AdAUC: End-to-end Adversarial AUC Optimization Against Long-tail Problems

Wenzheng Hou [1 2] **Qianqian Xu** [1] **Zhiyong Yang** [2] **Shilong Bao** [3 4] **Yuan He** [5] **Qingming Huang** [1 2 6 7]

Abstract

It is well-known that deep learning models are vulnerable to adversarial examples. Existing studies of adversarial training have made great progress against this challenge. As a typical trait, they often assume that the class distribution is overall balanced. However, long-tail datasets are ubiquitous in a wide spectrum of applications, where the amount of head class instances is larger than the tail classes. Under such a scenario, AUC is a much more reasonable metric than accuracy since it is insensitive toward class distribution. Motivated by this, we present an early trial to explore adversarial training methods to optimize AUC. The main challenge lies in that the positive and negative examples are tightly coupled in the objective function. As a direct result, one cannot generate adversarial examples without a full scan of the dataset. To address this issue, based on a concavity regularization scheme, we reformulate the AUC optimization problem as a saddle point problem, where the objective becomes an instance-wise function. This leads to an end-to-end training protocol. Furthermore, we provide a convergence guarantee of the proposed algorithm. Our analysis differs from the existing studies since the algorithm is asked to generate adversarial examples by calculating the gradient of a min-max problem. Finally, the extensive experimental results show the performance and robustness of our algorithm in three long-tail datasets.

[1]Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, CAS, Beijing, China. [2]School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing, China. [3]State Key Laboratory of Information Security, Institute of Information Engineering, CAS, Beijing, China. [4]School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China. [5]Alibaba Group, Beijing, China [6]Key Laboratory of Big Data Mining and Knowledge Management, Chinese Academy of Sciences, Beijing, China. [7]Artificial Intelligence Research Center, Peng Cheng Laboratory, Shenzhen, China. Correspondence to: Qianqian Xu <xuqianqian@ict.ac.cn>, Qingming Huang <qmhuang@ucas.ac.cn>.

Proceedings of the 39th International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

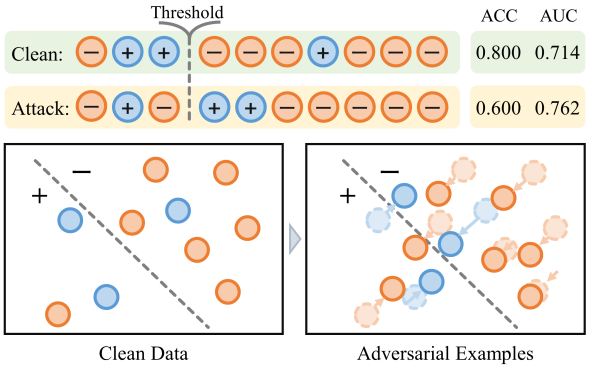


Figure 1. Diagram of ACC and AUC change when the model is attacked. The **upper** rectangular boxes represent the score rank before and after the attack occurs; The **lower** plots represent the change of score in the embedding space when the model is attacked.

1. Introduction

Deep learning has recently achieved significant progress on various machine learning tasks, such as computer vision (Voulodimos et al., 2018) and natural language processing (Strubell et al., 2019; Sorin et al., 2020). However, recent work shows that deep learning models are vulnerable to adversarial attack (Szegedy et al., 2014; Biggio et al., 2013). For example, images with human imperceptible perturbations (i.e., adversarial examples) can easily fool even the well-trained models. The existence of adversarial examples has raised big security threats to deep neural networks, which impels extensive efforts to improve the adversarial robustness (Madry et al., 2018; Zhang et al., 2019b). For example, one can resist the adversarial examples by means of adversarial training (AT). Specifically, AT could be formulated as a min-max problem, where the inner maximization problem is employed to generate adversarial examples, and the outer minimization problem is used to learn the model under the adversarial noise. In this way, AT could be easily applied to most of the modern architectures in deep learning, making it one of the most effective measures against adversarial attack.

The prior art of adversarial training methods focuses on balanced benchmark datasets. On top of this, the learning objective is to increase the overall accuracy. However, real-

world datasets usually exhibit a long-tail distribution that the proportion of the majority classes examples significantly dominates the others. For such long-tail problems, accuracy (ACC) is considered to be a less appropriate performance metric than another metric named AUC (Area Under the ROC Curve). Specifically, AUC is the probability of observing a positive instance with a higher score than a negative one. It is well-known to be insensitive to class distributions and costs (Fawcett, 2006; Hand & Till, 2001). Comparing AUC with ACC, we then ask:

Can we improve the adversarial robustness of AUC by employing the traditional ACC-based methods?

Unfortunately, the answer might be negative. As shown in Fig.1, the adversarial examples generated by minimizing ACC (ACC drops from 0.8 to 0.6), may fail to attack AUC (AUC here increases from 0.714 to 0.762). In this sense, the model trained on such adversarial examples cannot improve the adversarial robustness of AUC. Therefore, more attention should be paid to AUC when studying the adversarial robustness against long-tail problems.

Inspired by this fact, we present a very early trial to study adversarial training in AUC optimization with an end-to-end framework.

Existing AT methods can be easily implemented in an end-to-end manner since the inner maximization problem for generating the adversarial examples can be solved instance-wisely. However, this is not the case for AUC optimization. Specifically, in the expression of AUC, every positive instance is coupled with all the negative instances and vice versa. For a binary class classification problem, this means that we need to spend $O(n^- \cdot T)$ time for generating the adversarial example of a positive instance and $O(n^+ \cdot T)$ for a negative one, where n^+, n^- are the number of positive and negative instances, and T is complexity for calculating the gradient for a single positive-negative instance pair. In this sense, we can hardly implement such a naive training method on top of even the simplest deep learning framework.

To solve the challenge, this paper proposes an end-to-end adversarial AUC optimization framework with a convergence guarantee. Specifically, our contribution is as follows:

First, based on a reformulation technique and a concavity regularizer, we show that the original problem is equivalent to a min-max problem where the objective function can be expressed in an instance-wise manner.

Second, we propose an AT algorithm to optimize the min-max problem, where we alternately invoke a projected-gradient-descent-like protocol to generate the adversarial examples, and a stochastic gradient descent-ascent protocol to train the model parameters. Meanwhile, we also present a convergence analysis to show the correctness of our algorithm. The proof here is non-trivial since we have to simultaneously estimate the gradient of the min player and the max player.

Finally, we conduct a series of empirical analyses of our proposed algorithm on long-tail datasets. The results demonstrate the effectiveness of our proposed method.

2. Related Work

2.1. AUC Optimization

As a motivating study, (Cortes & Mohri, 2003) investigates the inconsistency between AUC maximization and error rate minimization, which shows the necessity to study direct AUC optimization methods. After that, a series of algorithms are designed for off-line AUC optimization (Herschtal & Raskutti, 2004; Joachims, 2005). To extend the scalability of AUC optimization, researchers start to explore the online and stochastic optimization extensions of the AUC maximization problem. (Zhao et al., 2011) makes the first attempt for this direction based on the reservoir sampling technique. (Gao et al., 2013) proposes a one-pass AUC optimization algorithm based on the squared surrogate loss. After that, (Ying et al., 2016) reformulates the minimization problem of the pairwise square loss into an equivalent stochastic saddle point problem, where the objective function could be expressed in an instance-wise manner. On top of the reformulation framework, (Natole et al., 2018) proposes an accelerated version with a faster convergence rate and (Liu et al., 2019) explores its extension in deep neural networks. Meanwhile, many researchers provide theoretical guarantees for AUC optimization algorithms from different aspects, such as generalization analysis (Agarwal et al., 2005; Clémençon et al., 2008; Usunier et al., 2005) and consistency analysis (Agarwal, 2014; Gao & Zhou, 2015). Beyond the optimization algorithms and theoretical supports for AUC, in practice, AUC optimization demonstrates its effectiveness in various class-imbalanced tasks, such as disease prediction (Westcott et al., 2019; Gola et al., 2020; Ren et al., 2018), rare event detection (Feizi, 2020; Robles et al., 2020) and etc.

Compared with the existing study, we present a very early trial for the adversarial training problem.

2.2. Adversarial Training

For a long time, machine learning models have proved vulnerable to adversarial examples (Biggio et al., 2013; Szegedy et al., 2014; Goodfellow et al., 2014). Numerous defenses have been proposed to address the security concern raised by the issue (Athalye & Carlini, 2018; Athalye et al., 2018). Among such studies, adversarial training is one of the most popular methods (Kurakin et al., 2017; Madry

et al., 2018; Zhang et al., 2019b). The majority of studies in this direction follows the min-max formulation proposed in (Madry et al., 2018), which so far has been improved in various way (Shafahi et al., 2020; Cai et al., 2018; Tramèr et al., 2017; Pang et al., 2019; Wang et al., 2019b; Zhang et al., 2020; Maini et al., 2020; Tramèr & Boneh, 2019). Furthermore, due to the heavy computational burden of AT, accelerating the training procedure of AT becomes increasingly urgent. Recently, there has been a new wave to explore the acceleration of AT, which includes reusing the computations (Shafahi et al., 2019; Zhang et al., 2019a), adaptive adversarial steps (Wang et al., 2019a) and one-step training (Wong et al., 2019). Besides the practical improvements, there are also some recent advances in theoretical investigations from the perspective of optimization (Wang et al., 2019a; Bai et al., 2022), generalization (Xing et al., 2021; Tu et al., 2019), and consistency (Bao et al., 2020).

In this paper, we will present an AT algorithm on top of the AUC optimization. As shown in the introduction, the complicated expression of AUC brings new elements into our model formulation and theoretical analysis.

3. Preliminaries

In this section, we briefly introduce the AUC optimization problem and the adversarial training framework.

3.1. AUC Optimization Problem

Let X be the feature set. Based on (Hanley & McNeil, 1982), AUC of a scoring function $h_\theta : X \to [0, 1]$ is equivalent to the probability that a positive instance is predicted with a higher score compared to a negative instance:

$$\mathsf{AUC}(h_\theta) = \Pr\left(h_\theta(\boldsymbol{x}^+) \geq h_\theta(\boldsymbol{x}^-) | y^+ = 1, y^- = 0\right),$$

where $(\boldsymbol{x}^+, y^+)$ and $(\boldsymbol{x}^-, y^-)$ represent positive and negative examples, respectively, and θ is the model parameters. By employing a differentiable loss ℓ as the surrogate loss, the unbiased estimation of $\mathsf{AUC}(h_\theta)$ could be expressed as:

$$\hat{\mathsf{AUC}}(h_\theta) = 1 - \sum_{i=1}^{n^+}\sum_{j=1}^{n^-} \frac{\ell\left(h_\theta(\boldsymbol{x}^+) - h_\theta(\boldsymbol{x}^-)\right)}{n^+ n^-}.$$

where n^+ and n^- denote the number of positive and negative examples, respectively. Then AUC maximization problem is equivalent to the following minimization problem:

$$\textbf{(OP0)} \quad \min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{x}, y) := \sum_{i=1}^{n^+}\sum_{j=1}^{n^-} \frac{\ell\left(h_\theta(\boldsymbol{x}^+) - h_\theta(\boldsymbol{x}^-)\right)}{n^+ n^-}.$$

3.2. Adversarial Training Framework

Adversarial training is one of the most effective defensive strategies against adversarial examples (Goodfellow et al., 2015; Madry et al., 2018), the key idea of which is to directly optimize the model performance based on the perturbed examples. Generally speaking, the adversarial training framework can be formalized as

$$\min_{\boldsymbol{\theta}} \frac{1}{n}\sum_{i=1}^{n} \max_{\|\boldsymbol{\delta}_i\|_\infty \leq \epsilon} \ell(h_\theta(\boldsymbol{x}_i^0 + \boldsymbol{\delta}_i), y_i). \tag{1}$$

Here $\boldsymbol{\delta}_i$ is the perturbation on clean feature vector $\boldsymbol{x}_i^0$, and $\boldsymbol{x}_i = \boldsymbol{x}_i^0 + \boldsymbol{\delta}_i$ is the resulting adversarial example for the instance $(\boldsymbol{x}_i^0, y_i), i = 1, 2, \cdots, n$. The inner maximization problem generates such adversarial examples by trying to hurt the model performance (by maximizing the loss $\ell(h_\theta(\boldsymbol{x}_i^0 + \boldsymbol{\delta}_i), y)$). The constraint $\|\boldsymbol{\delta}_i\|_\infty \leq \epsilon$ makes sure that the adversarial perturbation is small enough to be imperceptible. In this sense, the adversarial example lives in $\mathcal{X}_i = \left\{\boldsymbol{x} | \left\|\boldsymbol{x} - \boldsymbol{x}_i^0\right\|_\infty \leq \epsilon\right\}$. Finally, the outer minimization problem is to find a robust model that can resist the adversarial perturbation.

For the inner maximization problem, K-PGD (Madry et al., 2018) is a widely used attack method that perturbs the clean examples $\boldsymbol{x}^0$ iteratively with a total of K steps. At the end of each iteration, the example will be projected to the ϵ-ball of $\boldsymbol{x}^0$. Specifically, the adversarial examples generated in $k+1$ step are as follows:

$$\boldsymbol{x}^{k+1} = \mathrm{Proj}\left\{\boldsymbol{x}^k + \beta \cdot \mathrm{sign}(\nabla_{\boldsymbol{x}} \ell(h_\theta(\boldsymbol{x}^k), y^k))\right\}, \tag{2}$$

where Proj is the projection function, and β is the step size. For the outer minimization problem, gradient descent is

Table 1. Notations and their description

Notations	Description
n	Number of total examples
n^+, n^-	Number of positive (negative) examples
p	Proportion of positive examples
$\boldsymbol{x}^0$	Clean Examples
$\boldsymbol{x}^k$	Adversarial examples generated in step k
y	The label of example
$\boldsymbol{\delta}$	Perturbation on samples
$\mathcal{X}_i$	$\mathcal{X}_i = \{\boldsymbol{x} \| \left\|\boldsymbol{x} - \boldsymbol{x}_i^0\right\|_\infty \leq \epsilon\}$
$\boldsymbol{\theta}$	Parameters of model
a, b, α	Learnable parameters of loss function
$\boldsymbol{w}$	$\boldsymbol{w} = (\boldsymbol{\theta}, a, b)$
$f(\boldsymbol{w}, \alpha, \boldsymbol{x})$	The surrogate objective function
$L(\boldsymbol{w}, \alpha)$	$\frac{1}{n}\sum_{i=1}^n f(\boldsymbol{w}, \alpha, \boldsymbol{x}_i^*)$
$\Phi(\boldsymbol{w})$	$\Phi(\boldsymbol{w}) = \max_\alpha L(\boldsymbol{w}, \alpha)$
$\mathcal{B}$	The mini-batch
M	The batch size
L	$\max\{L_{ww}, L_{wx}, L_{\alpha\alpha}, L_{\alpha x}, L_{xw}, L_{x\alpha}\}$
T	The total of training epochs
$\hat{g}(\alpha), \hat{g}(\boldsymbol{w})$	Stochastic gradient
$g(\alpha), g(\boldsymbol{w})$	Stochastic gradient

usually used to solve it. A more detailed introduction of adversarial attack methods is shown in the Appendix B.

4. Methodology

Before entering into the methodology, we summarize some useful notations in Tab.3.1 to make our argument easier to follow.

4.1. Reformulation of Optimization Problem

A naive idea to perform AUC adversarial training is to directly combine **(OP0)** with the standard AT framework (Madry et al., 2018), resulting in the following problem:

$$\min_{\boldsymbol{\theta}} \max_{\boldsymbol{\delta}_1,\boldsymbol{\delta}_2,\cdots,\boldsymbol{\delta}_n} \sum_{i=1}^{n^+}\sum_{j=1}^{n^-} \frac{\ell(h_\theta(\boldsymbol{x}_i^+ + \boldsymbol{\delta}_i) - h_\theta(\boldsymbol{x}_j^- + \boldsymbol{\delta}_j))}{n^+n^-},$$

According to the definition of AUC optimization objective function $\mathcal{L}$ in **(OP0)** , we know that each pair of positive examples is inter-dependent with all negative examples, and vice versa for the negative examples. Thus, the inner maximization problem for $\boldsymbol{\delta}_1, \boldsymbol{\delta}_2, \cdots, \boldsymbol{\delta}_n$ cannot be decoupled into a series of instance-wise maximization problems. In other words, the following inequality holds in general:

$$\min_{\boldsymbol{\theta}} \max_{\boldsymbol{\delta}_1,\boldsymbol{\delta}_2,\cdots,\boldsymbol{\delta}_n} \sum_{i=1}^{n^+}\sum_{j=1}^{n^-} \frac{\ell(h_\theta(\boldsymbol{x}_i^+ + \boldsymbol{\delta}_i) - h_\theta(\boldsymbol{x}_j^- + \boldsymbol{\delta}_j))}{n^+n^-} \neq \min_{\boldsymbol{\theta}} \sum_{i=1}^{n^+}\sum_{j=1}^{n^-} \max_{\boldsymbol{\delta}_i,\boldsymbol{\delta}_j} \frac{\ell(h_\theta(\boldsymbol{x}_i^+ + \boldsymbol{\delta}_i) - h_\theta(\boldsymbol{x}_j^- + \boldsymbol{\delta}_j))}{n^+n^-}.$$

In this sense, the generation of adversarial examples cannot be carried out in a mini-batch fashion. Instead, one update $\boldsymbol{\delta}$ requires a full scan of $O(n^+n^-)$. This brings a heavy computational burden towards its application. Therefore, we need to reformulate the optimization problem.

Fortunately, if we adopt the square loss $\ell(t) = (1-t)^2$ as the surrogate loss function, then (Ying et al., 2016; Liu et al., 2019) proved that **(OP0)** could be converted in a min-max problem, as shown in the following proposition:

Proposition 1. *The empirical risk of AUC in **(OP0)** is equivalent to*

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{x}, y) = \min_{a,b} \max_{\alpha} \frac{1}{n}\sum_{i=1}^{n} g(\boldsymbol{\theta}, a, b, \alpha, (\boldsymbol{x}_i, y_i)), \quad (3)$$

where

$$\begin{aligned} & g(\boldsymbol{\theta}, a, b, \alpha, (\boldsymbol{x}_i, y_i)) \\ =& (1-p)(h_\theta(\boldsymbol{x}_i) - a)^2 \mathbb{I}_{[y_i=1]} + p(h_\theta(\boldsymbol{x}_i) - b)^2 \mathbb{I}_{[y_i=0]} \\ & + 2(1+\alpha)\left(p h_\theta(\boldsymbol{x}_i)\mathbb{I}_{[y_i=0]} - (1-p)h_\theta(\boldsymbol{x}_i)\mathbb{I}_{[y_i=1]}\right) \\ & - p(1-p)\alpha^2. \end{aligned} \quad (4)$$

where $a, b, \alpha \in \mathbb{R}$ are learnable parameters, and $p = \Pr(y=1)$.

Remark 1. *According to (Ying et al., 2016), a, b, α has the following closed-form solution: $a = \hat{\mathbb{E}}[h_\theta(\boldsymbol{x})|y=1]$, $b = \hat{\mathbb{E}}[h_\theta(\boldsymbol{x})|y=0]$ and $\alpha = \hat{\mathbb{E}}[h_\theta(\boldsymbol{x})|y=0] - \hat{\mathbb{E}}[h_\theta(\boldsymbol{x})|y=1]$, where $\hat{\mathbb{E}}$ is a shorthand for sample mean. If the score h_θ is normalized to the set $[0,1]$, we can restrict a, b and α to the following bounded domains:*

$$\Omega_{a,b} = \{a, b \in \mathbb{R} | 0 \le a, b \le 1\}, \Omega_\alpha = \{\alpha \in \mathbb{R} | |\alpha| \le 1\}.$$

And we can easily verify that g is μ-strongly concave w.r.t. α in Ω_α, i.e., for any $\alpha_1, \alpha_2 \in \Omega_\alpha$, it holds that

$$\begin{aligned} g(\boldsymbol{\theta}, a, b, \alpha_1, (\boldsymbol{x}, y)) \le & \, g(\boldsymbol{\theta}, a, b, \alpha_2, (\boldsymbol{x}, y)) + \\ & \langle \nabla_\alpha g(\boldsymbol{\theta}, a, b, \alpha_2, (\boldsymbol{x}, y)), \alpha_1 - \alpha_2 \rangle - \frac{\mu}{2}\|\alpha_1 - \alpha_2\|_2^2. \end{aligned}$$

And we can also verified that g is locally strongly convex in $\Omega_{a,b}$ w.r.t. a and b.

Hence, if we in turn construct an AT problem based on Prop.1, we can obtain the following optimization problem with ease:

$$\min_{\boldsymbol{\theta}} \max_{\boldsymbol{\delta}} \min_{a,b \in \Omega_{a,b}} \max_{\alpha \in \Omega_\alpha} \frac{1}{n}\sum_{i=1}^{n} g(\boldsymbol{\theta}, a, b, \alpha, (\boldsymbol{x}_i + \boldsymbol{\delta}_i, y_i)).$$

The good news here is that in the new loss function g, positive samples and negative samples are independent of each other. However, the bad news is that the min-max-min-max problem is still hardly tractable. Through a careful investigation, if we can swap the order of $\min_{a,b}$ and $\max_{\boldsymbol{\delta}}$, we can then obtain a min-max problem which can be solved in an end-to-end fashion. To realize the idea, we could resort to the von Neumann's Minimax theorem (Neumann, 1928; Sion, 1958):

Theorem 1. *Let $X \subset \mathbb{R}^n$ and $Y \subset \mathbb{R}^m$ be compact convex sets. If $f : X \times Y \to \mathbb{R}$ is a continuous function that is concave-convex, i.e.*

$$f(\cdot, \boldsymbol{y}) : X \to \mathbb{R} \text{ is concave for fixed } \boldsymbol{y},$$
$$f(\boldsymbol{x}, \cdot) : Y \to \mathbb{R} \text{ is convex for fixed } \boldsymbol{x}.$$

Then we have that $\max_{\boldsymbol{x} \in X} \min_{\boldsymbol{y} \in Y} f(\boldsymbol{x}, \boldsymbol{y}) = \min_{\boldsymbol{y} \in Y} \max_{\boldsymbol{x} \in X} f(\boldsymbol{x}, \boldsymbol{y})$.

Moreover, we resort to the definition of weak-concavity (Böhm & Wright, 2021; Liu et al., 2021):

Definition 1. *$f(\boldsymbol{x}) : \mathbb{R}^d \to \mathbb{R}$ is said to be a γ-weakly concave ($\gamma > 0$) function w.r.t. $\boldsymbol{x}$, if*

$$f(\boldsymbol{x}) - \frac{\gamma}{2}||\boldsymbol{x}||_2^2$$

is a concave function w.r.t. $\boldsymbol{x}$.

In the following proposition, we find a surrogate objective function $f(\boldsymbol{w}, \alpha, \boldsymbol{x}_i + \boldsymbol{\delta}_i)$ such that the resulting optimization problem could be reformulated as a min-max problem:

Proposition 2. *Define:*

$$r(a, b, \boldsymbol{x}) = \max_{\alpha} \frac{1}{n} \sum_{i=1}^{n} g(\boldsymbol{\theta}, a, b, \alpha, (\boldsymbol{x}_i + \boldsymbol{\delta}_i, y_i)),$$
$$f(\boldsymbol{w}, \alpha, \boldsymbol{x}_i + \boldsymbol{\delta}_i) = g(\boldsymbol{\theta}, a, b, \alpha, (\boldsymbol{x}_i + \boldsymbol{\delta}_i, y_i)) - \gamma \|\boldsymbol{x}_i + \boldsymbol{\delta}_i\|_2^2$$

If $r(a, b, \boldsymbol{x})$ is γ_*-weakly concave w.r.t. $\boldsymbol{\delta}_1, \boldsymbol{\delta}_2, \cdots, \boldsymbol{\delta}_n$, then for all $\gamma > \gamma_\star$, we have the following problem:

$$\min_{\boldsymbol{\theta}} \max_{\boldsymbol{\delta}} \min_{a,b \in \Omega_{a,b}} \max_{\alpha \in \Omega_\alpha} \frac{1}{n} \sum_{i=1}^{n} f(\boldsymbol{w}, \alpha, (\boldsymbol{x}_i + \boldsymbol{\delta}_i, y_i))$$

is equivalent to:

$$\textit{(OP)} \quad \min_{\boldsymbol{w}} \max_{\alpha} \max_{\boldsymbol{\delta}} \frac{1}{n} \sum_{i=1}^{n} [f(\boldsymbol{w}, \alpha, \boldsymbol{x}_i + \boldsymbol{\delta}_i)]$$
$$= \min_{\boldsymbol{w}} \max_{\alpha} \frac{1}{n} \sum_{i=1}^{n} \max_{\boldsymbol{\delta}_i} [f(\boldsymbol{w}, \alpha, \boldsymbol{x}_i + \boldsymbol{\delta}_i)],$$

where $\boldsymbol{w} = (\boldsymbol{\theta}, a, b)$. Moreover, $f(\boldsymbol{w}, \alpha, \boldsymbol{x}_i + \boldsymbol{\delta}_i)$ is strongly concave w.r.t. $\boldsymbol{\delta}_i$.

Remark 2. *According to (Böhm & Wright, 2021; Liu et al., 2021), if $\max_\alpha \frac{1}{n} \sum_{i=1}^{n} [f(\boldsymbol{w}, \alpha, \boldsymbol{x}_i + \boldsymbol{\delta}_i)]$ is smooth and L gradient Lipschitz, then it is also L-weakly concave. Hence, the weakly concavity assumption is much weaker than the strongly-concave assumption (Wang et al., 2019a).*

In this sense, we could turn to optimize **(OP)** in the next subsection.

4.2. Training Strategy

In this subsection, we continue to propose an adversarial AUC optimization framework to solve **(OP)**. Specifically, we design solutions for the *inner maximization problem* and the *outer min-max problem*, respectively.

Inner Maximization Problem: Adversarial Attack. In this paper, we choose K-PGD (Madry et al., 2018) to generate adversarial examples. To better control the quality of adversarial examples, we introduce the First-Order Stationary Condition (FOSC) (Wang et al., 2019a) about the inner maximization problem, which is as follows:

$$c(\boldsymbol{x}^k) = \max_{\boldsymbol{x} \in \mathcal{X}} \langle \boldsymbol{x} - \boldsymbol{x}^k, \nabla_{\boldsymbol{x}} f(\boldsymbol{w}, \alpha, \boldsymbol{x}^k) \rangle \tag{5}$$

Algorithm 1 Adversarial Training for AUC Optimization

Input: Neural network h_θ; initial parameters $\boldsymbol{w}_0 = \{\boldsymbol{\theta}^0, a^0, b^0\}$ and α^0; step size η_w, η_α; mini-batch $\mathcal{B}$ and its size M; max FOSC value c_{max}; training epochs T; control epoch T'; PGD step K; PGD step size β; maximum perturbation boundary ϵ.
for $t = 0$ **to** T **do**
  $c_t = \max(0, c_{max} - t \cdot c_{max}/T')$
  for Each batch $\boldsymbol{x}_{\mathcal{B}}^0$ **do**
    $M_c = \mathbb{1}_{\mathcal{B}}; k = 0$
    while $\sum M_c > 0$ & $k < K$ **do**
      $\boldsymbol{x}_{\mathcal{B}}^{k+1} = \boldsymbol{x}_{\mathcal{B}}^k + M_c \cdot \beta \cdot sign(\nabla_{\boldsymbol{x}} \ell(h_\theta(\boldsymbol{x}_{\mathcal{B}}^k), y))$
      $\boldsymbol{x}_{\mathcal{B}}^{k+1} = clip(\boldsymbol{x}_{\mathcal{B}}^{k+1}, \boldsymbol{x}_{\mathcal{B}}^{k+1} - \epsilon, \boldsymbol{x}_{\mathcal{B}}^{k+1} + \epsilon)$
      $M_c = \mathbb{1}_{\mathcal{B}}(c(x_{1\ldots M}^{k+1}) \leq c_t)$
      $k = k + 1$
    end while
    $\alpha^{t+1} = \alpha^t + \eta_\alpha \hat{g}(\alpha)$
    $\boldsymbol{w}^{t+1} = \boldsymbol{w}^t - \eta_w \hat{g}(\boldsymbol{w})$ # $\hat{g}$: stochastic gradient
  end for
end for
return $\boldsymbol{w}^T, \alpha^T$

When $c(\boldsymbol{x}^k) = 0$, the optimization problem reaches the convergence state. Specifically, such a condition can be achieved when **a)** $\nabla_{\boldsymbol{x}} f(\boldsymbol{w}, \alpha, \boldsymbol{x}^k) = 0$, or **b)** $\boldsymbol{x}^k - \boldsymbol{x}^0 = \epsilon \cdot \text{sign}\left(\nabla_{\boldsymbol{x}} f(\boldsymbol{w}, \alpha, \boldsymbol{x}^k)\right)$. Here **a)** implies $\boldsymbol{x}^k$ is a stationary point in the inner maximization problem, **b)** shows that local maximum point of $f(\boldsymbol{w}, \alpha, \boldsymbol{x}^k)$ reaches the boundary of $\mathcal{X}$. The proof process is shown in Lem.1.

Outer min-max Problem. For the outer min-max problem, we apply *Stochastic Gradient Descent Ascent* (SGDA) to solve the problem. At each iteration, SGDA performs stochastic gradient descent over the parameter $\boldsymbol{w}$ with the stepsize η_w, and stochastic gradient ascent over the parameter α with the stepsize η_α.

The total training strategy is presented in Alg.1. This is an extension of the algorithm proposed in (Wang et al., 2019a), where the outer level minimization problem now becomes a min-max problem. The value of FOSC can imply the adversarial strength of adversarial examples, whereas a small FOSC value implies a high adversarial strength. Due to this fact, through the FOSC value of current epoch c_t, we can dynamically control the strength of adversarial examples. Specifically, in the initial stage of training, the value of c_t is large, which means the generated adversarial examples are not so hard. In the later stage of training, the value of c_t is 0, which allows model to be trained on much stronger adversarial examples. Consequently, such an algorithm will allow the model to learn from in a progressive manner. Here, we use M_c to mask the examples that satisfy the condition in the `Line 9` in Alg.1. By doing so, we can ensure that the FOSC values of the adversarial examples generated by

Alg. 1 are all less than c_{max}. When the adversarial examples are obtained, we calculate the stochastic gradient of the parameters $\boldsymbol{w}$ and α. Then we perform stochastic gradient descent-ascent on $\boldsymbol{w}$ and α respectively.

4.3. Convergence Analysis

Next, we provide a convergence analysis of our proposed adversarial AUC optimization framework.

We first give the definition and description of some notations. In detail, let $\boldsymbol{x}_i^*(\boldsymbol{w},\alpha) = \arg\max_{\boldsymbol{x}_i\in\mathcal{X}_i} f(\boldsymbol{w},\alpha,\boldsymbol{x}_i)$ where $\mathcal{X}_i = \left\{\boldsymbol{x} \middle| \left\|\boldsymbol{x}-\boldsymbol{x}_i^0\right\|_\infty \le \epsilon\right\}$. And

$$L(\boldsymbol{w},\alpha) = \frac{1}{n}\sum_{i=1}^{n}\max_{\boldsymbol{x}_i\in\mathcal{X}_i} f(\boldsymbol{w},\alpha,\boldsymbol{x}_i) = \frac{1}{n}\sum_{i=1}^{n} f(\boldsymbol{w},\alpha,\boldsymbol{x}_i^*).$$

Then $\hat{\boldsymbol{x}}_i(\boldsymbol{w},\alpha)$ is a δ-approximate solution to $\boldsymbol{x}_i^*(\boldsymbol{w},\alpha)$, if it satisfies that

$$\max_{\boldsymbol{x}\in\mathcal{X}_i}\langle \boldsymbol{x}-\hat{\boldsymbol{x}}_i(\boldsymbol{w},\alpha), \nabla_{\boldsymbol{x}} f(\boldsymbol{w},\alpha,\hat{\boldsymbol{x}}_i(\boldsymbol{w},\alpha))\rangle \le \delta. \quad (6)$$

Furthermore, let $\nabla L(\alpha)$ denote the gradient of $L(\boldsymbol{w},\alpha)$ w.t.r. α. And let $g(\alpha) = \frac{1}{M}\sum_{i\in\mathcal{B}}\nabla_\alpha f(\boldsymbol{w},\alpha,\boldsymbol{x}_i^*)$ be the stochastic gradient of $L(\boldsymbol{w},\alpha)$ w.r.t. α, where $\mathcal{B}$ is mini-batch and $M = |\mathcal{B}|$. Meanwhile, let $\nabla_\alpha f(\boldsymbol{w},\alpha,\hat{\boldsymbol{x}}(\boldsymbol{w},\alpha))$ be the gradient of $f(\boldsymbol{w},\alpha,\hat{\boldsymbol{x}}(\boldsymbol{w},\alpha))$ w.r.t. α, and let $\hat{g}(\alpha) = \frac{1}{M}\sum_{i\in\mathcal{B}} f(\boldsymbol{w},\alpha,\hat{\boldsymbol{x}}_i)$ be the approximate stochastic gradient of $L(\boldsymbol{w},\alpha)$ w.r.t. α. And for $\boldsymbol{w}$, we have the same definition as α. In addition, let

$$\Phi(\boldsymbol{w}) = \max_{\alpha} L(\boldsymbol{w},\alpha), \quad \nabla\Phi(\boldsymbol{w}) = \nabla_{\boldsymbol{w}} L(\boldsymbol{w},\alpha^*(\boldsymbol{w})).$$

Then, before giving the convergence analysis, we list some assumptions needed to the analysis.

Assumption 1. *The function $f(\boldsymbol{w},\alpha,\boldsymbol{x})$ satisfies the gradient Lipschitz conditions as follows:*

$$\sup_{\alpha,\boldsymbol{x}}\|\nabla_{\boldsymbol{w}} f(\boldsymbol{w},\alpha,\boldsymbol{x}) - \nabla_{\boldsymbol{w}} f(\boldsymbol{w}',\alpha,\boldsymbol{x})\|_2 \le L_{ww}\|\boldsymbol{w}-\boldsymbol{w}'\|_2$$
$$\sup_{\alpha,\boldsymbol{w}}\|\nabla_{\boldsymbol{w}} f(\boldsymbol{w},\alpha,\boldsymbol{x}) - \nabla_{\boldsymbol{w}} f(\boldsymbol{w},\alpha,\boldsymbol{x}')\|_2 \le L_{wx}\|\boldsymbol{x}-\boldsymbol{x}'\|_2$$
$$\sup_{\alpha,\boldsymbol{x}}\|\nabla_{\boldsymbol{x}} f(\boldsymbol{w},\alpha,\boldsymbol{x}) - \nabla_{\boldsymbol{x}} f(\boldsymbol{w}',\alpha,\boldsymbol{x})\|_2 \le L_{xw}\|\boldsymbol{w}-\boldsymbol{w}'\|_2$$
$$\sup_{\boldsymbol{w},\boldsymbol{x}}\|\nabla_{\alpha} f(\boldsymbol{w},\alpha,\boldsymbol{x}) - \nabla_{\alpha} f(\boldsymbol{w},\alpha',\boldsymbol{x})\|_2 \le L_{\alpha\alpha}\|\alpha-\alpha'\|_2$$
$$\sup_{\boldsymbol{w},\alpha}\|\nabla_{\alpha} f(\boldsymbol{w},\alpha,\boldsymbol{x}) - \nabla_{\alpha} f(\boldsymbol{w},\alpha,\boldsymbol{x}')\|_2 \le L_{\alpha x}\|\boldsymbol{x}-\boldsymbol{x}'\|_2$$
$$\sup_{\boldsymbol{w},\boldsymbol{x}}\|\nabla_{\boldsymbol{x}} f(\boldsymbol{w},\alpha,\boldsymbol{x}) - \nabla_{\boldsymbol{x}} f(\boldsymbol{w},\alpha',\boldsymbol{x})\|_2 \le L_{x\alpha}\|\alpha-\alpha'\|_2$$

where $L_{\alpha\alpha}, L_{\alpha x}, L_{x\alpha}, L_{ww}, L_{wx}, L_{xw}$ are positive constants.

Remark 3. *The first three gradient Lipschitz conditions in Asm.1 are made in (Sinha et al., 2018), and the last three gradient Lipschitz conditions are made in (Liu et al., 2019). Meanwhile, for the overparameterized deep neural network, the loss function is semi-smooth (Allen-Zhu et al., 2019; Du et al., 2019), which helps to justify Asm.1.*

Assumption 2. *$\|\nabla_{\boldsymbol{w}} f(\boldsymbol{w},a,\boldsymbol{x})\|_2$ is upper bounded by l_w.*

Remark 4. *Asm.2 is widely used in minimax optimization problems (Sinha et al., 2018).*

Assumption 3. *$f(\boldsymbol{w},\alpha,\boldsymbol{x})$ is locally μ-strongly concave in $\mathcal{X}_i$ for all $i\in[n]$, i.e. for any $\boldsymbol{x}_1,\boldsymbol{x}_2\in\mathcal{X}_i$, it holds that*

$$f(\boldsymbol{w},\alpha,\boldsymbol{x}_1) \le f(\boldsymbol{w},\alpha,\boldsymbol{x}_2) + \langle\nabla_{\boldsymbol{x}} f(\boldsymbol{w},\alpha,\boldsymbol{x}_2), \boldsymbol{x}_1-\boldsymbol{x}_2\rangle - \frac{\mu}{2}\|\boldsymbol{x}_1-\boldsymbol{x}_2\|_2^2$$

Remark 5. *The strongly concave assumption is equivalent to weakly concave assumption of r, which is much easier to be achieved.*

Assumption 4. *The stochastic gradient $g(\alpha)$ satisfies*

$$\mathbb{E}[g(\alpha)-\nabla L(\alpha)] = 0, \quad \mathbb{E}\left[\|g(\alpha)-\nabla L(\alpha)\|_2^2\right] \le \sigma^2$$
$$\mathbb{E}[g(\boldsymbol{w})-\nabla L(\boldsymbol{w})] = 0, \quad \mathbb{E}\left[\|g(\boldsymbol{w})-\nabla L(\boldsymbol{w})\|_2^2\right] \le \sigma^2$$

Remark 6. *Asm.4 is a common assumption used to analyze the optimization algorithm based on stochastic gradient (Lin et al., 2020; Liu et al., 2019).*

Theorem 2. *Under the above assumptions and let the step-sizes be chosen as $\eta_w = \Theta\left(1/\kappa^2 L\right)$, $\eta_\alpha = \Theta(1/L)$ and $\kappa \le \frac{7}{6}$, then we have the following inequality:*

$$\frac{1}{T+1}\left(\sum_{t=0}^{T}\mathbb{E}\left[\|\nabla\Phi(\boldsymbol{w}_t)\|_2^2\right]\right) \le \frac{360\kappa^2 L\Delta_\Phi + 13\kappa L^2 D^2}{T+1} + \frac{26\kappa\sigma^2}{M} + h_\delta + h_\Delta$$

where $h_\delta = \frac{1024}{253}\left(\frac{3\kappa L\delta}{1024} + 6\kappa^4 L\delta + \frac{L\delta}{8} + L^2\sqrt{\frac{\delta}{\mu}}\right)$ and $h_\Delta = \frac{384\kappa^4 L^2\Delta}{253}$ and $\Delta_\Phi = \Phi(\boldsymbol{w}^0) - \min_{\boldsymbol{w}}\Phi(\boldsymbol{w})$, $D = |\Omega_\alpha|$, L denotes the maximum in Lipschitz constant, and the condition number $\kappa = L/\mu$.

Remark 7. *On the right side of the inequality, the first term is an $O(1/T)$ magnitude, the last three terms behave as the residuals. The second term $\frac{26\kappa\sigma^2}{M}$ is related to the batch size M. As M increases, its value gradually tends to 0. The third term h_δ is due to the use of an approximate solution $\hat{\boldsymbol{x}}$ to the inner maximization problem instead of the optimal solution $\boldsymbol{x}^*$, and the fourth term h_Δ is due to $\mathcal{X}$ being a bounded set. Since all of the residuals are small, we can find an ϵ-stationary point within a finite number of epochs.*

5. Experiments

In this section, we evaluate the performance of our AdAUC algorithms in three long-tail datasets.

Table 2. Test AUC and robustness of models trained with various methods. We mark the best performance with **Bold** and the second best performance with underscore.

Dataset	Method	Training	Evaluated Against						
			Clean	FSGM	PGD-5	PGD-10	PGD-20	C&W	AA
CIFAR-10-LT	CE	NT	0.7264	0.4038	0.0753	0.0206	0.0044	0.0009	0.0082
		AT_1	0.6659	0.5487	0.3335	0.2743	0.2344	0.2330	0.2678
		AT_2	0.6833	0.6296	0.4870	0.4417	0.4319	0.4310	0.4384
	AdAUC	NT	**0.7885**	0.6606	0.2671	0.1892	0.0064	0.0573	0.0740
		AT_1	0.7347	0.6646	0.5236	0.4625	0.4224	0.3927	0.4362
		AT_2	0.7528	**0.6952**	**0.5591**	**0.5309**	**0.5283**	**0.5283**	**0.5291**
CIFAR-100-LT	CE	NT	0.6382	0.1207	0.0271	0.0159	0.0110	0.0102	0.0123
		AT_1	0.6193	0.5183	0.3195	0.2750	0.2668	0.2630	0.2703
		AT_2	0.6198	0.5192	0.3183	0.2767	0.2681	0.2647	0.2712
	AdAUC	NT	**0.6462**	0.5161	0.3046	0.1818	0.1214	0.0035	0.1313
		AT_1	0.6302	0.5301	0.3815	0.3306	0.2989	0.2760	0.3102
		AT_2	0.6313	**0.5798**	**0.4644**	**0.4234**	**0.4065**	**0.3968**	**0.4122**
MNIST-LT	CE	NT	0.9736	0.7057	0.0116	0.0010	0.0002	0.0000	0.0003
		AT_1	0.9488	0.9302	0.8733	0.8626	0.8615	0.8611	0.8618
		AT_2	0.9547	0.9392	0.8912	0.8824	0.8816	0.8813	0.8818
	AdAUC	NT	**0.9904**	0.9309	0.5677	0.4419	0.3913	0.3645	0.4026
		AT_1	0.9772	0.9695	0.9422	**0.9395**	**0.9382**	**0.9381**	**0.9383**
		AT_2	0.9852	**0.9774**	**0.9436**	0.9347	0.9323	0.9310	0.9311

5.1. Competitors and Experiment Setting.

We compare the performance of our proposed algorithm and the AT methods with the classical **CE** classification loss function when the datasets have the long tail distribution.

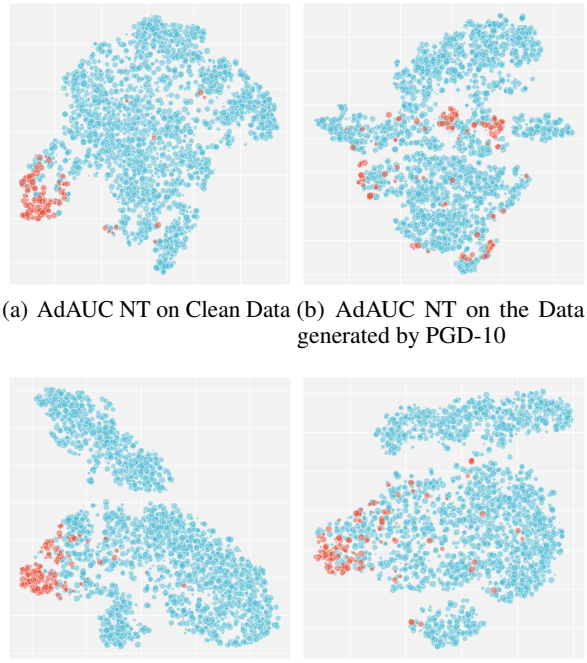
We adopt the WideResNet-28 (Zagoruyko & Komodakis, 2016) as the model architecture. The other detailed settings are shown in App.C.1. In Tab.2, **NT** means Natural Training without adversarial operations, $\mathbf{AT}_1$ means adversarial training without FOSC, and $\mathbf{AT}_2$ means algorithm in Alg.1.

To validate the robustness of our algorithm, we adopt FSGM (Goodfellow et al., 2014), iterative attack PGD (Madry et al., 2018), C&W (Carlini & Wagner, 2017) and ensemble attack AA (Croce & Hein, 2020) as attack methods.

5.2. Dataset Description

Binary CIFAR-10-LT Dataset. We construct a long-tail CIFAR-10 dataset, where the sample size across different classes decays exponentially and ensure the ratio of sample sizes of the least frequent to the most frequent class is set to 0.01. Then, we label the first 5 classes as the negative class and the last 5 classes as positive, which leads that the ratio of positive class size to negative class size $\rho \approx 1:9$.

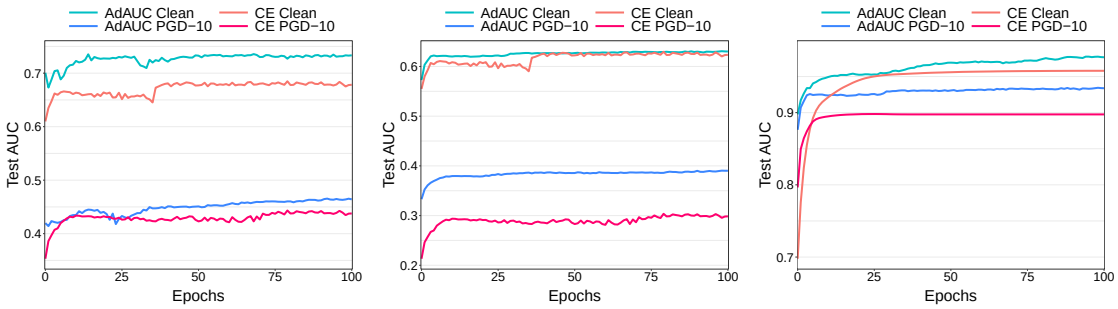
Binary CIFAR-100-LT Dataset. We construct a long-tail CIFAR-100 dataset in the same way as CIFAR-10-LT, where we label the first 50 classes as the negative class and the last 50 classes as positive, which leads that the ratio of positive



(a) AdAUC NT on Clean Data (b) AdAUC NT on the Data generated by PGD-10

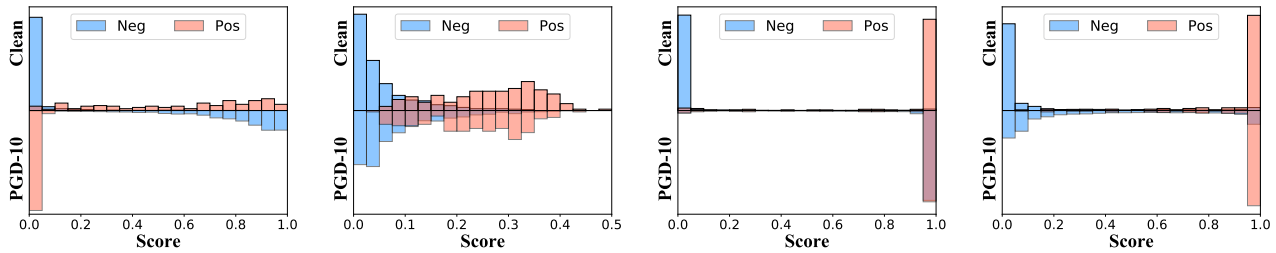
(c) AdAUC AT_2 on Clean Data (d) AdAUC AT_2 on the Data generated by PGD-10

Figure 2. The t-SNE projection of our methods on MNIST-LT dataset. **Red points** represent the positive examples, and **blue points** represent the negative examples.



(a) CIFAR-10-LT (b) CIFAR-100-LT (c) MNIST-LT

Figure 3. The convergence of AUC on testing data of CIFAR-10-LT, CIFAR-100-LT and MNIST-LT.



(a) CE NT (b) CE AT_2 (c) AdAUC NT (d) AdAUC AT_2

Figure 4. Score distribution of positive and negative examples on MNIST-LT dataset. In each subfigure, the **above part** represents the score distribution evaluated against clean data, and the **below part** represents the score distribution evaluated against PGD-10.

class size to negative class size $\rho \approx 1:9$.

Binary MNIST-LT Dataset. We construct a long-tail MNIST dataset from original MNIST dataset (LeCun et al., 1998) in the same way as CIFAR-10-LT, where the ratio of positive class size to negative class size $\rho \approx 1:9$.

5.3. Overall Performance

The performance and robustness of all the involved methods on three datasets are shown in Tab.2. Consequently, we have the following observations: **1)** On all the datasets, our methods achieve the best or competitive performance evaluated against all adversarial attack methods as shown in Tab.2. **2)** Even the AUC optimization with NT has certain robustness (the AUC will not drop to 0 when evaluated against adversarial examples). This is because the decision surface obtained by AUC optimization has a greater tolerance for minority classes than CE. Specifically, the decision surface is far away from the positive examples. To validate this argument, we show the score distribution in Fig.4 w.r.t MNIST-LT. For AdAUC NT, adversarial examples increase the score of the negative examples, while it has less impact on the positive examples. However, the perturbation becomes much more violent. As shown in Fig.4-(a), the adversarial examples simultaneously increase the score of the negative examples and decreases the score of the positive examples. The more results of other datasets are shown in the App.C.2, which show a similar trend. Moreover, we show the feature visualization of NT and AT_2 of our methods for clean data and adversarial examples. It implies that our AdAUC algorithm could separate the positive and negative instances well in the embedding space in Fig.2.

5.4. Convergence Analysis

We report the convergence of test AUC of CE-based methods and our proposed methods in Fig. 3. We can observe that our proposed method performs better both on clean data and adversarial examples. However, due to the high complexity of the outer min-max problem, it could be seen that our method converges slightly slower than CE methods, which is consistent with the analysis of Thm.2.

6. Conclusion

In this paper, we initiate the study on adversarial AUC optimization against long-tail problem. The complexity of AUC loss function makes the corresponding adversarial training hardly scalable. To address this issue, we first construct a reformulation of the AT problem of AUC optimization.

By further applying a concavity promoting regularizer, we can reformulate the original problem as a min-max problem where the objective function can be expressed instance-wisely. On top of the reformulation, we construct an end-to-end training algorithm with provable guarantee. Finally, we conduct a series of empirical studies on three long-tail benchmark datasets, the results of which demonstrate the effectiveness of our proposed method.

Acknowledgements

This work was supported in part by the National Key R&D Program of China under Grant 2018AAA0102000, in part by National Natural Science Foundation of China: U21B2038, 61931008, 6212200758 and 61976202, in part by the Fundamental Research Funds for the Central Universities, in part by Youth Innovation Promotion Association CAS, in part by the Strategic Priority Research Program of Chinese Academy of Sciences, Grant No. XDB28000000, and in part by the National Postdoctoral Program for Innovative Talents under Grant BX2021298.

References

Agarwal, S. Surrogate regret bounds for bipartite ranking via strongly proper losses. *The Journal of Machine Learning Research*, 15(1):1653–1674, 2014.

Agarwal, S., Graepel, T., Herbrich, R., Har-Peled, S., Roth, D., and Jordan, M. I. Generalization bounds for the area under the roc curve. *Journal of Machine Learning Research*, 6(4), 2005.

Allen-Zhu, Z., Li, Y., and Song, Z. A convergence theory for deep learning via over-parameterization. In *International Conference on Machine Learning*, pp. 242–252, 2019.

Athalye, A. and Carlini, N. On the robustness of the cvpr 2018 white-box adversarial example defenses. *arXiv preprint arXiv:1804.03286*, 2018.

Athalye, A., Carlini, N., and Wagner, D. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International conference on machine learning*, pp. 274–283. PMLR, 2018.

Bai, Y., Gautam, T., and Sojoudi, S. Efficient global optimization of two-layer relu networks: Quadratic-time algorithms and adversarial training. *CoRR*, abs/2201.01965, 2022.

Bao, H., Scott, C., and Sugiyama, M. Calibrated surrogate losses for adversarially robust classification. In *Conference on Learning Theory*, pp. 408–451. PMLR, 2020.

Biggio, B., Corona, I., Maiorca, D., Nelson, B., Šrndić, N., Laskov, P., Giacinto, G., and Roli, F. Evasion attacks against machine learning at test time. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 387–402, 2013.

Böhm, A. and Wright, S. J. Variable smoothing for weakly convex composite functions. *Journal of optimization theory and applications*, 188(3):628–649, 2021.

Cai, Q.-Z., Liu, C., and Song, D. Curriculum adversarial training. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pp. 3740–3747, 2018.

Carlini, N. and Wagner, D. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy*, pp. 39–57, 2017.

Clémençon, S., Lugosi, G., and Vayatis, N. Ranking and empirical minimization of u-statistics. *The Annals of Statistics*, 36(2):844–874, 2008.

Cortes, C. and Mohri, M. Auc optimization vs. error rate minimization. *Advances in neural information processing systems*, 16:313–320, 2003.

Croce, F. and Hein, M. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International Conference on Machine Learning*, pp. 2206–2216, 2020.

Du, S., Lee, J., Li, H., Wang, L., and Zhai, X. Gradient descent finds global minima of deep neural networks. In *International Conference on Machine Learning*, pp. 1675–1685, 2019.

Fawcett, T. An introduction to roc analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.

Feizi, A. Hierarchical detection of abnormal behaviors in video surveillance through modeling normal behaviors based on auc maximization. *Soft Computing*, 24(14): 10401–10413, 2020.

Gao, W. and Zhou, Z.-H. On the consistency of auc pairwise optimization. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

Gao, W., Jin, R., Zhu, S., and Zhou, Z.-H. One-pass auc optimization. In *International conference on machine learning*, pp. 906–914, 2013.

Gola, D., Erdmann, J., Müller-Myhsok, B., Schunkert, H., and König, I. R. Polygenic risk scores outperform machine learning methods in predicting coronary artery disease status. *Genetic epidemiology*, 44(2):125–138, 2020.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In *Proceedings of the*

27th International Conference on Neural Information Processing Systems, pp. 2672–2680, 2014.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015.

Hand, D. J. and Till, R. J. A simple generalisation of the area under the roc curve for multiple class classification problems. *Machine Learning*, 45(2):171–186, 2001.

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

Herschtal, A. and Raskutti, B. Optimising area under the roc curve using gradient descent. In *Proceedings of the twenty-first international conference on Machine learning*, pp. 49, 2004.

Joachims, T. A support vector method for multivariate performance measures. In *Proceedings of the 22nd international conference on Machine learning*, pp. 377–384, 2005.

Kurakin, A., Goodfellow, I., and Bengio, S. Adversarial machine learning at scale. In *International Conference on Learning Representations*, 2017.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Lin, T., Jin, C., and Jordan, M. On gradient descent ascent for nonconvex-concave minimax problems. In *International Conference on Machine Learning*, pp. 6083–6093, 2020.

Liu, M., Yuan, Z., Ying, Y., and Yang, T. Stochastic auc maximization with deep neural networks. In *International Conference on Learning Representations*, 2019.

Liu, M., Rafique, H., Lin, Q., and Yang, T. First-order convergence theory for weakly-convex-weakly-concave min-max problems. *Journal of Machine Learning Research*, 22(169):1–34, 2021.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

Maini, P., Wong, E., and Kolter, Z. Adversarial robustness against the union of multiple perturbation models. In *International Conference on Machine Learning*, pp. 6640–6650. PMLR, 2020.

Natole, M., Ying, Y., and Lyu, S. Stochastic proximal algorithms for auc maximization. In *International Conference on Machine Learning*, pp. 3710–3719. PMLR, 2018.

Nesterov, Y. Introductory lectures on convex programming volume i: Basic course. *Lecture notes*, 3(4):5, 1998.

Neumann, J. v. Zur theorie der gesellschaftsspiele. *Mathematische Annalen*, 100(1):295–320, 1928.

Pang, T., Xu, K., Du, C., Chen, N., and Zhu, J. Improving adversarial robustness via promoting ensemble diversity. In *International Conference on Machine Learning*, pp. 4970–4979. PMLR, 2019.

Ren, K., Yang, H., Zhao, Y., Chen, W., Xue, M., Miao, H., Huang, S., and Liu, J. A robust auc maximization framework with simultaneous outlier detection and feature selection for positive-unlabeled classification. *IEEE transactions on neural networks and learning systems*, 30 (10):3072–3083, 2018.

Robles, E., Zaidouni, F., Mavromoustaki, A., and Refael, P. Threshold optimization in multiple binary classifiers for extreme rare events using predicted positive data. In *AAAI Spring Symposium: Combining Machine Learning with Knowledge Engineering (1)*, 2020.

Shafahi, A., Najibi, M., Ghiasi, A., Xu, Z., Dickerson, J., Studer, C., Davis, L. S., Taylor, G., and Goldstein, T. Adversarial training for free! In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, pp. 3358–3369, 2019.

Shafahi, A., Najibi, M., Xu, Z., Dickerson, J., Davis, L. S., and Goldstein, T. Universal adversarial training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 5636–5643, 2020.

Sinha, A., Namkoong, H., and Duchi, J. Certifying some distributional robustness with principled adversarial training. In *International Conference on Learning Representations*, 2018.

Sion, M. On general minimax theorems. *Pacific Journal of Mathematics*, 8(1):171–176, 1958.

Sorin, V., Barash, Y., Konen, E., and Klang, E. Deep learning for natural language processing in radiology—fundamentals and a systematic review. *Journal of the American College of Radiology*, 17(5):639–648, 2020.

Strubell, E., Ganesh, A., and McCallum, A. Energy and policy considerations for deep learning in nlp. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 3645–3650, 2019.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. In *International Conference on Learning Representations*, 2014.

Tramèr, F. and Boneh, D. Adversarial training and robustness for multiple perturbations. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, pp. 5866–5876, 2019.

Tramèr, F., Kurakin, A., Papernot, N., Boneh, D., and McDaniel, P. Ensemble adversarial training: Attacks and defenses. *stat*, 1050:30, 2017.

Tu, Z., Zhang, J., and Tao, D. Theoretical analysis of adversarial learning: A minimax approach. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Usunier, N., Amini, M.-R., and Gallinari, P. A data-dependent generalisation error bound for the auc. In *Proceedings of the ICML 2005 Workshop on ROC Analysis in Machine Learning*. Citeseer, 2005.

Voulodimos, A., Doulamis, N., Doulamis, A., and Protopapadakis, E. Deep learning for computer vision: A brief review. *Computational intelligence and neuroscience*, 2018, 2018.

Wang, Y., Ma, X., Bailey, J., Yi, J., Zhou, B., and Gu, Q. On the convergence and robustness of adversarial training. In *International Conference on Machine Learning*, pp. 6586–6595, 2019a.

Wang, Y., Zou, D., Yi, J., Bailey, J., Ma, X., and Gu, Q. Improving adversarial robustness requires revisiting misclassified examples. In *International Conference on Learning Representations*, 2019b.

Westcott, A., Capaldi, D. P., McCormack, D. G., Ward, A. D., Fenster, A., and Parraga, G. Chronic obstructive pulmonary disease: thoracic ct texture analysis and machine learning to predict pulmonary ventilation. *Radiology*, 293(3):676–684, 2019.

Wong, E., Rice, L., and Kolter, J. Z. Fast is better than free: Revisiting adversarial training. In *International Conference on Learning Representations*, 2019.

Xing, Y., Song, Q., and Cheng, G. On the generalization properties of adversarial training. In *International Conference on Artificial Intelligence and Statistics*, pp. 505–513, 2021.

Ying, Y., Wen, L., and Lyu, S. Stochastic online auc maximization. *Advances in Neural Information Processing Systems*, 29:451–459, 2016.

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

Zhang, D., Zhang, T., Lu, Y., Zhu, Z., and Dong, B. You only propagate once: Accelerating adversarial training via maximal principle. *Advances in Neural Information Processing Systems*, 32:227–238, 2019a.

Zhang, H., Yu, Y., Jiao, J., Xing, E., El Ghaoui, L., and Jordan, M. Theoretically principled trade-off between robustness and accuracy. In *International Conference on Machine Learning*, pp. 7472–7482. PMLR, 2019b.

Zhang, J., Zhu, J., Niu, G., Han, B., Sugiyama, M., and Kankanhalli, M. Geometry-aware instance-reweighted adversarial training. In *International Conference on Learning Representations*, 2020.

Zhao, P., Hoi, S. C., Jin, R., and Yang, T. Online auc maximization. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, pp. 233–240, 2011.

A. Proofs of Main Results

In this section, we provide the proofs of the main results.

A.1. Proof of Proposition 2

Proof. According to (Ying et al., 2016), a, b, α have the following closed-form solution:

$$a = \hat{\mathbb{E}}[h_\theta(\boldsymbol{x})|y=1], \;\; b = \hat{\mathbb{E}}[h_\theta(\boldsymbol{x})|y=0], \;\; \alpha = \hat{\mathbb{E}}[h_\theta(\boldsymbol{x})|y=0] - \hat{\mathbb{E}}[h_\theta(\boldsymbol{x})|y=1].$$

Then it is easy to check that

$$r(a,b) = \max_\alpha \frac{1}{n}\sum_{i=1}^n g(\boldsymbol{\theta}, a, b, \alpha, (\boldsymbol{x}_i + \boldsymbol{\delta}_i, y_i))$$

is a strongly-convex problem w.r.t. (a,b). However, r is in general not concave w.r.t. $\boldsymbol{\delta}_i$. In this sense, we then try to find a surrogate objective to induce the concavity w.r.t $\boldsymbol{\delta}_i$. Specifically, we adopt a concavity regularization term $-\gamma||\boldsymbol{x}_i + \boldsymbol{\delta}_i||_2^2$, and define a surrogate objective:

$$f(\boldsymbol{w}, \alpha, \boldsymbol{x}_i + \boldsymbol{\delta}_i) = g(\boldsymbol{\theta}, a, b, \alpha, (\boldsymbol{x}_i + \boldsymbol{\delta}_i, y_i)) - \gamma \left\|\boldsymbol{x}_i + \boldsymbol{\delta}_i\right\|_2^2.$$

Therefore, if r is $\gamma_\star$-weakly concave w.r.t. $\boldsymbol{\delta}_i, i = 1, 2, \cdots, n$ (Liu et al., 2021; Böhm & Wright, 2021), then we can define $\gamma > \gamma_\star$ to obtain an objective $f(\boldsymbol{w}, \alpha, \boldsymbol{x} + \boldsymbol{\delta})$ such that $\max_\alpha \frac{1}{n}\sum_{i=1}^n f(\boldsymbol{w}, \alpha, \boldsymbol{x}_i + \boldsymbol{\delta}_i)$ is strongly concave w.r.t. $\boldsymbol{\delta}_i, i = 1, 2, \cdots, n$. Above all, with the weakly concavity assumption, we can instead solve the following surrogate problem by the von Neumann's minimax theorem:

$$\begin{aligned} \textbf{(OP)} \quad & \min_{\boldsymbol{w}} \max_\alpha \max_{\boldsymbol{\delta}} \frac{1}{n}\sum_{i=1}^n [f(\boldsymbol{w}, \alpha, \boldsymbol{x}_i + \boldsymbol{\delta}_i)] \\ & = \min_{\boldsymbol{w}} \max_\alpha \frac{1}{n}\sum_{i=1}^n \max_{\boldsymbol{\delta}_i} [f(\boldsymbol{w}, \alpha, \boldsymbol{x}_i + \boldsymbol{\delta}_i)], \end{aligned} \tag{7}$$

where $\boldsymbol{w} = (\boldsymbol{\theta}, a, b)$. □

A.2. Proof of Lemma 1

Lemma 1. *For all $x \in \mathcal{X}$, $c(\boldsymbol{x}^k) = 0$ when 1) $\nabla_{\boldsymbol{x}} f(\boldsymbol{w}, \alpha, \boldsymbol{x}^k) = 0$, or 2) $\boldsymbol{x}^k - \boldsymbol{x}^0 = \epsilon \cdot \mathrm{sign}(\nabla_{\boldsymbol{x}} f(\boldsymbol{w}, \alpha, \boldsymbol{x}^k))$.*

Proof.

$$\begin{aligned} c(\boldsymbol{x}^k) &= \max_{\boldsymbol{x}\in\mathcal{X}} \langle \boldsymbol{x} - \boldsymbol{x}^k, \nabla_{\boldsymbol{x}} f(\boldsymbol{w}, \alpha, \boldsymbol{x}^k)\rangle \\ &= \max_{\boldsymbol{x}\in\mathcal{X}} \langle \boldsymbol{x} - \boldsymbol{x}^0 + \boldsymbol{x}^0 - \boldsymbol{x}^k, \nabla_{\boldsymbol{x}} f(\boldsymbol{w}, \alpha, \boldsymbol{x}^k)\rangle \\ &= \max_{\boldsymbol{x}\in\mathcal{X}} \langle \boldsymbol{x} - \boldsymbol{x}^0, \nabla_{\boldsymbol{x}} f(\boldsymbol{w}, \alpha, \boldsymbol{x}^k)\rangle - \langle \boldsymbol{x}^k - \boldsymbol{x}^0, \nabla_{\boldsymbol{x}} f(\boldsymbol{w}, \alpha, \boldsymbol{x}^k)\rangle \\ &= \epsilon \cdot \left\|\nabla_{\boldsymbol{x}} f(\boldsymbol{w}, \alpha, \boldsymbol{x}^k)\right\|_1 - \langle \boldsymbol{x}^k - \boldsymbol{x}^0, \nabla_{\boldsymbol{x}} f(\boldsymbol{w}, \alpha, \boldsymbol{x}^k)\rangle \end{aligned} \tag{8}$$

This completes the proof. □

A.3. Proof of Technical Lemmas

In this subsections, we present seven key lemmas which are important for the proof of Theorem 2.

Lemma 2. *Under the Asm.1 and 3, we have $L(\alpha)$ is L_α-smooth where $L_\alpha = \frac{L_{\alpha x} L_{x\alpha}}{\mu} + L_{\alpha\alpha}$. For any α_1, α_2, it holds*

$$\begin{aligned} L(\alpha_1) \le L(\alpha_2) + \langle \nabla L(\alpha_2), \alpha_1 - \alpha_2\rangle + \frac{L_\alpha}{2}\left\|\alpha_1 - \alpha_2\right\|_2^2 \\ \left\|\nabla L(\alpha_1) - \nabla L(\alpha_2)\right\|_2 \le L_\alpha \left\|\alpha_1 - \alpha_2\right\|_2. \end{aligned} \tag{9}$$

We also have that $L(\boldsymbol{w})$ is L_w-smooth where $L_w = \frac{L_{wx} L_{xw}}{\mu} + L_{ww}$.

Proof. Here, since we only focus on α and $\boldsymbol{x}$ when the w is fixed, we abbreviate $\boldsymbol{x}^*(w,\alpha)$ as $\boldsymbol{x}^*(\alpha)$ for convenience.

By the Asm.3, we have

$$f(\boldsymbol{w},\alpha_2,\boldsymbol{x}_i^*(\alpha_1)) \leq f(\boldsymbol{w},\alpha_2,\boldsymbol{x}_i^*(\alpha_2)) + \langle \nabla_{\boldsymbol{x}} f(\boldsymbol{w},\alpha_2,\boldsymbol{x}_i^*(\alpha_2)), \boldsymbol{x}_i^*(\alpha_1) - \boldsymbol{x}_i^*(\alpha_2)\rangle - \frac{\mu}{2}\left\|\boldsymbol{x}_i^*(\alpha_1) - \boldsymbol{x}_i^*(\alpha_2)\right\|_2^2, \quad (10)$$

$$f(\boldsymbol{w},\alpha_2,\boldsymbol{x}_i^*(\alpha_2)) \leq f(\boldsymbol{w},\alpha_2,\boldsymbol{x}_i^*(\alpha_1)) + \langle \nabla_{\boldsymbol{x}} f(\boldsymbol{w},\alpha_2,\boldsymbol{x}_i^*(\alpha_1)), \boldsymbol{x}_i^*(\alpha_2) - \boldsymbol{x}_i^*(\alpha_1)\rangle - \frac{\mu}{2}\left\|\boldsymbol{x}_i^*(\alpha_1) - \boldsymbol{x}_i^*(\alpha_2)\right\|_2^2. \quad (11)$$

Since $\langle \nabla_{\boldsymbol{x}} f(\boldsymbol{w},\alpha_2,\boldsymbol{x}_i^*(\alpha_2)), \boldsymbol{x}_i^*(\alpha_1) - \boldsymbol{x}_i^*(\alpha_2)\rangle \leq 0$, combining (10) and (11), we obtain

$$\begin{aligned}
\mu\left\|\boldsymbol{x}_i^*(\alpha_1) - \boldsymbol{x}_i^*(\alpha_2)\right\|_2^2 &\leq \langle \nabla_{\boldsymbol{x}} f(\boldsymbol{w},\alpha_2,\boldsymbol{x}_i^*(\alpha_1)), \boldsymbol{x}_i^*(\alpha_2) - \boldsymbol{x}_i^*(\alpha_1)\rangle \\
&\leq \langle \nabla_{\boldsymbol{x}} f(\boldsymbol{w},\alpha_2,\boldsymbol{x}_i^*(\alpha_1)) - \nabla_{\boldsymbol{x}} f(\boldsymbol{w},\alpha_1,\boldsymbol{x}_i^*(\alpha_1)), \boldsymbol{x}_i^*(\alpha_2) - \boldsymbol{x}_i^*(\alpha_1)\rangle \\
&\leq \left\|\nabla_{\boldsymbol{x}} f(\boldsymbol{w},\alpha_2,\boldsymbol{x}_i^*(\alpha_1)) - \nabla_{\boldsymbol{x}} f(\boldsymbol{w},\alpha_1,\boldsymbol{x}_i^*(\alpha_1))\right\|_2 \left\|\boldsymbol{x}_i^*(\alpha_2) - \boldsymbol{x}_i^*(\alpha_1)\right\|_2 \\
&\leq L_{x\alpha}\left\|\alpha_1 - \alpha_2\right\|_2 \left\|\boldsymbol{x}_i^*(\alpha_1) - \boldsymbol{x}_i^*(\alpha_2)\right\|_2,
\end{aligned} \quad (12)$$

where the second inequality holds because $\langle \nabla_{\boldsymbol{x}} f(\boldsymbol{w},\alpha_1,\boldsymbol{x}_i^*(\alpha_1)), \boldsymbol{x}_i^*(\alpha_2) - \boldsymbol{x}_i^*(\alpha_1)\rangle \leq 0$.

Then (12) immediately yields

$$\left\|\boldsymbol{x}_i^*(\alpha_1) - \boldsymbol{x}_i^*(\alpha_2)\right\|_2 \leq \frac{L_{x\alpha}}{\mu}\left\|\alpha_1 - \alpha_2\right\|_2. \quad (13)$$

Then for $i \in [n]$, we have

$$\begin{aligned}
\left\|\nabla_\alpha f(\boldsymbol{w},\alpha_1,\boldsymbol{x}_i^*(\alpha_1)) - \nabla_\alpha f(\boldsymbol{w},\alpha_2,\boldsymbol{x}_i^*(\alpha_2))\right\|_2 &\leq \left\|\nabla_\alpha f(\boldsymbol{w},\alpha_1,\boldsymbol{x}_i^*(\alpha_1)) - \nabla_\alpha f(\boldsymbol{w},\alpha_1,\boldsymbol{x}_i^*(\alpha_2))\right\|_2 \\
&\quad + \left\|\nabla_\alpha f(\boldsymbol{w},\alpha_1,\boldsymbol{x}_i^*(\alpha_2)) - \nabla_\alpha f(\boldsymbol{w},\alpha_2,\boldsymbol{x}_i^*(\alpha_2))\right\|_2 \\
&\leq L_{\alpha x}\left\|\boldsymbol{x}_i^*(\alpha_1) - \boldsymbol{x}_i^*(\alpha_2)\right\|_2 + L_{\alpha\alpha}\left\|\alpha_1 - \alpha_2\right\|_2 \\
&\leq \left(\frac{L_{\alpha x}L_{x\alpha}}{\mu} + L_{\alpha\alpha}\right)\left\|\alpha_1 - \alpha_2\right\|_2.
\end{aligned} \quad (14)$$

Finally, by the definition of $L(\alpha)$, we have

$$\begin{aligned}
\left\|\nabla L(\alpha_1) - \nabla L(\alpha_2)\right\|_2 &\leq \left\|\frac{1}{n}\sum_{i=1}^n \nabla_\alpha f(\boldsymbol{w},\alpha_1,\boldsymbol{x}_i^*(\alpha_1)) - \frac{1}{n}\sum_{i=1}^n \nabla_\alpha f(\boldsymbol{w},\alpha_2,\boldsymbol{x}_i^*(\alpha_2))\right\|_2 \\
&\leq \frac{1}{n}\sum_{i=1}^n \left\|\nabla_\alpha f(\boldsymbol{w},\alpha_1,\boldsymbol{x}_i^*(\alpha_1)) - \nabla_\alpha f(\boldsymbol{w},\alpha_2,\boldsymbol{x}_i^*(\alpha_2))\right\|_2 \\
&\leq \left(\frac{L_{\alpha x}L_{x\alpha}}{\mu} + L_{\alpha\alpha}\right)\left\|\alpha_1 - \alpha_2\right\|_2.
\end{aligned} \quad (15)$$

This completes the proof. □

Lemma 3. *Under the Asm.1, we have $\Phi(\boldsymbol{w})$ is L_w-smooth where$L_w = \frac{L_{w\alpha}L_{\alpha w}}{\mu} + L_{ww}$. For any $\boldsymbol{w}_1, \boldsymbol{w}_2$, it holds*

$$\begin{aligned}
\Phi(\boldsymbol{w}_1) &\leq \Phi(\boldsymbol{w}_2) + \langle \nabla\Phi(\boldsymbol{w}_2), \boldsymbol{w}_1 - \boldsymbol{w}_2\rangle + \frac{L_w}{2}\left\|\boldsymbol{w}_1 - \boldsymbol{w}_2\right\|_2^2 \\
\left\|\nabla\Phi(\boldsymbol{w}_1) - \nabla\Phi(\boldsymbol{w}_2)\right\|_2 &\leq L_w\left\|\boldsymbol{w}_1 - \boldsymbol{w}_2\right\|_2.
\end{aligned} \quad (16)$$

Proof. By the Rem.1, we have

$$L(\boldsymbol{w}_2,\alpha^*(\boldsymbol{w}_1)) \leq L(\boldsymbol{w}_2,\alpha^*(\boldsymbol{w}_2)) + \langle \nabla_\alpha L(\boldsymbol{w}_2,\alpha^*(\boldsymbol{w}_2)), \alpha^*(\boldsymbol{w}_1) - \alpha^*(\boldsymbol{w}_2)\rangle - \frac{\mu}{2}\left\|\alpha^*(\boldsymbol{w}_1) - \alpha^*(\boldsymbol{w}_2)\right\|_2^2, \quad (17)$$

$$L(\boldsymbol{w}_2,\alpha^*(\boldsymbol{w}_2)) \leq L(\boldsymbol{w}_2,\alpha^*(\boldsymbol{w}_1)) + \langle \nabla_\alpha L(\boldsymbol{w}_2,\alpha^*(\boldsymbol{w}_1)), \alpha^*(\boldsymbol{w}_2) - \alpha^*(\boldsymbol{w}_1)\rangle - \frac{\mu}{2}\left\|\alpha^*(\boldsymbol{w}_1) - \alpha^*(\boldsymbol{w}_2)\right\|_2^2. \quad (18)$$

Since $\langle \nabla_\alpha L(\boldsymbol{w}_2, \alpha^*(\boldsymbol{w}_2)), \alpha^*(\boldsymbol{w}_1) - \alpha^*(\boldsymbol{w}_2)\rangle \leq 0$, combining (17) and (18), we obtain

$$\begin{aligned}
\mu \left\|\alpha^*(\boldsymbol{w}_1) - \alpha^*(\boldsymbol{w}_2)\right\|_2^2 &\leq \langle \nabla_\alpha L(\boldsymbol{w}_2, \alpha^*(\boldsymbol{w}_1)), \alpha^*(\boldsymbol{w}_2) - \alpha^*(\boldsymbol{w}_1)\rangle \\
&\leq \langle \nabla_\alpha L(\boldsymbol{w}_2, \alpha^*(\boldsymbol{w}_1),) - \nabla_\alpha L(\boldsymbol{w}_1, \alpha^*(\boldsymbol{w}_1)), \alpha^*(\boldsymbol{w}_2) - \alpha^*(\boldsymbol{w}_1)\rangle \\
&\leq \left\|\nabla_\alpha L(\boldsymbol{w}_2, \alpha^*(\boldsymbol{w}_1)) - \nabla_\alpha L(\boldsymbol{w}_1, \alpha^*(\boldsymbol{w}_1))\right\|_2 \left\|\alpha^*(\boldsymbol{w}_2) - \alpha^*(\boldsymbol{w}_1)\right\|_2 \\
&\leq L_{\alpha w} \left\|\boldsymbol{w}_2 - \boldsymbol{w}_1\right\|_2 \left\|\alpha^*(\boldsymbol{w}_2) - \alpha^*(\boldsymbol{w}_1)\right\|_2 .
\end{aligned} \tag{19}$$

Then (19) yields

$$\left\|\alpha^*(\boldsymbol{w}_1) - \alpha^*(\boldsymbol{w}_2)\right\|_2 \leq \frac{L_{\alpha w}}{\mu} \left\|\boldsymbol{w}_1 - \boldsymbol{w}_2\right\|_2 . \tag{20}$$

Then we have for $i \in [n]$,

$$\begin{aligned}
\left\|\nabla_{\boldsymbol{w}} L(\boldsymbol{w}_1, \alpha^*(\boldsymbol{w}_1)) - \nabla_{\boldsymbol{w}} L(\boldsymbol{w}_2, \alpha^*(\boldsymbol{w}_2))\right\|_2 &\leq \left\|\nabla_{\boldsymbol{w}} L(\boldsymbol{w}_1, \alpha^*(\boldsymbol{w}_1))) - \nabla_{\boldsymbol{w}} L(\boldsymbol{w}_1, \alpha^*(\boldsymbol{w}_2))\right\|_2 \\
&\quad + \left\|\nabla_{\boldsymbol{w}} L(\boldsymbol{w}_1, \alpha^*(\boldsymbol{w}_2)) - \nabla_{\boldsymbol{w}} L(\boldsymbol{w}_2, \alpha^*(\boldsymbol{w}_1))\right\|_2 \\
&\leq L_{w\alpha} \left\|\alpha^*(\boldsymbol{w}_1) - \alpha^*(\boldsymbol{w}_2)\right\|_2 + L_{ww} \left\|\boldsymbol{w}_1 - \boldsymbol{w}_2\right\|_2 \\
&\leq \big(\frac{L_{w\alpha} L_{\alpha w}}{\mu} + L_{ww}\big) \left\|\boldsymbol{w}_1 - \boldsymbol{w}_2\right\|_2 .
\end{aligned} \tag{21}$$

Finally, by the definition of $\Phi(\boldsymbol{w})$, we have

$$\begin{aligned}
\left\|\nabla\Phi(\boldsymbol{w}_1) - \nabla\Phi(\boldsymbol{w}_2)\right\|_2 &\leq \left\|\frac{1}{n}\sum_{i=1}^n \nabla_{\boldsymbol{w}} L(\boldsymbol{w}_1, \alpha^*(\boldsymbol{w}_1)) - \frac{1}{n}\sum_{i=1}^n \nabla_{\boldsymbol{w}} L(\boldsymbol{w}_2, \alpha^*(\boldsymbol{w}_2))\right\|_2 \\
&\leq \frac{1}{n}\sum_{i=1}^n \left\|\nabla_\alpha L(\boldsymbol{w}_1, \alpha^*(\boldsymbol{w}_1)) - \nabla_\alpha L(\boldsymbol{w}_2, \alpha^*(\boldsymbol{w}_2))\right\|_2 \\
&\leq \big(\frac{L_{w\alpha} L_{\alpha w}}{\mu} + L_{ww}\big) \left\|\boldsymbol{w}_1 - \boldsymbol{w}_2\right\|_2 .
\end{aligned} \tag{22}$$

This completes the proof. □

Lemma 4. *Under Asm.1 and 3, the approximate stochastic gradient $\hat{g}(\alpha)$ satisfies*

$$\left\|\hat{g}(\alpha) - g(\alpha)\right\|_2 \leq L_{\alpha x}\sqrt{\frac{\delta}{\mu}}. \tag{23}$$

We also have

$$\left\|\hat{g}(\boldsymbol{w}) - g(\boldsymbol{w})\right\|_2 \leq L_{wx}\sqrt{\frac{\delta}{\mu}}. \tag{24}$$

Proof. We have

$$\begin{aligned}
\left\|\hat{g}(\alpha) - g(\alpha)\right\|_2 &\leq \left\|\frac{1}{|\mathcal{B}|}\sum_{i\in\mathcal{B}} \big(\nabla_\alpha f(\boldsymbol{w}, \alpha, \hat{\boldsymbol{x}}_i(\alpha)) - \nabla_\alpha f(\boldsymbol{w}, \alpha, \boldsymbol{x}_i^*(\alpha))\big)\right\|_2 \\
&\leq \frac{1}{|\mathcal{B}|}\sum_{i\in\mathcal{B}} \left\|\nabla_\alpha f(\boldsymbol{w}, \alpha, \hat{\boldsymbol{x}}_i(\alpha)) - \nabla_\alpha f(\boldsymbol{w}, \alpha, \boldsymbol{x}_i^*(\alpha))\right\|_2 \\
&\leq \frac{1}{|\mathcal{B}|}\sum_{i\in\mathcal{B}} L_{\alpha x} \left\|\hat{\boldsymbol{x}}_i(\alpha) - \boldsymbol{x}_i^*(\alpha)\right\|_2 .
\end{aligned} \tag{25}$$

By the Asm.3, we have

$$\mu \left\|\hat{\boldsymbol{x}}_i(\alpha) - \boldsymbol{x}_i^*(\alpha)\right\|_2^2 \leq \langle \nabla_\alpha f(\boldsymbol{w}, \alpha, \hat{\boldsymbol{x}}_i(\alpha)) - \nabla_\alpha f(\boldsymbol{w}, \alpha, \boldsymbol{x}_i^*(\alpha)), \hat{\boldsymbol{x}}_i(\alpha) - \boldsymbol{x}_i^*(\alpha)\rangle. \tag{26}$$

Since the $\hat{\boldsymbol{x}}_i(\alpha)$ is a δ-approximate solution to $\boldsymbol{x}_i^*(\alpha)$, if it satisfies that

$$\langle \boldsymbol{x}_i^*(\alpha) - \hat{\boldsymbol{x}}_i(\alpha), \nabla_{\boldsymbol{x}} f(\boldsymbol{w}, \alpha, \hat{\boldsymbol{x}}_i(\alpha)) \rangle \leq \delta. \tag{27}$$

Furthermore, we have

$$\langle \hat{\boldsymbol{x}}_i(\alpha) - \boldsymbol{x}_i^*(\alpha), \nabla_{\boldsymbol{x}} f(\boldsymbol{w}, \alpha, \boldsymbol{x}_i^*(\alpha)) \rangle \leq 0. \tag{28}$$

Combining (27) and (28), we obtain

$$\langle \hat{\boldsymbol{x}}_i(\alpha) - \boldsymbol{x}_i^*(\alpha), \nabla_{\boldsymbol{x}} f(\boldsymbol{w}, \alpha, \boldsymbol{x}_i^*(\alpha)) - \nabla_{\boldsymbol{x}} f(\alpha, \hat{\boldsymbol{x}}_i)(\alpha) \rangle \leq \delta. \tag{29}$$

Combining (26) and (29), we obtain

$$\mu \left\| \hat{\boldsymbol{x}}_i(\alpha) - \boldsymbol{x}_i^*(\alpha) \right\|_2^2 \leq \delta \tag{30}$$

Combining (25) and (30), we obtain

$$\left\| \hat{g}(\alpha) - g(\alpha) \right\|_2 \leq L_{\alpha x} \sqrt{\frac{\delta}{\mu}}. \tag{31}$$

□

Lemma 5. *$g(\boldsymbol{w}) = \frac{1}{M}\sum_{i=1}^{M} G_w(\boldsymbol{w}_t, \alpha_t, \xi_i)$ and $g(\alpha) = \frac{1}{M}\sum_{i=1}^{M} G_\alpha(\boldsymbol{w}_t, \alpha_t, \xi_i)$ are unbiased and have bounded variance:*

$$\begin{aligned} &\mathbb{E}[g(\boldsymbol{w})] = \nabla_{\boldsymbol{w}} L(\boldsymbol{w}_t, \alpha_t), \quad \mathbb{E}\left[\left\| g(\boldsymbol{w}) \right\|_2^2 \right] = \left\| \nabla_{\boldsymbol{w}} L(\boldsymbol{w}_t, \alpha_t) \right\|_2^2 + \frac{\sigma^2}{M}, \\ &\mathbb{E}[g(\alpha)] = \nabla_{\alpha} L(\boldsymbol{w}_t, \alpha_t), \quad \mathbb{E}\left[\left\| g(\alpha) \right\|_2^2 \right] = \left\| \nabla_{\alpha} L(\boldsymbol{w}_t, \alpha_t) \right\|_2^2 + \frac{\sigma^2}{M}. \end{aligned} \tag{32}$$

Proof. Since $\hat{g}$ is unbiased, we have

$$\mathbb{E}[g(\boldsymbol{w})] = \nabla_{\boldsymbol{w}} L(\boldsymbol{w}_t, \alpha_t), \quad \mathbb{E}[g(\alpha)] = \nabla_{\alpha} L(\boldsymbol{w}_t, \alpha_t).$$

Furthermore, we have

$$\begin{aligned} \mathbb{E}[\left\| g(\boldsymbol{w}) - \nabla_{\boldsymbol{w}} L(\boldsymbol{w}_t, \alpha_t) \right\|_2^2] &= \mathbb{E}\left[\left\| \frac{1}{M}\sum_{i=1}^{M} G_w(\boldsymbol{w}_t, \alpha_t, \xi_i) - \nabla_{\boldsymbol{w}} L(\boldsymbol{w}_t, \alpha_t) \right\|_2^2 \right] \\ &= \frac{\sum_{i=1}^{M} \mathbb{E}\left[\left\| G_w(\boldsymbol{w}_t, \alpha_t, \xi_i) - \nabla_{\boldsymbol{w}} L(\boldsymbol{w}_t, \alpha_t) \right\|_2^2 \right]}{M^2} \leq \frac{\sigma^2}{M}. \\ \mathbb{E}[\left\| g(\alpha) - \nabla_{\alpha} L(\boldsymbol{w}_t, \alpha_t) \right\|_2^2] &= \mathbb{E}\left[\left\| \frac{1}{M}\sum_{i=1}^{M} G_\alpha(\boldsymbol{w}_t, \alpha_t, \xi_i) - \nabla_{\alpha} L(\boldsymbol{w}_t, \alpha_t) \right\|_2^2 \right] \\ &= \frac{\sum_{i=1}^{M} \mathbb{E}\left[\left\| G_\alpha(\boldsymbol{w}_t, \alpha_t, \xi_i) - \nabla_{\alpha} L(\boldsymbol{w}_t, \alpha_t) \right\|_2^2 \right]}{M^2} \leq \frac{\sigma^2}{M}. \end{aligned}$$

□

Lemma 6. *The iterates $\{\boldsymbol{w}_t\}_{t\geq 1}$ satisfy the following inequality:*

$$\begin{aligned} \mathbb{E}[\Phi(\boldsymbol{w}_t)] \leq \mathbb{E}[\Phi(\boldsymbol{w}_{t-1})] &- \left(\frac{\eta_w}{2} - 2L_w \eta_w^2 \right) \mathbb{E}\left[\left\| \nabla \Phi(\boldsymbol{w}_{t-1}) \right\|_2^2 \right] \\ &+ (\frac{\eta_w}{2} + 2L_w \eta_w^2) \mathbb{E}\left[\left\| \nabla \Phi(\boldsymbol{w}_{t-1}) - \nabla_{\boldsymbol{w}} L(\boldsymbol{w}_{t-1}, \alpha_{t-1}) \right\|_2^2 \right] + \frac{L_w \eta_w^2 \sigma^2}{M} + \frac{\delta L_w L_{wx}^2 \eta_w^2}{\mu} + L_{wx} \ell_w \sqrt{\frac{\delta}{\mu}}. \end{aligned} \tag{33}$$

Proof. By Lem.2, we have

$$\Phi(\boldsymbol{w}_t) \leq \Phi(\boldsymbol{w}_{t-1}) + \langle \nabla\Phi(\boldsymbol{w}_{t-1}), \boldsymbol{w}_t - \boldsymbol{w}_{t-1}\rangle + \frac{L_w}{2}\left\|\boldsymbol{w}_t - \boldsymbol{w}_{t-1}\right\|_2^2. \tag{34}$$

Plugging $\boldsymbol{w}_t - \boldsymbol{w}_{t-1} = -\eta_w \hat{g}(\boldsymbol{w})$ into (34) yields that

$$\begin{aligned}
\Phi(\boldsymbol{w}_t) &\leq \Phi(\boldsymbol{w}_{t-1}) - \eta_w\langle \nabla\Phi(\boldsymbol{w}_{t-1}), \hat{g}(\boldsymbol{w})\rangle + \frac{L_w\eta_w^2}{2}\left\|\hat{g}(\boldsymbol{w})\right\|_2^2 \\
&= \Phi(\boldsymbol{w}_{t-1}) - \eta_w\left\|\nabla\Phi(\boldsymbol{w}_{t-1})\right\|_2^2 - \eta_w\langle \nabla\Phi(\boldsymbol{w}_{t-1}), \hat{g}(\boldsymbol{w}) - \nabla\Phi(\boldsymbol{w}_{t-1})\rangle + \frac{L_w\eta_w^2}{2}\left\|\hat{g}(\boldsymbol{w})\right\|_2^2 \\
&\leq \Phi(\boldsymbol{w}_{t-1}) - \eta_w\left\|\nabla\Phi(\boldsymbol{w}_{t-1})\right\|_2^2 - \eta_w\langle \nabla\Phi(\boldsymbol{w}_{t-1}), g(\boldsymbol{w}) - \nabla\Phi(\boldsymbol{w}_{t-1})\rangle + L_w\eta_w^2\left\|g(\boldsymbol{w})\right\|_2^2 \\
&\quad + L_w\eta_w^2\left\|\hat{g}(\boldsymbol{w}) - g(\boldsymbol{w})\right\|_2^2 - \eta_w\langle \nabla\Phi(\boldsymbol{w}_{t-1}), \hat{g}(\boldsymbol{w}) - g(\boldsymbol{w})\rangle \\
&= \Phi(\boldsymbol{w}_{t-1}) - \eta_w\left\|\nabla\Phi(\boldsymbol{w}_{t-1})\right\|_2^2 - \eta_w\langle \nabla\Phi(\boldsymbol{w}_{t-1}), g(\boldsymbol{w}) - \nabla_{\boldsymbol{w}}L(\boldsymbol{w}_{t-1}, \alpha_{t-1})\rangle \\
&\quad - \eta_w\langle \nabla\Phi(\boldsymbol{w}_{t-1}), \nabla_{\boldsymbol{w}}L(\boldsymbol{w}_{t-1}, \alpha_{t-1}) - \nabla\Phi(\boldsymbol{w}_{t-1})\rangle + L_w\eta_w^2\left\|g(\boldsymbol{w})\right\|_2^2 \\
&\quad + L_w\eta_w^2\left\|\hat{g}(\boldsymbol{w}) - g(\boldsymbol{w})\right\|_2^2 - \eta_w\langle \nabla\Phi(\boldsymbol{w}_{t-1}), \hat{g}(\boldsymbol{w}) - g(\boldsymbol{w})\rangle \\
&\leq \Phi(\boldsymbol{w}_{t-1}) - \frac{\eta_w}{2}\left\|\nabla\Phi(\boldsymbol{w}_{t-1})\right\|_2^2 - \eta_w\langle \nabla\Phi(\boldsymbol{w}_{t-1}), g(\boldsymbol{w}) - \nabla_{\boldsymbol{w}}L(\boldsymbol{w}_{t-1}, \alpha_{t-1})\rangle \\
&\quad + \frac{\eta_w}{2}\left\|\nabla\Phi(\boldsymbol{w}_{t-1}) - \nabla_{\boldsymbol{w}}L(\boldsymbol{w}_{t-1}, \alpha_{t-1})\right\|_2^2 + L_w\eta_w^2\left\|g(\boldsymbol{w})\right\|_2^2 + \frac{\delta L_w L_{wx}^2\eta_w^2}{\mu} + \eta_w L_{wx}\ell_w\sqrt{\frac{\delta}{\mu}}.
\end{aligned} \tag{35}$$

By Lem.5, taking an expectation on the both sides yields that,

$$\begin{aligned}
\mathbb{E}\left[\Phi(\boldsymbol{w}_t)\right] \leq \mathbb{E}[\Phi(\boldsymbol{w}_{t-1})] &- \frac{\eta_w}{2}\mathbb{E}\left[\left\|\nabla\Phi(\boldsymbol{w}_{t-1})\right\|_2^2\right] + \frac{\eta_w}{2}\mathbb{E}\left[\left\|\nabla\Phi(\boldsymbol{w}_{t-1}) - \nabla_{\boldsymbol{w}}L(\boldsymbol{w}_{t-1}, \alpha_{t-1})\right\|_2^2\right] \\
&+ L_w\eta_w^2\left\|\nabla_{\boldsymbol{w}}L(\boldsymbol{w}_{t-1}, \alpha_{t-1})\right\|_2^2 + \frac{L_w\eta_w^2\sigma^2}{M} + \frac{\delta L_w L_{wx}^2\eta_w^2}{\mu} + \eta_w L_{wx}\ell_w\sqrt{\frac{\delta}{\mu}}.
\end{aligned} \tag{36}$$

By the Cauchy-Schwartz inequality, we have

$$\left\|\nabla_{\boldsymbol{w}}L(\boldsymbol{w}_{t-1}, \alpha_{t-1})\right\|_2^2 \leq 2\left(\left\|\nabla_{\boldsymbol{w}}L(\boldsymbol{w}_{t-1}, \alpha_{t-1}) - \nabla\Phi(\boldsymbol{w}_{t-1})\right\|_2^2 + \left\|\nabla\Phi(\boldsymbol{w}_{t-1})\right\|_2^2\right). \tag{37}$$

Plugging (37) into (36) and taking the expectation of both side, yields that

$$\begin{aligned}
\mathbb{E}[\Phi(\boldsymbol{w}_t)] \leq \mathbb{E}[\Phi(\boldsymbol{w}_{t-1})] &- \left(\frac{\eta_w}{2} - 2L_w\eta_w^2\right)\mathbb{E}\left[\left\|\nabla\Phi(\boldsymbol{w}_{t-1})\right\|_2^2\right] \\
&+ \left(\frac{\eta_w}{2} + 2L_w\eta_w^2\right)\mathbb{E}\left[\left\|\nabla\Phi(\boldsymbol{w}_{t-1}) - \nabla_{\boldsymbol{w}}L(\boldsymbol{w}_{t-1}, \alpha_{t-1})\right\|_2^2\right] + \frac{L_w\eta_w^2\sigma^2}{M} + \frac{\delta L_w L_{wx}^2\eta_w^2}{\mu} + \eta_w L_{wx}\ell_w\sqrt{\frac{\delta}{\mu}}.
\end{aligned}$$

This completes the proof. □

Lemma 7. *let* $\delta_t = \mathbb{E}\left[\left\|\alpha^*(\boldsymbol{w}_t, x_t^*) - \alpha_t(\boldsymbol{w}_t, \hat{x}_t)\right\|_2^2\right]$, *the following statements holds that*

$$\begin{aligned}
\delta_t \leq \left(1 - \frac{\mu}{2L_{\alpha w}} + \frac{L_{ww}^2 L_{\alpha w}^3\eta_w^2}{2\mu^3}\right)\delta_{t-1} &+ \frac{L_{\alpha w}^3\eta_w^2}{4\mu^3}\mathbb{E}\left[\left\|\nabla\Phi(\boldsymbol{w}_{t-1})\right\|_2^2\right] \\
&+ \frac{2\sigma^2}{L_{\alpha w}^2 M} + \frac{L_{\alpha w}^3\eta_w^2\sigma^2}{4\mu^3 M} + \frac{L_{\alpha w}^3 L_{wx}^2\eta_w^2\sigma^2}{4\mu^4} + \frac{L_{\alpha w}L_{\alpha x}^2\Delta}{8\mu^3} + \frac{2L_{\alpha x}^2\delta}{\mu^3}.
\end{aligned}$$

Proof.

$$\begin{aligned}
\delta_t &= \mathbb{E}\left[\left\|\alpha^*(\boldsymbol{w}_t, x_t^*) - \alpha^*(\boldsymbol{w}_t, \hat{x}_t) + \alpha^*(\boldsymbol{w}_t, \hat{x}_t) - \alpha_t(\boldsymbol{w}_t, \hat{x}_t)\right\|_2^2\right] \\
&\leq 2\mathbb{E}\left[\left\|\alpha^*(\boldsymbol{w}_t, x_t^*) - \alpha^*(\boldsymbol{w}_t, \hat{x}_t)\right\|_2^2\right] + 2\mathbb{E}\left[\left\|\alpha^*(\boldsymbol{w}_t, \hat{x}_t) - \alpha_t(\boldsymbol{w}_t, \hat{x}_t)\right\|_2^2\right].
\end{aligned} \tag{38}$$

By Young's inequality, for any $\epsilon_0 \leq 0$, we have

$$\begin{aligned}\mathbb{E}\left[\|\alpha^*(\boldsymbol{w}_t,\hat{x}_t)-\alpha_t(\boldsymbol{w}_t,\hat{x}_t)\|_2^2\right] \leq \left(1+\frac{1}{\epsilon_0}\right)\mathbb{E}\left[\|\alpha^*(\boldsymbol{w}_{t-1},\hat{x}_{t-1})-\alpha_t(\boldsymbol{w}_t,\hat{x}_t)\|_2^2\right] \\ +(1+\epsilon_0)\,\mathbb{E}\left[\|\alpha^*(\boldsymbol{w}_t,\hat{x}_t)-\alpha^*(\boldsymbol{w}_{t-1},\hat{x}_{t-1})\|_2^2\right].\end{aligned} \tag{39}$$

By the Rem.1 and $\eta_\alpha = 1/L_{\alpha w}$ (refer to (Nesterov, 1998) Theorem 2.3.4), we have

$$\mathbb{E}\left[\|\alpha^*(\boldsymbol{w}_{t-1},\hat{x}_t)-\alpha_t(\boldsymbol{w}_t,\hat{x}_t)\|_2^2\right] \leq \left(1-\frac{\mu}{L_{\alpha w}}\right)\mathbb{E}\left[\|\alpha^*(\boldsymbol{w}_{t-1},\hat{x}_{t-1})-\alpha_{t-1}(\boldsymbol{w}_{t-1},\hat{x}_{t-1})\|_2^2\right]+\frac{\sigma^2}{L_{\alpha w}^2 M}. \tag{40}$$

By triangle inequality, we have

$$\begin{aligned}\mathbb{E}\left[\|\alpha^*(\boldsymbol{w}_{t-1},\hat{x}_{t-1})-\alpha_{t-1}(\boldsymbol{w}_{t-1},\hat{x}_{t-1})\|_2^2\right] \leq & 2\mathbb{E}\left[\left\|\alpha^*(\boldsymbol{w}_{t-1},\hat{x}_{t-1})-\alpha^*(\boldsymbol{w}_{t-1},x_{t-1}^*)\right\|_2^2\right] \\ & +2\mathbb{E}\left[\left\|\alpha^*(\boldsymbol{w}_{t-1},x_{t-1}^*)-\alpha_{t-1}(\boldsymbol{w}_{t-1},\hat{x}_{t-1})\right\|_2^2\right].\end{aligned} \tag{41}$$

Plugging (41) into (40), yields that

$$\mathbb{E}\left[\|\alpha^*(\boldsymbol{w}_{t-1},\hat{x}_{t-1})-\alpha_t(\boldsymbol{w}_t,\hat{x}_t)\|_2^2\right] \leq 2\left(1-\frac{\mu}{L_{\alpha w}}\right)\left[\delta_{t-1}+\mathbb{E}\left[\left\|\alpha^*(\boldsymbol{w}_{t-1},\hat{x}_{t-1})-\alpha^*(\boldsymbol{w}_{t-1},x_{t-1}^*)\right\|_2^2\right]\right]+\frac{\sigma^2}{L_{\alpha w}^2 M}. \tag{42}$$

By triangle inequality, we have

$$\begin{aligned}\mathbb{E}\left[\|\alpha^*(\boldsymbol{w}_t,\hat{x}_t)-\alpha^*(\boldsymbol{w}_{t-1},\hat{x}_{t-1})\|_2^2\right] \leq & 2\mathbb{E}\left[\|\alpha^*(\boldsymbol{w}_t,\hat{x}_t)-\alpha^*(\boldsymbol{w}_{t-1},\hat{x}_t)\|_2^2\right] \\ & +2\mathbb{E}\left[\|\alpha^*(\boldsymbol{w}_{t-1},\hat{x}_t)-\alpha^*(\boldsymbol{w}_{t-1},\hat{x}_{t-1})\|_2^2\right].\end{aligned} \tag{43}$$

Since $\alpha^*(\cdot)$ is $\frac{L_{\alpha w}}{\mu}$-Lipschitz, we have

$$\mathbb{E}\left[\|\alpha^*(\boldsymbol{w}_t,\hat{x}_t)-\alpha^*(\boldsymbol{w}_{t-1},\hat{x}_t)\|_2^2\right] \leq \frac{L_{\alpha w}^2}{\mu^2}\mathbb{E}\left[\|\boldsymbol{w}_t-\boldsymbol{w}_{t-1}\|_2^2\right]. \tag{44}$$

Furthermore, we have

$$\begin{aligned}\mathbb{E}\left[\|\boldsymbol{w}_t-\boldsymbol{w}_{t-1}\|_2^2\right] &= \eta_w^2\mathbb{E}\left[\|\hat{g}(\boldsymbol{w})\|_2^2\right] \\ &= 2\eta_w^2\mathbb{E}\left[\|g(\boldsymbol{w})\|_2^2\right]+2\eta_w^2\mathbb{E}\left[\|\hat{g}(\boldsymbol{w})-g(\boldsymbol{w})\|_2^2\right] \\ &\leq 2\eta_w^2\left\|\nabla_{\boldsymbol{w}}L(\boldsymbol{w}_{t-1},\alpha_{t-1})\right\|_2^2+\frac{2\eta_w^2\sigma^2}{M}+\frac{2\delta L_{wx}^2\eta_w^2}{\mu} \\ &\overset{(37)}{\leq} 4\eta_w^2\left(\|\nabla_{\boldsymbol{w}}L(\boldsymbol{w}_{t-1},\alpha_{t-1})-\nabla\Phi(\boldsymbol{w}_{t-1})\|_2^2+\|\nabla\Phi(\boldsymbol{w}_{t-1})\|_2^2\right)+\frac{2\eta_w^2\sigma^2}{M}+\frac{2\delta L_{wx}^2\eta_w^2}{\mu} \\ &\leq 4L_{ww}^2\eta_w^2\delta_{t-1}+4\eta_w^2\left\|\nabla\Phi(\boldsymbol{w}_{t-1})\right\|_2^2+\frac{2\eta_w^2\sigma^2}{M}+\frac{2\delta L_{wx}^2\eta_w^2}{\mu}.\end{aligned} \tag{45}$$

Plugging (44) and (45) into (43) and taking the exception of both side, yields that

$$\begin{aligned}\mathbb{E}\left[\|\alpha^*(\boldsymbol{w}_t,\hat{x}_t)-\alpha^*(\boldsymbol{w}_{t-1},\hat{x}_{t-1})\|_2^2\right] \leq & \frac{8L_{\alpha w}^2L_{ww}^2\eta_w^2}{\mu^2}\delta_{t-1}+\frac{8L_{\alpha w}^2\eta_w^2}{\mu^2}\mathbb{E}\left[\|\nabla\Phi(\boldsymbol{w}_{t-1})\|_2^2\right]+\frac{4L_{\alpha w}^2\eta_w^2\sigma^2}{\mu^2 M} \\ & +\frac{4L_{\alpha w}^2L_{wx}^2\eta_w^2\delta}{\mu^3}+2\mathbb{E}\left[\|\alpha^*(\boldsymbol{w}_{t-1},\hat{x}_t)-\alpha^*(\boldsymbol{w}_{t-1},\hat{x}_{t-1})\|_2^2\right].\end{aligned} \tag{46}$$

Then, plugging (42) and (46) into (39) and let $\epsilon_0 = \frac{8(\kappa-1)}{7-6\kappa}$ and $\kappa \leq \frac{7}{6}$, yields that

$$\begin{aligned}\mathbb{E}\left[\|\alpha^*(\boldsymbol{w}_t, \hat{x}_t) - \alpha_t(\boldsymbol{w}_t, \hat{x}_t)\|_2^2\right] \leq & \left(\frac{1}{2} - \frac{\mu}{4L_{\alpha w}} + \frac{L_{ww}^2 L_{\alpha w}^3 \eta_w^2}{4\mu^3}\right)\delta_{t-1} + \frac{L_{\alpha w}^3 \eta_w^2}{4\mu^3}\mathbb{E}\left[\|\nabla\Phi(\boldsymbol{w}_{t-1})\|_2^2\right] \\ & + \frac{\sigma^2}{L_{\alpha w}^2 M} + \frac{L_{\alpha w}^3 \eta_w^2 \sigma^2}{8\mu^3 M} + \frac{L_{\alpha w}^3 L_{wx}^2 \eta_w^2 \delta}{8\mu^4} \\ & + \frac{L_{\alpha w}}{16\mu}\mathbb{E}\left[\|\alpha^*(\boldsymbol{w}_{t-1}, \hat{x}_t) - \alpha^*(\boldsymbol{w}_{t-1}, \hat{x}_{t-1})\|_2^2\right].\end{aligned} \tag{47}$$

Since $\alpha^*(\cdot)$ is $\frac{L_{\alpha x}}{\mu}$-Lipschitz, we have

$$\mathbb{E}\left[\|\alpha^*(\boldsymbol{w}_{t-1}, \hat{x}_t) - \alpha^*(\boldsymbol{w}_{t-1}, \hat{x}_{t-1})\|_2^2\right] \leq \frac{L_{\alpha x}^2}{\mu^2}\mathbb{E}\left[\|\hat{x}_t - \hat{x}_{t-1}\|_2^2\right] \leq \frac{L_{\alpha x}^2}{\mu^2}\Delta, \tag{48}$$

$$\mathbb{E}\left[\|\alpha^*(\boldsymbol{w}_t, x_t^*) - \alpha^*(\boldsymbol{w}_t, \hat{x}_t)\|_2^2\right] \leq \frac{L_{\alpha x}^2}{\mu^2}\mathbb{E}\left[\|x_t^* - \hat{x}_t\|_2^2\right] \leq \frac{L_{\alpha x}^2 \delta}{\mu^3}, \tag{49}$$

where Δ is the maximum distance between two adversarial samples on the same sample. We can get the value of Δ by the diameter if $\mathcal{X}_i$.

Plugging (46) (47) and (48) into (38), yields that

$$\begin{aligned}\delta_t \leq & \left(1 - \frac{\mu}{2L_{\alpha w}} + \frac{L_{ww}^2 L_{\alpha w}^3 \eta_w^2}{2\mu^3}\right)\delta_{t-1} + \frac{L_{\alpha w}^3 \eta_w^2}{4\mu^3}\mathbb{E}\left[\|\nabla\Phi(\boldsymbol{w}_{t-1})\|_2^2\right] \\ & + \frac{2\sigma^2}{L_{\alpha w}^2 M} + \frac{L_{\alpha w}^3 \eta_w^2 \sigma^2}{4\mu^3 M} + \frac{L_{\alpha w}^3 L_{wx}^2 \eta_w^2 \delta}{4\mu^4} + \frac{L_{\alpha w} L_{\alpha x}^2 \Delta}{8\mu^3} + \frac{2L_{\alpha x}^2 \delta}{\mu^3}.\end{aligned}$$

□

Lemma 8. *let $\delta_t = \mathbb{E}[\|\alpha^*(\boldsymbol{w}_t, x_t^*) - \alpha_t(\boldsymbol{w}_t, \hat{x}_t)\|_2^2]$, the following statements holds that*

$$\mathbb{E}[\Phi(\boldsymbol{w}_t)] \leq \mathbb{E}\left[\Phi(\boldsymbol{w}_{t-1})\right] - \frac{\eta_w}{4}\mathbb{E}\left[\|\nabla\Phi(\boldsymbol{w}_{t-1})\|_2^2\right] + \frac{3\eta_w L^2 \delta_{t-1}}{4} + \frac{L_w \eta_w^2 \sigma^2}{M} + \frac{\delta L_w L^2 \eta_w^2}{\mu} + \eta_w L \ell_w \sqrt{\frac{\delta}{\mu}}.$$

Proof. Let $\eta_w = \frac{1}{16(\kappa+1)^2 L}$, where L is the maximum value in the set $\{L_{ww}, L_{w\alpha}, L_{\alpha w}, L_{xw}, L_{wx}, L_{x\alpha}, L_{\alpha x}\}$, and $\kappa = \frac{L}{\mu}$, hence

$$\frac{1}{4}\eta_w \leq \frac{\eta_w}{2} - 2L_w \eta_w^2 \leq \frac{\eta_w}{2} + 2L_w \eta_w^2 \leq \frac{3}{4}\eta_w. \tag{50}$$

Then we have

$$\mathbb{E}\left[\|\nabla\Phi(\boldsymbol{w}_{t-1}) - \nabla_{\boldsymbol{w}} L(\boldsymbol{w}_{t-1}, \alpha_{t-1})\|_2^2\right] \leq L^2 \delta_{t-1}. \tag{51}$$

Combining (50) (51) and Lem.6 yields that

$$\mathbb{E}[\Phi(\boldsymbol{w}_t)] \leq \mathbb{E}\left[\Phi(\boldsymbol{w}_{t-1})\right] - \frac{\eta_w}{4}\mathbb{E}\left[\|\nabla\Phi(\boldsymbol{w}_{t-1})\|_2^2\right] + \frac{3\eta_w L^2 \delta_{t-1}}{4} + \frac{L_w \eta_w^2 \sigma^2}{M} + \frac{\delta L_w L^2 \eta_w^2}{\mu} + \eta_w L^2 \sqrt{\frac{\delta}{\mu}}.$$

□

A.4. Proof of Theorem 1

Proof. Throughout this subsection, we define $\gamma = 1 - \frac{1}{2\kappa} + \frac{L^2\kappa^3\eta_w^2}{2}$. Since $\delta_0 \leq D^2$, where $D = |\Omega_\alpha|$,we have

$$\begin{aligned}\delta_t \leq & \gamma^t D^2 + \frac{\kappa^3\eta_w^2}{4}\left(\sum_{j=0}^{t-1}\gamma^{t-1-j}\mathbb{E}\left[\|\nabla\Phi(\boldsymbol{w}_{t-1})\|_2^2\right]\right) \\ & + \left(\frac{2\sigma^2}{L^2M} + \frac{\kappa^3\eta_w^2\sigma^2}{4M} + \frac{L\kappa^4\eta_w^2\delta}{4} + \frac{2\kappa^2\delta}{\mu} + \frac{\kappa^3\Delta}{8}\right)\left(\sum_{j=0}^{t-1}\gamma^{t-1-j}\right).\end{aligned} \tag{52}$$

Combining (52) and Lem.8 yields that

$$\begin{aligned}\mathbb{E}[\Phi(\boldsymbol{w}_t)] \leq & \mathbb{E}\left[\Phi(\boldsymbol{w}_{t-1})\right] - \frac{\eta_w}{4}\mathbb{E}\left[\|\nabla\Phi(\boldsymbol{w}_{t-1})\|_2^2\right] + \frac{L_w\eta_w^2\sigma^2}{M} + \frac{\delta L_w L^2\eta_w^2}{\mu} + \eta_w L^2\sqrt{\frac{\delta}{\mu}} \\ & + \frac{3\eta_w L^2\gamma^t D^2}{4} + \frac{3L^2\kappa^3\eta_w^3}{16}\left(\sum_{j=0}^{t-2}\gamma^{t-2-j}\mathbb{E}\left[\|\nabla\Phi(\boldsymbol{w}_j)\|_2^2\right]\right) \\ & + \frac{3\eta_w L^2}{4}\left(\frac{2\sigma^2}{L^2M} + \frac{\kappa^3\eta_w^2\sigma^2}{4M} + \frac{L\kappa^4\eta_w^2\delta}{4} + \frac{2\kappa^2\delta}{\mu} + \frac{\kappa^3\Delta}{8}\right)\left(\sum_{j=0}^{t-2}\gamma^{t-2-j}\right)\end{aligned} \tag{53}$$

Summing up (53) over $t = 1, 2, \cdots, T+1$ and rearranging the terms yields that

$$\begin{aligned}\mathbb{E}[\Phi(\boldsymbol{w}_{T+1})] \leq & \mathbb{E}\left[\Phi(\boldsymbol{w}_0)\right] - \frac{\eta_w}{4}\sum_{t=0}^{T}\mathbb{E}\left[\|\nabla\Phi(\boldsymbol{w}_t)\|_2^2\right] + (T+1)\left(\frac{L_w\eta_w^2\sigma^2}{M} + \frac{\delta L_w L^2\eta_w^2}{\mu} + \eta_w L^2\sqrt{\frac{\delta}{\mu}}\right) \\ & + \frac{3\eta_w L^2 D^2}{4}\sum_{t=0}^{T}\gamma^t + \frac{3L^2\kappa^3\eta_w^3}{16}\left(\sum_{t=1}^{T+1}\sum_{j=0}^{t-2}\gamma^{t-2-j}\mathbb{E}\left[\|\nabla\Phi(\boldsymbol{w}_j)\|_2^2\right]\right) \\ & + \frac{3\eta_w L^2}{4}\left(\frac{2\sigma^2}{L^2M} + \frac{\kappa^3\eta_w^2\sigma^2}{4M} + \frac{L\kappa^4\eta_w^2\delta}{4} + \frac{2\kappa^2\delta}{\mu} + \frac{\kappa^3\Delta}{8}\right)\left(\sum_{t=1}^{T+1}\sum_{j=0}^{t-2}\gamma^{t-2-j}\right)\end{aligned} \tag{54}$$

Since $\eta_w = \frac{1}{16(\kappa+1)^2L}$, we have $\gamma \leq 1 - \frac{1}{4\kappa}$ and $\frac{3L^2\kappa^3\eta_w^3}{16} \leq \frac{3\eta_w}{4096\kappa}$. This implies that $\sum_{t=0}^{T}\gamma^t \leq 4\kappa$ and

$$\begin{aligned}\sum_{t=1}^{T+1}\sum_{j=0}^{t-2}\gamma^{t-2-j}\mathbb{E}\left[\|\nabla\Phi(\boldsymbol{w}_j)\|_2^2\right] &\leq 4\kappa\sum_{t=0}^{T}\mathbb{E}\left[\|\nabla\Phi(\boldsymbol{w}_t)\|_2^2\right] \\ \sum_{t=1}^{T+1}\sum_{j=0}^{t-2}\gamma^{t-2-j} &\leq 4\kappa(T+1)\end{aligned} \tag{55}$$

Putting these pieces together yields that

$$\begin{aligned}\mathbb{E}[\Phi(\boldsymbol{w}_{T+1})] \leq & \mathbb{E}\left[\Phi(\boldsymbol{w}_0)\right] - \frac{253\eta_w}{1024}\sum_{t=0}^{T}\mathbb{E}\left[\|\nabla\Phi(\boldsymbol{w}_t)\|_2^2\right] + \eta_w(T+1)\left(\frac{\sigma^2}{8\kappa M} + \frac{L\delta}{8} + L^2\sqrt{\frac{\delta}{\mu}}\right) \\ & + 3\eta_w\kappa L^2D^2 + \eta_w(T+1)\left(\frac{6\sigma^2\kappa}{M} + \frac{3\sigma^2}{1024M} + \frac{3\kappa L\delta}{1024} + 6\kappa^4L\delta + \frac{3\kappa^4L^2\Delta}{8}\right)\end{aligned} \tag{56}$$

Futhermore, we have

$$\frac{\sigma^2}{8\kappa M} + \frac{6\sigma^2\kappa}{M} + \frac{3\sigma^2}{1024M} \leq \frac{6403\kappa\sigma^2}{1024M} \tag{57}$$

By definition of Δ_Φ and plugging (57) into (56), we have

$$\begin{aligned}\frac{1}{T+1}\left(\sum_{t=0}^{T}\mathbb{E}\left[\|\nabla\Phi(\boldsymbol{w}_t)\|_2^2\right]\right) &\leq \frac{1024\Delta_\Phi}{253\eta_w(T+1)}+\frac{3072L^2D^2\kappa}{253(T+1)}+\frac{6403\kappa\sigma^2}{253M}\\ &\quad+\frac{1024}{253}\left(\frac{3\kappa L\delta}{1024}+6\kappa^4L\delta+\frac{L\delta}{8}+L^2\sqrt{\frac{\delta}{\mu}}\right)+\frac{384\kappa^4L^2\Delta}{253}\\ &\leq \frac{5\Delta_\Phi}{\eta_w(T+1)}+\frac{13\kappa L^2D^2}{T+1}+\frac{26\kappa\sigma^2}{M}+h_\delta+h_\Delta\\ &\leq \frac{360\kappa^2L\Delta_\Phi+13\kappa L^2D^2}{T+1}+\frac{26\kappa\sigma^2}{M}+h_\delta+h_\Delta\end{aligned} \tag{58}$$

where $h_\delta=\frac{1024}{253}\left(\frac{3\kappa L\delta}{1024}+6\kappa^4L\delta+\frac{L\delta}{8}+L^2\sqrt{\frac{\delta}{\mu}}\right)$ and $h_\Delta=\frac{384\kappa^4L^2\Delta}{253}$ and $\Delta_\Phi=\Phi(\boldsymbol{w}^0)-\min_w\Phi(\boldsymbol{w})$. On the right side of (58), the second term $\frac{26\kappa\sigma^2}{M}$ is due to random sampling, the third term h_δ is due to the use of an approximate solution $\hat{\boldsymbol{x}}$ to the inner maximization problem instead of the optimal solution $\boldsymbol{x}^*$, and the fourth term h_Δ is due to $\mathcal{X}$ being a bounded set. And their values are very small.

□

B. Adversarial Attacks

Fast Gradient Sign Method (FGSM) (Goodfellow et al., 2015) is a single-step attack that generates adversarial examples through a permutation along the gradient of the loss function with respect to the clean image feature vector as:

$$\boldsymbol{x}=\boldsymbol{x}+\epsilon\cdot sign\left(\nabla_{\boldsymbol{x}}\ell(h_\theta(\boldsymbol{x},y))\right) \tag{59}$$

Projected Gradient Descent (PGD) (Madry et al., 2018) starts from an initialization point that is uniformly sampled from the allowed ϵ-ball centered at the clean image, and it expends FSGM by iteratively applying multiple small steps of permutation updating with respect to the current gradient as:

$$\boldsymbol{x}^k=Proj\left(\boldsymbol{x}^{k-1}+\beta\cdot sign\left(\nabla_{\boldsymbol{x}}\ell(h_\theta(\boldsymbol{x}^{k-1},y))\right)\right) \tag{60}$$

Carlini & **Wagner (C&W)** (Carlini & Wagner, 2017) is another powerful attack based on optimization, where an auxiliary variable ω is induced and an adversarial example constrained by l_2 norm is represented by $x'=\frac{1}{2}(\tanh\omega+1)$. It can be optimized by:

$$\arg\min_{\omega}\left\{c\cdot f(\boldsymbol{x}')+\|\boldsymbol{x}'-\boldsymbol{x}\|_2^2\right\} \tag{61}$$

where

$$f(\boldsymbol{x})=\max\left(\max_{i\neq y}Z(\boldsymbol{x}'-Z(\boldsymbol{x}')_y),-\kappa\right) \tag{62}$$

and here κ controls the confidence of the adversarial examples. It can also be extended to other l_∞.

Auto Attack (AA) (Croce & Hein, 2020) is a combination of multiple attacks that forms a parameter-free and computationally affordable ensemble of attacks to evaluate adversarial robustness. The standard attacks include four selected attacks: APGD, targeted version of APGD-DLR and FAB, and Square Attack.

C. Additional Results

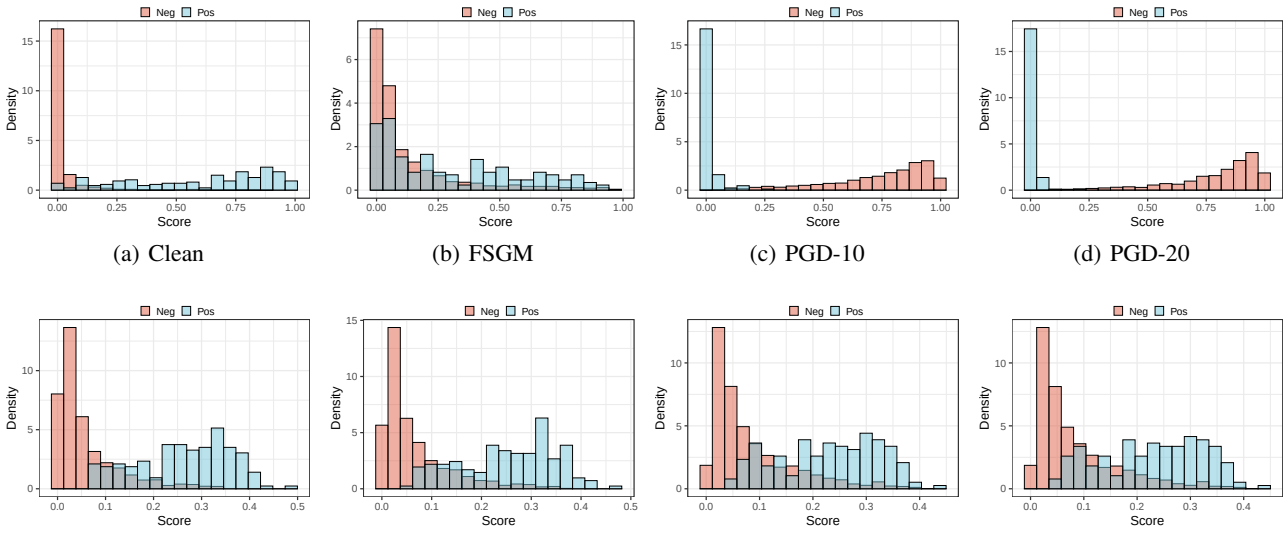
In this section, we provide additional results to further support the conclusions in the main text.

C.1. Experiment Setting

The adversarial training is applied with the maximal permutation ϵ of $8/255$ and a step size of $2/255$. The max number of iterations K is set as 10. For CE, we use SGD momentum optimizer, while for ours, we use SGDA momentum optimizer.

The initial learning rate η_w is set as 0.01 with decay 5×10^{-4}, and the batch size is 128. And the initial learning rate η_α is set as 0.1. In the training process, we adopt a learning rate step decay schedules, which cut the learning rate by a constant factor 0.001 every 30 constant number of epochs for all methods.

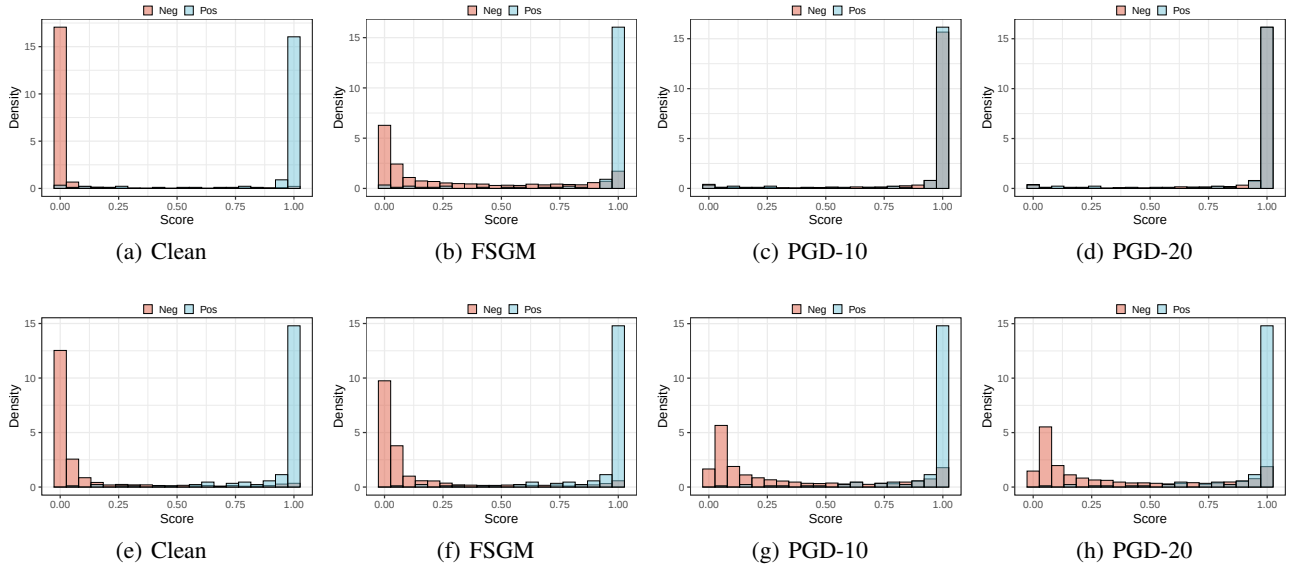
C.2. Score Distribution



(a) Clean (b) FSGM (c) PGD-10 (d) PGD-20

(e) Clean (f) FSGM (g) PGD-10 (h) PGD-20

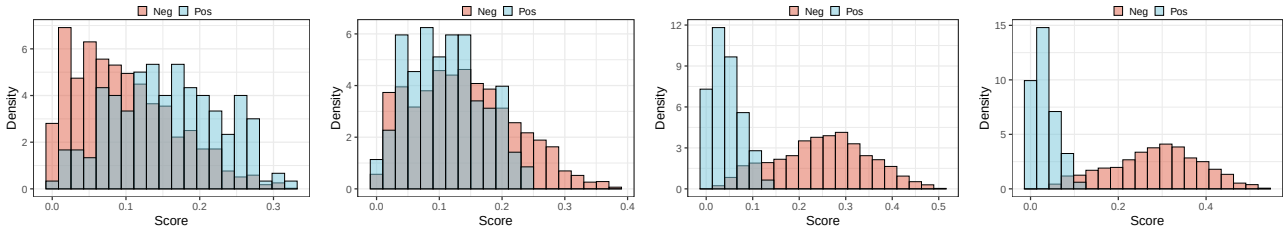
Figure 5. Distribution of positive and negative example scores of CE on MNIST-LT dataset. The first row represents the score distribution against different attacks under Natural Training, and the second row represents the score distribution under Adversarial Training.



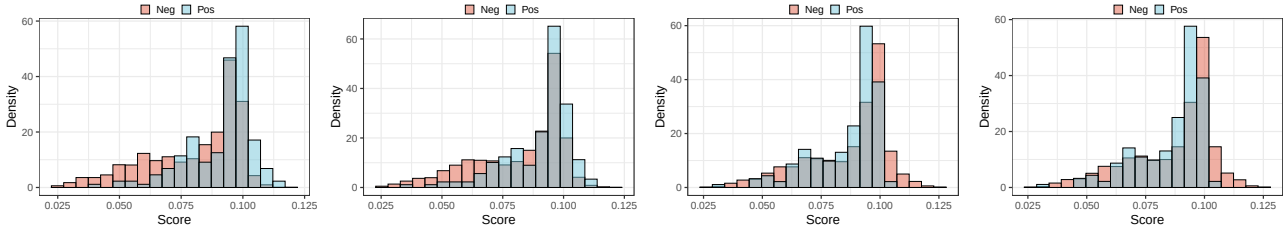
(a) Clean (b) FSGM (c) PGD-10 (d) PGD-20

(e) Clean (f) FSGM (g) PGD-10 (h) PGD-20

Figure 6. Distribution of positive and negative example scores of our proposed AdAUC on MNIST-LT dataset. The first row represents the score distribution against different attacks under Natural Training, and the second row represents the score distribution under Adversarial Training.

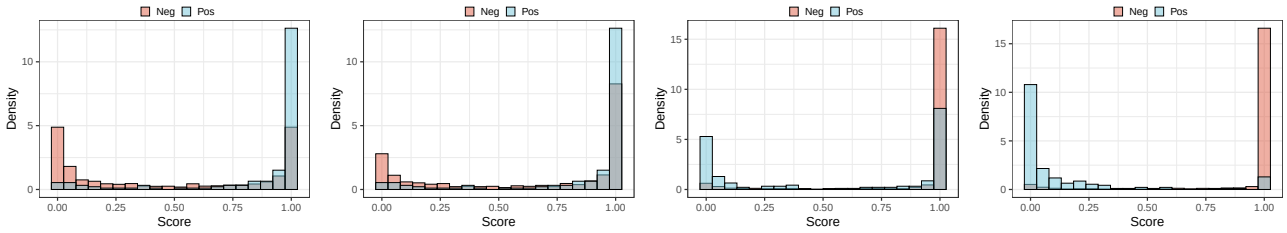


(a) Clean (b) FSGM (c) PGD-10 (d) PGD-20

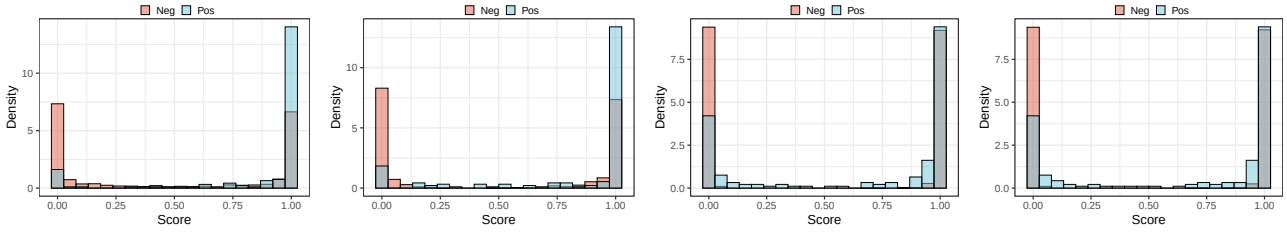


(e) Clean (f) FSGM (g) PGD-10 (h) PGD-20

Figure 7. Distribution of positive and negative example scores of CE on CIFAR-10-LT dataset. The first row represents the score distribution against different attacks under Natural Training, and the second row represents the score distribution under Adversarial Training.

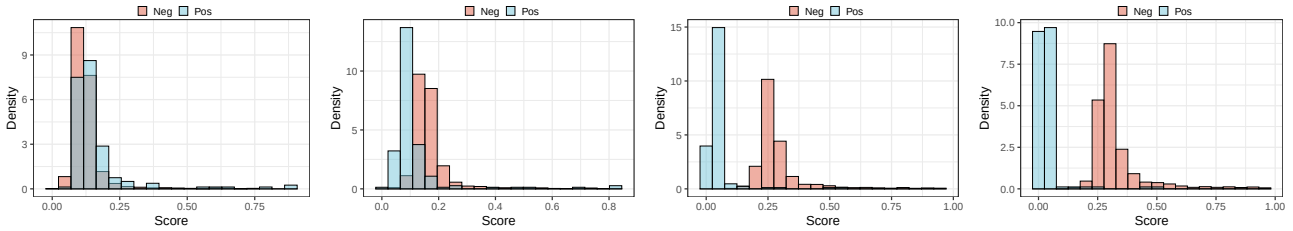


(a) Clean (b) FSGM (c) PGD-10 (d) PGD-20

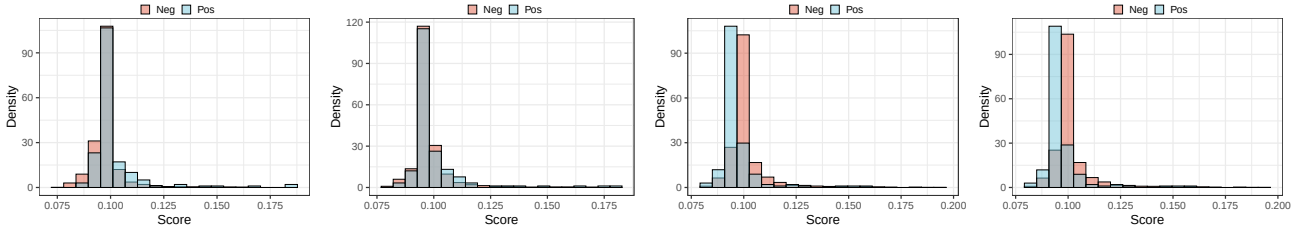


(e) Clean (f) FSGM (g) PGD-10 (h) PGD-20

Figure 8. Distribution of positive and negative example scores of AdAUC on CIFAR-10-LT dataset. The first row represents the score distribution against different attacks under Natural Training, and the second row represents the score distribution under Adversarial Training.

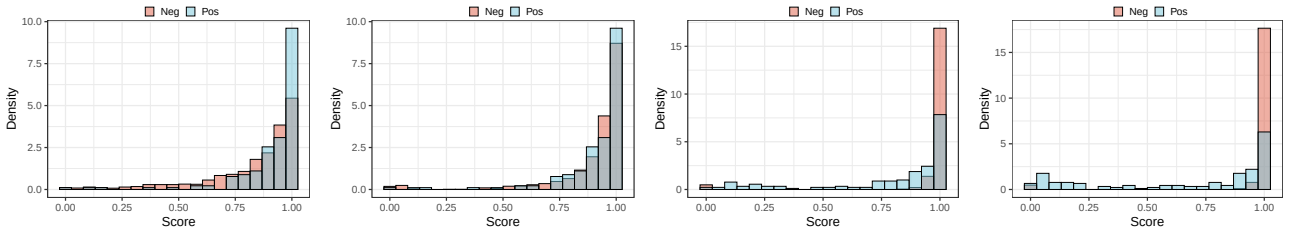


(a) Clean (b) FSGM (c) PGD-10 (d) PGD-20

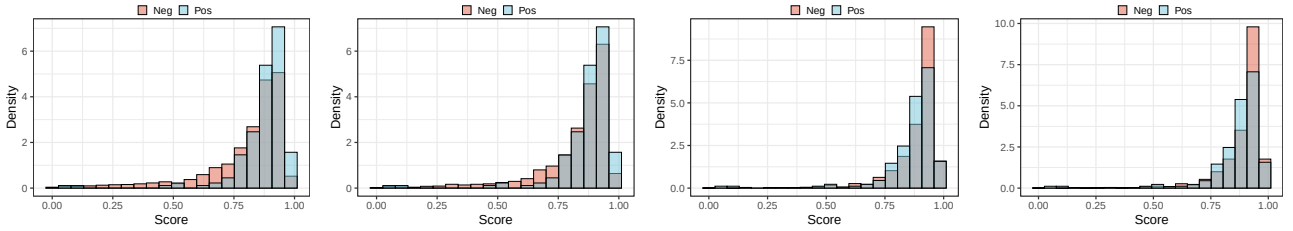


(e) Clean (f) FSGM (g) PGD-10 (h) PGD-20

Figure 9. Distribution of positive and negative example scores of CE on CIFAR-100-LT dataset. The first row represents the score distribution against different attacks under Natural Training, and the second row represents the score distribution under Adversarial Training.



(a) Clean (b) FSGM (c) PGD-10 (d) PGD-20



(e) Clean (f) FSGM (g) PGD-10 (h) PGD-20

Figure 10. Distribution of positive and negative example scores of AdAUC on CIFAR-100-LT dataset. The first row represents the score distribution against different attacks under Natural Training, and the second row represents the score distribution under Adversarial Training.