On Global-view Based Defense via Adversarial Attack and Defense Risk Guaranteed Bounds

Trung Le[1] **Anh Bui**[1] **Tue Le**[3]
He Zhao[1] **Paul Montague**[2] **Quan Tran**[4] **Dinh Phung**[1]
[1]Monash University, Australia [2]Defence Science and Technology Group, Australia
[3]JVN Institute, Vietnam [4]Adobe Research

Abstract

It is well-known that deep neural networks (DNNs) are susceptible to adversarial attacks, which presents the most severe fragility of deep learning system. Despite achieving impressive performance, most of the current state-of-the-art classifiers remain highly vulnerable to crafted imperceptible, adversarial perturbations. Recent research attempts to understand neural network attack and defense have become increasingly urgent and important. While rapid progress has been made on this front, there is still an important theoretical gap in achieving guaranteed bounds on attack/defense models, leaving uncertainty in the certified guarantees of these models. To this end, we systematically address this problem in this paper. More specifically, we formulate attack and defense in a generic setting where there exists a family of adversaries (i.e., attackers) for attacking a family of classifiers (i.e., defenders). We develop a novel class of f-divergences suitable for measuring divergence among multiple distributions. This equips us to study the interactions between attackers and defenders in a countervailing game where we formulate a joint risk on attack and defense schemes. This is followed by our key results on guaranteed upper and lower bounds on this risk that can provide a better understanding of the behaviors of those parties from the attack and defense perspectives, thereby having important implications to both attack and defense sides. Finally, benefited from our theory, we propose an empirical approach that bases on global view to defend against adversarial attacks. The experimental results conducted on benchmark datasets show that the global-view if exploited appropriately can help to improve adversarial robustness.

1 Introduction

Deep neural networks are powerful models that achieve impressive performance across various domains such as bioinformatics (Spencer et al., 2015), speech recognition (Hinton et al., 2012), computer vision (He et al., 2016), and natural language processing (Vaswani et al., 2017). Though achieving state-of-the-art performance, these models are quite brittle in the sense that one can easily craft small, imperceptible, adversarial perturbations of input data to fool them, hence resulting in highly incorrect classification (Szegedy et al., 2013; Goodfellow et al., 2014b). This finding of the surprising vulnerability of classifiers to perturbations has led to a large body of work to study the robustness of models from both attack (Goodfellow et al., 2014b; Madry et al., 2017; Carlini and Wagner, 2017; Kurakin et al., 2016) and defense (Goodfellow et al., 2014b; Madry et al., 2017; Carlini and Wagner, 2017; Kannan et al., 2018; Qin et al., 2019; Zhang et al., 2019; Xie et al., 2019) perspectives.

In a real-world scenario, attackers leverage various attack techniques to exploit machine learning systems, and a robust system is required to be resilient to those divergent attacks. Motivated by this real-world scenario, in this work, we study attack and defense from a game theory perspective of two players: attackers and defenders. More specifically, we assume that there is a family of adversaries (i.e., attackers) that tries to attack a family of classifiers (i.e., defenders). Under this assumption, we develop attack- and defense-guaranteed bounds that can be meaningfully and intuitively interpreted from the perspective of both attacks and defenses. Technically, the lower bound, which is useful for the attack side, reveals that to attack more effi-

Proceedings of the 25th International Conference on Artificial Intelligence and Statistics (AISTATS) 2022, Valencia, Spain. PMLR: Volume 151. Copyright 2022 by the author(s).

ciently, adversaries need to globally push adversarial examples to be more intermingled so as to increase damage on classifiers. Meanwhile, the upper bound, which is useful for the defensive side, shows that to defend more efficiently, classifiers should be trained so as to keep adversarial examples as close to the original data as possible. Additionally, besides offering a better understanding of the attack and defense perspectives, our proposed theory has appealing implications for both the attack and defense sides, which potentially sheds light on devising and developing new attack and defense methods. We summarize our main contributions in this work as follows:

- We study the problem of attack and defense from a game theory perspective and develop upper and lower guaranteed bounds that can be meaningfully and intuitively interpreted. Our proposed theory has appealing implications for both the attack and defense sides, thereby potentially providing support for devising and developing new attack and defense methods.
- Inspired from the theoretical implications, we develop two defense methods that take advantage from global-view attack and defense to improve the current state-of-the-art adversarial training methods PGD (Madry et al., 2017) and TRADES (Zhang et al., 2019). Comparing to PGD (Madry et al., 2017) and TRADES (Zhang et al., 2019), our ***G****lobal-****V****iew based* ***P****GD and* ***T****RADES* (GV-PGD and GV-TRADES) exploit global view to craft more dangerous adversarial examples and use them to improve their models. Specifically, when crafting adversarial examples, we aim to generate those simultaneously far from benign data distribution and more class intermingling to increase damage on classifiers. Subsequently, we use those adversarial examples to improve model robustness by keeping them closer to benign data distribution and more class separated.
- We establish experiments to demonstrate the usefulness of the theoretical global views for attack and defense by comparing our GV-PGD and GV-TRADES with their counterparts. Experimental results show that GV-PGD and GV-TRADES significantly outperform PGD (Madry et al., 2017) and TRADES (Zhang et al., 2019), which validates our theoretical findings.

Related works. Efficient attack and defense methods which are key ingredients to improve the robustness of deep learning models were proposed in (Goodfellow et al., 2014b; Madry et al., 2017; Carlini and Wagner, 2017; Kannan et al., 2018; Qin et al., 2019; Zhang et al., 2019; Xie et al., 2019; Hoang et al., 2020; Bui et al., 2021b, 2020, 2022; Nguyen-Duc et al., 2022). Besides, the theoretical studies in adversarial machine learning are crucial to provide a better understanding from the theoretical perspective. To this end, existing approaches have been proposed to study attack and defense from various perspectives, notably (Schmidt et al., 2018; Tsipras et al., 2019; Fawzi et al., 2018; Zhang et al., 2019; Cranko et al., 2019; Cullina et al., 2018; Bubeck et al., 2019; Wei and Ma, 2020).

The body of works (Schmidt et al., 2018; Tsipras et al., 2019; Zhang et al., 2019) examined the inherent tradeoff between the natural and robust accuracies (i.e., with and without using adversarial samples) and reached a consensus: the generalization of adversarial robustness requires more data and we need to sacrifice natural accuracy to make models robust. In addition, Fawzi et al. (2018) examined attacking scenario where data were assumed to be generated from a generative model, but without any ground-truth labeling function. Moreover, Cullina et al. (2018) developed a PAC learnability in the presence of an evasion attack. More recently, Wei and Ma (2020) inspected the generalization capacity of a deep net in relation to the all-layer margin.

Most closely related to ours is Cranko et al. (2019), in which the authors devised a lower bound in the context of using a family of transformations trying to generate adversarial examples for attacking a family of classifiers. Our work advances Cranko et al. (2019) in the following ways: i) we consider the case of multi-class classification, ii) we consider a family of adversaries instead of a family of transformations, and iii) we develop the lower and upper bounds w.r.t. the data and latent spaces.

2 Preliminaries

Consider a general setting wherein we have a family of adversaries $\mathcal{A}$ trying to attack a family of classifiers $\mathcal{H}$. Given a data example/label pair $(\mathbf{x}, y) \sim \mathcal{D}$, which is the joint distribution generating data and labels, an adversary $a \in \mathcal{A}$ (e.g., a PGD (Madry et al., 2017), BIM (Kurakin et al., 2016), FGSM (Goodfellow et al., 2014b), Auto-Attack (Croce and Hein, 2020a), SpdAdv (Zhao et al., 2019), or PVAdv (Zhao et al., 2021) attack) undertakes an attack on $h \in \mathcal{H}$ by transforming $\mathbf{x}$ to $a_h(\mathbf{x})$, causing the general loss:

$$\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell\left(y, h\left(a_h\left(\mathbf{x}\right)\right)\right)\right],$$

where $\ell : \Delta_M \times \Delta_M \to \mathbb{R}$ is a loss function in which M is the number of classes, $\triangle_M := \left\{\boldsymbol{\beta} \in \mathbb{R}^M : \left\|\boldsymbol{\beta}\right\|_1 = 1 \text{ and } \boldsymbol{\beta} \geq \mathbf{0}\right\}$ is the M-simplex, and we are considering a multi-class setting where the output lies in the $M-$simplex: $h(\mathbf{x}) \in \triangle_M$ for any $h \in \mathcal{H}$. Here we note that given a categorical label

$y \in \{1, ..., M\}$, we denote $\mathbf{1}_y$ as the corresponding one-hot vector and with a slight abuse of notation, we write the loss function as $\ell\left(y, \boldsymbol{\alpha}\right) = \ell\left(\mathbf{1}_y, \boldsymbol{\alpha}\right)$ for $\boldsymbol{\alpha} \in \triangle_M$.

The *attack transformation* $a_h\left(\mathbf{x}\right)$ returns an *adversarial example* corresponding to the clean example $\mathbf{x}$ with the label y formed by using the adversary a to attack the classifier h. For example, in the case of a PGD attack with the distortion radius $\epsilon > 0$ and T-step update, a_h can be expressed as: $a_h\left(\mathbf{x}\right) = \mathbf{x}_T$ where $\mathbf{x}_0 = \mathbf{x} + \boldsymbol{\mu}$ with $\boldsymbol{\mu} \sim \mathcal{N}\left(\mathbf{0}, \epsilon\mathbb{I}\right)$ and $\mathbf{x}_{t+1} = \Pi_{\mathcal{B}_\epsilon(\mathbf{x})}\left(\mathbf{x}_t + \eta\nabla_{\mathbf{x}} L\left(y, h\left(\mathbf{x}_t\right)\right)\right)$ $(0 \leq t \leq T-1)$ for which $\Pi_{\mathcal{B}_\epsilon(\mathbf{x})}$ is the projection onto the ball $\mathcal{B}_\epsilon\left(\mathbf{x}\right) := \left\{\mathbf{x}' : \|\mathbf{x}' - \mathbf{x}\| \leq \epsilon\right\}$ and $\eta > 0$ is the learning rate.

We formulate our problem as follows: we first find an adversary $a \in A$ that most efficiently attacks a given classifier $h \in \mathcal{H}$, then find the best h that minimizes its worst-case general loss as shown in the following min-max problem

$$\mathcal{L}\left(\mathcal{A}, \mathcal{H}, \mathcal{D}\right) := \inf_{h \in \mathcal{H}} \sup_{a \in \mathcal{A}} \mathcal{J}\left(a, h\right), \qquad (1)$$

where $\mathcal{J}\left(a, h\right) := \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell\left(y, h\left(a_h\left(\mathbf{x}\right)\right)\right)\right]$.

The min-max problem in Eq. (1) can be viewed from a game theory perspective involving two parties: (1) the *classifier family* $\mathcal{H}$ and (2) the *adversary family* $\mathcal{A}$. Particularly, given a classifier $h \in \mathcal{H}$, the second party (i.e., the adversary family $\mathcal{A}$) attempts to select the best adversary a_h^* to attack $\mathcal{H}$ by performing $\sup_{a\in\mathcal{A}} \mathcal{J}\left(a, h\right)$ causing the worst general loss, whereas the first party (i.e., the classifier family $\mathcal{H}$) attempts to select the optimal classifier h^* minimizing the worst general loss caused by $a_{h^*}^*$.

We term $\mathcal{L}\left(\mathcal{A}, \mathcal{H}, \mathcal{D}\right)$ in Eq. (1) as the *attack/defense risk* which constitutes our central quantity of interest. To facilitate our theory developed in the sequel, we assume the following generative mechanism for generating a data example $\mathbf{x}$ and its labels y:

$$k \sim \text{Cat}\left(\boldsymbol{\pi}\right), \mathbf{x} \sim p_k\left(\mathbf{x}\right) = p\left(\mathbf{x} \mid y = k\right),$$

where $\pi_k = p\left(y = k\right)$, $\boldsymbol{\pi} > \mathbf{0}, \|\boldsymbol{\pi}\|_1 = 1$, and $\text{Cat}\left(\cdot\right)$ specifies the categorical distribution, hence the data density is a mixture $p\left(\mathbf{x}\right) = \sum_{k=1}^M \pi_k p_k\left(\mathbf{x}\right)$ with $p_k(\mathbf{x})$ is the k-th conditional class density.

Central to our developed theory is the loss function $\ell : \Delta_M \times \Delta_M \to \mathbb{R}$, we focus on the following family of loss functions:

$$\ell\left(\boldsymbol{\beta}, \boldsymbol{\alpha}\right) := D_f\left(\boldsymbol{\beta}, \boldsymbol{\alpha}\right) = \sum_{i=1}^M f\left(\frac{\beta_i}{\alpha_i}\right)\alpha_i, \qquad (2)$$

for any $\boldsymbol{\alpha}, \boldsymbol{\beta} \in \triangle_M$, where f is a lower semi-continuous and convex function with $f(1) = 0$ with the convention that $0 \times f\left(\frac{\cdot}{0}\right) = 0$. While this family of loss functions is very large, we emphasize that by setting $f(t) = t\log t$, we obtain the *cross-entropy loss* widely used in deep learning and machine learning.

We interest in the functions f such that the corresponding loss function ℓ is a *proper* loss (Reid and Williamson, 2010; Williamson et al., 2016) in the sense that

$$\text{argmin}_{\boldsymbol{\alpha}\in\triangle_M} \sum_{y=1}^M \ell\left(y, \boldsymbol{\alpha}\right)\boldsymbol{\beta}_y = \boldsymbol{\beta},$$

for any $\boldsymbol{\beta} = \left[\boldsymbol{\beta}_y\right]_{i=1}^M \in \triangle_M$.

Reid and Williamson (2010); Williamson et al. (2016) indicate the necessary and sufficient conditions for which a loss ℓ is a *proper* loss. Building upon the results of (Reid and Williamson, 2010; Williamson et al., 2016), we find sufficient conditions imposed on the function f so that the loss function ℓ is a proper loss.

Proposition 1. *If the function f is strictly convex and increasing, the loss function ℓ is a proper loss. In addition, the cross-entropy loss is a proper loss belonging to the family of interest.*

A proof can be found in Appendix A. We note that comprehensive studies of proper loss can be found in (Williamson et al., 2016) for interested readers. In what follows, we develop the guaranteed bounds on attack/defense w.r.t the loss function ℓ.

3 Theoretical results on guaranteed bounds

We first establish key results on the loss function and divergence followed by our guaranteed bounds on attack/defense. We then provide a more general form of the bounds on an intermediate space in which we assume that the hypotheses h decompose into the composition of two functions (i.e., a feature extractor and a classifier). Finally, detailed implications of our theory will be developed and presented together with the experiments in the experimental section.

3.1 Results for the loss function and divergence

We depart with a definition of *multi-distributional* $f-$divergence to measure the divergence among M distributions $p_1, ..., p_M$ as follows:

$$D_\ell^{\boldsymbol{\pi}}\left(p_1, ..., p_M\right) = g_\ell^{\boldsymbol{\pi}}\left(1, ..., 1\right) \qquad (3)$$
$$- \int g_\ell^{\boldsymbol{\pi}}\left(\frac{p_1\left(\mathbf{x}\right)}{p\left(\mathbf{x}\right)}, \ldots, \frac{p_M\left(\mathbf{x}\right)}{p\left(\mathbf{x}\right)}\right) p\left(\mathbf{x}\right) d\mathbf{x},$$

where $p(\mathbf{x}) := \sum_{k=1}^M \pi_k p_k\left(\mathbf{x}\right)$ with $\boldsymbol{\pi} > \mathbf{0}$ and $g_\ell^{\boldsymbol{\pi}} : \mathbb{R}^M \to \mathbb{R}$ is defined as

$$g_{\ell}^{\boldsymbol{\pi}}\left(\boldsymbol{\tau}\right):=\min_{\boldsymbol{\alpha}\in\triangle_{M}}\sum_{k=1}^{M}\boldsymbol{\pi}_{k}\ell\left(y=k,\boldsymbol{\alpha}\right)\tau_{k},\tag{4}$$

for any $\boldsymbol{\tau}\in\mathbb{R}^{M}$.

Theoretically, $D_{\ell}^{\boldsymbol{\pi}}$ defined as above is a proper divergence in the sense that $D_{\ell}^{\boldsymbol{\pi}}\left(p_{1:M}\right)\geq0$ and $D_{\ell}^{\boldsymbol{\pi}}\left(p_{1:M}\right)=0$ if only if $p_{1}=p_{2}=...=p_{M}$, given appropriate conditions on ℓ as shown in Lemma 2.

Lemma 2. *Assume the loss function ℓ is as in Eq. (2) in which the function $f\left(t\right)$ and $-t^{-1}f\left(t\right)$ $(t>0)$ are convex, lower semi-continuous, and $f\left(1\right)=f\left(0\right)=0$, then for any $p_{1:M}$, we have $D_{\ell}^{\boldsymbol{\pi}}\left(p_{1:M}\right)\geq0$ and $D_{\ell}^{\boldsymbol{\pi}}\left(p_{1:M}\right)=0$ if only if $p_{1}=p_{2}=...=p_{M}$. Moreover, $D_{\ell}^{\boldsymbol{\pi}}\left(p_{1:M}\right)$ can be explicitly represented as*

$$D_{\ell}^{\boldsymbol{\pi}}\left(p_{1:M}\right)=\sum_{k=1}^{M}\pi_{k}D_{u_{k}}\left(p,p_{k}\right),$$

where $D_{u_{k}}$ is the standard f-divergence w.r.t. u_{k} and the function u_{k} is defined as

$$\begin{aligned}u_{k}\left(t\right)&:=-\pi_{k}t^{-1}f\left(t\pi_{k}^{-1}\right)+f\left(\pi_{k}^{-1}\right)\pi_{k}\\&=-\left(t\pi_{k}^{-1}\right)^{-1}f\left(t\pi_{k}^{-1}\right)+f\left(\pi_{k}^{-1}\right)\pi_{k}.\end{aligned}\tag{5}$$

Remark 3. u_{k} defined in Eq. (5) is a convex function since $z(t)=-t^{-1}f\left(t\right)$ is convex, $v_{k}\left(t\right)=t\pi_{k}^{-1}$ is a linear function, and $u_{k}=z\circ v_{k}+\text{const}$ ($\circ$ specifies the function composition operator). We note that due to $\boldsymbol{\pi}>\mathbf{0}$ the definitions of $u_{1:M}$ are valid.

Remark 4. It is worth noting that our multi-distributional $f-$divergence uses the standard f-divergence as its building block. Especially, when $M=2$, the multi-distributional $f-$divergence between two distributions reduces to a sum of two standard f-divergences as $D_{\ell}^{\boldsymbol{\pi}}\left(p_{1},p_{2}\right)=\pi_{1}D_{u_{1}}\left(p,p_{1}\right)+\pi_{2}D_{u_{2}}\left(p,p_{2}\right)$ where $p:=\pi_{1}p_{1}+\pi_{2}p_{2}$.

For the cross-entropy loss ℓ, the corresponding function $f\left(t\right)=t\log t$ satisfies the conditions stated in Lemma 2; hence the concrete form for $D_{\ell}^{\boldsymbol{\pi}}\left(p_{1},...,p_{M}\right)$ can be derived in the next lemma.

Lemma 5. *For the cross-entropy loss ℓ, the multi-f divergence has the following form*

$$D_{\ell}^{\boldsymbol{\pi}}\left(p_{1},...,p_{M}\right)=JS^{\boldsymbol{\pi}}\left(p_{1},...,p_{M}\right),$$

where $JS^{\boldsymbol{\pi}}\left(p_{1},...,p_{M}\right):=\sum_{k=1}^{M}\pi_{k}D_{KL}\left(p_{k},p\right)$ with $p=\sum_{k=1}^{M}\pi_{k}p_{k}$ is the Jensen-Shannon divergence.

Note that our developed f**-**divergences has the same characteristic with that developed in Duchi et al. (2018) by being aware of the divergence between some distributions using a family of classifiers. However, the technical detail and specification of our f**-**divergences is totally different. In addition, our divergence is more specific and intuitive by linking with our specific loss $\ell\left(\beta,\alpha\right)$ and the standard f-divergence as shown in Lemma 2.

3.2 Attack/defense guaranteed bounds w.r.t. a data space

Recap that given a classifier $h\in\mathcal{H}$, an adversary $a\in\mathcal{A}$ when performing its attack on h transforms a clean example $\mathbf{x}$ to an adversarial example $\mathbf{x}_{a}=a_{h}\left(\mathbf{x}\right)$. Assume that this clean example has ground-truth label $y=k$ (i.e., $\mathbf{x}\sim p_{k}\left(\cdot\right)$), the attack transformation a_{h} transforms $\mathbf{x}\sim p_{k}\left(\cdot\right)$ to $\mathbf{x}_{a}=a_{h}\left(\mathbf{x}\right)$ sampled from another distribution with a density function $p_{k}^{a,h}$, named as the k-th *adversarial conditional class distribution.* In other words, $p_{k}^{a,h}$ is the density function of the distribution formed by pushing forward the k-th conditional class distribution $p_{k}\left(\cdot\right)$ via a_{h}.

In what follows, we present our main body of theory regarding the attack/defense guaranteed bounds. Since the adversaries in $\mathcal{A}$ always attempt to increase the attack/defense risk $\mathcal{L}\left(\mathcal{A},\mathcal{H},\mathcal{D}\right)$ (cf. Eq. (1)), from the attack perspective, it is more interesting to investigate a tight lower bound of this quantity given in the following theorem.

Theorem 6. *For any adversary family $\mathcal{A}$, classifier family $\mathcal{H}$, and a loss function ℓ w.r.t. an increasing function f satisfying the conditions in Lemma 2, we have*

$$\begin{aligned}\mathcal{L}\left(\mathcal{A},\mathcal{H},\mathcal{D}\right)\geq&\sum_{k=1}^{M}\ell\left(y=k,\boldsymbol{\pi}\right)\pi_{k}\\&-\inf_{a\in\mathcal{A}}\sup_{h\in\mathcal{H}}D_{\ell}^{\boldsymbol{\pi}}\left(p_{1}^{a,h},...,p_{M}^{a,h}\right),\end{aligned}\tag{6}$$

where for any $1\leq k\leq M$ and $p_{k}^{a,h}$ is the corresponding adversarial distribution.

Remark 7. We can obtain more explicit expressions for the lower bound in Eq. (6) as follows:

i) Referring to Lemma 2, the lower bound can be rewritten in a more explicit form

$$\sum_{k=1}^{M}\ell\left(y=k,\boldsymbol{\pi}\right)\pi_{k}-\inf_{a\in\mathcal{A}}\sup_{h\in\mathcal{H}}\sum_{k=1}^{M}\pi_{k}D_{u_{k}}\left(p^{a,h},p_{k}^{a,h}\right),$$

where we recall the definition of the convex function u_{k} as $u_{k}\left(t\right)=-\pi_{k}t^{-1}f\left(t\pi_{k}^{-1}\right)+f\left(\pi_{k}^{-1}\right)\pi_{k}$.

ii) In particular, for the case of the cross-entropy loss (i.e., the D_{f} in the loss definition is the KL divergence), in light of Lemma 5, we can come with a more specific form for the lower bound

$$\mathbb{H}\left(\boldsymbol{\pi}\right)-\inf_{a\in\mathcal{A}}\sup_{h\in\mathcal{H}}JS^{\boldsymbol{\pi}}\left(p_{1}^{a,h},...,p_{M}^{a,h}\right),$$

where $\mathbb{H}(\cdot)$ represents the Shannon entropy (Shannon, 1948).

Remark 8. Our lower bound is very appealing and intuitive to interpret from a game theory perspective. Given an adversary $a \in \mathcal{A}$, the classifier family $\mathcal{H}$ tries to defend by selecting the optimal classifier h_a^* for simultaneously maximizing $D_\ell^{\boldsymbol{\pi}}\left(p_1^{a,h}, ..., p_M^{a,h}\right)$ and decreasing the lower bound. By maximizing $D_\ell^{\boldsymbol{\pi}}\left(p_1^{a,h}, ..., p_M^{a,h}\right)$, we aim to choose the classifier h_a^* so that when a performs its attack to this classifier, the corresponding adversarial distributions w.r.t. the classes become more distant (i.e., the classes of adversarial examples become more separate for better robust accuracy). Meanwhile, from an attack perspective, the adversaries desire to increase the attack/defense risk $\mathcal{L}(\mathcal{A}, \mathcal{H}, \mathcal{D})$ by decreasing $D_\ell^{\boldsymbol{\pi}}\left(p_1^{a,h}, ..., p_M^{a,h}\right)$, thereby making $p_1^{a,h}, ..., p_M^{a,h}$ more intermingling. More specifically, for a given specific adversary $a \in \mathcal{A}$, we choose the best classifier $h_a^* \in \mathcal{H}$ to approximate $\sup_{h\in\mathcal{H}} D_\ell^{\boldsymbol{\pi}}\left(p_1^{a,h}, ..., p_M^{a,h}\right)$ that can best defend its attack and then choose the adversary a^* to approximate $\inf_{a\in\mathcal{A}} D_\ell^{\boldsymbol{\pi}}\left(p_1^{a,h_a^*}, ..., p_M^{a,h_a^*}\right)$ (cf. Figure 1).

From the defense perspective, it is appealing to develop an upper bound for the attack/defense risk $\mathcal{L}(\mathcal{A}, \mathcal{H}, \mathcal{D})$, which is provided in the following theorem.

Theorem 9. *We can upper-bound the attack/defense risk:*

$$\mathcal{L}(\mathcal{A}, \mathcal{H}, \mathcal{D}) \leq \inf_{h\in\mathcal{H}} \Bigg(\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell(y, h(\mathbf{x}))^2\right]^{1/2} + \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell(y, h(\mathbf{x}))^2\right]^{1/2} \sup_{a\in\mathcal{A}} \left[\sum_{k=1}^{M} \pi_k D_v\left(p_k^{a,h}, p_k\right)\right]^{1/2} \Bigg),$$

where D_v is the standard f-divergence w.r.t. v with $v(t) = (t-1)^2$.

Remark 10. The obtained upper bound is also intuitive and might help us to devise robust defense models. As indicated by Theorem 9, to minimize the attack/defense risk $\mathcal{L}(\mathcal{A}, \mathcal{H}, \mathcal{D})$, we need to find the optimal classifier h to minimize $\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell(y, h(\mathbf{x}))^2\right]$, while minimizing $\sup_{a\in\mathcal{A}}\left[\sum_{k=1}^{M} \pi_k D_v\left(p_k^{a,h}, p_k\right)\right]^{1/2}$. This implies finding the classifier h that can predict well the clean examples, while keeping the conditional class adversarial distributions (i.e., $p_{1:K}^{a,h}$) as close to their corresponding conditional class distributions ($p_{1:K}$) as possible (cf. Figure 1).

3.3 Attack/defense guaranteed bounds w.r.t. a latent space

We impose a structure on the classifiers in the family $\mathcal{H}$. More specifically, we consider and investigate the composite classifiers $h = h_2 \circ h_1$ (i.e., $h(\mathbf{x}) = h_2(h_1(\mathbf{x}))$), where $h_1 : \mathcal{X} \to \mathcal{Z}$ and $h_2 : \mathcal{Z} \to \Delta_M$ in which $\mathcal{X}$ is the data space and $\mathcal{Z}$ is an intermediate space. Given a composite classifier h, h_1 is known as a *feature extractor* that maps input data to intermediate representations (i.e., on an intermediate layer of h) whose space is $\mathcal{Z}$, while h_2 acting on $\mathcal{Z}$ is known as a *classifier*. In the sequel, we develop our bounds w.r.t. the latent space $\mathcal{Z}$, which allows us to develop attack and defense methods on the latent space. We note that the bounds w.r.t. the latent space are really useful and necessary because defense on the latent space corresponding to an intermediate layer of a deep net shows advantages (see (Xie et al., 2019; Bui et al., 2020, 2021a) for the comprehensive discussions).

We endow some new notions w.r.t. the latent space. Given an clean example $\mathbf{x}$ and a classifier h, when performing an attack, an adversary a moves $\mathbf{x}$ to $\mathbf{x}_a = a_h(\mathbf{x})$ on the data space $\mathcal{X}$, which is analogical to move $\mathbf{z} = h_1(\mathbf{x})$ to $\mathbf{z}_a = h_1(\mathbf{x}_a) = h_1(a_h(\mathbf{x}))$ on the latent space. Recap that for a class k, $p_k^{a,h}$ is the density of the distribution induced by push-forwarding the conditional class distribution p_k via a_h, the feature extractor h_1 push-forwards $p_k^{a,h}$ to another distribution on the latent space with the density function $q_k^{a,h}$. Similarly, h_1 also push-forwards the conditional class distribution p_k to a distribution on the latent space with the density function q^k. We are now ready to restate our upper and lower bounds w.r.t. the latent space.

Theorem 11. *For any adversary family $\mathcal{A}$, classifier family $\mathcal{H}$, and a loss function ℓ w.r.t. an increasing function f satisfying the conditions in Lemma 2, we have*

$$\mathcal{L}(\mathcal{A}, \mathcal{H}, \mathcal{D}) \geq \sum_{k=1}^{M} \ell(y = k, \boldsymbol{\pi}) \pi_k - \inf_{a\in\mathcal{A}} \sup_{h\in\mathcal{H}} D_\ell^{\boldsymbol{\pi}}\left(q_1^{a,h}, ..., q_M^{a,h}\right). \tag{7}$$

Remark 12. Theorem 11 generalizes the lower bound w.r.t. a latent space. It turns out that the lower bound w.r.t. the latent space totally depends on the divergence among the adversarial conditional class distributions (i.e., $q_{1:M}^{a,h}$) on the latent space (cf. Figure 1).

The following theorem restates the upper bound w.r.t. the latent space.

Theorem 13. *We can upper-bound the attack/defense risk:*

$$\mathcal{L}(\mathcal{A}, \mathcal{H}, \mathcal{D}) \leq \inf_{h\in\mathcal{H}} \Bigg(\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell(y, h(\mathbf{x}))^2\right]^{1/2} + \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell(y, h(\mathbf{x}))^2\right]^{1/2} \sup_{a\in\mathcal{A}} \left[\sum_{k=1}^{M} \pi_k D_v\left(q_k^{a,h}, q_k\right)\right]^{1/2} \Bigg),$$

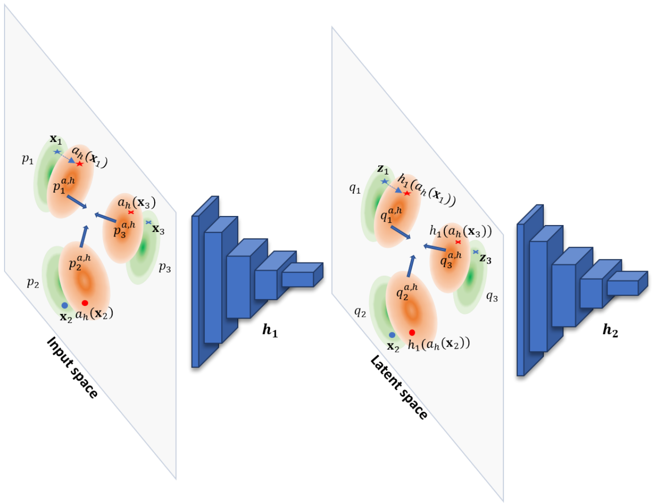


Figure 1: An illustration of our proposed theory on attack/defense risk bound. On the input space, to attack a classifier h, an adversary a tries to mix up adversarial examples by minimizing a divergence among the conditional class adversarial distributions $p_{1:3}^{a,h}$, while the classifier h tries to keep $p_k^{a,h}$ as closest to p_k possible ($1 \le k \le 3$) to defend. Similar phenomenons happen on the latent space but w.r.t. the induced distributions $q_{1:3}$ and $q_{1:3}^{a,h}$ via the feature extractor h_1.

where D_v is the standard f-divergence w.r.t. v with $v(t) = (t-1)^2$.

Remark 14. Theorem 13 generalizes the upper bound w.r.t. a latent space. It keeps the spirit of Theorem 9 in which we need to learn a classifier h that can predict well the clean examples, while keeping the conditional class adversarial distributions (i.e., $q_{1:K}^{a,h}$) as close to their corresponding conditional class distributions $q_{1:K}$ as possible on latent space (cf. Figure 1).

The implications of our theory. To promote defense methods inspired from our theoretical findings in the next section, we consider the case of a single adversarial a trying to attack a family of classifiers $\mathcal{H}$. We recap how a can benefit from the global view for attack to craft dangerous adversarial examples and how the classifiers can advantage from this global view to improve adversarial robustness.

- **Implications for adversary**. Adversary a should generate adversarial examples globally far away benign data distribution to maximize $\sum_{k=1}^{M} D_v(q_k^{a,h}, q_k)$ and globally class intermingling to minimize $D_\ell^{\boldsymbol{\pi}}\left(q_1^{a,h}, ..., q_M^{a,h}\right)$.

- **Implications for classifiers**. In contrast, for composite defense classifiers $h = h_2 \circ h_1$, the feature extractor should produce latent representations to minimize $\sum_{k=1}^{M} D_v(q_k^{a,h}, q_k)$ and maximize $D_\ell^{\boldsymbol{\pi}}\left(q_1^{a,h}, ..., q_M^{a,h}\right)$, while the classifier h_2 on top is trained to predict accurately adversarial and benign examples.

4 Global-view based defense

In this section, we try to exploit the implications of our theory to propose a global-view based defense method. We emphasize that we do not seek the state-of-the-art results. In fact, our aim here is to demonstrate that it is beneficial to exploit the global-view attack and defense to further improve the current state-of-the-art adversarial training methods, .e.g., PGD (Madry et al., 2017) and TRADES (Zhang et al., 2019). As suggested by Theorems 11 and 13, a dangerous adversary a tries to produce dangerous and harmful adversarial examples by maximizing

$$\sum_{k=1}^{M} D_v(q_k^{a,h}, q_k) - \lambda D_\ell^{\boldsymbol{\pi}}\left(q_1^{a,h}, ..., q_M^{a,h}\right) \qquad (8)$$

on a latent space, while the objective of defense is to learn the feature extractor h_1 and the classifier h_2 so that $h = h_2 \circ h_1$ can mitigate the danger and defend well those adversarial examples.

It is transparent that by maximizing $\sum_{k=1}^{M} D_v(q_k^{a,h}, q_k)$, we globally keep adversarial examples further to benign examples, hence making them harder to be recognized. Moreover, by minimizing $D_\ell^{\boldsymbol{\pi}}\left(q_1^{a,h}, ..., q_M^{a,h}\right)$, we globally push adversarial examples with different classes more intermingling, hence making the task of the classifier h_2 more challenging. We reckon this viewpoint as a global-view attack, which is useful for us to craft more dangerous and harmful adversarial examples.

Furthermore, to tackle $D_\ell^{\boldsymbol{\pi}}\left(q_1^{a,h}, ..., q_M^{a,h}\right)$, we use the cross-entropy loss, hence $D_\ell^{\boldsymbol{\pi}}\left(q_1^{a,h}, ..., q_M^{a,h}\right)$ reduces exactly to $JS^{\pi}\left(q_1^{a,h},...,q_M^{a,h}\right)$. We employ a *multi-class discriminator* h_d (i.e., a multi-class classifier) that can distinguish class labels of adversarial examples. It is well-known that the following inequality in (9) holds (Hoang et al., 2018)

$$\begin{aligned}&JS^{\pi}\left(q_1^{a,h},...,q_M^{a,h}\right)\\ &\geq -\min_{h_d}\left\{\sum_{k=1}^{M}\pi_k E_{\mathbf{z}_a\sim q_k^{a,h}}\left[\mathrm{CE}(h_d(\mathbf{z}_a),k)\right]\right\}+\mathbb{H}\left(\boldsymbol{\pi}\right)\\ &=-\min_{h_d}\left\{\sum_{k=1}^{M}\pi_k E_{\mathbf{x}_a\sim p_k^{a,h}}\left[\mathrm{CE}(h_d(h_1\left(\mathbf{x}_a\right)),k)\right]\right\}+\mathbb{H}\left(\boldsymbol{\pi}\right),\end{aligned} \tag{9}$$

where CE represents the cross-entropy divergence.

In addition, the inequality (see proof in Appendix A) in (9) is tight if the family to search for h_d has members to approach h_d^* with $h_d^*\left(\mathbf{z},k\right)=\frac{q_k^{a,h}(\mathbf{z})}{\sum_{j=1}^{M}q_j^{a,h}(\mathbf{z})},\forall k=1,...,M$ up to any level of precision or contains h_d^*. Note that we use $h_d^*\left(\mathbf{z},k\right)$ to represent the k-th component of $h_d^*\left(\mathbf{z}\right)$. Similar to the derivation in GAN (Goodfellow et al., 2014a), we use the right hand side of the inequality in (9) to approximate the quantity of interest $JS^{\pi}\left(q_1^{a,h},...,q_M^{a,h}\right)$.

To tackle $\sum_{k=1}^{M}D_v(q_k^{a,h},q_k)$, to take advantage from the well-known efficiency of GAN (Goodfellow et al., 2014a) in estimating the JS divergence, we replace it with $\sum_{k=1}^{M}JS^{0.5,0.5}(q_k^{a,h},q_k)$ and derive $\sum_{k=1}^{M}JS^{0.5,0.5}(q_k^{a,h},q_k)$ as

$$\begin{aligned}&\sum_{k=1}^{M}JS^{0.5,0.5}(q_k^{a,h},q_k)\\ &\geq 0.5\sum_{k=1}^{M}\max_{T_k}\Big\{\mathbb{E}_{\mathbf{z}\sim q_k}\left[\log T_k\left(\mathbf{z}\right)\right]\\ &\quad+\mathbb{E}_{\mathbf{z}_a\sim q_k^{a,h}}\left[\log\left(1-T_k\left(\mathbf{z}_a\right)\right)\right]+2\log 2\Big\}\\ &=0.5\sum_{k=1}^{M}\max_{T_k}\Big\{\mathbb{E}_{\mathbf{x}\sim p_k}\left[\log T_k\left(h_1\left(\mathbf{x}\right)\right)\right]\\ &\quad+\mathbb{E}_{\mathbf{x}_a\sim p_k^{a,h}}\left[\log\left(1-T_k\left(h_1\left(\mathbf{x}_a\right)\right)\right)\right]+2\log 2\Big\},\end{aligned} \tag{10}$$

where $T_k,k=1,...,M$ is the discriminator that distinguishes the samples from q_k and $q_k^{a,h}$.

In addition, the inequality (see proof in Appendix A) in (10) is tight if the family to search each $T_k,k=1,...,M$ has members to approach T_k^* with $T_k^*\left(\mathbf{z}\right)=\frac{q_k(\mathbf{z})}{q_k(\mathbf{z})+q_k^{a,h}(\mathbf{z})}$ up to any level of precision or contains T_k^*. Similar to the derivation in GAN (Goodfellow et al., 2014a), we use the right hand side of the inequality in (10) to approximate the quantity of interest $\sum_{k=1}^{M}JS^{0.5,0.5}(q_k^{a,h},q_k)$.

Using the above approximations, assume that we use sufficiently strong families for $T_{1:M}$ and h_d, and can train them to reach closely their optimums, we then find the adversarial examples by maximizing $\sum_{k=1}^{M}JS^{0.5,0.5}(q_k^{a,h},q_k)-\lambda JS^{\pi}\left(q_1^{a,h},...,q_M^{a,h}\right)$ as

$$\max_{\mathbf{x}'\in\mathcal{B}(\mathbf{x},\epsilon)}\left\{F(\mathbf{x},\mathbf{x}';h_d,T_{1:M})\right\},$$

where $\mathcal{B}(\mathbf{x},\epsilon)$ is a ball with center x and radius ϵ w.r.t. a norm $\|\cdot\|$ and we have defined

$$\begin{aligned}F(\mathbf{x},\mathbf{x}';h_d,T_{1:M})&:=\lambda\sum_{k=1}^{M}\mathrm{CE}(h_d(h_1\left(\mathbf{x}'\right)),k)\\ &+\sum_{k=1}^{M}\left[\log T_k\left(h_1\left(\mathbf{x}\right)\right)+\log\left(1-T_k\left(h_1\left(\mathbf{x}'\right)\right)\right)\right].\end{aligned}$$

We summarize the key steps of our proposed methods as follows.

Crafting adversarial examples. Given a benign example with label $(\mathbf{x},y)$, we generate the corresponding adversarial example in two ways: (1) GV-TRADES in Eq. (11) and (2) GV-PGD in Eq. (12) as follows:

$$\begin{aligned}\mathbf{x}_a=\mathrm{argmax}_{x'\in\mathcal{B}(\mathbf{x},\epsilon)}\Big\{&\mathrm{KL}\left(h_2\left(h_1(\mathbf{x}')\right),h_2\left(h_1(\mathbf{x})\right)\right)\\ &+\alpha F\left(\mathbf{x},\mathbf{x}';h_d,T_{1:M}\right)\Big\},\end{aligned} \tag{11}$$

$$\begin{aligned}\mathbf{x}_a=\mathrm{argmax}_{x'\in\mathcal{B}(\mathbf{x},\epsilon)}\Big\{&\mathrm{CE}\left(h_2\left(h_1(\mathbf{x}')\right),y\right)\\ &+\alpha F\left(\mathbf{x},\mathbf{x}';h_d,T_{1:M}\right)\Big\},\end{aligned} \tag{12}$$

where KL represents Kullback-Leibler (KL) divergence, $\mathcal{B}(\mathbf{x},\epsilon)$ is a ball with center $\mathbf{x}$ and radius ϵ w.r.t. a norm $\|\cdot\|$, and $\alpha>0$ is a trade-off parameter.

Updating the discriminator h_d, the generator h_1, and $T_{1:M}$. We update h_d,h_1, and $T_{1:M}$ for the current mini–batch of the benign examples and the corresponding mini-batch of adversarial examples crafted in the previous step. In addition, the purpose of h_d is to classify the labels of adversarial examples, while the purpose of each $T_k,k=1,...,M$ is to distinguish the samples from q_k and $q_k^{a,h}$.

$$\begin{aligned}&\min_{h_1,h_d,T_{1:M}}\Big\{\lambda\sum_{k=1}^{M}E_{p_k^{a,h}}\left[\mathrm{CE}(h_d(h_1\left(\mathbf{x}_a\right)),k)\right]+\\ &\sum_{k=1}^{M}\Big[-\mathbb{E}_{p_k}\left[\log T_k\left(h_1\left(\mathbf{x}\right)\right)\right]-\mathbb{E}_{p_k^{a,h}}\left[\log\left(1-T_k\left(h_1\left(\mathbf{x}_a\right)\right)\right)\right]\Big]\Big\},\end{aligned}$$

Update the classifier h_2. Finally, we update the classifier h_2 to classify class labels of adversarial and benign examples for the current min–batch of the benign examples and the corresponding mini-batch of adversarial examples crafted in the previous step as follows

$$\min_{h_2}\Bigg\{\sum_{k=1}^{M}E_{p_k^{a,h}}\left[\mathrm{CE}(h_d(h_1\left(\mathbf{x}_a\right)),k)\right] + \sum_{k=1}^{M}E_{p_k}\left[\mathrm{CE}(h_d(h_1\left(\mathbf{x}\right)),k)\right]\Bigg\}.$$

In our implementation, we build up h_d and $T_{1:M}$ on top of the output of h_1. In addition, $T_{1:M}$ are shared up to the penultimate layer and only different in the last layers.

5 Experiments

5.1 Experimental Setting

In this section, we briefly summarize the experimental setting whose details can be found in Appendix B.

General Setting. We demonstrate performances on MNIST (Lecun et al., 1998) and CIFAR10 (Krizhevsky et al., 2009). The inputs were normalized to $[0, 1]$. We apply padding 4 pixels at all borders before randomly cropping and doing horizontal flips. We use a standard CNN (Carlini and Wagner, 2017) and ResNet(He et al., 2016) architecture for the MNIST and CIFAR10 dataset, respectively.

Baseline Setting. We contrast our method with the standard AT methods, i.e., PGD-AT (Madry et al., 2017), TRADES (Zhang et al., 2019). For TRADES, we use the original trade-off ratio between natural loss and robust loss as reported in (Zhang et al., 2019) (i.e., $\beta = 6$ for CIFAR10 and $\beta = 1$ for MNIST). The AT setting are $\{k = 40, \epsilon = 0.3, \eta = 0.01\}$ for the MNIST dataset, $\{k = 10, \epsilon = 8/255, \eta = 2/255\}$ for CIFAR10 dataset, where k is the iteration steps, ϵ is the distortion bound and η is the step size.

Attack Setting. We use different SOTA attacks to evaluate the defense methods including: (i) **PGD attack** (Madry et al., 2017) which is gradient based attack. The parameters $\{k, \epsilon, \eta\}$ will be described in each individual experiment. (ii) **B&B attack** (Brendel et al., 2019) which is a decision based attack. We initialized with the PGD attack with $k = 20$ and corresponding $\{\epsilon, \eta\}$ then apply B&B attack with 200 steps. (iii) **Auto-Attack (AA)** (Croce and Hein, 2020b) which is an ensemble based attack. We use $\epsilon = 0.3$ for the MNIST dataset and $\epsilon = 8/255$ for the CIFAR10 dataset, all with standard version which is an ensemble of four different attacks. The distortion metric we use in our experiments is l_∞ for all measures. We use the full test set for the PGD attack and 1,000 test samples for other attacks.

Table 1: Robustness evaluation on MNIST and CIFAR10. PGD attack with $\{k = 200, \epsilon = 0.3, \eta = 0.01\}$ for MNIST and $\{k = 200, \epsilon = 8/255, \eta = 2/255\}$ for CIFAR10.

	MNIST				CIFAR10			
	Nat	PGD	AA	B&B	Nat	PGD	AA	B&B
PGD-AT	99.4	94.0	88.9	91.3	86.4	46.0	42.5	44.2
GV-PGD	**99.5**	**96.5**	**90.5**	**93.9**	86.4	**46.4**	**45.7**	**46.9**
TRADES	**99.4**	95.1	90.9	92.2	80.8	51.9	49.1	50.2
GV-TRADES	99.3	**96.0**	**92.3**	**94.3**	**83.3**	**53.3**	**49.9**	50.2

5.2 Robustness Evaluation

We report robust accuracies of our GV-PGD against PGD-AT and GV-TRADES against TRADES. Comparing with their counterparts, GV-PGD and GV-TRADES further exploit a global view for improving robustness. We wish to show that a global view if exploited properly can improve both natural and robust accuracies. This claim is convincingly demonstrated in Table 1 in which our methods outperform their counterparts in terms of both natural and robust accuracies in all datasets.

6 Conclusion

We propose attack and defense guaranteed bounds from a game theory perspective. More specifically, we have put attackers (i.e., adversaries) and defenders (i.e., classifiers) in a game theory context and further developed guaranteed bounds for the attack and defense risk. Our guaranteed bounds have implications to both attack and defense sides, thereby being potentially useful in developing new attack and defense methods. Inspired by the theoretical findings, we propose GV-PGD and GV-TRADES which are the counterparts of PGD and TRADES in which the global-view attack/defense is employed to further improve those SOTA adversarial training techniques. Finally, we conduct experiments to demonstrate the usefulness of the global view where our GV-PGD and GV-TRADES quite significantly outperform its counterparts.

Acknowledgments. This work was partially supported by the Australian Defence Science and Technology (DST) Group under the Next Generation Technology Fund (NGTF) scheme.

Broader impact

The fact that deep learning methods are vulnerable and error-prone to small, imperceptible, adversarial perturbations, which presents one of most urgent fragility of machine learning systems on which our modern society is increasingly depending from face recognition in online digital identity verification, automatic car number plate extraction for toll charges to credit scoring in fintech sector. While we do not solve any of these problems directly in this work, its theoretical and practical results can have a significant implication in all of them. Our present work is solving a theoretical research problem in strengthening deep learning models, we believe that it does not put anyone at disadvantages.

References

Brendel, W., Rauber, J., Kümmerer, M., Ustyuzhaninov, I., and Bethge, M. (2019). Accurate, reliable and fast robustness evaluation. In *Advances in Neural Information Processing Systems*, pages 12861–12871. 5.1

Bubeck, S., Lee, Y. T., Price, E., and Razenshteyn, I. (2019). Adversarial examples from computational constraints. In Chaudhuri, K. and Salakhutdinov, R., editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 831–840, Long Beach, California, USA. PMLR. 1

Bui, A., Le, T., Zhao, H., Montague, P., Camtepe, S., and Phung, D. (2021a). Understanding and achieving efficient robustness with adversarial supervised contrastive learning. 3.3

Bui, A., Le, T., Zhao, H., Montague, P., deVel, O., Abraham, T., and Phung, D. (2020). Improving adversarial robustness by enforcing local and global compactness. In Vedaldi, A., Bischof, H., Brox, T., and Frahm, J.-M., editors, *Computer Vision – ECCV 2020*, pages 209–223, Cham. Springer International Publishing. 1, 3.3

Bui, A. T., Le, T., Tran, Q. H., Zhao, H., and Phung, D. (2022). A unified wasserstein distributional robustness framework for adversarial training. In *International Conference on Learning Representations*. 1

Bui, A. T., Le, T., Zhao, H., Montague, P., deVel, O., Abraham, T., and Phung, D. (2021b). Improving ensemble robustness by collaboratively promoting and demoting adversarial robustness. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 6831–6839. 1

Carlini, N. and Wagner, D. (2017). Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pages 39–57. IEEE. 1, 5.1, B.1

Cranko, Z., Menon, A., Nock, R., Ong, C. S., Shi, Z., and Walder, C. (2019). Monge blunts bayes: Hardness results for adversarial training. In Chaudhuri, K. and Salakhutdinov, R., editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 1406–1415, Long Beach, California, USA. PMLR. 1

Croce, F. and Hein, M. (2020a). Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *Proceedings of the 37th International Conference on Machine Learning (ICML) 2020*, volume 119 of *Proceedings of Machine Learning Research*, pages 2206–2216. PMLR. 2

Croce, F. and Hein, M. (2020b). Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. *arXiv preprint arXiv:2003.01690*. 5.1

Cullina, D., Bhagoji, A. N., and Mittal, P. (2018). Pac-learning in the presence of evasion adversaries. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 228–239, Red Hook, NY, USA. Curran Associates Inc. 1

Duchi, J., Khosravi, K., Ruan, F., et al. (2018). Multiclass classification, information, divergence and surrogate risk. *Annals of Statistics*, 46(6B):3246–3275. 3.1

Fawzi, A., Fawzi, H., and Fawzi, O. (2018). Adversarial vulnerability for any classifier. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 1186–1195. 1

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014a). Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680. 4, 4

Goodfellow, I. J., Shlens, J., and Szegedy, C. (2014b). Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*. 1, 2

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778. 1, 5.1

Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T. N., and Kingsbury, B. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97. 1

Hoang, Q., Le, T., and Phung, D. (2020). Parameterized rate-distortion stochastic encoder. In III, H. D. and Singh, A., editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 4293–4303. PMLR. 1

Hoang, Q., Nguyen, T. D., Le, T., and Phung, D. (2018). MGAN: Training generative adversarial nets with multiple generators. In *International Conference on Learning Representations*. 4

Kannan, H., Kurakin, A., and Goodfellow, I. (2018). Adversarial logit pairing. *arXiv preprint arXiv:1803.06373*. 1

Krizhevsky, A. et al. (2009). Learning multiple layers of features from tiny images. 5.1

Kurakin, A., Goodfellow, I., and Bengio, S. (2016). Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*. 1, 2

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, pages 2278–2324. 5.1

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2017). Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*. 1, 2, 4, 5.1

Nguyen-Duc, T., Le, T., Zhao, H., Cai, J., and Phung, D. (2022). Particle-based adversarial local distribution regularization. In *International Conference on Artificial Intelligence and Statistics*. PMLR. 1

Qin, C., Martens, J., Gowal, S., Krishnan, D., Dvijotham, K., Fawzi, A., De, S., Stanforth, R., and Kohli, P. (2019). Adversarial robustness through local linearization. In *Advances in Neural Information Processing Systems*, pages 13824–13833. 1

Reid, M. D. and Williamson, R. C. (2010). Composite binary losses. *Journal of Machine Learning Research*, 11(Sep):2387–2422. 2, A.1

Schmidt, L., Santurkar, S., Tsipras, D., Talwar, K., and Madry, A. (2018). Adversarially robust generalization requires more data. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 31*, pages 5014–5026. 1

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423, 623–656. 7

Spencer, M., Eickholt, J., and Cheng, J. (2015). A deep learning network approach to ab initio protein secondary structure prediction. *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, 12(1):103–112. 1

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. (2013). Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*. 1

Tsipras, D., Santurkar, S., Engstrom, L., Turner, A., and Madry, A. (2019). Robustness may be at odds with accuracy. In *International Conference on Learning Representations*. 1

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008. 1

Wei, C. and Ma, T. (2020). Improved sample complexities for deep neural networks and robust classification via an all-layer margin. In *International Conference on Learning Representations*. 1

Williamson, R. C., Vernet, E., and Reid, M. D. (2016). Composite multiclass losses. *Journal of Machine Learning Research*, 17(222):1–52. 2, 2, A.1

Xie, C., Wu, Y., Maaten, L. v. d., Yuille, A. L., and He, K. (2019). Feature denoising for improving adversarial robustness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 501–509. 1, 3.3

Zhang, H., Yu, Y., Jiao, J., Xing, E., Ghaoui, L., and Jordan, M. I. (2019). Theoretically principled tradeoff between robustness and accuracy. *arXiv preprint arXiv:1901.08573*. 1, 4, 5.1

Zhao, H., Le, T., Montague, P., De Vel, O., Abraham, T., and Phung, D. (2019). Perturbations are not enough: Generating adversarial examples with spatial distortions. *arXiv preprint arXiv:1910.01329*. 2

Zhao, H., Nguyen, T., Le, T., Montague, P., De Vel, O., Abraham, T., and Phung, D. (2021). Learning to attack with fewer pixels: A probabilistic post-hoc framework for refining arbitrary dense adversarial attacks. *arXiv preprint arXiv:2010.06131*. 2

Appendix for On Global-view Based Defense via Adversarial Attack and Defense Risk Guaranteed Bounds

Since our paper relies heavily on the technical rigor to establish its main results and due to page limit on the main text, this appendix aims to provide additional details for all proofs presented in the main paper and additional experiments. Particularly, this appendix is organized as

- In Appendix A, we present all proofs regarding our theoretical development. At some places, to bring a coherent context, we briefly summarize key contexts from the main paper as well for readability.
- In Appendix B, we provide additional experiments, training specification, and ablation studies.

A Theoretical Development

A.1 Preliminaries

Reid and Williamson (2010); Williamson et al. (2016) indicate the necessary and sufficient conditions for which a loss ℓ is a *proper* loss. Building upon the results of (Reid and Williamson, 2010; Williamson et al., 2016), we find sufficient conditions imposed on the function f so that the loss function ℓ is a proper loss. We now introduce the notions and theory in Williamson et al. (2016) that are necessary for our sufficient conditions.

Given $c \in \bar{\Delta}_M$ (i.e., the interior of the M-simplex), we define

$$\mathcal{J}_i\left(\boldsymbol{c}\right) = \left\{\boldsymbol{p} \in \Delta_M : p_i c_j \geq p_j c_i,\, \forall j \neq i\right\} \text{ for} 1 \leq i \leq M.$$

Definition. We say a loss function ℓ to be c- calibrated at $\boldsymbol{p} \in \Delta_M$ if for $i \in \{1, ..., M\}$ such that $\boldsymbol{p} \notin \mathcal{J}_i\left(\boldsymbol{c}\right)$ then $\forall \boldsymbol{q} \in \mathcal{J}_i\left(\boldsymbol{c}\right)$

$$\underline{\ell}\left(\boldsymbol{p}\right) = \inf_{\boldsymbol{r} \in \Delta_M} \ell\left(\boldsymbol{r}, \boldsymbol{p}\right) < \ell\left(\boldsymbol{q}, \boldsymbol{p}\right).$$

In addition, a loss function ℓ is said to be c- calibrated if it is c-calibrated at $\boldsymbol{p}$ for all $\boldsymbol{p} \in \Delta_M$.

The following proposition indicates the link of calibration and properness (i.e., Proposition 3.3 in Williamson et al. (2016)).

Proposition. *A continuous function loss ℓ is strictly proper if only if it is c-calibrated for all $\boldsymbol{c} \in \bar{\Delta}_M$.*

Proof of Proposition 1

Given $c \in \bar{\Delta}_M$, we prove that the loss function ℓ is c-calibrated. Take $\boldsymbol{p} \in \Delta_M$, denote $1 \leq i \leq M$ as largest index such that $\boldsymbol{p} \notin \mathcal{J}_i\left(\boldsymbol{c}\right)$, and let $\boldsymbol{q} \in \mathcal{J}_i\left(\boldsymbol{c}\right)$ We need to prove that

$$\underline{\ell}\left(\boldsymbol{p}\right) = \inf_{\boldsymbol{r} \in \Delta_M} \ell\left(\boldsymbol{r}, \boldsymbol{p}\right) < \ell\left(\boldsymbol{q}, \boldsymbol{p}\right).$$

Assume by contradiction that

$$\underline{\ell}\left(\boldsymbol{p}\right) = \inf_{\boldsymbol{r} \in \Delta_M} \ell\left(\boldsymbol{r}, \boldsymbol{p}\right) = \ell\left(\boldsymbol{q}, \boldsymbol{p}\right). \tag{13}$$

Without the loss of generalization, we assume that $0 \leq p_1 \leq p_2 \leq .... \leq p_{M-1} \leq p_M$. Given $1 \leq u < v \leq M$, we then have $p_u < p_v$ or $p_u = p_v$. We examine these cases.

* **Case 1** ($p_u < p_v$): we further assume that $q_u > q_v$.

Since f is strictly increasing, we have

$$(p_u - p_v)\left[f\left(\frac{1}{q_u}\right) - f\left(\frac{1}{q_v}\right)\right] > 0$$
$$p_u f\left(\frac{1}{q_u}\right) + p_v f\left(\frac{1}{q_v}\right) > p_u f\left(\frac{1}{q_v}\right) + p_v f\left(\frac{1}{q_u}\right).$$

Therefore, by choosing $\boldsymbol{q}'$ such that $q'_1 = q_1, ..., q'_{u-1} = q_{u-1}, q'_u = q_v, q'_{u+1} = q_{u+1}, ..., q'_{v-1} = q_{v-1}, q'_v = q_u, ..., q'_M = q_M$, we then have

$$\ell\left(\boldsymbol{q}, \boldsymbol{p}\right) > \ell\left(\boldsymbol{q}', \boldsymbol{p}\right),$$

which is contradict to (13). Therefore, $q_u \leq q_v$.

* **Case 2** ($p_u = p_v$): we assume that $q_u \neq q_v$.

Since f is strictly convex and increasing, we have

$$\begin{aligned} p_u f\left(\frac{1}{q_u}\right) + p_v f\left(\frac{1}{q_v}\right) &= 2p_u\left[\frac{1}{2}f\left(\frac{1}{q_u}\right) + \frac{1}{2}f\left(\frac{1}{q_v}\right)\right] \\ &> 2p_u\left[f\left(\frac{1}{2}\left(\frac{1}{q_u} + \frac{1}{q_v}\right)\right)\right] \\ &\overset{(1)}{>} 2p_u\left[f\left(\frac{2}{q_u + q_v}\right)\right] \\ &= p_u f\left(\frac{1}{\left(q_u + q_v\right)/2}\right) + p_v f\left(\frac{1}{\left(q_u + q_v\right)/2}\right). \end{aligned}$$

Note that we have the inequality (1) due to

$$\frac{1}{2}\left(\frac{1}{q_u} + \frac{1}{q_v}\right) > \frac{2}{q_u + q_v}.$$

Therefore, by choosing $\boldsymbol{q}'$ such that $q'_1 = q_1, ..., q'_{u-1} = q_{u-1}, q'_u = \frac{q_u + q_v}{2}, q'_{u+1} = q_{u+1}, ..., q'_{v-1} = q_{v-1}, q'_v = \frac{q_u + q_v}{2}, ..., q'_M = q_M$, we then have

$$\ell\left(\boldsymbol{q}, \boldsymbol{p}\right) > \ell\left(\boldsymbol{q}', \boldsymbol{p}\right),$$

which is contradict to (13). Therefore, $q_u = q_v$.

We reach the property if $u < v$ then $q_u \leq q_v$. This means that $q_1 \leq q_2 \leq .... \leq q_M$.

We further prove that $\frac{p_u}{q_u} \leq \frac{p_v}{q_v}$. Indeed, it is obvious if $q_u = q_v$ due to $p_u \leq p_v$. We consider the case $q_u < q_v$. We further assume that $\frac{p_u}{q_u} > \frac{p_v}{q_v}$ and derive as follows:

$$\begin{aligned} p_u f\left(\frac{1}{q_u}\right) + p_v f\left(\frac{1}{q_v}\right) &= \left(p_u + p_v\right)\left[\frac{p_u}{p_u + p_v} f\left(\frac{1}{q_u}\right) + \frac{p_v}{p_u + p_v} f\left(\frac{1}{q_v}\right)\right] \\ &\geq \left(p_u + p_v\right)\left[f\left(\frac{p_u}{p_u + p_v}\frac{1}{q_u} + \frac{p_v}{p_u + p_v}\frac{1}{q_v}\right)\right] \\ &= \left(p_u + p_v\right)\left[f\left(\frac{q_u + q_v}{p_u + p_v}\left(\frac{p_u}{q_u} + \frac{p_v}{q_v}\right)\frac{1}{q_u + q_v}\right)\right] \\ &\overset{(2)}{>} \left(p_u + p_v\right)\left[f\left(\frac{2}{\left(q_u + q_v\right)}\right)\right] \\ &= p_u f\left(\frac{1}{\left(q_u + q_v\right)/2}\right) + p_v f\left(\frac{1}{\left(q_u + q_v\right)/2}\right). \end{aligned}$$

Note that we have the inequality (2) due to

$$\frac{p_u}{q_u}q_v + \frac{p_v}{q_v}q_u > \frac{p_u}{q_u}q_u + \frac{p_v}{q_v}q_v = p_u + p_v.$$

$$\left(\frac{p_u}{q_u}+\frac{p_v}{q_v}\right)\left(q_u+q_v\right)>2\left(p_u+p_v\right).$$

Therefore, by choosing $\boldsymbol{q}'$ such that $q'_1=q_1,...,q'_{u-1}=q_{u-1},q'_u=\frac{q_u+q_v}{2},q'_{u+1}=q_{u+1},...,q'_{v-1}=q_{v-1},q'_v=\frac{q_u+q_v}{2},...,q'_M=q_M$, we then have

$$\ell\left(\boldsymbol{q},\boldsymbol{p}\right)>\ell\left(\boldsymbol{q}',\boldsymbol{p}\right),$$

which is contradict to (13). Therefore, $\frac{p_u}{q_u}\leq\frac{p_v}{q_v}$. This follows that

$$\frac{p_1}{q_1}\leq\frac{p_2}{q_2}\leq....\leq\frac{p_M}{q_M}.$$

Recall that i is the largest index such that $\boldsymbol{p}\notin\Delta_i\left(\boldsymbol{c}\right)$ and $\boldsymbol{q}\in\Delta_i\left(\boldsymbol{c}\right)$. Therefore, $\boldsymbol{p}\in\Delta_k\left(\boldsymbol{c}\right),\forall k=1,...,i-1$. Note that $\boldsymbol{p}\in\Delta_k\left(\boldsymbol{c}\right),\forall k=1,...,i-1$ leads to $\frac{p_k}{c_k}=\max_{1\leq j\leq M}\frac{p_j}{c_j}$ and $\boldsymbol{q}\in\Delta_i\left(\boldsymbol{c}\right)$ leads to $\frac{q_i}{c_i}=\max_{1\leq j\leq M}\frac{q_j}{c_j}$. We further derive as

$$\frac{p_1}{c_1}=\frac{p_1}{q_1}\times\frac{q_1}{c_1}\leq\frac{p_i}{q_i}\times\frac{q_i}{c_i}=\frac{p_i}{c_i}.$$

Note that we have used $\frac{p_1}{q_1}\leq\frac{p_i}{q_i}$ and $\frac{q_i}{c_i}=\max_{1\leq j\leq M}\frac{q_j}{c_j}$. By referring to the fact

$$\frac{p_k}{c_k}=\max_{1\leq j\leq M}\frac{p_j}{c_j},\forall k=1,...,i-1$$

, we achieve

$$\frac{p_i}{c_i}=\max_{1\leq j\leq M}\frac{p_j}{c_j}$$

, which further implies that

$$p_ic_j\geq p_jc_i,\forall 1\leq j\leq M$$

, hence we reach $\boldsymbol{p}\in\Delta_i\left(\boldsymbol{c}\right)$ which is a contradiction. Therefore, we achieve

$$\underline{\ell}\left(\boldsymbol{p}\right)=\inf_{\boldsymbol{r}\in\Delta_n}\ell\left(\boldsymbol{r},\boldsymbol{p}\right)<\ell\left(\boldsymbol{q},\boldsymbol{p}\right).$$

In addition, the function $f\left(t\right)=t\log\left(t\right),t>0$ is strictly convex and increasing. Therefore, the cross-entropy loss is a proper loss. In the following lemma, we independently prove that the cross-entropy loss is a proper loss.

Proof of Lemma 5

For the cross-entropy loss, we have $f(t)=t\log t$ and obtain

$$\sum_{y=1}^{M}\ell\left(y,\boldsymbol{\alpha}\right)\boldsymbol{\beta}_y=-\sum_{i=1}^{M}\beta_i\log\alpha_i=\sum_{i=1}^{M}\beta_i\log\frac{\beta_i}{\alpha_i}+\mathbb{H}\left(\boldsymbol{\beta}\right)$$

$$=D_{KL}\left(\boldsymbol{\beta},\boldsymbol{\alpha}\right)+\mathbb{H}\left(\boldsymbol{\beta}\right)\geq\mathbb{H}\left(\boldsymbol{\beta}\right)=\sum_{y=1}^{M}\ell\left(y,\boldsymbol{\beta}\right)\boldsymbol{\beta}_y,$$

for which the equality happens when $\boldsymbol{\beta}$=$\boldsymbol{\alpha}$. Note that $\mathbb{H}\left(\boldsymbol{\beta}\right)=-\sum_{i=1}^{M}\beta_i\log\beta_i$ denotes the Shannon entropy and $D_{KL}\left(\boldsymbol{\beta},\boldsymbol{\alpha}\right)$ represents the Kullback-Leibler (KL) divergence.

A.2 Theoretical results on guaranteed bounds

We first establish key results on the loss function and divergence followed by our guaranteed bounds on attack/defense. We then provide a more general form of the bounds on an intermediate space in which we assume that the hypotheses h decompose into the composition of two functions (i.e., a feature extractor and a classifier). Finally, detailed implications of our theory will be developed and presented together with the experiments in the experimental section.

A.2.1 Results for the loss function and divergence

Proof of Lemma 2

We first observe that

$$g^{\ell,\boldsymbol{\pi}}\left(\frac{p_1(\mathbf{x})}{p(\mathbf{x})},\ldots,\frac{p_M(\mathbf{x})}{p(\mathbf{x})}\right)=\min_{\boldsymbol{\alpha}}\sum_{k=1}^{M}\pi_k\ell\left(y=k,\boldsymbol{\alpha}\right)\frac{p_k(\mathbf{x})}{p(\mathbf{x})}=\min_{\boldsymbol{\alpha}}\sum_{k=1}^{M}\ell\left(y=k,\boldsymbol{\alpha}\right)\frac{p\left(\mathbf{x},y=k\right)}{p(\mathbf{x})}$$
$$=\min_{\boldsymbol{\alpha}}\sum_{k=1}^{M}\ell\left(y=k,\boldsymbol{\alpha}\right)p\left(y=k\mid\mathbf{x}\right)=\sum_{k=1}^{M}\ell\left(y=k,p\left(\cdot\mid\mathbf{x}\right)\right)p\left(y=k\mid\mathbf{x}\right),$$

where we denote $p\left(\cdot\mid\mathbf{x}\right)=\left[p\left(y=k\mid\mathbf{x}\right)\right]_{k=1}^{M}$. Note that we use the property ℓ is a proper loss for deriving the last step.

$$g^{\ell,\boldsymbol{\pi}}\left(1,...,1\right)=\min_{\boldsymbol{\alpha}}\sum_{k=1}^{M}\ell\left(y=k,\boldsymbol{\alpha}\right)\pi_k=\sum_{k=1}^{M}\ell\left(y=k,\boldsymbol{\pi}\right)\pi_k.$$

We now consider

$$\ell\left(y=k,\boldsymbol{\pi}\right)-\ell\left(y=k,p\left(\cdot\mid\mathbf{x}\right)\right)=f\left(\pi_k^{-1}\right)\pi_k-f\left(p\left(y=k\mid\mathbf{x}\right)^{-1}\right)p\left(y=k\mid\mathbf{x}\right)$$
$$=-f\left(\frac{p(\mathbf{x})}{p_k(\mathbf{x})\pi_k}\right)\frac{p_k(\mathbf{x})\pi_k}{p(\mathbf{x})}+f\left(\pi_k^{-1}\right)\pi_k=u_k\left(\frac{p(\mathbf{x})}{p_k(\mathbf{x})}\right),$$

where we define

$$u_k(t)=-\pi_k t^{-1}f\left(t\pi_k^{-1}\right)+f\left(\pi_k^{-1}\right)\pi_k=-\left(t\pi_k^{-1}\right)^{-1}f\left(t\pi_k^{-1}\right)+f\left(\pi_k^{-1}\right)\pi_k,\tag{14}$$

which is a convex function since $z(t)=-t^{-1}f(t)$ is convex, $v_k(t)=t\pi_k^{-1}$ is linear, and $u_k=z\circ v_k+\text{const}$ ($\circ$ specifies the function composition).

It appears that

$$D_\ell^{\boldsymbol{\pi}}\left(p_1,...,p_M\right)=\sum_{k=1}^{M}\int\ell\left(y=k,\boldsymbol{\pi}\right)\pi_k p_k(\mathbf{x})\,d\mathbf{x}$$
$$-\int\sum_{k=1}^{M}\ell\left(y=k,p\left(\cdot\mid\mathbf{x}\right)\right)p\left(y=k\mid\mathbf{x}\right)p(\mathbf{x})\,d\mathbf{x}$$
$$=\sum_{k=1}^{M}\int\left[\ell\left(y=k,\boldsymbol{\pi}\right)-\ell\left(y=k,p\left(\cdot\mid\mathbf{x}\right)\right)\right]p\left(\mathbf{x},y=k\right)d\mathbf{x}$$
$$=\sum_{k=1}^{M}\pi_k\int\left[\ell\left(y=k,\boldsymbol{\pi}\right)-\ell\left(y=k,p\left(\cdot\mid\mathbf{x}\right)\right)\right]p_k(\mathbf{x})\,d\mathbf{x}$$
$$=\sum_{k=1}^{M}\pi_k\int u_k\left(\frac{p(\mathbf{x})}{p_k(\mathbf{x})}\right)p_k(\mathbf{x})\,d\mathbf{x}=\sum_{k=1}^{M}\pi_k D_{u_k}\left(p,p_k\right)\geq 0,$$

since $u_k(1)=0,\forall k$.

The equality occurs if only if $p_k=p$, $\forall k$.

Proof of Lemma 5

This result is trivial from Lemma 2 with noting that

$$u_k(t)=\log\left(\pi_k t^{-1}\right)-\log\pi_k^{-1}=-\log t,$$

here we refer to Eq. (5) for evaluating u_k.

A.2.2 Attack/defense guaranteed bounds w.r.t. a data space

Proof of Theorem 6

We first turn the min-max problem to the max-min one as

$$\inf_{h\in\mathcal{H}}\sup_{a\in\mathcal{A}}\mathcal{J}\left(a,h\right)\geq\sup_{a\in\mathcal{A}}\inf_{h\in\mathcal{H}}\mathcal{J}\left(a,h\right).$$

Let $\mathcal{H}'$ be the family of all measurable functions, we then inspect

$$\begin{aligned}\mathcal{I}\left(a,h\right)&:=\inf_{h'\in\mathcal{H}'}\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell\left(y,h'\left(a_h\left(\mathbf{x}\right)\right)\right)\right]\\&=\inf_{h'\in\mathcal{H}'}\sum_{k=1}^{M}\pi_k\mathbb{E}_{p_k(\mathbf{x})}\left[\ell\left(y=k,h'\left(a_h\left(\mathbf{x}\right)\right)\right)\right]\\&=\inf_{h'\in\mathcal{H}'}\sum_{k=1}^{M}\pi_k\mathbb{E}_{p_k^{a,h}(\mathbf{x})}\left[\ell\left(y=k,h'\left(\mathbf{x}\right)\right)\right],\end{aligned}$$

here we note that we use the push-forward measure property of the expectation in the last derivation.

To continue, we further derive

$$\begin{aligned}\mathcal{I}\left(a,h\right)&=\inf_{h'\in\mathcal{H}'}\sum_{k=1}^{M}\pi_k\mathbb{E}_{p_k^{a,h}(\mathbf{x})}\left[\ell\left(y=k,h'\left(\mathbf{x}\right)\right)\right]\\&=\inf_{h'\in\mathcal{H}'}\int\sum_{k=1}^{M}\pi_k\ell\left(y=k,h'\left(\mathbf{x}\right)\right)p_k^{a,h}\left(\mathbf{x}\right)d\mathbf{x}\\&=\inf_{h'\in\mathcal{H}'}\int\left(\sum_{k=1}^{M}\pi_k\ell\left(y=k,h'\left(\mathbf{x}\right)\right)\frac{p_k^{a,h}\left(\mathbf{x}\right)}{p^{a,h}\left(\mathbf{x}\right)}\right)p^{a,h}\left(\mathbf{x}\right)d\mathbf{x}\\&=\int\left(\min_{\boldsymbol{\alpha}\in\Delta_M}\sum_{k=1}^{M}\pi_k\ell\left(y=k,\boldsymbol{\alpha}\right)\frac{p_k^{a,h}\left(\mathbf{x}\right)}{p^{a,h}\left(\mathbf{x}\right)}\right)p^{a,h}\left(\mathbf{x}\right)d\mathbf{x},\end{aligned}$$

here we note that the last derivation is due to the infinite capacity of $\mathcal{H}'$ and $p^{a,h}\left(\mathbf{x}\right)=\sum_{k=1}^{M}\pi_k p_k^{a,h}\left(\mathbf{x}\right)$, which is the data distribution induced from the adversary transformation $a_h\left(\mathbf{x}\right)$.

Using the results in Eqs. (4,3), we have

$$\begin{aligned}\mathcal{I}\left(a,h\right)=&\int g^{\ell,\boldsymbol{\pi}}\left(\frac{p_1^{a,h}\left(\mathbf{x}\right)}{p^{a,h}\left(\mathbf{x}\right)},...,\frac{p_M^{a,h}\left(\mathbf{x}\right)}{p^{a,h}\left(\mathbf{x}\right)}\right)p^{a,h}\left(\mathbf{x}\right)d\mathbf{x}\\=&g^{\ell,\boldsymbol{\pi}}\left(1,...,1\right)-D_\ell^{\boldsymbol{\pi}}\left(p_1^{a,h},...,p_M^{a,h}\right)\\&=\sum_{k=1}^{M}\ell\left(y=k,\boldsymbol{\pi}\right)\pi_k-D_\ell^{\boldsymbol{\pi}}\left(p_1^{a,h},...,p_M^{a,h}\right).\end{aligned}$$

We finally reach the conclusion by

$$\begin{aligned}&\mathcal{J}\left(a,h\right)\geq\mathcal{I}\left(a,h\right),\\&\inf_{h\in\mathcal{H}}\mathcal{J}\left(a,h\right)\geq\inf_{h\in\mathcal{H}}\mathcal{I}\left(a,h\right)=\sum_{k=1}^{M}\ell\left(y=k,\boldsymbol{\pi}\right)\pi_k-\sup_{h\in\mathcal{H}}D_\ell^{\boldsymbol{\pi}}\left(p_1^{a,h},...,p_M^{a,h}\right),\\&\sup_{a\in\mathcal{A}}\inf_{h\in\mathcal{H}}\mathcal{J}\left(a,h\right)\geq\sum_{k=1}^{M}\ell\left(y=k,\boldsymbol{\pi}\right)\pi_k-\inf_{a\in\mathcal{A}}\sup_{h\in\mathcal{H}}D_\ell^{\boldsymbol{\pi}}\left(p_1^{a,h},...,p_M^{a,h}\right).\end{aligned}$$

That concludes our proof.

Proof of Theorem 9

For any $a \in \mathcal{A}$, we have

$$\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell\left(y,h\left(a_h\left(\mathbf{x}\right)\right)\right)\right]=\sum_{k=1}^{M}\pi_k\mathbb{E}_{p_k(\mathbf{x})}\left[\ell\left(y=k,h\left(a_h\left(\mathbf{x}\right)\right)\right)\right]$$
$$=\sum_{k=1}^{M}\pi_k\mathbb{E}_{p_k^{a,h}(\mathbf{x})}\left[\ell\left(y=k,h\left(\mathbf{x}\right)\right)\right]=\int\sum_{k=1}^{M}\pi_k\ell\left(y=k,h\left(\mathbf{x}\right)\right)p_k^{a,h}\left(\mathbf{x}\right)d\mathbf{x}$$
$$=\int\sum_{k=1}^{M}\pi_k\ell\left(y=k,h\left(\mathbf{x}\right)\right)p_k\left(\mathbf{x}\right)d\mathbf{x}+\int\sum_{k=1}^{M}\pi_k\ell\left(y=k,h\left(\mathbf{x}\right)\right)\left[p_k^{a,h}\left(\mathbf{x}\right)-p_k\left(\mathbf{x}\right)\right]d\mathbf{x}.$$

Using the Cauchy–Schwarz inequality, we obtain

$$\int\sum_{k=1}^{M}\pi_k\ell\left(y=k,h\left(\mathbf{x}\right)\right)\left[p_k^{a,h}\left(\mathbf{x}\right)-p_k\left(\mathbf{x}\right)\right]d\mathbf{x}=\sum_{k=1}^{M}\pi_k\int\ell\left(y=k,h\left(\mathbf{x}\right)\right)\left[p_k^{a,h}\left(\mathbf{x}\right)-p_k\left(\mathbf{x}\right)\right]d\mathbf{x}$$
$$=\sum_{k=1}^{M}\pi_k\int\ell\left(y=k,h\left(\mathbf{x}\right)\right)p_k\left(\mathbf{x}\right)^{1/2}\frac{p_k^{a,h}\left(\mathbf{x}\right)-p_k\left(\mathbf{x}\right)}{p_k\left(\mathbf{x}\right)}p_k\left(\mathbf{x}\right)^{1/2}d\mathbf{x}$$
$$\overset{(1)}{\leq}\sum_{k=1}^{M}\pi_k\left[\int\ell\left(y=k,h\left(\mathbf{x}\right)\right)^2p_k\left(\mathbf{x}\right)d\mathbf{x}\right]^{1/2}\left[\int\left(\frac{p_k^{a,h}\left(\mathbf{x}\right)-p_k\left(\mathbf{x}\right)}{p_k\left(\mathbf{x}\right)}\right)^2p_k\left(\mathbf{x}\right)d\mathbf{x}\right]^{1/2}$$
$$=\sum_{k=1}^{M}\left[\pi_k\int\ell\left(y=k,h\left(\mathbf{x}\right)\right)^2p_k\left(\mathbf{x}\right)d\mathbf{x}\right]^{1/2}\left[\pi_k\int\left(\frac{p_k^{a,h}\left(\mathbf{x}\right)-p_k\left(\mathbf{x}\right)}{p_k\left(\mathbf{x}\right)}\right)^2p_k\left(\mathbf{x}\right)d\mathbf{x}\right]^{1/2}$$
$$\overset{(2)}{\leq}\left[\sum_{k=1}^{M}\pi_k\int\ell\left(y=k,h\left(\mathbf{x}\right)\right)^2p_k\left(\mathbf{x}\right)d\mathbf{x}\right]^{1/2}\left[\sum_{k=1}^{M}\pi_k\int\left(\frac{p_k^{a,h}\left(\mathbf{x}\right)-p_k\left(\mathbf{x}\right)}{p_k\left(\mathbf{x}\right)}\right)^2p_k\left(\mathbf{x}\right)d\mathbf{x}\right]^{1/2}$$
$$=\left[\sum_{k=1}^{M}\pi_k\int\ell\left(y=k,h\left(\mathbf{x}\right)\right)^2p_k\left(\mathbf{x}\right)d\mathbf{x}\right]^{1/2}\left[\sum_{k=1}^{M}\pi_k\int\left(\frac{p_k^{a,h}\left(\mathbf{x}\right)-p_k\left(\mathbf{x}\right)}{p_k\left(\mathbf{x}\right)}\right)^2p_k\left(\mathbf{x}\right)d\mathbf{x}\right]^{1/2}$$
$$=\left[\sum_{k=1}^{M}\int\ell\left(y=k,h\left(\mathbf{x}\right)\right)^2p\left(\mathbf{x},y=k\right)d\mathbf{x}\right]^{1/2}\left[\sum_{k=1}^{M}\pi_k\int\left(\frac{p_k^{a,h}\left(\mathbf{x}\right)}{p_k\left(\mathbf{x}\right)}-1\right)^2p_k\left(\mathbf{x}\right)d\mathbf{x}\right]^{1/2}$$
$$=\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell\left(y,h\left(\mathbf{x}\right)\right)^2\right]^{1/2}\left[\sum_{k=1}^{M}\pi_kD_v\left(p_k^{a,h},p_k\right)\right]^{1/2}.$$

note that we use the inequality $\int u\left(\mathbf{x}\right)v\left(\mathbf{x}\right)d\mathbf{x}\leq\left[\int u\left(\mathbf{x}\right)^2d\mathbf{x}\right]^{1/2}\left[\int v\left(\mathbf{x}\right)d\mathbf{x}\right]^{1/2}$ in (1) and the Cauchy–Schwarz inequality in (2).

Using the Cauchy–Schwarz inequality again, we obtain

$$\int\sum_{k=1}^{M}\pi_k\ell\left(y=k,h\left(\mathbf{x}\right)\right)p_k\left(\mathbf{x}\right)d\mathbf{x}=\sum_{k=1}^{M}\pi_k\int\ell\left(y=k,h\left(\mathbf{x}\right)\right)p_k\left(\mathbf{x}\right)^{1/2}p_k\left(\mathbf{x}\right)^{1/2}d\mathbf{x}$$
$$\overset{(1)}{\leq}\sum_{k=1}^{M}\pi_k\left[\int\ell\left(y=k,h\left(\mathbf{x}\right)\right)^2p_k\left(\mathbf{x}\right)d\mathbf{x}\right]^{1/2}\left[\int p_k\left(\mathbf{x}\right)d\mathbf{x}\right]^{1/2}$$
$$=\sum_{k=1}^{M}\left[\pi_k\int\ell\left(y=k,h\left(\mathbf{x}\right)\right)^2p_k\left(\mathbf{x}\right)d\mathbf{x}\right]^{1/2}\left[\pi_k\int p_k\left(\mathbf{x}\right)d\mathbf{x}\right]^{1/2}$$

$$\begin{aligned}
&\int\sum_{k=1}^{M}\pi_k\ell\left(y=k,h\left(\mathbf{x}\right)\right)p_k\left(\mathbf{x}\right)d\mathbf{x}=\\
&\overset{(2)}{\leq}\left[\sum_{k=1}^{M}\pi_k\int\ell\left(y=k,h\left(\mathbf{x}\right)\right)^2p_k\left(\mathbf{x}\right)d\mathbf{x}\right]^{1/2}\left[\sum_{k=1}^{M}\pi_k\int p_k\left(\mathbf{x}\right)d\mathbf{x}\right]\\
&=\left[\sum_{k=1}^{M}\pi_k\int\ell\left(y=k,h\left(\mathbf{x}\right)\right)^2p_k\left(\mathbf{x}\right)d\mathbf{x}\right]^{1/2}\left[\int p\left(\mathbf{x}\right)d\mathbf{x}\right]\\
&=\left[\sum_{k=1}^{M}\pi_k\int\ell\left(y=k,h\left(\mathbf{x}\right)\right)^2p_k\left(\mathbf{x}\right)d\mathbf{x}\right]^{1/2}=\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell\left(y,h\left(\mathbf{x}\right)\right)^2\right]^{1/2}.
\end{aligned}$$

Finally, we arrive at

$$\begin{aligned}
\mathcal{L}\left(\mathcal{A},\mathcal{H},\mathcal{D}\right)&=\inf_{h\in\mathcal{H}}\sup_{a\in\mathcal{A}}\mathcal{J}\left(a,h\right)=\inf_{h\in\mathcal{H}}\sup_{a\in\mathcal{A}}\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell\left(y,h\left(a_h\left(\mathbf{x}\right)\right)\right)\right]\\
&\leq\inf_{h\in\mathcal{H}}\left(\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell\left(y,h\left(\mathbf{x}\right)\right)^2\right]^{1/2}+\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell\left(y,h\left(\mathbf{x}\right)\right)^2\right]^{1/2}\sup_{a\in\mathcal{A}}\left[\sum_{k=1}^{M}\pi_kD_v\left(p_k^{a,h},p_k\right)\right]^{1/2}\right).
\end{aligned}$$

That concludes our proof.

A.2.3 Attack/defense guaranteed bounds w.r.t. a latent space

Proof of Theorem 11

We first turn the min-max problem to the max-min one as

$$\inf_{h\in\mathcal{H}}\sup_{a\in\mathcal{A}}\mathcal{J}\left(a,h\right)\geq\sup_{a\in\mathcal{A}}\inf_{h\in\mathcal{H}}\mathcal{J}\left(a,h\right).$$

We consider the classifier family $\mathcal{H}$ as

$$\mathcal{H}:=\left\{h=h_1\circ h_2:h_1\in\mathcal{H}_1\,\text{and}\,h_2\in\mathcal{H}_2\right\}.$$

Let $\mathcal{H}_2^{'}$ be the family all measurable functions on the latent space. Given $h=h_1\circ h_2\in\mathcal{H}$, we denote

$$\mathcal{H}_h^{'}:=\left\{h'=h_1\circ h_2^{'}:h_2^{'}\in\mathcal{H}_2^{'}\right\}.$$

We then inspect

$$\begin{aligned}
\mathcal{I}\left(a,h\right)&:=\inf_{h'\in\mathcal{H}_h^{'}}\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell\left(y,h'\left(a_h\left(\mathbf{x}\right)\right)\right)\right]\\
&=\inf_{h'\in\mathcal{H}_h^{'}}\sum_{k=1}^{M}\pi_k\mathbb{E}_{p_k(\mathbf{x})}\left[\ell\left(y=k,h_2^{'}\left(h_1\left(a_h\left(\mathbf{x}\right)\right)\right)\right)\right]\\
&=\inf_{h'\in\mathcal{H}_h^{'}}\sum_{k=1}^{M}\pi_k\mathbb{E}_{q_k^{a,h}(\mathbf{z})}\left[\ell\left(y=k,h_2'\left(\mathbf{z}\right)\right)\right],
\end{aligned}$$

here we note that we use the push-forward measure property of the expectation in the last derivation.

To continue, we further derive

$$\begin{aligned}\mathcal{I}\left(a,h\right) &= \inf_{h'\in\mathcal{H}'}\sum_{k=1}^{M}\pi_{k}\mathbb{E}_{q_{k}^{a,h}\left(\mathbf{z}\right)}\left[\ell\left(y=k,h_{2}'\left(\mathbf{z}\right)\right)\right]\\ &= \inf_{h'\in\mathcal{H}'}\int\sum_{k=1}^{M}\pi_{k}\ell\left(y=k,h_{2}'\left(\mathbf{z}\right)\right)q_{k}^{a,h}\left(\mathbf{z}\right)d\mathbf{z}\\ &= \inf_{h'\in\mathcal{H}'}\int\left(\sum_{k=1}^{M}\pi_{k}\ell\left(y=k,h_{2}'\left(\mathbf{z}\right)\right)\frac{q_{k}^{a,h}\left(\mathbf{z}\right)}{q^{a,h}\left(\mathbf{z}\right)}\right)q^{a,h}\left(\mathbf{z}\right)d\mathbf{z}\\ &= \int\left(\min_{\boldsymbol{\alpha}\in\Delta_{M}}\sum_{k=1}^{M}\pi_{k}\ell\left(y=k,\boldsymbol{\alpha}\right)\frac{q_{k}^{a,h}\left(\mathbf{z}\right)}{q^{a,h}\left(\mathbf{z}\right)}\right)q^{a,h}\left(\mathbf{z}\right)d\mathbf{z},\end{aligned}$$

here we note that the last derivation is due to the infinite capacity of $\mathcal{H}_{2}^{'}$.

Using the results in Eqs. (4,3), we have

$$\begin{aligned}\mathcal{I}\left(a,h\right) &= \int g^{\ell,\boldsymbol{\pi}}\left(\frac{q_{1}^{a,h}\left(\mathbf{z}\right)}{q^{a,h}\left(\mathbf{z}\right)},...,\frac{q_{M}^{a,h}\left(\mathbf{z}\right)}{q^{a,h}\left(\mathbf{z}\right)}\right)q^{a,h}\left(\mathbf{z}\right)d\mathbf{z}\\ &= g^{\ell,\boldsymbol{\pi}}\left(1,...,1\right)-D_{\ell}^{\boldsymbol{\pi}}\left(q_{1}^{a,h},...,q_{M}^{a,h}\right)\\ &= \sum_{k=1}^{M}\ell\left(y=k,\boldsymbol{\pi}\right)\pi_{k}-D_{\ell}^{\boldsymbol{\pi}}\left(q_{1}^{a,h},...,q_{M}^{a,h}\right).\end{aligned}$$

We finally reach the conclusion by

$$\begin{aligned}&\mathcal{J}\left(a,h\right)\geq\mathcal{I}\left(a,h\right),\\ &\inf_{h\in\mathcal{H}}\mathcal{J}\left(a,h\right)\geq\inf_{h\in\mathcal{H}}\mathcal{I}\left(a,h\right)=\sum_{k=1}^{M}\ell\left(y=k,\boldsymbol{\pi}\right)\pi_{k}-\sup_{h\in\mathcal{H}}D_{\ell}^{\boldsymbol{\pi}}\left(q_{1}^{a,h},...,q_{M}^{a,h}\right),\\ &\sup_{a\in\mathcal{A}}\inf_{h\in\mathcal{H}}\mathcal{J}\left(a,h\right)\geq\sum_{k=1}^{M}\ell\left(y=k,\boldsymbol{\pi}\right)\pi_{k}-\inf_{a\in\mathcal{A}}\sup_{h\in\mathcal{H}}D_{\ell}^{\boldsymbol{\pi}}\left(q_{1}^{a,h},...,q_{M}^{a,h}\right).\end{aligned}$$

That concludes our proof.

Proof of Theorem 13 .

For any $a\in\mathcal{A}$, we have

$$\begin{aligned}\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell\left(y,h\left(a_{h}\left(\mathbf{x}\right)\right)\right)\right] &= \sum_{k=1}^{M}\pi_{k}\mathbb{E}_{p_{k}\left(\mathbf{x}\right)}\left[\ell\left(y=k,h\left(a_{h}\left(\mathbf{x}\right)\right)\right)\right]\\ &= \sum_{k=1}^{M}\pi_{k}\mathbb{E}_{p_{k}^{a,h}\left(\mathbf{x}\right)}\left[\ell\left(y=k,h\left(\mathbf{x}\right)\right)\right]=\int\sum_{k=1}^{M}\pi_{k}\ell\left(y=k,h\left(\mathbf{x}\right)\right)p_{k}^{a,h}\left(\mathbf{x}\right)d\mathbf{x}\\ &= \int\sum_{k=1}^{M}\pi_{k}\ell\left(y=k,h\left(\mathbf{x}\right)\right)p_{k}\left(\mathbf{x}\right)d\mathbf{x}+\int\sum_{k=1}^{M}\pi_{k}\ell\left(y=k,h\left(\mathbf{x}\right)\right)\left[p_{k}^{a,h}\left(\mathbf{x}\right)-p_{k}\left(\mathbf{x}\right)\right]d\mathbf{x}\\ &= \int\sum_{k=1}^{M}\pi_{k}\ell\left(y=k,h\left(\mathbf{x}\right)\right)p_{k}\left(\mathbf{x}\right)d\mathbf{x}+\int\sum_{k=1}^{M}\pi_{k}\ell\left(y=k,h_{2}\left(h_{1}\left(\mathbf{x}\right)\right)\right)\left[p_{k}^{a,h}\left(\mathbf{x}\right)-p_{k}\left(\mathbf{x}\right)\right]d\mathbf{x}\\ &= \int\sum_{k=1}^{M}\pi_{k}\ell\left(y=k,h\left(\mathbf{x}\right)\right)p_{k}\left(\mathbf{x}\right)d\mathbf{x}+\int\sum_{k=1}^{M}\pi_{k}\ell\left(y=k,h_{2}\left(\mathbf{z}\right)\right)\left[q_{k}^{a,h}\left(\mathbf{z}\right)-q_{k}\left(\mathbf{z}\right)\right]d\mathbf{z}. \quad (15)\end{aligned}$$

With respect to the latent space via a feature extractor h_1, we define $\mathbb{P}_z := h_1 \# \mathbb{P}$ as the push-forward distribution via h_1, where $\mathbb{P}$ is the data distribution with the density $p\left(\mathbf{x}\right) = \sum_{k=1}^{M} \pi_k p_k(\mathbf{x})$. It appears that $\mathbb{P}_z$ is the data distribution on the latent space. We now equip the latent space a labeling mechanism $p_z\left(y \mid \mathbf{z}\right) = \frac{\int_{h_1^{-1}(\mathbf{z})} p(y|\mathbf{x})p(\mathbf{x})d\mathbf{x}}{\int_{h_1^{-1}(\mathbf{z})} p(\mathbf{x})d\mathbf{x}}$ induced by $p\left(y \mid \mathbf{x}\right)$ via h_1. Denote $\mathcal{D}_z$ as the joint distribution of data-label $(\mathbf{z}, y)$ on the latent space, where $\mathbf{z} \sim \mathbb{P}_z$ and $y \sim p_z\left(y \mid \mathbf{z}\right)$. Using the same derivation as in the proof of Theorem 9, we obtain:

$$\int \sum_{k=1}^{M} \pi_k \ell\left(y = k, h_2\left(\mathbf{z}\right)\right) \left[q_k^{a,h}\left(\mathbf{z}\right) - q_k\left(\mathbf{z}\right)\right] d\mathbf{z} \leq \mathbb{E}_{(\mathbf{z},y)\sim\mathcal{D}_z}\left[\ell\left(y, h_2\left(\mathbf{z}\right)\right)^2\right]^{1/2} \left[\sum_{k=1}^{M} \pi_k D_v\left(q_k^{a,h}, q_k\right)\right]^{1/2}. \tag{16}$$

Next, to move from the latent space to the input space, we prove that

$$\mathbb{E}_{(\mathbf{z},y)\sim\mathcal{D}_z}\left[\ell\left(y, h_2\left(\mathbf{z}\right)\right)^2\right] = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell\left(y, h\left(\mathbf{x}\right)\right)^2\right]. \tag{17}$$

First, let denote $p_z\left(\mathbf{z}\right)$ as the density of $\mathbb{P}_z$, then we have

$$p_z\left(\mathbf{z}\right) = \int_{h_1^{-1}(\mathbf{z})} p\left(\mathbf{x}\right) d\mathbf{x}.$$

Therefore, denote $\mathcal{X}$ as the data space and $\mathcal{Z}$ as the latent space, it follows that

$$\begin{aligned}
\mathbb{E}_{(\mathbf{z},y)\sim\mathcal{D}_z}\left[\ell\left(y, h_2\left(\mathbf{z}\right)\right)^2\right] &= \sum_{k=1}^{M} \int_{\mathcal{Z}} \ell\left(y = k, h_2\left(\mathbf{z}\right)\right)^2 p_z\left(y = k \mid \mathbf{z}\right) p_z\left(\mathbf{z}\right) d\mathbf{z} \\
&= \sum_{k=1}^{M} \int_{\mathcal{Z}} \ell\left(y = k, h_2\left(\mathbf{z}\right)\right)^2 \frac{\int_{h_1^{-1}(\mathbf{z})} p\left(y \mid \mathbf{x}\right) p\left(\mathbf{x}\right) d\mathbf{x}}{\int_{h_1^{-1}(\mathbf{z})} p\left(\mathbf{x}\right) d\mathbf{x}} \int_{h_1^{-1}(\mathbf{z})} p\left(\mathbf{x}\right) d\mathbf{x} d\mathbf{z} \\
&= \sum_{k=1}^{M} \int_{\mathcal{Z}} \ell\left(y = k, h_2\left(\mathbf{z}\right)\right)^2 \int_{h_1^{-1}(\mathbf{z})} p\left(y \mid \mathbf{x}\right) p\left(\mathbf{x}\right) d\mathbf{x} d\mathbf{z} \\
&= \sum_{k=1}^{M} \int_{\mathcal{Z}} \int_{h_1^{-1}(\mathbf{z})} \ell\left(y = k, h_2\left(\mathbf{z}\right)\right)^2 p\left(y \mid \mathbf{x}\right) p\left(\mathbf{x}\right) d\mathbf{x} d\mathbf{z} \\
&= \sum_{k=1}^{M} \int_{\mathcal{Z}} \int_{\mathcal{X}} \mathbb{I}_{\mathbf{x} \in h_1^{-1}(\mathbf{z})} \ell\left(y = k, h_2\left(\mathbf{z}\right)\right)^2 p\left(y \mid \mathbf{x}\right) p\left(\mathbf{x}\right) d\mathbf{x} d\mathbf{z}
\end{aligned}$$

$$\begin{aligned}
\mathbb{E}_{(\mathbf{z},y)\sim\mathcal{D}_z}\left[\ell\left(y, h_2\left(\mathbf{z}\right)\right)^2\right] &= \sum_{k=1}^{M} \int_{\mathcal{Z}} \int_{\mathcal{X}} \mathbb{I}_{\mathbf{x} \in h_1^{-1}(\mathbf{z})} \ell\left(y = k, h_2\left(\mathbf{z}\right)\right)^2 p\left(y \mid \mathbf{x}\right) p\left(\mathbf{x}\right) d\mathbf{x} d\mathbf{z} \\
&\overset{(1)}{=} \sum_{k=1}^{M} \int_{\mathcal{X}} \int_{\mathcal{Z}} \mathbb{I}_{\mathbf{x} \in h_1^{-1}(\mathbf{z})} \ell\left(y = k, h_2\left(\mathbf{z}\right)\right)^2 p\left(y \mid \mathbf{x}\right) p\left(\mathbf{x}\right) d\mathbf{z} d\mathbf{x} \\
&= \sum_{k=1}^{M} \int_{\mathcal{X}} \int_{\mathcal{Z}} \mathbb{I}_{\mathbf{z} = h_1(\mathbf{x})} \ell\left(y = k, h_2\left(\mathbf{z}\right)\right)^2 p\left(y \mid \mathbf{x}\right) p\left(\mathbf{x}\right) d\mathbf{z} d\mathbf{x} \\
&= \sum_{k=1}^{M} \int_{\mathcal{X}} \ell\left(y = k, h_2\left(h_1\left(\mathbf{x}\right)\right)\right)^2 p\left(y \mid \mathbf{x}\right) p\left(\mathbf{x}\right) d\mathbf{x} \\
&= \sum_{k=1}^{M} \int_{\mathcal{X}} \ell\left(y = k, h\left(\mathbf{x}\right)\right)^2 p\left(y \mid \mathbf{x}\right) p\left(\mathbf{x}\right) d\mathbf{x} = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\ell\left(y, h\left(\mathbf{x}\right)\right)^2\right].
\end{aligned}$$

Here we note that $\mathbb{I}_A$ is the indicator function which returns 1 if A is true and 0 otherwise and in $\overset{(1)}{=}$, we use Fubini theorem to interchange two integrals.

Finally, combining (15), (16), and (17), we reach the conclusion.

A.3 Proof of Inequalities in Section 4 of the Main Paper

We have missed the factor π_k on the first inequality. Although it does not affect our result, we apologize for this inconvenience.

Proof of the first inequality

$$
\begin{aligned}
JS^{\pi}\left(q_1^{a,h},...,q_M^{a,h}\right) &\geq -\min_{h_d}\left\{\sum_{k=1}^{M}\pi_k E_{\mathbf{z}_a\sim q_k^{a,h}}[\mathrm{CE}(h_d(\mathbf{z}_a),k)]\right\}+\mathbb{H}\left(\boldsymbol{\pi}\right)\\
&=-\min_{h_d}\left\{\sum_{k=1}^{M}\pi_k E_{\mathbf{x}_a\sim p_k^{a,h}}[\mathrm{CE}(h_d(h_1\left(\mathbf{x}_a\right)),k)]\right\}+\mathbb{H}\left(\boldsymbol{\pi}\right). \qquad (18)
\end{aligned}
$$

Proof. We consider the optimization problem

$$
\min_{h_d\in\mathcal{A}}\left\{\sum_{k=1}^{M}\pi_k E_{\mathbf{z}_a\sim q_k^{a,h}}[\mathrm{CE}(h_d(\mathbf{z}_a),k)]\right\}, \qquad (19)
$$

where h_d is searched in the family of all functions $\mathcal{A}$.

We have

$$
\sum_{k=1}^{M}\pi_k E_{\mathbf{z}_a\sim q_k^{a,h}}[\mathrm{CE}(h_d(\mathbf{z}_a),k)]=-\int\sum_{k=1}^{M}\pi_k\log h_d\left(\mathbf{z}_a,k\right)q_k^{a,h}\left(\mathbf{z}_a\right)d\mathbf{z}_a.
$$

For each $\mathbf{z}_a$, we point-wisely solve the following optimization problem

$$
\begin{aligned}
&\max_{h}\sum_{k=1}^{M}\pi_k q_k^{a,h}\left(\mathbf{z}_a\right)\log h_k\\
\text{s.t.}:&\sum_{k=1}^{M}h_k=1,\\
&h_k\geq 0,\forall k=1,...,M.
\end{aligned}
$$

It is obvious that the above optimization problem has the solution

$$
h_k^*=\frac{\pi_k q_k^{a,h}\left(\mathbf{z}_a\right)}{\sum_{j=1}^{M}\pi_j q_j^{a,h}\left(\mathbf{z}_a\right)},\forall k=1,...,M.
$$

Therefore, the optimization problem in Eq. (19) has the solution:

$$
h_d^*\left(\mathbf{z}_a,k\right)=\frac{\pi_k q_k^{a,h}\left(\mathbf{z}_a\right)}{\sum_{j=1}^{M}\pi_j q_j^{a,h}\left(\mathbf{z}_a\right)},\forall k=1,...,M.
$$

Finally, we have

$$\begin{aligned}
\min_{h_d}\left\{\sum_{k=1}^{M}\pi_k E_{\mathbf{z}_a\sim q_k^{a,h}}[\text{CE}(h_d(\mathbf{z}_a),k)]\right\} &\geq \min_{h_d\in\mathcal{A}}\left\{\sum_{k=1}^{M}\pi_k E_{\mathbf{z}_a\sim q_k^{a,h}}[\text{CE}(h_d(\mathbf{z}_a),k)]\right\} \\
&= \sum_{k=1}^{M}\pi_k E_{\mathbf{z}_a\sim q_k^{a,h}}[\text{CE}(h_d^*(\mathbf{z}_a),k)] \\
&= -\sum_{k=1}^{M}\pi_k\int\log h_d^*(\mathbf{z}_a,k)q_k^{a,h}\left(\mathbf{z}_a\right)d\mathbf{z}_a \\
&= -\sum_{k=1}^{M}\pi_k\int\log\frac{\pi_k q_k^{a,h}\left(\mathbf{z}_a\right)}{\sum_{j=1}^{M}\pi_j q_j^{a,h}\left(\mathbf{z}_a\right)}q_k^{a,h}\left(\mathbf{z}_a\right)d\mathbf{z}_a \\
&= -JS^{\pi}\left(q_1^{a,h},...,q_M^{a,h}\right)+\mathbb{H}\left(\boldsymbol{\pi}\right).
\end{aligned}$$

The inequality becomes tight if the family to search for h_d has members to approach h_d^* with $h_d^*\left(\mathbf{z},k\right)=\frac{\pi_k q_k^{a,h}(\mathbf{z})}{\sum_{j=1}^{M}\pi_j q_j^{a,h}(\mathbf{z})},\forall k=1,...,M$ up to any level of precision or contains h_d^*. Note that we use $h_d^*\left(\mathbf{z},k\right)$ to represent the k-th component of $h_d^*\left(\mathbf{z}\right)$. □

Proof of the second inequality

$$\begin{aligned}
\sum_{k=1}^{M}JS^{0.5,0.5}(q_k^{a,h},q_k) &\geq 0.5\sum_{k=1}^{M}\max_{T_k}\left\{\mathbb{E}_{\mathbf{z}\sim q_k}\left[\log T_k\left(\mathbf{z}\right)\right]+\mathbb{E}_{\mathbf{z}_a\sim q_k^{a,h}}\left[\log\left(1-T_k\left(\mathbf{z}_a\right)\right)\right]+2\log 2\right\} \\
&= 0.5\sum_{k=1}^{M}\max_{T_k}\left\{\mathbb{E}_{\mathbf{x}\sim p_k}\left[\log T_k\left(h_1\left(\mathbf{x}\right)\right)\right]+\mathbb{E}_{\mathbf{x}_a\sim p_k^{a,h}}\left[\log\left(1-T_k\left(h_1\left(\mathbf{x}_a\right)\right)\right)\right]+2\log 2\right\},
\end{aligned}$$

Proof. The proof of this inequality depends the result of the previous proof. We have:

$$\begin{aligned}
JS^{0.5,0.5}(q_k^{a,h},q_k) &\geq -\min_{T_k}\left\{-0.5\mathbb{E}_{\mathbf{z}\sim q_k}\left[\log T_k\left(\mathbf{z}\right)\right]-0.5\mathbb{E}_{\mathbf{z}_a\sim q_k^{a,h}}\left[\log\left(1-T_k\left(\mathbf{z}_a\right)\right)\right]\right\}+\log 2 \\
&= 0.5\sum_{k=1}^{M}\max_{T_k}\left\{\mathbb{E}_{\mathbf{z}\sim q_k}\left[\log T_k\left(\mathbf{z}\right)\right]+\mathbb{E}_{\mathbf{z}_a\sim q_k^{a,h}}\left[\log\left(1-T_k\left(\mathbf{z}_a\right)\right)\right]+2\log 2\right\}.
\end{aligned}$$

We now take sum over k to reach the final conclusion. □

B Additional Experimental Results

B.1 Experiment Setting

For toy2D dataset. The toy2D dataset consists of three clusters A, B1, B2 where A, B are two classes. The data points are sampled from normal distributions, i.e., $A\sim\mathcal{N}\left((-2,0),\Sigma\right),B1\sim\mathcal{N}\left((2,0),\Sigma\right)$ and $B2\sim\mathcal{N}\left((6,0),\Sigma\right)$ where $\Sigma=0.5*I$ with I is the identity matrix. There are total 10k training samples and 2k testing samples with densities of three clusters are 10%, 50% and 40%, respectively.

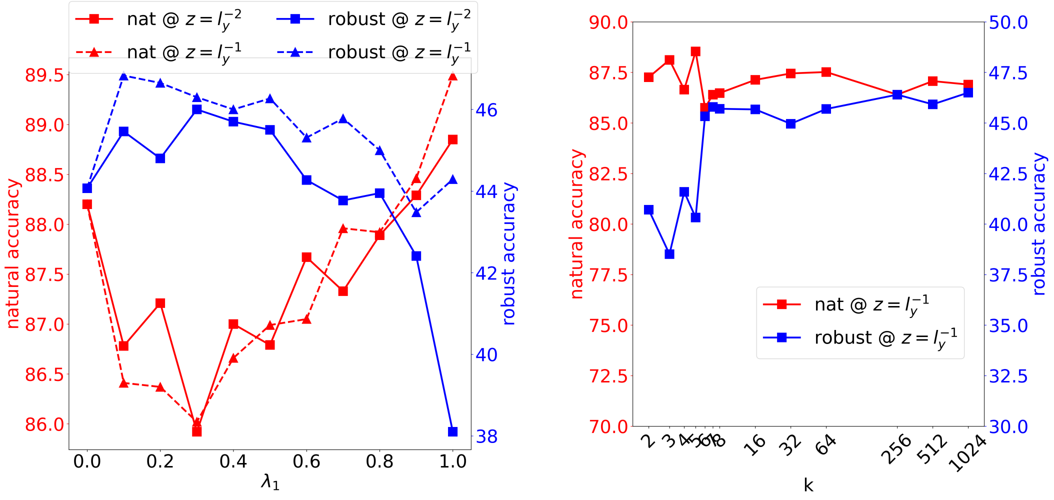
For MNIST dataset. We use a standard CNN architecture for the MNIST dataset which is identical with that in (Carlini and Wagner, 2017). We use SGD optimizer with momentum 0.9, starting learning rate 1e-2 and reduce the learning rate (×0.1) at epoch {55, 75, 90}. We train with 100 epochs.

For CIFAR10 dataset. We use the ResNet18 for the CIFAR10 dataset. We use SGD optimizer with momentum 0.9, weight decay 3.5e-3. The starting learning rate 1e-2 and reduce the learning rate (×0.1) at epoch {75, 90}. We train with 100 epochs.

Setting for the framework. The architecture for the binary discriminator h_d^b as follow: Input –> ReLU(FC(2k)) –> ReLU(FC(k)) –> Sigmoid(FC(1)), while that for the multi-class discriminator h_d^m: Input –> ReLU(FC(2k)) –> ReLU(FC(k)) –> FC(M), with FC(k) represents for a Fully-Connected layer with k hidden units, M is number of classes, Input represents for the discriminator's input (i.e., latent vector). Two first layers have been shared between two discriminators h_d^b, h_d^m. We use $k = 256$ in default while provide an analytical experiment with different value of k in Section B3. The optimizer and learning rate for the discriminator are similar to the classifier.

B.2 Analytical Experiments

Sensitivity to the tradeoff parameter. Here we would like to provide a study on the sensitivity on the tradeoff parameter λ_1, while fixing $\lambda_2 = 0$. The experiment has been conducted on the CIFAR10 dataset, with our GV-PGD variant. We compare two options of the latent space, when choosing the second last ($z = l_y^{-2}$) or the last hidden layer ($z = l_y^{-1}$) before the softmax layer. The result has been shown in Figure 2a. It can be seen that: (i) the robust accuracy increases then decreases when increasing the tradeoff parameter from 0.0 to 1.0. In contrast, the natural accuracy decreases then increases in the same range of the tradeoff parameter. The highest robust accuracy is 46.8% at $\lambda_1 = 0.1$ (ii) using the last hidden layer achieves a better robustness than using the second last hidden layer. In other experiments, we use the last hidden layer as the input of discriminator, with the tradeoff parameter $\lambda_1 = 0.3$ as default.

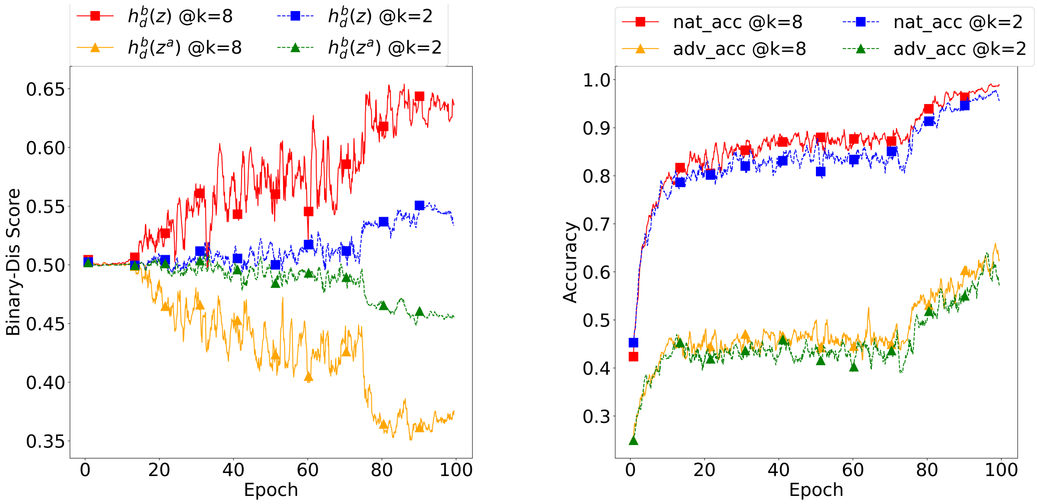


(a)

(b)

Figure 2: (a) Natural/Robust performance in correlation with the trade-off parameter λ_1. l_y^{-1}, l_y^{-2} represent for the last hidden layer and second last hidden layer before the softmax. (b) The Natural/Robustness performance in correlation with the discriminator capacity k. The x-axis is in log scale.

Sensitivity to the discriminator's capacity. We would like to provide a study on the impact of discriminator's capacity to the performance by varying number of hidden units k as described in Section B1. The result has been shown in Figure 2b. It can be seen that the robustness increases when increasing the discriminator's capacity. Specifically, the performance is not good with overly small discriminator (i.e., robust accuracy is less than 43% with $k < 6$ or less than 287 parameters). In contrast, increasing discriminator's capacity slightly improves the robustness of the model. In addition, we provide the training progress with the binary discriminator score with two values of discriminator's capacity k as shown in Figure 3. It can be seen that, the higher discriminator's capacity (i.e., $k = 8$) the higher distinguishable between natural input and adversarial examples. As a consequence, the model achieves better both natural and robustness by leveraging better knowledge from the bigger discriminator. In other experiments, we use $k = 256$ as the default setting.



(a) Binary discriminator score.

(b) Training natural/robust accuracy.

Figure 3: Training progress.