Entropic Regularization for Adversarial Robust Learning

Jie Wang, Yifan Lin, Song Wei, Rui Gao and Yao Xie*

Despite the growing prevalence of artificial neural networks in real-world applications, their vulnerability to adversarial attacks remains to be a significant concern, which motivates us to investigate the robustness of machine learning models. While various heuristics aim to optimize the distributionally robust risk using the ∞-Wasserstein metric, such a notion of robustness frequently encounters computation intractability. To tackle the computational challenge, we develop a novel approach to adversarial training that integrates entropic regularization into the distributionally robust risk function. This regularization brings a notable improvement in computation compared with the original formulation. We develop stochastic gradient methods with near-optimal sample complexity to solve this problem efficiently. Moreover, we establish the regularization effects and demonstrate this formulation is asymptotic equivalence to a regularized empirical risk minimization (ERM) framework, by considering various scaling regimes of the entropic regularization η and robustness level ρ. These regimes yield gradient norm regularization, variance regularization, or a smoothed gradient norm regularization that interpolates between these extremes. We numerically validate our proposed method in supervised learning and reinforcement learning applications and showcase its state-of-the-art performance against various adversarial attacks.

1. Introduction

Machine learning models are highly vulnerable to potential *adversarial attack* on their input data, which intends to cause wrong outputs. Even if the adversarial input is slightly different from the clean input drawn from the data distribution, these machine learning models can make a wrong decision. Goodfellow et al. [5] provided an example that, after adding a tiny adversarial noise to an image, a well-trained classification model may make a wrong prediction, even when such data perturbations are imperceptible to visual eyes.

Given that modern machine learning models have been applied in many safety-critical tasks, such as autonomous driving, medical diagnosis, security systems, *etc*, improving the resilience of these models against adversarial attacks in such contexts is of great importance. Neglecting to do so could be risky or unethical and may result in severe consequences. For example, if we use machine learning models in self-driving cars, adversarial examples could allow an attacker to cause the car to take unwanted actions.

Adversarial training is a process of training machine learning model to make it more robust to potential adversarial attacks. To be precise, it aims to optimize the following robust optimization-based formulation, called *adversarial risk minimization*:

$$\min_{\theta\in\Theta}\ \Big\{\mathbb{E}_{x\sim\widehat{\mathbb{P}}}\big[R_\rho(\theta;x)\big]\Big\},\quad \text{where } R_\rho(\theta;x)\triangleq\sup_{z\in\mathbb{B}_\rho(x)} f_\theta(z). \tag{1}$$

Here $\widehat{\mathbb{P}}$ represents the observed distribution on data, θ represents the machine learning model, $f_\theta(z)$ is a loss function, and the uncertainty set is defined as $\mathbb{B}_\rho(x)\triangleq\{z\in\mathcal{Z}:\ \|z-x\|\le\rho\}$ for some norm function $\|\cdot\|$ and some radius $\rho>0$. In other words, this formulation seeks to train a machine learning model based on adversarial perturbations of data, where the adversarial perturbations can be found by considering all possible inputs around the data with radius ρ and picking the one that

* J. Wang, Y. Lin, S. Wei, and Y. Xie are with H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology. R. Gao is with Department of Information, Risk, and Operations Management, University of Texas at Austin.

yields the worst-case loss. Unfortunately, problem (1) is typically intractable to solve because the inner supremum objective function is in general nonconcave in z. As pointed out by [13], solving the inner supremum problem in (1) with deep neural network loss functions is NP-hard. Several heuristic algorithms [5, 9, 11, 3, 10, 15] have been proposed to approximately find the optimal solution of (1), but they lack of global convergence guarantees and it remains an open question whether they can accurately and efficiently find the adversarial perturbations of data.

In this paper, we propose a new approach for adversarial risk minimization based on entropic regularization for distributionally robust optimization (DRO). Here we briefly review our approach. By [4, Lemma EC.2], problem (1) can be viewed as the dual reformulation of the following DRO problem:

$$\min_{\theta\in\Theta}\ \left\{\sup_{\mathbb{P}}\ \left\{\mathbb{E}_{z\sim\mathbb{P}}[f_\theta(z)]:\ \mathcal{W}_\infty(\mathbb{P},\widehat{\mathbb{P}})\le\rho\right\}\right\},\tag{∞-WDRO}$$

where $\mathcal{W}_\infty(\cdot,\cdot)$ is the ∞-Wasserstein metric defined as

$$\mathcal{W}_\infty(\mathbb{P},\mathbb{Q})=\inf_{\gamma}\ \left\{\text{ess.sup}\ \|\zeta_1-\zeta_2\|:\ \begin{matrix}\gamma\text{ is a joint distribution of }\zeta_1\text{ and }\zeta_2\\ \text{with marginals }\mathbb{P}\text{ and }\mathbb{Q}\text{, respectively}\end{matrix}\right\}.$$

It is also convenient to introduce the optimal transport mapping γ to re-write problem (∞-WDRO) as

$$\min_{\theta\in\Theta}\ \left\{\sup_{\mathbb{P},\gamma}\ \left\{\mathbb{E}_{\mathbb{P}}[f_\theta(z)]:\ \begin{matrix}\text{Proj}_{1\#\gamma}=\widehat{\mathbb{P}},\text{Proj}_{2\#\gamma}=\mathbb{P}\\ \text{ess.sup}_\gamma\|\zeta_1-\zeta_2\|\le\rho\end{matrix}\right\}\right\}.\tag{2}$$

As long as the loss $f_\theta(z)$ is nonconcave in z, such as neural networks and other complex machine learning models, problem (2) is intractable for arbitrary radius $\rho>0$. Instead, we add *entropic regularization* to the objective in (2), and focus on solving the following formulation:

$$\min_{\theta\in\Theta}\ \left\{\sup_{\mathbb{P},\gamma}\ \left\{\mathbb{E}_{z\sim\mathbb{P}}\left[f_\theta(z)\right]-\eta H(\gamma\mid\pi):\ \begin{matrix}\text{Proj}_{1\#\gamma}=\widehat{\mathbb{P}},\text{Proj}_{2\#\gamma}=\mathbb{P}\\ \text{ess.sup}_\gamma\|\zeta_1-\zeta_2\|\le\rho\end{matrix}\right\}\right\},\tag{E-∞-WDRO}$$

where π is the refenence measure satisfying $\mathrm{d}\pi(x,z)=\mathrm{d}\widehat{\mathbb{P}}(x)\,\mathrm{d}\nu_x(z)$, with ν_x being the Lebesgue measure on $\mathbb{B}_\rho(x)$, and $H(\gamma\mid\pi)$ is the relative entropy of γ with respect to π, e.g, $H(\gamma\mid\pi)=\mathbb{E}_{(x,z)\sim\gamma}\left[\log\left(\frac{\mathrm{d}\gamma(x,z)}{\mathrm{d}\pi(x,z)}\right)\right]$. We demonstrate several notable features of the formulation (E-∞-WDRO), which are summarized below:

(I) Based on our duality result in Theorem 1, problem (E-∞-WDRO) admits the strong dual reformulation:

$$\min_{\theta\in\Theta}\quad \mathbb{E}_{x\sim\widehat{\mathbb{P}}}[\phi_\eta(x)],\tag{3a}$$

$$\text{where}\quad \phi_\eta(x)=\eta\log\mathbb{E}_{z\sim\nu_x}\left[\exp\left(\frac{f_\theta(z)}{\eta}\right)\right].\tag{3b}$$

Compared with the original formulation (1), we replace the worst-case loss $R_\rho(\theta;x)=\sup_{z\in\mathbb{B}_\rho(x)}f_\theta(z)$ by $\phi_\eta(x)$. Based on the well-known Laplace's method [2], it can be shown that $\phi_\eta(x)$ is a smooth approximation of the optimal value $R_\rho(\theta;x)$ defined in (1).

(II) We characterize the worst-case distribution for problem (E-∞-WDRO) in Remark 1. In contrast to the conventional formulation (∞-WDRO) that *deterministically* transports each data from $\widehat{\mathbb{P}}$ to its extreme perturbation, the worst-case distribution of our formulation transports each data x towards the entire domain set $\mathbb{B}_\rho(x)$ through specific absolutely continuous distributions. This observation indicates that our formulation (E-∞-WDRO) is well-suited for adversarial defense where the data distribution after adversarial attack manifests as absolutely continuous, such as through the addition of white noise to the data.

(III) Our reformulation (3) is a variant of *conditional stochastic optimization*, recently studied in [6, 8, 7]. We introduce and analyze stochastic gradient methods (see Section 3) to solve this problem efficiently, achieving $\widetilde{\mathcal{O}}(\epsilon^{-2})$ sample complexity for finding ϵ-optimal solution for convex $f_\theta(z)$, and $\widetilde{\mathcal{O}}(\epsilon^{-4})$ sample complexity for finding ϵ-stationary point for nonconvex $f_\theta(z)$. These sample complexity results are near-optimal up to a near-constant factor.

(IV) We develop regularization effects for problem (E-∞-WDRO) in Section 4. Specifically, we show that it is asymptotically equivalent to a regularized ERM formulation for a certain type of regularizer for different scalings of the entropic regularization value ϵ and radius ρ: when $\rho/\eta \to \infty$, (E-∞-WDRO) corresponds to the gradient norm regularized ERM formulation; when $\rho/\eta \to 0$, (E-∞-WDRO) corresponds to a special gradient variance regularized ERM formulation; when $\rho/\eta \to C$, (E-∞-WDRO) corresponds to a regularized ERM formulation that interpolates between these two extreme cases.

(V) Finally, we provide numerical experiments in Section 5 on supervised learning and robust Markov decision process, demonstrating the state-of-the-art performance attained by our algorithm against various adversarial attacks.

In the following, we compare existing references on related topics and list some notations to be used throughout this paper.

Literature Review. Ever since the seminal work [5] illustrated the vulnerability of neural networks to adversarial perturbations, the research on adversarial attack and defense has progressively gained much attention in literature. Numerous approaches for adversarial defense have been put forth [5, 9, 11, 3, 10, 15], aiming to develop heuristic algorithms to approximately optimize the formulation (1), whereas these algorithms may not efficiently find the global optimum solution. Sinha et al. [13] showed that replacing ∞-Wasserstein distance with 2-Wasserstein distance in (∞-WDRO) yields more tractable formulations. Unfortunately, their proposed algorithm necessitates a sufficiently small robustness level such that the involved subproblem becomes strongly convex, which is not well-suited for adversarial training in scenarios with large perturbations. Wang et al. [16] added entropic regularization regarding the p-WDRO formulation to develop more efficient algorithms. We highlight that their result cannot be applied to the entropic regularization for ∞-WDRO setup because the associated transport cost function is not finite-valued. Besides, we provide theoretical justification for adding entropic regularization into the distributionally robust risk function in Section 4, which, to our best knowledge, is new in literature.

Notations. Denote by $\mathrm{Proj}_{1\#}\gamma, \mathrm{Proj}_{2\#}\gamma$ the first and the second marginal distributions of γ, respectively. For a measurable set $\mathcal{Z}$, denote by $\mathcal{P}(\mathcal{Z})$ the set of probability measures on $\mathcal{Z}$. Denote by $\mathrm{supp}\,\mathbb{P}$ the support of probability distribution $\mathbb{P}$. Given a measure μ and a measurable variable $f:\ \mathcal{Z} \to \mathbb{R}$, we write $\mathbb{E}_{z\sim\mu}[f]$ for $\int f(z)\,\mathrm{d}\mu(z)$. Given a subset E in Euclidean space, let $\mathrm{vol}(E)$ denote its volume. Let $\omega:\ \Theta \to \mathbb{R}$ be a distance generating function that is continuously differentiable and κ-strongly convex on Θ with respect to norm $\|\cdot\|_\omega$. This induces the Bregman divergence $D_\omega(\theta,\theta'):\ \Theta\times\Theta\to\mathbb{R}_+$: $D_\omega(\theta,\theta') = \omega(\theta') - \omega(\theta) - \langle\nabla\omega(\theta), \theta'-\theta\rangle$. Next, we define the *prox-mapping* $\mathrm{Prox}:\ \mathbb{R}^{d_\theta} \to \Theta$ as $\mathrm{Prox}_\theta(y) = \arg\min_{\theta'\in\Theta}\left\{\langle y, \theta'-\theta\rangle + D_\omega(\theta,\theta')\right\}$.

2. Strong Duality Result

With a measurable variable $f:\ \mathcal{Z}\to\mathbb{R}$, we associate value

$$\sup_{\mathbb{P},\gamma}\ \left\{\mathbb{E}_{z\sim\mathbb{P}}\left[f(z)\right] - \eta H(\gamma\,|\,\pi):\ \begin{array}{l}\mathrm{Proj}_{1\#\gamma} = \widehat{\mathbb{P}}, \mathrm{Proj}_{2\#\gamma} = \mathbb{P}\\ \text{ess.sup}_\gamma\|\zeta_1-\zeta_2\|\le\rho\end{array}\right\}. \tag{Primal}$$

Problem (Primal) is an infinite-dimensional convex program. The main goal in this section is to derive its strong dual reformulation, which is a more tractable form. Define the dual problem

$$V_{\text{Dual}} = \mathbb{E}_{x\sim\widehat{\mathbb{P}}}\left[\eta\log\mathbb{E}_{z\sim\nu_x}\left[\exp\left(\frac{f(z)}{\eta}\right)\right]\right] = \mathbb{E}_{x\sim\widehat{\mathbb{P}}}\left[\eta\log\mathbb{E}_{z\sim\mathbb{Q}_x}\left[\exp\left(\frac{f(z)}{\eta}\right)\right]\right] + C, \tag{Dual}$$

where ν_x is a Lebesgue measure on $\mathbb{B}_\rho(x)$, $\mathbb{Q}_x$ is the uniform distribution on $\mathbb{B}_\rho(x)$, and the constant $C = \mathbb{E}_{x\sim\widehat{\mathbb{P}}}[\eta\log(\mathrm{vol}(\mathbb{B}_\rho(x)))]$. The following reveals that problems (Primal) and (Dual) are equivalent under mild technical conditions.

Theorem 1. *Assume that $\mathcal{Z}$ is a measurable space, $f:\ \mathcal{Z}\to\mathbb{R}\cup\{\infty\}$ is a measurable function, and for every joint distribution $\gamma\in\mathcal{P}(\mathcal{Z}\times\mathcal{Z})$ with $\mathrm{Proj}_{1\#}\gamma = \widehat{\mathbb{P}}$, it has a regular conditional distribution γ_x given the value of the first marginal equals x. Then for any $\eta > 0$, it holds that $V_{\mathrm{Primal}} = V_{\mathrm{Dual}}$.*

Remark 1 (Worst-case Distribution). The key to prove Theorem 1 is to construct the worst-case distribution of (Primal) whose objective value equals the dual optimal value (Dual). In detail, we construct such a distribution $\mathbb{P}_*$ with density

$$\mathrm{d}\mathbb{P}_*(z) = \mathbb{E}_{x\sim\widehat{\mathbb{P}}}\left[\alpha_x\cdot\exp\left(\frac{f(z)}{\eta}\right)\mathrm{d}\nu_x(z)\right].$$

The worst-case distribution $\mathbb{P}_*$ is supported on $\cup_{x\in\mathrm{supp}\ \widehat{\mathbb{P}}}\ \mathbb{B}_\rho(x)$, which is the whole space where the adversarial perturbations of data may happen. In contrast, the worst-case distribution of classical ∞-WDRO model $\sup_{\mathbb{P}:\ \mathcal{W}(\mathbb{P},\widehat{\mathbb{P}})\le\rho}\ \mathbb{E}_{z\sim\widehat{\mathbb{P}}}[f(z)]$ only moves each data from $\widehat{\mathbb{P}}$ towards its most extreme perturbation.

Now we visualize the worst-case distribution for ∞-WDRO and entropy-regularized ∞-WDRO models by considering $\widehat{\mathbb{P}} = \delta_{x=0}$ with radius $\rho = 3$ and the following multi-layer LeakyReLU neural network with quadratic loss function and 1-dimensional input:

$$\psi(z) = \phi(2\phi(x) - 4\phi(x-0.5)), \qquad f(z) = \Big(\psi(\psi(z+2)) - 1\Big)^2,$$

where $\phi(z) = \mathrm{LeakyReLU}(z) = \max(z,0) - 0.1\min(z,0), z\in\mathbb{R}$. Figure 1(a) presents the landscape of the loss $f(z)$ together with the adversarial transport mapping using ∞-WDRO. Specifically, the ∞-WDRO deterministically moves the nominal distribution $\widehat{\mathbb{P}} = \delta_0$ to another one-point distribution δ_{z^*} with $z^* = \arg\max_{z\in[-3,3]}\ f(z) = 2.5$. Figure 1(b) presents the worst-case distribution from the entropy-regularized ∞-WDRO model for different choices of η. As $\eta\to 0$, the worst-case distribution tends to be the one-point extreme distribution generated by ∞-WDRO; as $\eta\to\infty$, the worst-case distribution tends to be the uniform distribution on $[-3,3]$. Figure 1(b) shows how the entropic regularization value generates the worst-case distribution that interpolates between these two extreme cases.

In this numerical example, finding the global maximum of $f(z)$ over interval $[-3,3]$ using gradient method with initial guess $z = 0$ is difficult, whereas many heuristic adversarial training approaches [9, 5, 10] run this procedure. This also reveals why solving ∞-WDRO problem exactly is computationally challenging.

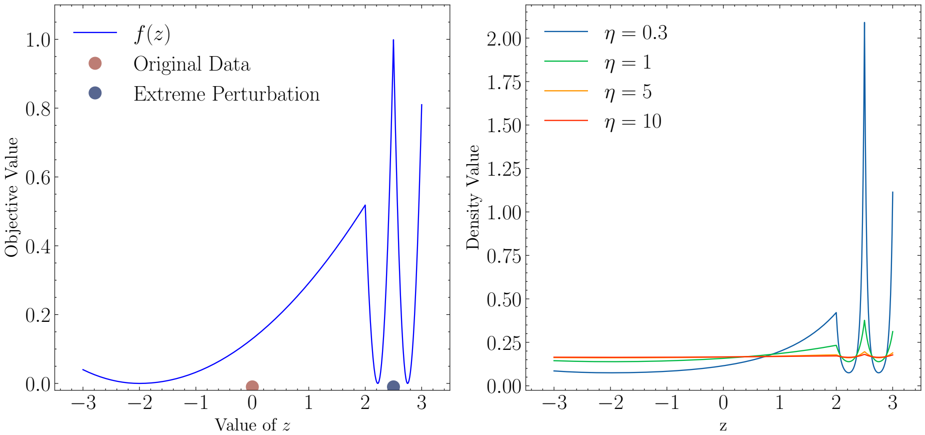

Figure 1 (a) Visualization of $f(z)$ and the most adversarial location from ∞-WDRO; (b) Visualization of density values of the worst-case distribution from entropy-regularized ∞-WDRO for different choices of η.

3. Optimization by Stochastic Gradient Method

We now develop stochastic gradient-type methods to solve the proposed formulation (E-∞-WDRO). Based on our strong duality result in Theorem 1, it can be reformulated as the single minimization problem

$$\min_{\theta\in\Theta}\ \left\{F(\theta)\triangleq\mathbb{E}_{x\sim\widehat{\mathbb{P}}}\left[\eta\log\mathbb{E}_{z\sim\mathbb{Q}_x}\left[\exp\left(\frac{f_\theta(z)}{\eta}\right)\right]\right]\right\}, \tag{4}$$

where $\mathbb{Q}_x$ is an uniform distribution supported on $\mathbb{B}_\rho(x)$. Problem (4) is a special case of conditional stochastic optimization studied in literature: the objective in (4) involves an expectation of nonlinear transformation of the expectation taken with respect to a conditional distribution. Unlike many stochastic programs, one difficulty of solving (4) is that it is computationally challenging to obtain an unbiased gradient estimate of the objective. Inspired by the state-of-the-art algorithms studied in literature, we propose a stochastic mirror descent (SMD) algorithm with biased gradient estimators to solve this problem. By properly balancing the statistics of the biased gradient estimators, we demonstrate that our proposed algorithm can solve this problem with near-optimal sample complexity.

3.1. Main Algorithm

Our main algorithm, summarized in Algorithm 1, operates by iteratively generating a stochastic gradient estimator and then perform prox-mapping update. Specifically, we use the randomized truncation multi-level Monte-Carlo (RT-MLMC) approach to simulate (biased) gradient estimators in Step 3 of Algorithm 1.

Algorithm 1 BSMD for solving (4)

```
Require: maximum iterations $T$, constant step size $\gamma$, initial guess $\theta_0$.
1: for $t=0,1,\dots,T-1$ do
2:     Formulate (biased) gradient estimate of $F(\theta_t)$, denoted as $v(\theta_t)$.
3:     Update $\theta_{t+1}=\mathrm{Prox}_{\theta_t}\big(\gamma v(\theta_t)\big)$.
4: end for
   Output $\tilde{\theta}$ randomly selected from $\{\theta_0,\theta_1,\dots,\theta_T\}$.
```

Construction of RT-MLMC Estimator First, we construct a sequence of approximation functions $\{F^\ell(\theta)\}_{\ell\geq 0}$ instead, where

$$F^\ell(\theta)=\mathbb{E}_{x^\ell}\ \mathbb{E}_{\{z_j^\ell\}_{j\in[2^\ell]}|x^\ell}\left[\eta\log\left(\frac{1}{2^\ell}\sum_{j\in[2^\ell]}\exp\left(\frac{f_\theta(z_j^\ell)}{\eta}\right)\right)\right]. \tag{5}$$

Here the random variable x^ℓ follows distribution $\widehat{\mathbb{P}}$, and for fixed value of x^ℓ, $\{z_j^\ell\}_{j\in[2^\ell]}$ are independent and identically distributed samples from $\mathbb{Q}_{x^\ell}$. Unlike the original objective $F(\theta)$, unbiased gradient estimators of its approximation $F^\ell(\theta)$ can be easily obtained. RT-MLMC estimator is an computationally efficient approach to obtain the unbiased gradient estimator.

Denote by $\zeta^\ell=(x^\ell,\{z_j^\ell\}_{j\in[2^\ell]})$ the collection of random sampling parameters, and

$$U_{n_1:n_2}(\theta,\zeta^\ell)=\eta\log\left(\frac{1}{n_2-n_1+1}\sum_{j\in[n_1:n_2]}\exp\left(\frac{f_\theta(z_j^\ell)}{\eta}\right)\right).$$

For fixed parameter θ, we define

$$G^\ell(\theta,\zeta^\ell)=\nabla_\theta\left[U_{1:2^\ell}(\theta,\zeta^\ell)-\frac{1}{2}U_{1:2^{\ell-1}}(\theta,\zeta^\ell)-\frac{1}{2}U_{2^{\ell-1}+1:2^\ell}(\theta,\zeta^\ell)\right]. \tag{6}$$

To obtain the RT-MLMC estimator, for $i=1,\dots,n_L^{\mathrm{o}}$, we sample random level ι_i from the truncated geometric $\mathbb{Q}_{\mathrm{RT}}$ with probability mass value

$$\mathbb{Q}_{\mathrm{RT}}(\iota=\ell)=q_\ell\propto 2^{-\ell}, \ell=0,\dots,L,$$

and then construct $G^{\iota_i}(\theta,\zeta^{\iota_i})$. Then we take RT-MLMC estimator as the mini-batch estimator

$$v^{\text{RT-MLMC}}(\theta)=\frac{1}{n_L^{\mathrm{o}}}\sum_{i=1}^{n_L^{\mathrm{o}}}\frac{1}{q_{\iota_i}}G^{\iota_i}(\theta,\zeta^{\iota_i}).$$

The RT-MLMC estimator has the following key features:

(I) It constitutes an unbiased gradient estimator of the approximation function $F^L(\theta)$:

$$\begin{aligned}\mathbb{E}[v^{\text{RT-MLMC}}(\theta)]&=\mathbb{E}_{\iota_1}\left[\frac{1}{q_{\iota_1}}\mathbb{E}_{\zeta^{\iota_1}}[G^{\iota_1}(\theta,\zeta^{\iota_1})]\right]=\sum_{\ell=0}^{L}q_\ell\cdot\left[\frac{1}{q_\ell}\mathbb{E}_{\zeta^\ell}[G^\ell(\theta,\zeta^\ell)]\right]\\&=\sum_{\ell=0}^{L}\mathbb{E}_{\zeta^\ell}[G^\ell(\theta,\zeta^\ell)]=\nabla F^L(\theta).\end{aligned}$$

(II) Since $U_{1:2^\ell}(\theta,\zeta^\ell), U_{1:2^{\ell-1}}(\theta,\zeta^\ell)$, and $U_{2^{\ell-1}+1:2^\ell}(\theta,\zeta^\ell)$ are generated using the same random sampling parameters ζ^ℓ, they are highly correlated, which implies the stochastic estimator $G^\ell(\theta,\zeta^\ell)$ defined in (6) has small second-order moment and variance thanks to the control variate effect, making it a suitable recipe for gradient simulation.

3.2. Convergence Analysis

In this subsection, we provide convergence analysis of our algorithm. We first consider the following assumptions regarding the loss function $f_\theta(z)$.

ASSUMPTION 1. (I) *(Convexity): The loss function $f_\theta(z)$ is convex in θ.*

(II) *(Boundedness): The loss function $f_\theta(z)$ satisfies $0\le f_\theta(z)\le B$ for any $\theta\in\Theta$ and $z\in\mathcal{Z}$.*

(III) *(Lipschitz Continuity): For fixed z and θ_1,θ_2, it holds that $|f_{\theta_1}(z)-f_{\theta_2}(z)|\le L_f\|\theta_1-\theta_2\|_2$.*

(IV) *(Lipschitz Smoothness): The loss function $f_\theta(z)$ is continuously differentiable and for fixed z and θ_1,θ_2, it holds that $\|\nabla f_{\theta_1}(z)-\nabla f_{\theta_2}(z)\|_2\le S_f\|\theta_1-\theta_2\|_2$.*

To quantify the quality of the obtained solution for solving (4), we say a given random vector θ is ϵ-optimal if $\mathbb{E}[F(\theta)-F(\theta^*)]\le\epsilon$, where θ^* is a global optimum solution of $\min_\theta\ F(\theta)$, and θ is a ϵ-stationary point if for some step size $\gamma>0$, it holds that $\mathbb{E}\left\|\left[\theta-\mathrm{Prox}_\theta\big(\gamma\nabla F(\theta)\big)\right]/\gamma\right\|_2^2\le\epsilon$. We quantify the computation cost of the proposed algorithm as the number of generated random parameters from $\widehat{\mathbb{P}}$ or $\mathbb{Q}_x$ for any $x\in\mathcal{Z}$; and quantify the storage cost as the number of parameters saved in the data buffer. By properly tuning hyper-parameters of RT-MLMC estimator, we establish the convergence guarantees of our algorithm in Theorem 2.

THEOREM 2 **(Complexity Analysis of BSMD)**. *Under Assumption 1(II) and 1(III), with properly chosen hyper-parameters of the RT-MLMC estimator as in Table 1, the following results hold:*

(I) *(Nonsmooth Convex Optimization) Additionally assume Assumption 1(I) holds, the the computation cost of RT-MLMC scheme for finding ϵ-optimal solution is of $\widetilde{\mathcal{O}}(\epsilon^{-2})$, with memory cost $\widetilde{\mathcal{O}}(1)$.*

(II) *(Smooth Nonconvex Optimization) Additionally assume Assumption 1(IV) holds, the computation cost of RT-MLMC scheme for finding ϵ-stationary point is of $\widetilde{\mathcal{O}}(\epsilon^{-4})$ with memory cost $\widetilde{\mathcal{O}}(\epsilon^{-2})$.*

(III) *(Unconstrainted Smooth Nonconvex Optimization) Under the setup of Part (II) and additionally assume the constraint set $\Theta=\mathbb{R}^{d_\theta}$, then the memory cost of RT-MLMC improves to $\widetilde{\mathcal{O}}(1)$.*

Table 1 Hyper-parameters, computational cost (Comp.), and memory cost (Memo.) of Algorithm 1.

Scenarios	Hyper-parameters	Comp./Memo.
Nonsmooth Convex Optimization	$L=\mathcal{O}(\log\frac{1}{\epsilon}),\quad T=\widetilde{\mathcal{O}}(\epsilon^{-2})$ $n_L^{\mathrm{o}}=\mathcal{O}(1),\quad \gamma=\widetilde{\mathcal{O}}(\epsilon)$	Comp. $=\mathcal{O}(T(n_L^{\mathrm{o}}L))=\widetilde{\mathcal{O}}(\epsilon^{-2})$ Memo. $=\mathcal{O}(n_L^{\mathrm{o}}L)=\widetilde{\mathcal{O}}(1)$
Smooth Nonconvex Optimization	$L=\mathcal{O}(\log\frac{1}{\epsilon^2}),\quad T=\widetilde{\mathcal{O}}(\epsilon^{-2})$ $n_L^{\mathrm{o}}=\widetilde{\mathcal{O}}(\epsilon^{-2}),\quad \gamma=O(1)$	Comp. $=\mathcal{O}(T(n_L^{\mathrm{o}}L))=\widetilde{\mathcal{O}}(\epsilon^{-4})$ Memo. $=\mathcal{O}(n_L^{\mathrm{o}}L)=\widetilde{\mathcal{O}}(\epsilon^{-2})$
Unconstrainted Smooth Nonconvex Optimization	$L=\mathcal{O}(\log\frac{1}{\epsilon^2}),\quad T=\widetilde{\mathcal{O}}(\epsilon^{-4})$ $n_L^{\mathrm{o}}=\mathcal{O}(1),\quad \gamma=\widetilde{\mathcal{O}}(\epsilon^2)$	Comp. $=\mathcal{O}(T(n_L^{\mathrm{o}}L))=\widetilde{\mathcal{O}}(\epsilon^{-4})$ Memo. $=\mathcal{O}(n_L^{\mathrm{o}}L)=\widetilde{\mathcal{O}}(1)$

Remark 2 (Comparison with ∞-WDRO). When solving the classical ∞-WDRO problem (∞-WDRO), the involved subproblems are finding the global optimal value of the supremum $\sup_{z\in\mathbb{B}_\rho(x)}\ f(z)$ for $x\in\operatorname{supp}\widehat{\mathbb{P}}$, which are computationally challenging in general. Various heuristics [9, 5, 10] have been proposed to approximately solve it by replacing $f(z)$ with its linear approximation $f(x)+\nabla f(x)^{\mathrm{T}}z$. It is worth noting that such an approximation is not accurate, especially when the radius ρ of domain set $\mathbb{B}_\rho(x)$ is moderately large, which corresponds to large adversarial perturbation scenarios. For example, for the loss $f(z)$ depicted in Figure 1, its linear approximation around $x=0$ will yield a wrong global maximum estimate. In contrast, we proposed stochastic gradient methods to solve the regularized formulation (E-∞-WDRO) with provable convergence guarantees, which avoids solving such a hard maximization subproblem. Numerical comparisons in Section 5.1 also suggests that our method outperforms those heuristics when adversarial perturbations are moderately large.

4. Regularization Effects of (E-∞-WDRO)

In this section, we provide an interpretation on how our proposed formulation (E-∞-WDRO) works by showing its close connection to the regularized ERM problem:

$$\min_{\theta\in\Theta}\ \mathbb{E}_{z\sim\widehat{\mathbb{P}}}[f_\theta(z)]+\mathcal{R}(f_\theta;\rho,\eta)$$

for certain regularization $\mathcal{R}(f_\theta;\rho,\eta)$. In this part we focus on small-perturbation attacks, which motivates us to assume hyper-parameters $\rho\to 0$ and $\eta\to 0$. In the following we consider the regularization effects of (E-∞-WDRO) by considering different scaling between ρ and η. To begin with, we define the entropic regularizer $\mathcal{E}$ as the difference between entropic robust loss in (Primal) and non-robust loss:

$$\mathcal{E}_{\widehat{\mathbb{P}}}(f;\rho,\eta)=\mathrm{Optval}(\text{Primal})-\mathbb{E}_{\widehat{\mathbb{P}}}[f].$$

Besides, we define the following regularizations. Let μ and ν be the uniform probability distribution and Lebesgue measure supported on $\mathbb{B}_1(0)$, respectively. Next, define

$$\mathcal{R}_1(f;\rho,\eta)=\rho\mathbb{E}_{x\sim\widehat{\mathbb{P}}}[\|\nabla f(x)\|_*], \tag{7a}$$

$$\mathcal{R}_2(f;\rho,\eta)=\eta\log(\alpha)+\frac{\rho^2}{\eta}\mathbb{E}_{x\sim\widehat{\mathbb{P}}}\left[\mathbb{V}\mathrm{ar}_{z\sim\mu}[\nabla f(x)^{\mathrm{T}}z]\right],\quad \text{where } \alpha=\mathrm{vol}(\mathbb{B}_1(0)), \tag{7b}$$

$$\mathcal{R}_3(f;\rho,\eta)=\frac{\rho}{C}\mathbb{E}_{\widehat{\mathbb{P}}}\left[\log\mathbb{E}_{z\sim\nu}\left[\exp\left(C\nabla f(x)^{\mathrm{T}}z\right)\right]\right], \tag{7c}$$

where $C>0$ is some constant to be specified. These three regularizations correspond to the asymptotic approximations of the entropic regularizer under the following three cases.

Case 1: $\rho/\eta\to\infty$. In this case, the convergence rate of the entropic regularization η is faster than that of ρ. As such, most contributions to the integral of the dual objective function in (Dual) will come only from points in a neighbourhood of $\arg\max_{z\in\mathbb{B}_\rho(x)} f(z)$, and we can apply Laplace's method to give the error estimate. The following proposition reveals that problem (E-∞-WDRO) is asymptotically equivalent to the classical formulation (∞-WDRO).

Proposition 1. *Assume for any $x \in \operatorname{supp} \widehat{\mathbb{P}}$, $f(\cdot)$ is twicely differentiable and has a unique global maximizer z_x^* in the domain set $\mathbb{B}_\rho(x)$. Then it holds that*

$$\mathrm{Optval}(\text{Primal}) = \sup_{\mathbb{P}:\ \mathcal{W}_\infty(\mathbb{P},\widehat{\mathbb{P}})\le\rho} \Big\{\mathbb{E}_{z\sim\mathbb{P}}[f(z)]\Big\} - \frac{\eta}{2}\mathbb{E}_{x\sim\widehat{\mathbb{P}}}\left[\log\det\big(-\nabla^2 f(z_x^*)\big)\right] + \frac{d\eta}{2}\log(2\pi\eta) + o(\eta).$$

Recall that the gradient norm regularization $\mathcal{R}_1(\theta;\rho,\eta)$ defined in (7a) is the first-order regularization effect of the ∞-WDRO formulation (see, e.g., [4, Theorem 1]). This fact, together with Proposition 1, further implies that

$$\Big|\mathcal{E}_{\widehat{\mathbb{P}}}(f;\rho,\eta) - \mathcal{R}_1(f;\rho,\eta)\Big| = O\left(\rho^2 \vee \eta\log\eta\right).$$

Case 2: $\rho/\eta \to 0$. Next, we consider the case where the convergence rate of radius ρ is faster than that of the entropic regularization η. The following proposition reveals that (Primal) is asymptotically equivalent to ERM formulation with regularization $\mathcal{R}_2$ defined in (7b). Based on the first-order Taylor expansion of the loss function $f_\theta(z)$ and second order Taylor expansion of the exponential function in the dual formulation (Dual), we obtain such a result.

Proposition 2. *Assume $f(\cdot)$ is smooth such that for any x, x', $\|\nabla f(x) - \nabla f(x')\|_* \le S(x)\cdot\|x - x'\|$. Then it holds that*

$$\Big|\mathcal{E}_{\widehat{\mathbb{P}}}(f;\rho,\eta) - \mathcal{R}_2(f;\rho,\eta)\Big| = o(\rho),$$

where the small-o notation $o(\cdot)$ hides constant related to $S(x)$.

Case 3: $\rho/\eta \to C$. In this case, we have the following asymptotic expansion result.

Proposition 3. *Assume $f(\cdot)$ is smooth such that for any x, x', $\|\nabla f(x) - \nabla f(x')\|_* \le S(x)\cdot\|x - x'\|$. Then it holds that*

$$\Big|\mathcal{E}_{\widehat{\mathbb{P}}}(f;\rho,\eta) - \mathcal{R}_3(f;\rho,\eta)\Big| = O(\rho^2),$$

where the big-O notation $O(\cdot)$ hides constant related to $S(x)$.

Interestingly, as the constant $C \to \infty$, the regularization $\mathcal{R}_3(f;\rho,\eta) \to \mathcal{R}_1(f;\rho,\eta)$; as the constant $C \to 0$, the regularization $\mathcal{R}_3(f;\rho,\eta) \to \mathcal{R}_2(f;\rho,\eta)$. When $\rho/\eta \to C$ for some $C > 0$, the corresponding regularized ERM is an interpolation between the regularized ERM formulations corresponding to other two extreme cases.

5. Numerical Study

In this section, we examine the numerical performance of our proposed algorithm on two applications: supervised learning and robust Markov decision process. We compare our method with the following baselines: (i) fast-gradient method (FGM) [5], (ii) its iterated variant (IFGM) [9], (iii) and projected-gradient method (PGM) [10]. Those baseline methods are heuristic approaches to approximately solving the ∞-WDRO model.

5.1. Supervised Learning

We validate our method on the MNIST handwritten digit dataset, by training the classifer using a neural network with 8×8 and 6×6 convolutional filter layers and ELU activations, and followed by a connected layer and softmax output. After the training process with those listed methods, we then add various perturbations to the testing datasets, such as the ℓ_2-norm and ℓ_∞-norm adversarial projected gradient method attacks [10], and white noises uniformly distributed in a ℓ_2 or ℓ_∞ norm ball. We use the mis-classification rate on testing dataset to quantify the performance for the obtained classifers.

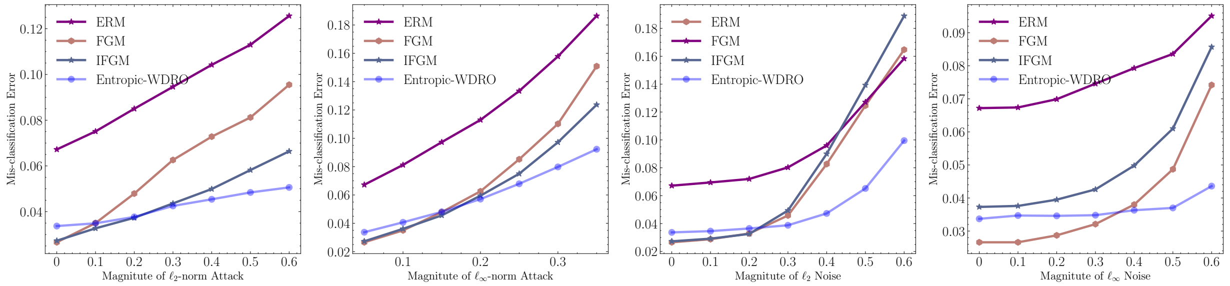

Figure 2 Results of adversarial training on MNIST dataset. From left to right, the figures correspond to (a) ℓ_2-norm PGD attack; (b) ℓ_∞-norm PGD attack; (c) uniform noise in ℓ_2-norm ball; and (d) uniform noise in ℓ_∞-norm ball.

For fair comparison, we take the same level of robustness parameter $\rho = 0.45$ for all approaches. Since the scaling of η that satisfies $\rho/\eta \to C$ for some constant C corresponds to a smoothed norm regularized ERM training as suggested by Section 4, we specify the regularization value $\eta = 0.1 \cdot \rho$ in this experiment.

Figure 2 presents mis-classification results of various methods for different types of adversarial attacks with different levels of adversarial perturbations on the testing dataset. From the plot we can see that all methods tend to have worse performance as the perturbation level increases, but our proposed method tends to outperform other baselines, especially for cases in large perturbation levels. This suggests that our model has superior performance for adversarial training in scenarios with large perturbations.

5.2. Reliable Reinforcement Learning

Next, we provide a robust algorithm for reinforcement learning (RL). Consider an infinite-horizon discounted finite state MDP represented by a tuple $\langle \mathcal{S}, \mathcal{A}, \mathbb{P}, R, \gamma \rangle$, where $\mathcal{S}, \mathcal{A}$ denotes the state and action space, respectively; $\mathbb{P} = \{\mathbb{P}(s' \mid s, a)\}_{s,s',a}$ is the set of transition probability metrics; $R = \{r(s,a)\}_{s,a}$ is the reward table with (s,a)-th entry being the reward for taking the action a at state s; and $\gamma \in (0,1)$ is the discounted factor. Similar to problem (∞-WDRO), robust reinforcement learning seeks to maximize the worst-case risk function $\sup_{\mathbb{P} \in \mathfrak{R}} \mathbb{E}[\sum_t \gamma^t r(s_t, a_t)]$, with $\mathfrak{R}$ represents the ambiguity set for state-action transitions. For simplicity, we consider a tabular Q-learning setup in this subsection. The standard Q-learning algorithm in RL learns a Q-function $Q: \ \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ with iterations

$$Q(s^t, a^t) \leftarrow (1-\alpha_t) Q(s^t, a^t) + \alpha_t r(s^t, a^t) - \gamma \alpha_t \min_a (-Q(s^{t+1}, a)), \quad s^{t+1} \sim \mathbb{P}(\cdot \mid s^t, a^t). \tag{8}$$

We modify the last term of the update (8) with an adversarial state perturbation to take ∞-Wasserstein distributional robustness with entropic regularization into account, leading to the new update

$$Q(s^t, a^t) \leftarrow (1-\alpha_t) Q(s^t, a^t) + \alpha_t r(s^t, a^t) - \gamma \alpha_t \min_a \left\{ \eta \log \mathbb{E}_{s \sim \nu(s^{t+1};\rho)} e^{-Q(s,a)/\eta} \right\},$$

where $\nu(s^{t+1}; \rho)$ denotes an uniform distribution supported on a norm ball of s^{t+1} with radius ρ. Standard convergence analysis on Q-learning [14] can be modified to show the convergence of the modified Q-learning iteration. Our proposed algorithm in Section 3 can be naturally applied to proceed the updated Q-learning iteration.

We test our algorithm in the cart-pole environment [1], where the objective is to balance a pole on a cart by moving the cart to left or right, with state space including the physical parameters such as chart position, chart velocity, angle of pole rotation, and angular of pole velocity. To generate perturbed

Environment	Regular	Robust
Original MDP	469.42 ± 19.03	**487.11 ± 9.09**
Perturbed MDP (Heavy)	187.63 ± 29.40	**394.12 ± 12.01**
Perturbed MDP (Short)	355.54 ± 28.89	**443.17 ± 9.98**
Perturbed MDP (Strong g)	271.41 ± 20.7	**418.42± 13.64**

Table 2 Performance of Q-learning algorithms in original MDP and shifted MDP environments. Error bars are produced using 10 independent trials.

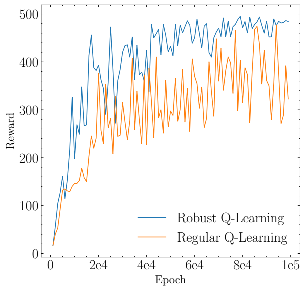

Figure 3 Episode lengths during training. The environment caps episodes to 400 steps.

MDP environments, we perturb the physical parameters of the system by magnifying the pole's mass by 2, or shrinking the pole length by 2, or magnifying the strength of gravity g by 5. We name those three perturbed environments as *Heavy*, *Short*, or *Strong* g MDP environments, respectively.

Figure 3 demonstrates the training process of regular and robust Q-learning algorithms on the original MDP environment. Interestingly, the robust Q-learning algorithm learns the optimal policy more efficiently than the regular MDP. One possible explanation is that taking account into adversarial perturbations increase the exploration ability of the learning algorithm. Next, we report the performance of trained policies in original and perturbed MDP environments in Table 2, from which we can see that our proposed robust Q-learning algorithm consistently outperforms the regular non-robust algorithm.

6. Conclusion

In this paper, we proposed an entropic regularized framework for adversarial robust training. From the computational perspective, this new formulation brings a notable improvement in tractability. From the statistical perspective, this framework is asymptotically equivalent to certain regularized ERM under different scaling regimes of the entropic regularization and robustness level hyper-parameters. Numerical experiments indicate that our proposed framework has state-of-the-art performance especially for large adversarial perturbation scenarios.

References

[1] Brockman G, Cheung V, Pettersson L, Schneider J, Schulman J, Tang J, Zaremba W (2016) Openai gym. *arXiv preprint arXiv:1606.01540* .

[2] Butler RW (2007) *Saddlepoint approximations with applications*, volume 22 (Cambridge University Press).

[3] Carlini N, Wagner D (2017) Towards evaluating the robustness of neural networks. *2017 ieee symposium on security and privacy (sp)*, 39–57 (Ieee).

[4] Gao R, Chen X, Kleywegt AJ (2022) Wasserstein distributionally robust optimization and variation regularization. *Operations Research* .

[5] Goodfellow IJ, Shlens J, Szegedy C (2014) Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* .

[6] Hu Y, Chen X, He N (2020) Sample complexity of sample average approximation for conditional stochastic optimization. *SIAM Journal on Optimization* 30(3):2103–2133.

[7] Hu Y, Chen X, He N (2021) On the bias-variance-cost tradeoff of stochastic optimization. *Advances in Neural Information Processing Systems*.

[8] Hu Y, Zhang S, Chen X, He N (2020) Biased stochastic first-order methods for conditional stochastic optimization and applications in meta learning. *Advances in Neural Information Processing Systems*, volume 33, 2759–2770.

[9] Kurakin A, Goodfellow I, Bengio S (2017) Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236* .

[10] Madry A, Makelov A, Schmidt L, Tsipras D, Vladu A (2017) Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083* .

[11] Papernot N, McDaniel P, Jha S, Fredrikson M, Celik ZB, Swami A (2016) The limitations of deep learning in adversarial settings. *2016 IEEE European symposium on security and privacy (EuroS&P)*, 372–387 (IEEE).

[12] Shapiro A (2017) Distributionally robust stochastic programming. *SIAM Journal on Optimization* 27(4):2258–2275.

[13] Sinha A, Namkoong H, Volpi R, Duchi J (2020) Certifying some distributional robustness with principled adversarial training. *arXiv preprint arXiv:1710.10571* .

[14] Szepesvári C, Littman ML (1999) A unified analysis of value-function-based reinforcement-learning algorithms. *Neural computation* 11(8):2017–2060.

[15] Tramèr F, Kurakin A, Papernot N, Goodfellow I, Boneh D, McDaniel P (2017) Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204* .

[16] Wang J, Gao R, Xie Y (2023) Sinkhorn distributionally robust optimization. *arXiv preprint arXiv:2109.11926* .