Adversarial Training Should Be Cast as a Non-Zero-Sum Game

Anonymous Authors[1]

Abstract

One prominent approach toward resolving the adversarial vulnerability of deep neural networks is the two-player zero-sum paradigm of adversarial training, in which predictors are trained against adversarially-chosen perturbations of data. Despite the promise of this approach, algorithms based on this paradigm have not engendered sufficient levels of robustness, and suffer from pathological behaviour like robust overfitting. To understand this shortcoming, we first show that the commonly used surrogate-based relaxation used in adversarial training algorithms voids all guarantees on the robustness of trained classifiers. The identification of this pitfall informs a novel non-zero-sum bilevel formulation of adversarial training, wherein each player optimizes a different objective function. Our formulation naturally yields a simple algorithmic framework that matches and in some cases outperforms state-of-the-art attacks, attains comparable levels of robustness to standard adversarial training algorithms, and does not suffer from robust overfitting.

1. Introduction

A longstanding disappointment in the machine learning (ML) community is that deep neural networks (DNNs) remain vulnerable to seemingly innocuous changes to their input data including nuisances in visual data (Hendrycks & Dietterich, 2019; Robey et al., 2020; Eykholt et al., 2018), sub-populations (Santurkar et al., 2021; Sohoni et al., 2020; Koh et al., 2021), and distribution shifts (Xiao et al., 2021; Arjovsky et al., 2019; Sagawa et al., 2020). Prominent amongst these vulnerabilities is the setting of *adversarial examples*, wherein it has been conclusively shown that imperceptible, adversarially-chosen perturbations can fool state-of-the-art classifiers parameterized by DNNs (Szegedy et al., 2013; Biggio et al., 2013; 2012; Carlini & Wagner, 2017). In response, a plethora of research has proposed so-called adversarial training (AT) algorithms (Huang et al., 2015; Wong & Kolter, 2018; Kurakin et al., 2017; Madry et al., 2018; Goodfellow et al., 2015), which are designed to improve robustness against adversarial examples.

AT is ubiquitously formulated as a *two-player zero-sum* game, where both players—often referred to as the *defender* and the *adversary*—respectively seek to minimize and maximize the classification error. However, this zero-sum game is not implementable in practice as the discontinuous nature of the classification error is not compatible with first-order optimization algorithms. To bridge this gap between theory and practice, it is commonplace to replace the classification error with a smooth surrogate loss (e.g., the cross-entropy loss) which is amenable to gradient-based optimization (Madry et al., 2018; Zhang et al., 2019). And while this seemingly harmless modification has a decades-long tradition in the ML literature due to the guarantees it imparts on non-adversarial objectives (Bartlett et al., 2006; Shalev-Shwartz & Ben-David, 2014; Roux, 2017), there is a pronounced gap in the literature regarding the implications of this relaxation on the standard formulation of AT.

As the field of robust ML has matured, surrogate-based AT algorithms (see, e.g., (Madry et al., 2018; Zhang et al., 2019; Goodfellow et al., 2015; Wang et al., 2020)) have collectively ushered in significant progress toward designing stronger attacks and obtaining more robust defenses (Croce et al., 2020a). However, despite these advances, recent years have witnessed a plateau in robustness measures on leaderboards such as RobustBench, resulting in the widely held beliefs that robustness and accuracy may be irreconcilable (Tsipras et al., 2019a; Dobriban et al., 2020; Javanmard et al., 2020) and that robust generalization requires significantly more data (Schmidt et al., 2018; Chen et al., 2020; Stutz et al., 2019). Moreover, various phenomena such as robust overfitting (Rice et al., 2020) and insufficient robustness evaluation (Carlini et al., 2019) have indicated that progress has been overestimated (Croce & Hein, 2020). To combat these pitfalls, state-of-the-art algorithms increasingly rely on ad-hoc regularization schemes (Kannan et al., 2018; Zhang et al., 2019; Chan et al., 2020; Hoffman et al., 2019; Finlay et al., 2018), weight perturbations (Wu et al., 2020; Sun et al., 2021; Foret et al., 2020), and heuristics such as multi-

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Under review by *AdvML-Frontiers'23: The 2nd Workshop on New Frontiers in Adversarial Machine Learning@ICML'23*, Honolulu, Hawaii, USA. Do not distribute.

ple restarts (Madry et al., 2018), carefully crafted learning rate schedules (Rice et al., 2020), and convoluted stopping conditions (Croce & Hein, 2020), all of which contribute to a growing literature concerned with identifying flaws in various AT schemes (Latorre et al., 2023).

Motivated by these challenges, we argue that the pervasive surrogate-based zero-sum approach to AT suffers from a fundamental flaw. Our analysis of the standard minimax formulation of AT reveals that maximizing a surrogate like the cross-entropy provides no guarantee that the the classification error will increase, resulting in weak adversaries and ineffective AT algorithms. In identifying this shortcoming, we prove that to preserve guarantees on the optimality of the classification error objective, the defender and the adversary must optimize different objectives, resulting in a *non-zero-sum* game. This leads to a novel, yet natural *bilevel* formulation (Bard, 2013) of AT in which the defender minimizes an upper bound on the classification error, while the attacker maximizes a continuous reformulation of the classification error. We then propose an algorithm based on our formulation which is free from ad hoc optimization techniques. Our empirical evaluations reveal that our approach matches the test robustness achieved by the state-of-the-art, yet highly heuristic approaches such as AutoAttack, and that it eliminates the problem of robust overfitting.

Contributions. We summarize our contributions as follows.

- **New formulation for adversarial robustness.** Starting from the discontinuous minmax formulation of AT WRT the 0-1 loss, we derive a novel continuous bilevel optimization formulation, the solution of which *guarantees* improved robustness against the optimal adversary.
- **New adversarial training algorithm.** We derive a new, heuristic-free algorithm (Algorithm 2) based on our bilevel formulation, and show that offers strong robustness on CIFAR-10.
- **Elimination of robust overfitting.** Without the need of heuristic modifications, our algorithm does not suffer from robust overfitting (RO). This suggest that RO is an artifact of the use of improper surrogates in the original AT paradigm, and that the use of a correct optimization formulation is enough to solve it.
- **State-of-the-art robustness evaluation.** We show that our proposed optimization objective for the adversary yields a simple algorithm that matches the performance of the state-of-the-art, yet highly complex AutoAttack method, on classifiers trained on CIFAR-10.

2. Promises and pitfalls of adversarial training

2.1. Preliminaries: Training DNNs with surrogate losses

We consider a k-way classification setting, wherein data arrives as instance-label pairs (X, Y) drawn i.i.d. from an unknown distribution $\mathcal{D}$ taking support over $\mathcal{X} \times \mathcal{Y} \subseteq \mathbb{R}^d \times [K]$, where $[K] := \{1, \ldots, K\}$. Given a suitable hypothesis class $\mathcal{F}$, one goal in this setting is to select an element $f \in \mathcal{F}$ which correctly predicts the label Y of a corresponding instance X. In practice, this hypothesis class $\mathcal{F}$ often comprises functions $f_\theta : \mathbb{R}^d \to \mathbb{R}^K$ which are parameterized by a vector $\theta \in \Theta \subset \mathbb{R}^p$, as is the case when training DNNs. In this scenario, the problem of learning a classifier that correctly predicts Y from X can written as follows:

$$\min_{\theta \in \Theta} \mathbb{E} \left\{ \operatorname*{arg\,max}_{i \in [K]} f_\theta(X)_i \neq Y \right\} \tag{1}$$

Here $f_\theta(X)_i$ denotes the i^{th} component of the logits vector $f_\theta(X) \in \mathbb{R}^K$ and we use the notation $\{A\}$ to denote the indicator function of an event A, i.e., $\{A\} := \mathbb{I}_A(\cdot)$. In this sense, $\{\arg\max_{i \in [K]} f_\theta(X)_i \neq Y\}$ denotes the *classification error* of f_θ on the pair (X, Y).

Prominent among the barriers to solving (1) in practice is the fact that the classification error is a discontinuous function of θ, which in turn renders continuous first-order methods (e.g., gradient descent) intractable. Fortunately, this pitfall can be resolved by minimizing a surrogate loss function $\ell : [k] \times [k] \to \mathbb{R}$ in place of the classification error (Shalev-Shwartz & Ben-David, 2014). For minimization problems, surrogate losses are chosen to be differentiable *upper bounds* of the classification error of f_θ, in the sense that

$$\left\{ \operatorname*{arg\,max}_{i \in [K]} f_\theta(X)_i \neq Y \right\} \leq \ell(f_\theta(X), Y). \tag{2}$$

This inequality gives rise to the following differentiable counterpart of (1) which is amenable to minimization via first-order methods:

$$\min_{\theta \in \Theta} \mathbb{E}\, \ell(f_\theta(X), Y). \tag{3}$$

Crucially, the inequality in (2) guarantees that the problem in (3) provides a solution that decreases the classification error (Bartlett et al., 2006), which, as discussed above, is the primary goal in supervised classification.

2.2. The pervasive setting of adversarial examples

For common hypothesis classes, it is well-known that classifiers obtained by solving (3) are sensitive to seemingly benign changes to their input data. Among these vulnerabilities, perhaps the most well-studied is the setting of adversarial examples, wherein a plethora of research has demonstrated that state-of-the-art classifiers can be fooled by small, adversarially-chosen perturbations (Szegedy et al., 2013; Biggio et al., 2013; 2012; Carlini & Wagner, 2017). In other words, given an instance label pair (X, Y), it is relatively straightforward to find perturbations $\eta \in \mathbb{R}^d$ with small norm $||\eta|| \le \epsilon$ for some fixed $\epsilon > 0$ such that the following equations simultaneously hold.

$$\arg\max_{i\in[K]} f_\theta(X)_i = Y \tag{4}$$

$$\arg\max_{i\in[K]} f_\theta(X+\eta)_i \neq \arg\max_{i\in[K]} f_\theta(X)_i \tag{5}$$

The task of finding such perturbations η which cause the classifier f_θ to misclassify perturbed data points $X+\eta$ can be compactly cast as the following maximization problem:

$$\eta^\star \in \arg\max_{\eta:\|\eta\|\le\epsilon} \left\{ \arg\max_{i\in[K]} f_\theta(X+\eta)_i \neq Y \right\} \tag{6}$$

Here, if both of the expressions in (4) hold for the perturbation $\eta = \eta^\star$, then the perturbed instance $X+\eta^\star$ is called an *adversarial example* for f_θ with respect to (X, Y).

Due to prevalence of adversarial examples, there has been pronounced interest in solving the robust analog of (1), which is designed to find classifiers that are insensitive to small perturbations. This robust analog is ubiquitously written as the following a two-player zero-sum game with respect to the discontinuous classification error:

$$\min_{\theta\in\Theta} \mathbb{E}\left[\max_{\eta:\|\eta\|\le\epsilon} \left\{ \arg\max_{i\in[K]} f_\theta(X+\eta)_i \neq Y \right\} \right] \tag{7}$$

An optimal solution $\theta^\star$ for (7) yields a model $f_{\theta^\star}$ that achieves the lowest possible classification error despite the presence of adversarial perturbations. For this reason, this problem—wherein the interplay between the maximization over η and the minimization over θ comprises a two-player zero-sum game— is the starting point for numerous algorithms which aim to improve robustness.

2.3. Surrogate-based approaches to robustness

As discussed in § 2.1, the discontinuity of the classification error complicates the task of finding adversarial examples, as in (6), and of training against these perturbed instances, as in (7). One appealing approach toward overcoming this pitfall is to simply deploy a surrogate loss in place of the classification error inside (7), which gives rise to the following pair of optimization problems:

$$\eta^\star \in \arg\max_{\eta:||\eta||\le\epsilon} \ell(f_\theta(X+\eta), Y) \tag{8}$$

$$\min_{\theta\in\Theta} \mathbb{E}\left[\max_{\eta:\|\eta\|\le\epsilon} \ell(f_\theta(X+\eta), Y) \right] \tag{9}$$

Indeed, this surrogate-based approach is pervasive in practice. Madry et al.'s seminal paper on the subject of adversarial training employs this formulation (Madry et al., 2018), which has subsequently been used as the starting point for numerous AT schemes (Wong & Kolter, 2018; Kurakin et al., 2017; Madry et al., 2018; Goodfellow et al., 2015).

Pitfalls of surrogate-based optimization. Despite the intuitive appeal of this paradigm, surrogate-based adversarial attacks are known to overestimate robustness (Mosbach et al., 2018; Croce et al., 2020b; Croce & Hein, 2020), and standard adversarial training algorithms are known to fail against strong attacks. Furthermore, this formulation suffers from pitfalls such as robust overfitting (Rice et al., 2020) and trade-offs between robustness and accuracy (Tsipras et al., 2019b). To combat these shortcomings, empirical adversarial attacks and defenses have increasingly relied on heuristics such as multiple restarts and variable learning rate schedules (Croce & Hein, 2020) resulting in a widening gap between the theory and practice of adversarial learning. In the next section, we argue that these pitfalls can be attributed to the fundamental limitations of (9).

3. Non-zero-sum adversarial training

3.1. Limitations of surrogates in adversarial learning

From an optimization perspective, the surrogate-based approaches to adversarial evaluation and training outlined in § 2.3 engenders two fundamental limitations.

Limitation I: Weak attackers. In the adversarial evaluation problem of (8), the adversary maximizes an *upper bound* on the classification error. This means that any solution $\eta^\star$ to (8) is not guaranteed to increase the classification error in (6), resulting in weakened adversaries which are misaligned with the goal of finding adversarial examples. Indeed, when the surrogate is an upper bound on the classification error, the only conclusion about the perturbation $\eta^\star$ obtained from (8) and its *true* objective (6) is:

$$\left\{ \arg\max_{i\in[K]} f_\theta(X+\eta^\star)_i \neq Y \right\} \le \max_{\eta:||\eta||\le\epsilon} \ell(f_\theta(X+\eta), Y) \tag{10}$$

Notably, the RHS of (10) can be arbitrarily large while the LHS can simultaneously be equal to zero, i.e., solving (8) can fail to produce an adversarial example, even at optimality. Thus, while it is known empirically that attacks based on (8) tend to overestimate robustness (Croce & Hein, 2020; Gowal et al., 2019), we show that this is evident *a priori*.

Limitation II: Ineffective defenders. Because attacks which seek to maximize upper bounds on the classification error are not proper surrogates for the classification error (c.f., Limitation I), training a model f_θ on such perturbations does not guarantee any improvement in robustness. Therefore, AT algorithms which seek to solve (9) are ineffective in that they do not optimize the worst-case classification error. Thus, it should not be surprising that robust overfitting (Rice et al., 2020) occurs for models trained to solve eq. (9).

Both of these limitations arise directly by virtue of rewriting (8) and (9) with the surrogate loss ℓ. Therefore, to summarize, there is a distinct tension between the efficient, yet misaligned paradigm of surrogate-based AT with the principled, yet intractable paradigm of minimax optimization on the classification error. In the remainder of this section, we resolve this tension by decoupling the optimization problems of the adversary and the training algorithm.

3.2. Decoupling adversarial attacks and defenses

Our starting point is the two-player zero-sum formulation in (7). Observe that this minimax optimization problem can be equivalently cast as a *bilevel* optimization problem[1]:

$$\min_{\theta\in\Theta} \quad \mathbb{E}\left\{ \operatorname*{arg\,max}_{i\in[K]} f_\theta(X+\eta^\star)_i \neq Y \right\} \tag{11}$$

$$\text{subject to} \quad \eta^\star \in \operatorname*{arg\,max}_{\eta:\, \|\eta\|\leq\epsilon} \left\{ \operatorname*{arg\,max}_{i\in[K]} f_\theta(X+\eta)_i \neq Y \right\} \tag{12}$$

While this problem still constitutes a zero-sum game, the role of the attacker (the constraint in (12)) and the role of the defender (the objective in (11)) are now decoupled. From this perspective, the tension engendered by introducing surrogate losses is laid bare: the attacker ought to maximize a *lower bound* of the classification error (c.f., Limitation I), whereas the defender ought to minimize an *upper bound* on the classification error (c.f., Limitation II). This implies that to preserve guarantees on optimality, the attacker and defender must optimize separate objectives. In what follows, we discuss these objectives for both players in detail.

The attacker's objective. We first address the role of the attacker. To do so, we define the *negative margin* $M_\theta(X,Y)$, $M_\theta : \mathcal{X}\times\mathcal{Y}\to\mathbb{R}^k$ of the classifier f_θ as follows:

$$M_\theta(X,Y)_j \triangleq f_\theta(X)_j - f_\theta(X)_Y \tag{13}$$

We call $M_\theta(X,Y)$ the negative margin because a positive value of (13) corresponds to a misclassification. As we show in the following proposition, the negative margin function (which is differentiable) provides an alternative characterization of the classification error.

Proposition 1. *Given data (X,Y), let $\eta^\star$ denote any maximizer of $M_\theta(X+\eta,Y)_j$ over the classes $j\in[K]-\{Y\}$ and perturbations $\eta\in\mathbb{R}^d$ satisfying $||\eta||\leq\epsilon$, i.e.,*

$$(j^\star,\eta^\star) \in \operatorname*{arg\,max}_{j\in[K]-\{Y\},\, \eta:\, ||\eta||\leq\epsilon} M_\theta(X+\eta,Y)_j. \tag{14}$$

Then if $M_\theta(X+\eta^\star,Y)_{j^\star}>0$, $\eta^\star$ induces a misclassification and satisfies the constraint in (12), so $X+\eta^\star$ is an adversarial example. Otherwise, if $M_\theta(X+\eta^\star,Y)_{j^\star}\leq 0$, then any $\eta : ||\eta|| < \epsilon$ satisfies (12), and no adversarial example exists for the pair (X,Y). In summary, if $\eta^\star$ is as in (14), then $\eta^\star$ solves the lower level problem in (12).

We present a proof in appendix C[2]. Proposition 1 implies that the non-differentiable constraint in (12) can be equivalently recast as an ensemble of K differentiable optimization problems that can be solved independently. This can collectively be expressed as

$$\eta^\star \in \operatorname*{arg\,max}_{\eta:\, ||\eta||<\epsilon} \max_{j\in[K]-\{Y\}} M_\theta(X+\eta,Y)_j. \tag{15}$$

Note that this does not constitute a relaxation; (12) and (15) are equivalent optimization problems. However, as (15) is differentiable almost everywhere, the attacker can maximize the classification error directly using first-order methods.

The defender's objective. Next, we consider the role of the defender. To handle the discontinuous upper-level problem in (11), note that this problem is equivalent to a perturbed version of the supervised learning problem in (1). As discussed in § 2.1, the strongest results for problems of this kind have historically been achieved via a surrogate-based relaxation. Subsequently, replacing the 0-1 loss with a differentiable upper bound like the cross-entropy is a principled, guarantee-preserving approach for the defender.

3.3. Putting the pieces together: Non-zero-sum AT

By combining the disparate problems discussed in the preceeding section, we arrive at a novel *non-zero-sum* (almost-everywhere) differentiable formulation of AT:

$$\min_{\theta\in\Theta} \quad \mathbb{E}\,\ell(f_\theta(X+\eta^\star),Y) \tag{16}$$

$$\text{subject to} \quad \eta^\star \in \operatorname*{arg\,max}_{\eta:\, \|\eta\|\leq\epsilon} \max_{j\in[K]-\{Y\}} M_\theta(X+\eta,y)_j \tag{17}$$

Notice that the second level of this bilevel problem remains non-smooth due to the maximization over the classes $j\in[K]-\{Y\}$. To impart smoothness on the problem without relaxing the constraint, observe that we can equivalently solve $K-1$ distinct smooth problems in the second level for each sample (X,Y), resulting in the following equivalent optimization problem:

$$\min_{\theta\in\Theta} \quad \mathbb{E}\,\ell(f_\theta(X+\eta^\star_{j^\star}),Y) \tag{18}$$

$$\text{subject to} \quad \eta^\star_j \in \operatorname*{arg\,max}_{\eta:\, \|\eta\|\leq\epsilon} M_\theta(X+\eta,y)_j \quad \forall j \tag{19}$$

$$j^\star \in \operatorname*{arg\,max}_{j\in[K]-\{Y\}} M_\theta(x+\eta^\star_j,y)_j \tag{20}$$

[1]To be precise, the optimal value $\eta^\star$ in (17) is a function of (X,Y), i.e., $\eta^\star=\eta^\star(X,Y)$, and the constraint must hold for almost every $(X,Y)\sim\mathcal{D}$.

[2]This result is similar in spirit to (Gowal et al., 2019, Theorem 3.1), although this prior result only holds for linear functions, whereas Proposition 1 holds for arbitrary functions.

Hence, in (20), we first obtain one perturbation $\eta_j^\star$ per class which maximizes the negative margin $M_\theta(X + \eta_j^\star, Y)$ for that particular class. Next, in (19), we select the class index $j^\star$ corresponding to the perturbation $\eta_j^\star$ that maximized the negative margin. And finally, in the upper level, the surrogate minimization over $\theta \in \Theta$ is on the perturbed data pair $(X + \eta_{j^\star}^\star, Y)$. The result is a non-zero-sum formulation for AT that is amenable to gradient-based optimization, and preserves the optimality guarantees engendered by surrogate loss minimization without weakening the adversary.

4. Algorithms

Given the non-zero-sum formulation of AT in the previous section, the next question is how one should solve this bilevel optimization problem in practice. Our starting point is the empirical version of this bilevel problem, wherein we assume access to a finite dataset $\{(x_i, y_i)\}_{i=1}^n$ of n instance-label pairs sampled i.i.d. from $\mathcal{D}$.

$$\min_{\theta \in \Theta} \quad \frac{1}{n} \sum_{i=1}^{n} \ell(f_\theta(x_i + \eta_{ij^\star}^\star), y_i) \tag{21}$$

$$\text{subject to} \quad \eta_{ij}^\star \in \operatorname*{arg\,max}_{\eta : \|\eta\| \leq \epsilon} M_\theta(x_i + \eta, y_i)_j \quad \forall i, j \tag{22}$$

$$j^\star \in \operatorname*{arg\,max}_{j \in [K] - \{y_i\}} M_\theta(x_i + \eta_{ij}^\star, y_i)_j \quad \forall i \tag{23}$$

To solve this empirical problem, we adopt a stochastic optimization based approach. That is, we first iteratively sample mini-batches from our dataset uniformly at random, and then obtain adversarial perturbations by solving the lower level problems in (22) and (23). Note that given the differentiability of the negative margin, the lower level problems can be solved iteratively with generic optimizers. This procedure is summarized in Algorithm 1, which we call the *BEst Targeted Attack (BETA)*, given that it directly maximizes the classification error.

After obtaining such perturbations, we calculate the perturbed loss in (21), and then differentiate through this loss with respect to the model parameters. By updating the model parameters θ in the negative direction of this gradient, our algorithm seeks classifiers that are robust against perturbations found by BETA. We call the full adversarial training procedure based on this attack *BETA Adversarial Training (BETA-AT)*, as it invokes BETA as a subroutine; see Algorithm 2 for details.

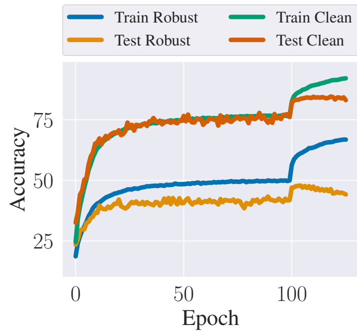
Smoothing the lower level. One potential limitation of the BETA-AT algorithm introduced in Algorithm 2 is its sample efficiency: BETA computes one adversarial perturbation per class, but only one of these perturbations is chosen for the upper level of the bilevel formulation (21). In this way, one could argue that there is wasted computational effort in discarding perturbations that achieve high values of the negative margin (13). This potential shortcoming is a byproduct of the non-smoothness of the max operator in (23). Fortunately, we can alleviate this limitation by using smooth under-approximations of the max (e.g., the softmax function), which is continuously differentiable. We explore this scheme in Appendix D.

5. Experiments

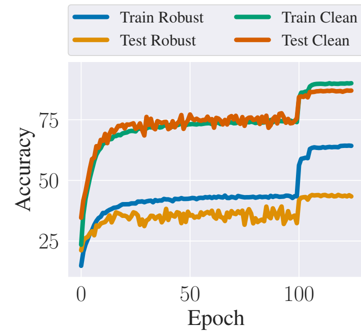
In this section, we evaluate the performance of BETA and BETA-AT on CIFAR-10 (Krizhevsky et al., 2009). Throughout, we consider a range of AT algorithms, including PGD (Madry et al., 2018), FGSM (Goodfellow et al., 2015), TRADES (Zhang et al., 2019), MART (Wang et al., 2020), as well as a range of adversarial attacks, including APGD and AutoAttack (Croce & Hein, 2020). We consider the standard perturbation budget of $\epsilon = 8/255$, and all training and test-time attacks use a step size of $\alpha = 2/255$. For both TRADES and MART, we set the trade-off parameter $\lambda = 5$, which is consistent with the original implementations (Wang et al., 2020; Zhang et al., 2019).

The bilevel formulation eliminates robust overfitting. Robust overfitting occurs when the robust test accuracy peaks immediately after the first learning rate decay, and then falls significantly in subsequent epochs as the model continues to train (Rice et al., 2020). This is illustrated in Figure 1a, in which we plot the learning curves (i.e., the clean and robust accuracies for the training and test sets) for a ResNet-18 (He et al., 2016) model trained using 10-step PGD against a 20-step PGD adversary. Notice that after the first learning rate decay step at epoch 100, the robust test accuracy spikes, before dropping off in subsequent epochs. On the other hand, BETA-AT does not suffer from robust overfitting, as shown in Figure 1b. We argue that this strength of our method is a direct result of our bilevel formulation, in which we train against a proper surrogate for the classification error.

BETA-AT outperforms baselines on the last iterate. We next compare the performance of ResNet-18 models trained using four different AT algorithms: FGSM, PGD, TRADES, MART, and BETA. PGD, TRADES, and MART used a 10-step adversary at training time. At test time, the models were evaluated against five different adversaries: FGSM, 10-step PGD, 40-step PGD, 10-step BETA, and APGD. We report the performance of two different checkpoints for each algorithm: the best performing checkpoint chosen by early stopping on a held-out validation set, and the performance of the last checkpoint from training. Note that while BETA performs comparably to the baseline algorithms with respect to early stopping, it outperforms these algorithms significantly when the test-time adversaries attack the last checkpoint of training. This owes to the fact that BETA does not suffer from robust overfitting, meaning that the last and best checkpoints perform similarly.



(a) **PGD^{10} learning curves.**



(b) **BETA-AT^{10} learning curves.**

Figure 1: **BETA does not suffer from robust overfitting.** We plot the learning curves against a PGD^{20} adversary for PGD^{10} and BETA-AT^{10}. Observe that although PGD displays robust overfitting after the first learning rate decay step, BETA-AT does not suffer from this pitfall.

Table 1: **Adversarial performance on CIFAR-10.** Ttest accuracies of various AT algorithms on the CIFAR-10 dataset.

Training algorithm	Test accuracy											
	Clean		FGSM		PGD^{10}		PGD^{40}		BETA^{10}		APGD	
	Best	Last	Best	Last	Best	Last	Best	Last	Best	Last	Best	Last
FGSM	81.96	75.43	94.26	94.22	42.64	1.49	42.66	1.62	40.30	0.04	41.56	0.00
PGD^{10}	83.71	83.21	51.98	47.39	46.74	39.90	45.91	39.45	43.64	40.21	44.36	42.62
TRADES^{10}	81.64	81.42	52.40	51.31	47.85	42.31	47.76	42.92	44.31	40.97	43.34	41.33
MART^{10}	78.80	77.20	53.84	53.73	49.08	41.12	48.41	41.55	44.81	41.22	45.00	42.90
BETA-AT^{5}	87.02	86.67	51.22	51.10	44.02	43.22	43.94	42.56	42.62	42.61	41.44	41.02
BETA-AT^{10}	85.37	85.30	51.42	51.11	45.67	45.39	45.22	45.00	44.54	44.36	44.32	44.12
BETA-AT^{20}	82.11	81.72	54.01	53.99	49.96	48.67	49.20	48.70	46.91	45.90	45.27	45.25

BETA matches the robustness estimate of AutoAttack. AutoAttack is a state-of-the-art adversarial attack which is widely used to estimate the robustness of trained models on leaderboards such as RobustBench (Croce et al., 2020a; Croce & Hein, 2020). In brief, AutoAttack comprises a collection of four disparate attacks and involves several heuristics, including multiple restarts and variable stopping conditions. In Table 2, we compare the performance of the top-performing models on RobustBench against AutoAttack, APGD-T, and BETA with RMSprop. Both APGD-T and BETA used thirty steps, whereas we used the default implementation of AutoAttack, which runs for 100 iterations. We also recorded the gap between AutoAttack and BETA. Notice that the 30-step BETA—a heuristic-free algorithm derived from our bilevel formulation of AT—performs almost identically to AutoAttack, despite the fact that AutoAttack runs for significantly more iterations and uses five restarts, which endows AutoAttack with an unfair computational advantage. That is, excepting for a negligible number of samples, BETA matches the robustness estimate of AutoPGD-targeted and AutoAttack, despite using an off-the-shelf optimizer.

Table 2: Estimated ℓ_∞ robustness (robust test accuracy). BETA+RMSprop (ours) vs APGD-targeted (APGD-T) vs AutoAttack (AA). CIFAR-10. BETA and APGD-T use 30 iterations + single restart. $\epsilon = 8/255$. AA uses 4 different attacks with 100 iterations and 5 restarts.

Model	BETA	APGD-T	AA	BETA/AA gap	Architecture
Wang et al. (2023)	70.78	70.75	70.69	0.09	WRN-70-16
Wang et al. (2023)	67.37	67.33	67.31	0.06	WRN-28-10
Rebuffi et al. (2021)	66.75	66.71	66.58	0.17	WRN-70-16
Gowal et al. (2021)	66.27	66.26	66.11	0.16	WRN-70-16
Huang et al. (2022)	65.88	65.88	65.79	0.09	WRN-A4
Rebuffi et al. (2021)	64.73	64.71	64.64	0.09	WRN-106-16
Rebuffi et al. (2021)	64.36	64.27	64.25	0.11	WRN-70-16
Gowal et al. (2021)	63.58	63.45	63.44	0.14	WRN-28-10
Pang et al. (2022)	63.38	63.37	63.35	0.03	WRN-70-16

References

Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019. 1

Bard, J. F. *Practical bilevel optimization: algorithms and applications*, volume 30. Springer Science & Business Media, 2013. 2, 10

Bartlett, P. L., Jordan, M. I., and McAuliffe, J. D. Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101(473):138–156, 2006. doi: 10.1198/016214505000000907. URL https://doi.org/10.1198/016214505000000907. 1, 2

Biggio, B., Nelson, B., and Laskov, P. Poisoning attacks against support vector machines. *arXiv preprint arXiv:1206.6389*, 2012. 1, 2

Biggio, B., Corona, I., Maiorca, D., Nelson, B., Srndic, N., Laskov, P., Giacinto, G., and Roli, F. Evasion attacks against machine learning at test time. In *ECML/PKKD*, 2013. 1, 2

Carlini, N. and Wagner, D. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pp. 39–57. Ieee, 2017. 1, 2

Carlini, N., Athalye, A., Papernot, N., Brendel, W., Rauber, J., Tsipras, D., Goodfellow, I., Madry, A., and Kurakin, A. On evaluating adversarial robustness. *arXiv preprint arXiv:1902.06705*, 2019. 1

Chan, A., Tay, Y., Ong, Y. S., and Fu, J. Jacobian adversarially regularized networks for robustness. *ICLR*, 2020. 1

Chen, L., Min, Y., Zhang, M., and Karbasi, A. More data can expand the generalization gap between adversarially robust and standard models. In *International Conference on Machine Learning*, pp. 1670–1680. PMLR, 2020. 1

Chen, T., Zhang, Z., Liu, S., Chang, S., and Wang, Z. Robust overfitting may be mitigated by properly learned smoothening. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=qZzy5urZw9. 10

Croce, F. and Hein, M. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, pp. 2206–2216. PMLR, 2020. 1, 2, 3, 5, 6, 10

Croce, F., Andriushchenko, M., Sehwag, V., Debenedetti, E., Flammarion, N., Chiang, M., Mittal, P., and Hein, M. Robustbench: a standardized adversarial robustness benchmark. *arXiv preprint arXiv:2010.09670*, 2020a. 1, 6

Croce, F., Rauber, J., and Hein, M. Scaling up the randomized gradient-free adversarial attack reveals overestimation of robustness using established attacks. *International Journal of Computer Vision*, 128:1028–1046, 2020b. 3, 10

Dobriban, E., Hassani, H., Hong, D., and Robey, A. Provable tradeoffs in adversarially robust classification. *arXiv preprint arXiv:2006.05161*, 2020. 1

Dong, Y., Xu, K., Yang, X., Pang, T., Deng, Z., Su, H., and Zhu, J. Exploring memorization in adversarial training. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=7gE9V9GBZaI. 10

Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., Prakash, A., Kohno, T., and Song, D. Robust physical-world attacks on deep learning visual classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1625–1634, 2018. 1

Finlay, C., Calder, J., Abbasi, B., and Oberman, A. Lipschitz regularized deep neural networks generalize and are adversarially robust. *arXiv preprint arXiv:1808.09540*, 2018. 1

Foret, P., Kleiner, A., Mobahi, H., and Neyshabur, B. Sharpness-aware minimization for efficiently improving generalization. *arXiv preprint arXiv:2010.01412*, 2020. 1

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In *ICLR*, 2015. 1, 3, 5

Gowal, S., Uesato, J., Qin, C., Huang, P.-S., Mann, T., and Kohli, P. An alternative surrogate loss for pgd-based adversarial testing. *arXiv preprint arXiv:1910.09338*, 2019. 3, 4, 10

Gowal, S., Rebuffi, S.-A., Wiles, O., Stimberg, F., Calian, D. A., and Mann, T. Improving robustness using generated data. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=0NXUSlb6oEu. 6

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *CVPR*, 2016. 5

Hendrycks, D. and Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019. 1

Hoffman, J., Roberts, D. A., and Yaida, S. Robust learning with jacobian regularization. *arXiv preprint arXiv:1908.02729*, 2019. 1

Huang, R., Xu, B., Schuurmans, D., and Szepesvari, C. Learning with a strong adversary. *ArXiv*, abs/1511.03034, 2015. 1

Huang, S., Lu, Z., Deb, K., and Naresh Boddeti, V. Revisiting Residual Networks for Adversarial Robustness: An Architectural Perspective. *arXiv e-prints*, art. arXiv:2212.11005, December 2022. doi: 10.48550/arXiv.2212.11005. 6

Javanmard, A., Soltanolkotabi, M., and Hassani, H. Precise tradeoffs in adversarial training for linear regression. In *Conference on Learning Theory*, pp. 2034–2078. PMLR, 2020. 1

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

Koh, P. W., Sagawa, S., Marklund, H., Xie, S. M., Zhang, M., Balsubramani, A., Hu, W., Yasunaga, M., Phillips, R. L., Gao, I., et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, pp. 5637–5664. PMLR, 2021. 1

Krizhevsky, A., Nair, V., and Hinton, G. Cifar datasets (canadian institute for advanced research). 2009. URL http://www.cs.toronto.edu/~kriz/cifar.html. 5

Kurakin, A., Goodfellow, I., and Bengio, S. Adversarial examples in the physical world. *ICLR Workshop*, 2017. URL https://openreview.net/forum?id=HJGU3Rodl. 1, 3

Latorre, F., Krawczuk, I., Dadi, L. T., Pethick, T., and Cevher, V. Finding actual descent directions for adversarial training. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=I3HCE7Ro78H. 2

Lee, S., Lee, H., and Yoon, S. Adversarial vertex mixup: Toward better adversarially robust generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 10

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018. 1, 2, 3, 5, 10

Mosbach, M., Andriushchenko, M., Trost, T., Hein, M., and Klakow, D. Logit pairing methods can fool gradient-based attacks. *arXiv preprint arXiv:1810.12042*, 2018. 3

Pang, T., Lin, M., Yang, X., Zhu, J., and Yan, S. Robustness and accuracy could be reconcilable by (Proper) definition. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., and Sabato, S. (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 17258–17277. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/pang22a.html. 6

Rebuffi, S.-A., Gowal, S., Calian, D. A., Stimberg, F., Wiles, O., and Mann, T. Fixing data augmentation to improve adversarial robustness. *arXiv preprint arXiv:2103.01946*, 2021. 6, 10

Rice, L., Wong, E., and Kolter, J. Z. Overfitting in adversarially robust deep learning. In *ICML*, 2020. 1, 2, 3, 5, 10

Robey, A., Hassani, H., and Pappas, G. J. Model-based robust deep learning: Generalizing to natural, out-of-distribution data. *arXiv preprint arXiv:2005.10247*, 2020. 1

Roux, N. L. Tighter bounds lead to improved classifiers. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=HyAbMKwxe. 1

Sagawa, S., Raghunathan, A., Koh, P. W., and Liang, P. An investigation of why overparameterization exacerbates spurious correlations. In *International Conference on Machine Learning*, pp. 8346–8356. PMLR, 2020. 1

Santurkar, S., Tsipras, D., and Madry, A. Breeds: Benchmarks for subpopulation shift. *International Conference on Learning Representations*, 2021. 1

Schmidt, L., Santurkar, S., Tsipras, D., Talwar, K., and Madry, A. Adversarially robust generalization requires more data. *Advances in neural information processing systems*, 31, 2018. 1

Shalev-Shwartz, S. and Ben-David, S. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014. 1, 2

Sohoni, N., Dunnmon, J., Angus, G., Gu, A., and Ré, C. No subclass left behind: Fine-grained robustness in coarse-grained classification problems. *Advances in Neural Information Processing Systems*, 33:19339–19352, 2020. 1

Stutz, D., Hein, M., and Schiele, B. Disentangling adversarial robustness and generalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6976–6987, 2019. 1

Sun, X., Zhang, Z., Ren, X., Luo, R., and Li, L. Exploring the vulnerability of deep neural networks: A study of parameter corruption. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 11648–11656, 2021. 1

Szegedy, C., Zaremba, W., Sutskever, I., Joan Bruna, D. E., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. In *ICLR*, 2013. 1, 2

Tsipras, D., Santurkar, S., Engstrom, L., Turner, A., and Madry, A. Robustness may be at odds with accuracy. In *ICLR*, 2019a. 1

Tsipras, D., Santurkar, S., Engstrom, L., Turner, A., and Madry, A. Robustness may be at odds with accuracy. In *ICLR*, 2019b. 3

Wang, Y., Zou, D., Yi, J., Bailey, J., Ma, X., and Gu, Q. Improving adversarial robustness requires revisiting misclassified examples. *ICLR*, 2020. 1, 5, 10

Wang, Z., Pang, T., Du, C., Lin, M., Liu, W., and Yan, S. Better diffusion models further improve adversarial training. *arXiv preprint arXiv:2302.04638*, 2023. 6

Wong, E. and Kolter, Z. Provable defenses against adversarial examples via the convex outer adversarial polytope. *ICML*, 2018. 1, 3

Wong, E., Rice, L., and Kolter, J. Z. Fast is better than free: Revisiting adversarial training. *ICLR*, 2020. 10

Wu, D., tao Xia, S., and Wang, Y. Adversarial weight perturbation helps robust generalization. *NeurIPS*, 2020. 1

Xiao, K., Engstrom, L., Ilyas, A., and Madry, A. Noise or signal: The role of image backgrounds in object recognition. *International Conference on Machine Learning*, 2021. 1

Yu, C., Han, B., Shen, L., Yu, J., Gong, C., Gong, M., and Liu, T. Understanding robust overfitting of adversarial training and beyond. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., and Sabato, S. (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 25595–25610. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/yu22b.html. 10

Zhang, H., Yu, Y., Jiao, J., Xing, E. P., Ghaoui, L. E., and Jordan, M. I. Theoretically principled trade-off between robustness and accuracy. In *ICML*, 2019. 1, 5

Zhang, Y., Zhang, G., Khanduri, P., Hong, M., Chang, S., and Liu, S. Revisiting and advancing fast adversarial training through the lens of bi-level optimization. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., and Sabato, S. (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 26693–26712. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/zhang22ak.html. 10

A. Additional related work

Robust overfitting. Several recent papers (see, e.g., (Rebuffi et al., 2021; Chen et al., 2021; Yu et al., 2022; Dong et al., 2022; Wang et al., 2020; Lee et al., 2020)) have attempted to explain and resolve robust overfitting (Rice et al., 2020). However, none of these works point to a fundamental limitation of adversarial training as the cause of robust overfitting. Rather, much of this past work has focused on proposing heuristics for algorithms specifically designed to reduce robust overfitting, rather than to improve adversarial training. In contrast, we posit that the lack of guarantees of the zero-sum surrogate-based AT paradigm (Madry et al., 2018) is at fault, as this paradigm is not designed to maximize robustness with respect to classification error. And indeed, our empirical evaluations in the previous section confirm that our non-zero-sum formulation eliminates robust overfitting.

Estimating adversarial robustness. There is empirical evidence that attacks based on surrogates (e.g., PGD) overestimate the robustness of trained classifiers (Croce & Hein, 2020; Croce et al., 2020b; Gowal et al., 2019). Indeed, this evidence served as motivation for the formulation of more sophisticated attacks like AutoAttack (Croce & Hein, 2020), which empirically tend to provide more accurate estimates of robustness. In contrast, we provide solid, theoretical evidence that commonly used attacks overestimate robustness due to the misalignment between standard surrogate losses and the adversarial classification error. Moreover, we show that optimizing the BETA objective with a standard optimizer (e.g., RMSprop) achieves the same robustness as AutoAttack without employing ad hoc training procedures such as multiple restarts. convoluted stopping conditions, or adaptive learning rates.

One notable feature of past work is an overservation made in (Gowal et al., 2019), which finds that multitargeted attacks tend to more accurately estimate robustness. However, their theoretical analysis only applies to linear functions, whereas our work extends these ideas to the nonlinear setting of DNNs. Moreover, (Gowal et al., 2019) do not explore *training* using a multitargeted attack, whereas we show that BETA-AT is an effective AT algorithm that mitigates the impact of robust overfitting.

Bilevel formulations of AT. Prior to our work, (Zhang et al., 2022) proposed a different *pseudo-bilevel*[3] formulation for AT, wherein the main objective was to justify the Fast AT algorithm introduced in (Wong et al., 2020). More specifically, the formulation in (Zhang et al., 2022) is designed to produce solutions that coincide with the iterates of Fast AT by linearizing the attacker's objective. In contrast, our bilevel formulation appears naturally following principled relaxations of the intractable classification error AT formulation. In this way, the formulation in (Zhang et al., 2022) applies only in the context of Fast AT, whereas our formulation deals more generally with the task of adversarial training.

[3]In a strict sense, the formulation of (Zhang et al., 2022) is not a bilevel problem. In general, the most concise way to write a bilevel optimization problem is $\min_\theta f(\theta, \delta^\star(\theta))$ subject to $\delta^\star(\theta) \in \arg\max g(\theta, \delta)$. In such problems the value $\delta^\star(\theta)$ only depends on θ, as the objective function $g(\theta, \cdot)$ is then uniquely determined. This is not the case in (Zhang et al., 2022, eq. (7)), where an additional variable z appears, corresponding to the random initialization of Fast-AT. Hence, in (Zhang et al., 2022) the function $g(\theta, \cdot)$ is not uniquely defined by θ, but is a random function realized at each iteration of the algorithm. Thus, it is not a true bilevel optimization problem in the sense of the textbook definition (Bard, 2013).

Algorithm 1 Best Targeted Attack (BETA)

Input: Data-label pair (x, y), perturbation size ϵ, model f_θ, number of classes K, iterations T
Output: Adversarial perturbation $\eta^\star$
for $j \in 1, \ldots, K$ **do**
  $\eta_j \leftarrow$ Unif$[\max(X - \epsilon, 0), \min(X + \epsilon, 1)]$ {(}assume images in $[0, 1]^d$)
end for
for $t = 1, \ldots, T$ **do**
  for $j \in 1, \ldots, K$ **do**
    $\eta_j \leftarrow$ OPTIM$(\eta_j, \nabla_{\eta_i} M_\theta(x + \eta_j, y)_j)$ {(}optimizer step, e.g., RMSprop)
    $\eta_j \leftarrow \Pi_{B_\epsilon(X) \cap [0,1]^d}(\eta_j)$
  end for
end for $j^\star \leftarrow \arg\max_{j \in [K] - \{y\}} M_\theta(x + \eta_j, y)$

Algorithm 2 BETA Adversarial Training (BETA-AT)

Input: Dataset $(X, Y) = (x_i, y_i)_{i=1}^n$, perturbation size ϵ, model f_θ, number of classes K, iterations T, attack iterations T'
Output: Robust model $f_{\theta^\star}$
for $t \in 1, \ldots, T$ **do**
  Sample $i \sim$ Unif$[n]$ $\eta^\star \leftarrow$ BETA$(x_i, y_i, \epsilon, f_\theta, T')$
  $L(\theta) \leftarrow \ell(f_\theta(x_i + \eta^\star), y_i)$
  $\theta \leftarrow$ OPTIM$(\theta, \nabla L(\theta))$
end for

B. Pseudocode for BETA

In this appendix, we provide the pseudocode for BETA in Algorithms 1 and 2.

C. Proof of proposition 1

Suppose there exists $\hat{\eta}$ satisfying $||\hat{\eta}|| \leq \epsilon$ such that for some $j \in [K]$, $j \neq Y$ we have $M_\theta(X+\hat{\eta}, Y)_j > 0$, i.e., assume

$$\max_{j \in [K]-\{Y\},\, \eta: \|\eta\| \leq \epsilon} M_\theta(X+\eta, Y)_j > 0 \tag{24}$$

for such $\hat{\eta}$ and such j we have $f_\theta(X+\hat{\eta})_j > f_\theta(X+\hat{\eta})_Y$ and thus $\arg\max_{j \in [K]} f_\theta(X+\hat{\eta})_j \neq Y$. Hence, such $\hat{\eta}$ induces a misclassification error i.e.,

$$\hat{\eta} \in \arg\max_{\eta: \|\eta\|_2 \leq \epsilon} \left\{ \arg\max_{j \in [K]} f_\theta(X+\eta)_j \neq Y \right\} \tag{25}$$

In particular if

$$(j^\star, \eta^\star) \in \arg\max_{j \in [K]-\{Y\},\, \eta: \|\eta\| \leq \epsilon} M_\theta(X+\eta, Y)_j \Rightarrow \eta^\star \in \arg\max_{\eta: \|\eta\|_2 \leq \epsilon} \left\{ \arg\max_{j \in [K]} f_\theta(X+\eta)_j \neq Y \right\} \tag{26}$$

Otherwise, assume

$$\max_{j \in [K]-\{Y\},\, \eta: \|\eta\| \leq \epsilon} M_\theta(X+\eta, Y)_j < 0, \tag{27}$$

then for all $\eta : ||\eta|| < \epsilon$ and all $j \neq Y$ we have $f_\theta(X+\eta)_j < f_\theta(X+\eta)_Y$, so that $\arg\max_{j \in [K]} f_\theta(x+\eta)_j = Y$ i.e., there is no adversarial example in the ball. In this case for any η, in particular In particular if

$$(j^\star, \eta^\star) \in \arg\max_{j \in [K]-\{Y\},\, \eta: \|\eta\| \leq \epsilon} M_\theta(X+\eta, Y)_j \tag{28}$$

Then

$$0 = \left\{ \arg\max_{j \in [K]} f_\theta(X+\eta^\star)_j \neq Y \right\} = \max_{\eta: \|\eta\|_2 \leq \epsilon} \left\{ \arg\max_{j \in [K]} f_\theta(X+\eta)_j \neq Y \right\} \tag{29}$$

In conclusion, the solution

$$(j^\star, \eta^\star) \in \arg\max_{j \in [K]-\{Y\},\, \eta: \|\eta\| \leq \epsilon} M_\theta(X+\eta, Y)_j \tag{30}$$

always yields a maximizer of the misclassification error.

D. Smooth reformulation of the lower level

First, note that the problem in eqs. (21) to (23) is equivalent to

$$\begin{aligned} &\min_{\theta\in\Theta} \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{K} \lambda_{ij}^{\star}\ell(f_\theta(x_i+\eta_{ij}^{\star}), y_i) \\ &\text{subject to } \lambda_{ij}^{\star}, \eta_{ij}^{\star} \in \underset{\substack{\|\eta_{ij}\|\le\epsilon \\ \lambda_{ij}\ge 0, \|\lambda_i\|_1=1, \lambda_{iy}=0}}{\arg\max} \sum_{j=1}^{K}\lambda_{ij}M_\theta(x_i+\eta_{ij}, y_i)_j \qquad \forall i\in[n] \end{aligned} \tag{31}$$

This is because the maximum over λ_i in eq. (31) is always attained at the coordinate vector $\mathbf{e}_j$ such that $M_\theta(x_i+\eta_{ij}^{\star}, y_i)$ is maximum.

An alternative is to smooth the lower level optimization problem by adding an entropy regularization:

$$\begin{aligned} \max_{\eta:\|\eta\|\le\epsilon} \max_{j\in[K]-\{y\}} M_\theta(x+\eta_j, y)_j &= \max_{\eta:\|\eta\|\le\epsilon} \max_{\lambda\ge 0, \|\lambda\|_1=1, \lambda_y=0} \langle \lambda, M_\theta(x+\eta_j, y)_{j=1}^{K}\rangle \\ &\ge \max_{\eta:\|\eta\|\le\epsilon} \max_{\lambda\ge 0, \|\lambda\|_1=1, \lambda_y=0} \langle \lambda, M_\theta(x+\eta_j, y)_{j=1}^{K}\rangle - \frac{1}{\mu}\sum_{j=1}^{K}\lambda_j\log(\lambda_j) \\ &= \max_{\eta:\|\eta\|\le\epsilon} \frac{1}{\mu}\log\left(\sum_{\substack{j=1 \\ j\ne y}}^{K} e^{\mu M_\theta(X+\eta,y)_j}\right) \end{aligned} \tag{32}$$

where $\mu > 0$ is some *temperature* constant. The inequality here is due to the fact that the entropy of a discrete probability λ is positive. The innermost maximization problem in (32) has the closed-form solution:

$$\lambda_j^{\star} = \frac{e^{\mu M_\theta(x+\eta_j, y)_j}}{\sum_{\substack{j=1 \\ j\ne y}}^{K} e^{\mu M_\theta(x+\eta_j, y)_j}} : j\ne y, \qquad \lambda_y^{\star} = 0 \tag{33}$$

Hence, after relaxing the second level maximization problem following eq. (32), and plugging in the optimal values for λ we arrive at:

$$\begin{aligned} &\min_{\theta\in\Theta} \frac{1}{n}\sum_{i=1}^{n}\sum_{\substack{j=1 \\ j\ne y_i}}^{K} \frac{e^{\mu M_\theta(x_i+\eta_{ij}, y_i)_j}}{\sum_{\substack{j=1 \\ j\ne y_i}}^{K} e^{\mu M_\theta(x_i+\eta_{ij}, y_i)_j}}\ell(f_\theta(x_i+\eta_{ij}^{\star}), y_i) \\ &\text{subject to } \eta_{ij}^{\star} \in \underset{\|\eta_{ij}\|\le\epsilon}{\arg\max} M_\theta(x_i+\eta_{ij}, y_i)_j \qquad \forall i\in[n], j\in[K] \end{aligned} \tag{34}$$

$$\min_{\theta\in\Theta} \quad \frac{1}{n}\sum_{i=1}^{n}\sum_{\substack{j=1 \\ j\ne y_i}}^{K} \frac{e^{\mu M_\theta(x_i+\eta_{ij}^{\star}, y_i)_j}}{\sum_{\substack{j=1 \\ j\ne y_i}}^{K} e^{\mu M_\theta(x_i+\eta_{ij}^{\star}, y_i)_j}}\ell(f_\theta(x_i+\eta_{ij}^{\star}), y_i) \tag{35}$$

$$\text{subject to} \quad \eta_{ij}^{\star} \in \underset{\eta:\|\eta\|\le\epsilon}{\arg\max} M_\theta(x_i+\eta, y_i)_j \qquad \forall i\in[n] \tag{36}$$

In this formulation, both upper- and lower-level problems are smooth (barring the possible use of nonsmooth components like ReLU). Most importantly (I) the smoothing is obtained through a lower bound of the original objective in eqs. (22) and (23), retaining guarantees that the adversary will increase the misclassification error and (II) all the adversarial perturbations obtained for each class now appear in the upper level (35), weighted by their corresponding negative margin. In this way, we make efficient use of all perturbations generated: if two perturbations from different classes achieve the same negative margin, they will affect the upper-level objective in fair proportion. This formulation gives rise to algorithm 3.

Algorithm 3 Smooth BETA Adversarial Training (SBETA-AT)

Input: Dataset $(X, Y) = (x_i, y_i)_{i=1}^n$, perturbation size ϵ, model f_θ, number of classes K, iterations T, attack iterations T', temperature $\mu > 0$
Output: Robust model $f_{\theta^\star}$
for $t \in 1, \ldots, T$ **do**
  Sample $i \sim \text{Unif}[n]$
  Initialize $\eta_j \sim \text{Unif}[\max(0, x_i - \epsilon), \min(x_i + \epsilon, 1)], \forall j \in [K]$
  for $j \in 1, \ldots, K$ **do**
    for $t \in 1, \ldots, T'$ **do**
      $\eta_j \leftarrow \text{OPTIM}(\eta_j, \nabla_\eta M_\theta(x_i + \eta_j, y_i)_j)$ {(}attack optimizer step, e.g., RMSprop)
      $\eta_j \leftarrow \Pi_{B_\epsilon(x_i) \cap [0,1]^d}(\eta_j)$ {(}projection onto valid perturbation set)
    end for
  end for
  Compute $L(\theta) = \sum_{j=1, j \neq y_i}^K \frac{e^{\mu M_\theta(x_i+\eta_j, y_i)_j}}{\sum_{j=1, j \neq y_i}^K e^{\mu M_\theta(x_i+\eta_j, y_i)_j}} \ell(f_\theta(x_i + \eta_j), y_i)$
  $\theta \leftarrow \text{OPTIM}(\theta, \nabla L(\theta))$ {(}model optimizer step)
end for