is Gaussian kernel function. MMD regularization loss is calculated by a batch of training and validation instances. We vary the weight of the MMD term across {0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1, 2, 4, 8}.

Adversary Regularization. Adversary regularization (Nasr et al., 2018) conducts a min-max game optimization and an adversarial training algorithm that minimizes classification loss while also reducing the maximum gain of potential membership inference attacks. We vary the weight of the adversarial loss over {0.8,1.0,1.2,1.4,1.6,1.8}.

Dropout. Dropout randomly deactivates a subset of neurons in a layer with a given probability during training. In our experiments, dropout is applied specifically to the last fully connected layer of each target model. We vary the probability of dropout over {0.01, 0.02, 0.05, 0.1, 0.3, 0.5, 0.7, 0.9}.

Label Smoothing. Label smoothing (Guo et al., 2017) modifies the target labels, making them slightly less confident by replacing the hard 0 and 1 targets with values slightly closer to a uniform distribution. We vary the smoothing parameter over {0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9 }.

Confidence Penalty. Confidence penalty (Pereyra et al., 2017) mitigates overfitting by penalizing low entropy in the output distributions of neural networks. In particular, it is implemented by adding an entropy regularization term to the objective. We vary the weight of the regularization term over {0.1, 0.3, 0. 5, 1, 2, 4, 8}.

Early Stopping. Early Stopping (Yao et al., 2007) monitors the model's performance on a validation set and stops the training process when performance begins to degrade. Following the implementation of Chen et al. (2022), our experiments save checkpoints at specific epochs: {25, 50, 75, 100, 125, 150, 175, 200, 225, 250, 275}