ON THE CONVERGENCE OF CERTIFIED ROBUST TRAINING WITH INTERVAL BOUND PROPAGATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Interval Bound Propagation (IBP) is so far the base of state-of-the-art methods for training neural networks with certifiable robustness guarantees when potential adversarial perturbations present, while the convergence of IBP training remains unknown in existing literature. In this paper, we present a theoretical analysis on the convergence of IBP training. With an overparameteried assumption, we analyze the convergence of IBP robust training. We show that for a randomly initialized two-layer ReLU neural network trained with IBP, with sufficiently small perturbation radius and large network width, gradient descent for IBP training can converge to zero training error with a linear convergence rate for logistic loss, and the training can converge to a point where the robustness certification by IBP accurately reflects the true robustness of the model, with a high probability.

1 INTRODUCTION

It has been shown that deep neural networks are vulnerable against adversarial examples (Szegedy et al., 2013; Goodfellow et al., 2015; Carlini & Wagner, 2017; Kurakin et al., 2016), where a human imperceptible adversarial perturbation can easily alter the network's prediction. This poses concerns to safety-critical applications such as autonomous vehicles, healthcare or finance systems. To combat adversarial examples, many defense mechanisms have been proposed in the past few years (Kurakin et al., 2016; Madry et al., 2018; Zhang et al., 2019; Guo et al., 2018; Song et al., 2018; Xiao et al., 2020). However, due to the lack of reliable measurement on adversarial robustness, many defense methods are later broken by stronger attacks (Carlini & Wagner, 2017; Athalye et al., 2018; Tramer et al., 2020).

There are recently a line of robust training works, known as **certified defense**, focusing on training neural networks with certified robustness, where the evaluation of robustness needs to be provable – the network is considered robust on a test example if and only if the prediction is provably correct for any perturbation in a predefined set (e.g., a small ℓ_∞ ball) (Wang et al., 2018b; Bunel et al., 2018; Zhang et al., 2018; Wang et al., 2018c; Wong & Kolter, 2018; Singh et al., 2018; 2019; Weng et al., 2018; Xu et al., 2020). Unlike empirical methods that evaluate robustness with adversarial attacks and thus the evaluation can be limited by the strength of the attack algorithm, certified defense methods provide provable robustness guarantees without referring to any specific attack.

To obtain a neural network with certified robustness, a common practice is to derive a neural network verification method that computes upper and lower bound of output neurons given an input region (e.g., a small ℓ_∞ ball around a test sample), and then train the neural network to optimize the loss defined not on the original output, but on the worst-case output produced by the verification method. Many methods along this line have been proposed in the past few years (Wong & Kolter, 2018; Wong et al., 2018; Mirman et al., 2018; Gowal et al., 2018; Raghunathan et al., 2018a; Zhang et al., 2020a). Among these methods, Interval Bound Propagation (IBP) (Mirman et al., 2018; Gowal et al., 2018) is a simple but most effective method for certified robust training. It simply propagates the interval of each neuron through the network in the forward pass to obtain lower and upper bounds of output neurons, and when conducting certified defense, back-propagates the certified robust loss to network weights to conduct SGD-based training. Most of the latest state-of-the-art certified defense methods are still at least partly based on IBP training (Zhang et al., 2020a; Shi et al., 2021; Zhang et al., 2021).

Despite being one of the most successful certified defense methods, the convergence properties of IBP training remained unknown. For natural neural network training (training without considering adversarial perturbation, aka standard training), it has been shown that gradient descent for overparameterized networks can provably converge to a global minimizer with random initialization (Li & Liang, 2018; Du et al., 2018b;a; Jacot et al., 2018; Allen-Zhu et al., 2019; Zou et al., 2018). Compared to natural neural network training, IBP-based robust training has a very different training scheme, and thus requires a different convergence analysis. First, in the robust training problem, the data contain adversarial perturbations, and the training objective is to minimize a robust loss rather than a natural loss. Second, IBP training essentially optimizes an augmented network which contains IBP bound computation rather than standard neural networks, as illustrated in Zhang et al. (2020a). Third, in IBP training, the activation state of each neuron depends on the certified bounds rather than the value in natural neural network computation, and this introduces additional perturbation-related terms in the convergence analysis for IBP.

In this paper, we conduct a theoretical analysis to study the convergence of IBP training. Following recent convergence analysis on SGD for natural training, we consider IBP robust training with gradient descent for a two-layer overparameterized neural network on a classification task. Our contributions can be summarized below:

- We provide the first convergence analysis for IBP-based certified robust training. On a two-layer overparameterized ReLU network, under certain conditions on perturbation radius and network width, we prove that gradient flow with IBP robust loss has a linear convergence rate, and is guaranteed to converge to zero training error with high probability.
- Our result implies that IBP converges to a point where certified robust accuracy by IBP bounds accurately reflects the true robustness of the model.
- We show additional perturbation-related conditions required to guarantee the convergence of IBP training so far, and propose additional challenges in the convergence analysis for IBP training compared to standard training.

2 Related Work

2.1 Certified Robust Training

The goal of certified robust training is to maximize the robust accuracy of the model under the evaluation by robustness verifiers. Some works add heuristic regularizations during adversarial training to improve certified robustness (Xiao et al., 2019; Balunovic & Vechev, 2020). Some others optimize a certified robust loss which is a certified upper bound of the loss function under any possible adversarial perturbation within in the range, computed from bounds by robustness verifiers. Among them, Wong & Kolter (2018); Mirman et al. (2018); Dvijotham et al. (2018); Wong et al. (2018); Wang et al. (2018a) used verification based on linear relaxation and Raghunathan et al. (2018b) used semidefinite relaxation. However, Interval Bound Propagation (IBP) (Mirman et al., 2018; Gowal et al., 2018) which computes and propagates an interval lower and upper bound for each neuron, has been shown as efficient and effective and can even outperform methods using more complicated relaxation as partly discussed by Lee et al. (2021); Jovanović et al. (2021). Zhang et al. (2020a) further combined IBP with linear relaxation bounds. In more recent works, Lyu et al. (2021) designed a parameterized activation; Zhang et al. (2021) designed a 1-Lipschitz layer with ℓ_∞-norm computation before layers using IBP; Shi et al. (2021) improved IBP training with shortened training schedules. Yet, the state-of-the-art methods still contain IBP as an important part, and thus we focus on analyzing the convergence of IBP training in this paper.

The aforementioned methods for certified robustness target at robustness with deterministic certification. There are also some other works on probabilistic certification such as randomized smoothing (Cohen et al., 2019; Li et al., 2019; Salman et al., 2019) which is out of our scope.

On the theoretical analysis for IBP training, Baader et al. (2019) analyzed the universal approximation of IBP verification bounds, and Wang et al. (2020) extended the analysis to other activation functions beyond ReLU. However, to the best of our knowledge, there is still no existing work analyzing the convergence of IBP training.

2.2 CONVERGENCE OF STANDARD NEURAL NETWORK TRAINING

There have been many works analyzing the convergence of standard neural network training. For two-layer ReLU neural networks with quadratic loss, Du et al. (2018b) proved that randomly initialized gradient descent can converge to a globally optima with a large enough network width m which is polynomial in the number of training examples n. Ji & Telgarsky (2019) pushed the requirement of network width m to a polylogarithmic function. For deep neural networks, Allen-Zhu et al. (2019) proved that for randomly initialized ReLU deep neural networks, gradient descent has a linear convergence rate for various loss functions with width m polynomial in network depth and number of training examples n. And Chen et al. (2019) proved that polylogarithmic width is also sufficient for deep neural networks to converge to the optimal point. However, these works only focus on standard training, and cannot be directly adapted to the robust training setting.

2.3 CONVERGENCE OF EMPIRICAL ADVERSARIAL TRAINING

Robust training including both certified training and empirical adversarial training is essentially a min-max optimization. For a dataset $\mathcal{X}$, the objective for learning a model f_θ parameterized by θ can be written as[1]:

$$\arg\min_{\theta} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{X}} \max_{\Delta\in\mathbb{S}} \ell(f_\theta(\mathbf{x}+\Delta), y), \tag{1}$$

where $(\mathbf{x}, y)$ is clean input sampled from $\mathcal{X}$, $\ell(\cdot, y)$ is the loss function for ground truth y, $\mathbb{S}$ is the space of all possible perturbations. For empirical adversarial training, the inner maxmization is achieved by running adversarial attacks to find adversarial examples. Some works have analyzed the convergence of adversarial training. Wang et al. (2019) analyzed the convergence and robustness of adversarial training through the lens of first-order stationary condition for solving the inner constrained maximization problem. Gao et al. (2019); Zhang et al. (2020b) showed that overparameterized networks with projected gradient descent can converge to a point where the surrogate robust loss by adversarial attack is close to the true robust loss (i.e., the inner maximization by adversarial attack is mostly optimal) and the robust loss is close to 0. Zou et al. (2021) showed that adversarial training provably learns robust halfspaces in the presence of noise.

However, there is a significant difference between empirical adversarial training and IBP training. In adversarial training, there is a concrete perturbation Δ generated for the inner maximization, and there is a concrete adversarial input $\mathbf{x}+\Delta$ to the neural network. However, in IBP-based training, the inner maximization is computed from certified bounds, where for each layer, the interval certified bounds of each neuron in the layer are computed independently. Thereby, the certified bounds of the network output generally no longer correspond to any concrete perturbation Δ. Due to this significant difference, prior theoretical analysis on empirical adversarial training, which are based on the assumption that there should be a concrete Δ for inner maxmization, is not directly applicable to IBP training.

3 PRELIMINARIES

3.1 NEURAL NETWORKS

We consider a similar network architecture as used in Du et al. (2018b) – a two-layer neural network with one hidden layer. Unlike Du et al. (2018b) which considered a regression task with a square loss, here we consider a binary classification task where IBP robust training is usually used. On a training dataset $\{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n)\}$ with input $\mathbf{x}_i (\mathbf{x}_i \in \mathbb{R}^d)$ and label $y_i (y_i \in \{\pm 1\})$ where d is the dimension of the input vector. The output of the network in standard training for the i-th example is:

$$f(\mathbf{W}, \mathbf{a}, \mathbf{x}_i) = \frac{1}{\sqrt{m}} \sum_{r=1}^{m} a_r \sigma(\mathbf{w}_r^\top \mathbf{x}_i), \tag{2}$$

where m is the width of hidden layer (the first layer) in the network, $\mathbf{W} \in \mathbb{R}^{m\times d}$ is the weight matrix of the hidden layer, $\mathbf{w}_r$ is the r-th row of $\mathbf{W}$, $\mathbf{a} \in \mathbb{R}^m$ is the weight vector of the second

[1]Here we use notations to denote the general robust training problem, but in our later theoretical analysis, we will have different notations for a simplified problem setting.

layer (output layer) with elements $a_1, \cdots, a_m$, and $\sigma(\cdot)$ is the activation function and we assume it is ReLU in this paper. For initialization, we set $a_r \sim \text{unif}[\{1, -1\}]$ and $\mathbf{w}_r \sim \mathbf{N}(0, \mathbf{I})$. We assume the second layer in the network is fixed after initialization, and we only train the first layer, following Du et al. (2018b). Since we consider a binary classification task, we use a logistic loss $l(u) = \log(1 + \exp(-u))$. Let $u_i(\mathbf{W}, \mathbf{a}, \mathbf{x}_i) = y_i f(\mathbf{W}, \mathbf{a}, \mathbf{x}_i)$, the standard training loss can be written as

$$L = \sum_{i=1}^{n} l(u_i) = \sum_{i=1}^{n} \log(1 + \exp(-u_i)). \tag{3}$$

3.2 Certified Robust Training

In the robust training setting, we consider that the actual input to the model may be a perturbed point $\mathbf{x}_i + \Delta_i$ rather than the original $\mathbf{x}_i$. For a widely adopted setting in the area, we consider ℓ_∞ perturbations, where the perturbation is bounded by an ℓ_∞ ball with radius $\epsilon (0 \leq \epsilon \leq 1)$, i.e., $\|\Delta_i\|_\infty \leq \epsilon$. For the convenience of subsequent analysis and without loss of generality, we set the following assumption on each $\mathbf{x}_i$:

Assumption 1. *For $1 \leq i \leq n$, we assume that:*

$$\mathbf{x}_i \in [\epsilon, 1]^d, \quad \|\mathbf{x}_i\|_2 \geq \xi. \tag{4}$$

Also, we need to assume data points are seperable. In Du et al. (2018b), they assume there are no parallel examples, and in our case we need to assume this separation holds under any ϵ perturbations to input data points. This can be mathematically formulated as follows.

Assumption 2. *For the dataset $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ and perturbation radius ϵ, we assume that*

$$\forall 1 \leq i \leq n, \forall \mathbf{x}_i' \in B_\infty(y_i \mathbf{x}_i, \epsilon), \forall \mathbf{x}_j' \in B_\infty(y_j \mathbf{x}_j, \epsilon), \quad \mathbf{x}_i' \nparallel \mathbf{x}_j', \tag{5}$$

where $B_\infty(\mathbf{x}_i, \epsilon)$ stands for the ℓ_∞ ball with radius ϵ centered at $\mathbf{x}_i$.

IBP training computes and optimizes a robust loss, which is an upper bound of the standard loss for any possible perturbation Δ_i:

$$\overline{L} = \sum_{i=1}^{n} \max_{\Delta_i} \log\big(1 + \exp(-y_i f(\mathbf{W}, \mathbf{a}, \mathbf{x}_i + \Delta_i))\big), \quad \text{s.t. } \|\Delta_i\|_\infty \leq \epsilon \ (i = 1, 2, \cdots, n). \tag{6}$$

To compute $\overline{L}$, since that $\log(\cdot)$ and $\exp(\cdot)$ are both monotonic, IBP first compute the lower bound of $u_i(\mathbf{W}, \mathbf{a}, \mathbf{x}_i + \Delta_i)$ for any possible Δ_i, namely $\underline{u}_i$, and then the robust loss can be computed as

$$\overline{L} = \sum_{i=1}^{n} \log(1 + \exp(-\underline{u}_i)), \quad \text{where } \underline{u}_i \leq \min_{\Delta_i} u_i(\mathbf{W}, \mathbf{a}, \mathbf{x}_i + \Delta_i) \ (i = 1, 2, \cdots, n). \tag{7}$$

IBP computes and propagates an interval lower and upper bound for each neuron in the network, and then $\underline{u}_i$ is equivalent to the lower bound of the final output neuron. Initially, the interval bound of the input is $[\mathbf{x}_i - \epsilon \cdot \mathbf{1}, \mathbf{x} + \epsilon \cdot \mathbf{1}]$. For the r-th neuron in the first layer, given constraints of Δ_i as mentioned above, we have its interval bounds as

$$\sigma\big(\mathbf{w}_r^\top \mathbf{x}_i - \epsilon |\mathbf{w}_r|^\top \cdot \mathbf{1}\big) \leq \sigma\big(\mathbf{w}_r^\top (\mathbf{x}_i + \Delta_i)\big) \leq \sigma\big(\mathbf{w}_r^\top \mathbf{x}_i + \epsilon |\mathbf{w}_r|^\top \cdot \mathbf{1}\big), \tag{8}$$

where $|\cdot|$ for vectors are element-wise. Then these interval bounds are propagated to the second layer. We only care about the lower bound of u_i, which can be computed from the interval bounds of the first layer in Eq. (8) by considering the sign of multiplier $y_i a_r$:

$$u_i = y_i \frac{1}{\sqrt{m}} \sum_{r=1}^{m} a_r \sigma(\mathbf{w}_r^\top (\mathbf{x}_i + \Delta_i)) \tag{9}$$

$$\geq \frac{1}{\sqrt{m}} \sum_{r=1}^{m} \mathbb{1}(y_i a_r = 1) \sigma\big(\mathbf{w}_r^\top \mathbf{x}_i - \epsilon \|\mathbf{w}_r\|_1\big) + \mathbb{1}(y_i a_r = -1) \sigma\big(\mathbf{w}_r^\top \mathbf{x}_i + \epsilon \|\mathbf{w}_r\|_1\big) \coloneqq \underline{u}_i. \tag{10}$$

Thereby, with $\underline{u}_i$ computed above, the IBP robust loss can be obtained as Eq. (7). And we define $\underline{\mathbf{u}} = (\underline{u}_1, \underline{u}_2, \cdots, \underline{u}_n)$ for later analysis.

For certified robust training, we can define a **certified robust accuracy** in IBP training, which is the percentage of examples that IBP bounds can successfully certify that the prediction is correct for any possible perturbation. Here example i is considered as classified correctly and robustly in the evaluation if and only if $\underline{u}_i \geq 0$ where $\underline{u}_i$ is computed from IBP. We can also define a **true robust accuracy** – where the robustness of prediction requires $\tilde{u}_i \geq 0$ and $\tilde{u}_i$ is the *exact* solution of the minimization in Eq. (7) rather than relaxed IBP bounds. The certified robust accuracy by IBP is a provable lower bound of the true robust accuracy.

3.3 Gradient Flow

To analyze the convergence of IBP training, we adopt continuous time analysis where we consider gradient descent with infinitesimal step size, namely gradient flow as used in prior works on standard neural network training (Arora et al., 2018; Du et al., 2018a;b). In the gradient flow for IBP training, the weights $\mathbf{w}_r$ are optimized as

$$\frac{d\mathbf{w}_r(t)}{dt} = -\frac{-\partial \overline{L}(\mathbf{W}(t), \mathbf{a})}{\partial \mathbf{w}_r(t)}, \tag{11}$$

where $\mathbf{W}(t)$ is weight matrix at time t with rows $\mathbf{w}_t(t)$, and $\overline{L}(\mathbf{W}(t), \mathbf{a})$ is the IBP robust loss defined as Eq. (7) using weights at time t.

3.4 Gram Matrix

Under the gradient flow setting as Eq. (11), we analyze the dynamics of $\underline{u}_i$ during IBP training, where we use $\underline{u}_i(t)$ to denote the value at time t (detailed derivation in Appendix B.1):

$$\frac{d}{dt}\underline{u}_i(t) = \sum_{r=1}^{m} \left\langle \frac{\partial \underline{u}_i(t)}{\partial \mathbf{w}_r(t)}, \frac{d\mathbf{w}_r(t)}{dt} \right\rangle = \sum_{j=1}^{n} -l'(\underline{u}_j)\mathbf{H}_{ij}(t), \tag{12}$$

where $\mathbf{H}(t)$ is defined as $\mathbf{H}_{ij}(t) = \sum_{r=1}^{m} \left\langle \frac{\partial \underline{u}_i(t)}{\partial \mathbf{w}_r(t)}, \frac{\partial \underline{u}_j(t)}{\partial \mathbf{w}_r(t)} \right\rangle$.

With the definition of $\mathbf{H}$, we can easily describe the dynamic of $\underline{u}_i$ using $\mathbf{H}$ under the gradient flow.

From Eq. (10), derivative $\frac{\partial \underline{u}_i(t)}{\partial \mathbf{w}_r(t)}$ can be computed as follows:

$$\frac{\partial \underline{u}_i(t)}{\partial \mathbf{w}_r(t)} = \frac{1}{\sqrt{m}} y_i a_r \Big(A_{ri}^{+}(t)(\mathbf{x}_i - \epsilon\, \mathrm{sign}(\mathbf{w}_r(t))) + A_{ri}^{-}(t)(\mathbf{x}_i + \epsilon\, \mathrm{sign}(\mathbf{w}_r(t))) \Big), \tag{13}$$

where $\mathrm{sign}(\mathbf{w}_r(t))$ is element-wise for $\mathbf{w}_r(t)$, and we define *indicators*

$$A_{ri}^{+}(t) = \mathbb{1}(y_i a_r = 1, \mathbf{w}_r(t)^\top \mathbf{x}_i - \epsilon \|\mathbf{w}_r(t)\|_1 > 0), \tag{14}$$

$$A_{ri}^{-}(t) = \mathbb{1}(y_i a_r = -1, \mathbf{w}_r(t)^\top \mathbf{x}_i + \epsilon \|\mathbf{w}_r(t)\|_1 > 0), \tag{15}$$

and terms $(\mathbf{w}_r(t)^\top \mathbf{x}_i - \epsilon \|\mathbf{w}_r(t)\|_1 > 0)$ and $(\mathbf{w}_r(t)\mathbf{x}_i + \epsilon \|\mathbf{w}_r(t)\|_1 > 0)$ in the indicators stand for the active status of the activation functions. Then elements in $\mathbf{H}$ can be written as:

$$\begin{aligned} \mathbf{H}_{ij}(t) = \frac{1}{m} y_i y_j \sum_{r=1}^{m} a_r^2 \Big(A_{ri}^{+}(t)(\mathbf{x}_i - \epsilon\, \mathrm{sign}(\mathbf{w}_r(t))) + A_{ri}^{-}(t)(\mathbf{x}_i + \epsilon\, \mathrm{sign}(\mathbf{w}_r(t))) \Big)^\top \\ \Big(A_{rj}^{+}(t)(\mathbf{x}_j - \epsilon\, \mathrm{sign}(\mathbf{w}_r(t))) + A_{rj}^{-}(t)(\mathbf{x}_j + \epsilon\, \mathrm{sign}(\mathbf{w}_r(t))) \Big) \end{aligned} \tag{16}$$

$$= \frac{1}{m} y_i y_j \Big(\mathbf{x}_i^\top \mathbf{x}_j \sum_{r=1}^{m} \alpha_{rij}(t) - \epsilon \big(\sum_{r=1}^{m} (\beta_{rij}(t)\mathbf{x}_i + \beta_{rji}(t)\mathbf{x}_j)^\top \mathrm{sign}(\mathbf{w}_r(t)) \big) + \epsilon^2 d \sum_{r=1}^{m} \gamma_{rij}(t) \Big), \tag{17}$$

$$\text{where} \quad \alpha_{rij}(t) = (A_{ri}^{+}(t) + A_{ri}^{-}(t))(A_{rj}^{+}(t) + A_{rj}^{-}(t)), \tag{18}$$

$$\beta_{rij}(t) = (A_{ri}^{+}(t) + A_{ri}^{-}(t))(A_{rj}^{+}(t) - A_{rj}^{-}(t)), \tag{19}$$

$$\gamma_{rij}(t) = (A_{ri}^{+}(t) - A_{ri}^{-}(t))(A_{rj}^{+}(t) - A_{rj}^{-}(t)). \tag{20}$$

Note that in Eq. (16) we have $a_r^2 = 1$ since $a_r \in \{\pm 1\}$ as defined in section 3.1, and thus a_r can be ignored from the results.

Further, we define $\mathbf{H}^{\infty}$ (known as the Gram matrix in Du et al. (2018b)), which is the elementwise expectation of $\mathbf{H}(0)$, to describe the characteristic of $\mathbf{H}(0)$ on the random initialization basis:

Definition 1. *We define*

$$\mathbf{H}_{ij}^{\infty} = \mathbb{E}_{\forall 1 \leq r \leq m, \mathbf{w}_r \sim \mathbf{N}(0,\mathbf{I}), a_r \sim \mathrm{unif}[\{-1,1\}]} \mathbf{H}_{ij}(0), \tag{21}$$

where $\mathbf{H}_{ij}(0)$ depends on the initialization of weights $\mathbf{w}_r$ and a_r. We also define $\lambda_0 = \lambda_{\min}(\mathbf{H}^{\infty})$ as the least eigenvalue of $\mathbf{H}^{\infty}$.

The Gram matrix will be used for proving that $\mathbf{H}(0)$ is positive definite with high probability, by showing that $\mathbf{H}^{\infty}$ is positive definite and bounding the difference between $\mathbf{H}(0)$ and $\mathbf{H}^{\infty}$.

4 Convergence Analysis for IBP Training

We present the following main theorem which shows the convergence of IBP training under certain conditions on perturbation radius and network width:

Theorem 1 (Convergence of IBP Training)**.** *Suppose Assumption 1 and 2 hold, if the ℓ_∞ perturbation radius $\epsilon \leq O(\min(\frac{\delta^2 \lambda_0^2}{d^{2.5} n^3}, \frac{\sqrt{2d}R}{\log(\sqrt{\frac{2\pi d}{R}}\xi)}))$, where $R = \frac{c\delta\lambda_0}{d^{1.5}n^2}$, $c = \frac{\sqrt{2\pi}\xi}{384}$, by gradient flow defined in Section 3.3, with probability at least $1-\delta$, the IBP certified robust accuracy of the two-layer ReLU neural network defined in Section 2 can converge to zero with a large enough network width $m \geq \Omega\left(\left(\frac{d^{1.5}n^4\delta\lambda_0}{\epsilon d^{2.5}n^4 - \delta^2\lambda_0^2}\right)^2\right)$.*

In the following part of this section, we provide the proof sketch for the main theorem.

4.1 Stability of the Gram Matrix during IBP Training

We first analyze the stability of the Gram matrix during training. We want to show that when the change of $\mathbf{w}_r$ during training is restricted to $\|\mathbf{w}_r(t) - \mathbf{w}_r(0)\|_2 \leq R$ for some R, we can guarantee that the minimum eigenvalue of $\mathbf{H}(t)$ remains positive during the training with high probability, which will be used in the later section to reach the conclusion on the convergence. Here we defer the derivation for the constraint on R to the later part of this section.

With the aforementioned constraint on $\mathbf{w}_r(t)$, we first show that during the IBP training most of $\alpha_{rij}(t), \beta_{rij}(t), \gamma_{rij}(t)$ terms in Eq. (17) remain the same values as their initialization ($t = 0$). It is because for for each of $\alpha_{rij}(t), \beta_{rij}(t), \gamma_{rij}(t)$, the probability that its value changes during training can be upper bounded and made sufficiently small, as the lemma below shows:

Lemma 1. *When $\|\mathbf{w}_r(t) - \mathbf{w}_r(0)\|_2 \leq R$ holds for some R, and $\epsilon \leq \frac{\sqrt{2d}R}{\log(\sqrt{\frac{2\pi d}{R}}\xi)}$ holds, then $\forall 1 \leq r \leq m,\ 1 \leq i, j \leq n$, we have*

$$\Pr(\alpha_{rij}(t) \neq \alpha_{rij}(0)), \Pr(\beta_{rij}(t) \neq \beta_{rij}(0)), \Pr(\gamma_{rij}(t) \neq \gamma_{rij}(0)) \leq \frac{12}{\sqrt{2\pi}\xi}(1+\epsilon)\sqrt{d}R := \tilde{R}. \tag{22}$$

The full proof for the lemma is provided in Appendix A.1. Given the definition of $\alpha_{rij}(t), \beta_{rij}(t), \gamma_{rij}(t)$, probabilities in Lemma 1 can be easily bounded as long as the probability that each of indicator $A_{ri}^{+}(t), A_{ri}^{-}(t), A_{rj}^{+}(t), A_{rj}^{-}(t)$ changes can be upper bounded respectively. And when the change of $\mathbf{w}_r(t)$ is bounded, the indicators can change during the training only if at initialization $|\mathbf{w}_r(0)^\top \mathbf{x}_i \pm \epsilon\|\mathbf{w}_r(0)\|_1|$ is sufficiently small with $|\mathbf{w}_r(0)^\top \mathbf{x}_i \pm \epsilon\|\mathbf{w}_r(0)\|_1| \leq$

$(1+\epsilon)\sqrt{d}R$ as we show in the proof, whose probability can be upper bounded (the notation $\pm$ here means the analysis is consistent for both $+$ and $-$ cases).

To upper bound $\Pr\left(|\mathbf{w}_r(0)^\top \mathbf{x}_i \pm \epsilon\|\mathbf{w}_r(0)\|_1| \leq (1+\epsilon)\sqrt{d}R\right)$ note that the analysis here needs to be different compared to the analysis for natural training in Du et al. (2018b), because we have additional perturbation-related terms $\epsilon\|\mathbf{w}_r(0)\|_1$ which come from IBP computation. Du et al. (2018b) simply used the anti-concentration of standard Gaussian distribution to bound $\Pr(|\mathbf{w}_r(0)^\top \mathbf{x}_i) \leq R$ (they did not have $\sqrt{d}$ on the right-hand-side due to a different assumption on $\|\mathbf{x}\|$). Here when $\epsilon\|\mathbf{w}_r(0)\|_1$ presents for the IBP case, we split the original probability into two parts, $\Pr(|\mathbf{w}_r(0)^\top \mathbf{x}_i|) \leq 2(1+\epsilon)\sqrt{d}R$ and $\Pr(\epsilon\|\mathbf{w}_r(0)\|_1 \leq (1+\epsilon)\sqrt{d}R)$, and then combine the results to upper bound the original probability. The first probability can still be bounded by anti-concentration inequality, while the second probability can be bounded by the tail bound of standard Gaussian. And thereby we can prove the lemma, with details in the appendix.

We can then bound $\|\mathbf{H}(t) - \mathbf{H}(0)\|_2$ as shown in the following lemma:

Lemma 2. *When $\|\mathbf{w}_r(t) - \mathbf{w}_r(0)\|_2 \leq R$ for some constant R, with probability at least $1-\delta$, it holds that*

$$\|\mathbf{H}(t) - \mathbf{H}(0)\|_2 \leq \frac{12(1+\epsilon)(1+2\epsilon+\epsilon^2)d^{1.5}n^2}{\sqrt{2\pi}\xi\delta}R. \tag{23}$$

This lemma can be proved by firstly upper bound $\mathbb{E}[|\mathbf{H}_{ij}(t) - \mathbf{H}_{ij}(0)|]$ using the probability that each of $\alpha_{rij}(t), \beta_{rij}(t), \gamma_{rij}(t)$ changes respectively as shown in Lemma 1, and then using Markov's inequality, we can upper bound $\|\mathbf{H}(t) - \mathbf{H}(0)\|_2$ with high probability. We provide the detailed proof in Appendix A.2. And by triangle inequality, we can also lower bound $\lambda_{\min}(\mathbf{H}(t))$:

Collary 1. *When $\|\mathbf{w}_r(t) - \mathbf{w}_r(0)\|_2 \leq R$ for some constant R, with probability at least $1-\delta$, it holds that*

$$\lambda_{\min}(\mathbf{H}(t)) \geq \lambda_{\min}(\mathbf{H}(0)) - \frac{12(1+\epsilon)(1+2\epsilon+\epsilon^2)d^{1.5}n^2}{\sqrt{2\pi}\xi\delta}R, \tag{24}$$

where $\lambda_{min}(\cdot)$ stands for the minimum eigenvalue.

We also need to lower bound $\lambda_{\min}(\mathbf{H}(0))$ to lower bound $\lambda_{\min}(\mathbf{H}(t))$. Given Assumption 2, we show that the minimum eigenvalue of $\mathbf{H}^\infty$ is positive.

Lemma 3. *When the dataset satisfies Assumption 2, $\lambda_0 = \lambda_{\min}(\mathbf{H}^\infty) > 0$ holds true.*

The lemma can be similar proved as Theorem 3.1 in Du et al. (2018b), as discussed in Appendix A.3. Then we can lower bound $\lambda_{\min}(\mathbf{H}(0))$ by Lemma 3.1 from Du et al. (2018b):

Lemma 4 (Lemma 3.1 from Du et al. (2018b))**.** *If $\lambda_0 > 0$, with $m = \Omega(\frac{n^2}{\lambda_0^2}\log(\frac{n}{\delta}))$, and with probability at least $1-\delta$, it holds true that $\lambda_{\min}(\mathbf{H}(0)) \geq \frac{3}{4}\lambda_0$.*

Although we have different values in the Gram matrix $\mathbf{H}(0)$ for IBP training, compared to the Gram matrix in Du et al. (2018b) for standard training, we can still adopt their original lemma because their proof by Hoeffding inequality is general for values in $\mathbf{H}(0)$ given the definition on λ_0.

Then we can guarantee that $\lambda_{\min}(\mathbf{H}(t))$ remains positive by solving an inequality and setting a proper R such that

$$\lambda_{\min}(\mathbf{H}(t)) \geq \lambda_{\min}(\mathbf{H}(0)) - \frac{12(1+\epsilon)(1+2\epsilon+\epsilon^2)d^{1.5}n^2}{\sqrt{2\pi}\xi\delta}R \geq \frac{\lambda_0}{2}, \tag{25}$$

as shown in the following lemma (proved in Appendix A.4):

Lemma 5. *When $\|\mathbf{w}_r(t) - \mathbf{w}_r(0)\|_2 \leq R$, where $R = \frac{c\delta\lambda_0}{d^{1.5}n^2}$ with $c = \frac{\sqrt{2\pi}\xi}{384}$, it holds that $\lambda_{\min}(\mathbf{H}(t)) \geq \frac{\lambda_0}{2}$, with probability at least $1-\delta$.*

Therefore, we have shown that with over-parameterization (required by Lemma 4), when $\mathbf{w}_r$ is relatively stable during training, i.e., the maximum change on $\mathbf{w}_r(t)$ is upper bounded during training (charactized by the ℓ_2 norm of weight change restricted by R as chosen in Lemma 5), $\mathbf{H}(t)$ is also relatively stable and remains positive definite with high probability (Lemma 1).

4.2 CONVERGENCE OF THE IBP ROBUST LOSS

Based on the result that $\mathbf{H}(t)$ remains positive definite under certain conditions, we further derive the upper bound of the IBP loss during training.

In the following lemma, we show that when $\mathbf{H}(t)$ remains positive definite, the IBP loss $\overline{L}(t)$ descends in a linear convergence rate, and meanwhile we have an upper bound on the change of $\mathbf{w}_r(t)$ w.r.t. time t.

Lemma 6. *Suppose for* $0 \leq s \leq t$, $\lambda_{\min}(\mathbf{H}(t)) \geq \frac{\lambda_0}{2}$, *we have*

$$\overline{L}(t) \leq \exp\big(2\overline{L}(\mathbf{u}(0))\big)\overline{L}(\mathbf{u}(0))\exp\bigg(-\frac{\lambda_0 t}{2}\bigg), \|\mathbf{w}_r(t)-\mathbf{w}_r(0)\|_2 \leq \frac{nt}{\sqrt{m}}. \tag{26}$$

This lemma is proved in Appendix A.5, which follows the proof of Lemma 5.4 in Zou et al. (2018).

To guarantee that $\lambda_{\min}(\mathbf{H}(s)) \geq \frac{\lambda_0}{2}$ for $0 \leq s \leq t$, by Lemma 5, we require

$$\|\mathbf{w}_r(t)-\mathbf{w}_r(0)\|_2 \leq \frac{nt}{\sqrt{m}} \leq R \leq \frac{c\delta\lambda_0}{d^{1.5}n^2} \Rightarrow t \leq \frac{c\delta\lambda_0\sqrt{m}}{d^{1.5}n^3}. \tag{27}$$

Meanwhile, for each example i, it is correctly classified if and only if $\underline{u}_i \geq 0$ and thereby $l(\underline{u}_i) \leq \kappa$, where $\kappa = \log(1+\exp(0))$. Therefore, to reach zero training error on the whole training set at time t, we can require $\overline{L}(t) \leq \kappa$, which implies that $\forall 1 \leq i \leq n, l(\underline{u}_i) \leq \kappa$. Then we want to make

$$\overline{L}(t) \leq \exp\big(2\overline{L}(\mathbf{u}(0))\big)\overline{L}(\mathbf{u}(0))\exp\bigg(-\frac{\lambda_0 t}{2}\bigg) \leq \kappa \quad \Rightarrow t \geq \frac{4}{\lambda_0}\bigg(\log\bigg(\frac{\overline{L}(0)}{\kappa}\bigg)+\overline{L}(0)\bigg). \tag{28}$$

Combining Eq. (27) and Eq. (28), we require

$$\frac{4}{\lambda_0}\bigg(\log\bigg(\frac{\overline{L}(0)}{\kappa}\bigg)+\overline{L}(0)\bigg) \leq t \leq \frac{c\delta\lambda_0\sqrt{m}}{d^{1.5}n^3}. \tag{29}$$

Since $\overline{L}(0)$ has the randomness from the randomly initialized weight $\mathbf{W}$, we need to upper bound the value of $\overline{L}(0)$ with probability at least $1-\delta$.

Lemma 7. *In natural training, with probability at least* $1-\delta$, $L(0) = O(\frac{n}{\delta})$. *In IBP training, with probability at least* $1-\delta$, $\overline{L}(0) = O(\frac{n\sqrt{m}d\epsilon}{\delta} + \frac{n}{\delta})$.

This lemma is proved in Appendix A.6 by concentration.

4.3 PROVING THE MAIN THEOREM

Finally, we are ready to prove the main theorem. Since $\overline{L}(0) = O(\frac{n\sqrt{m}d\epsilon}{\delta} + \frac{n}{\delta})$, with large n and m, there exist constants c_1, c_2, c_3 such that

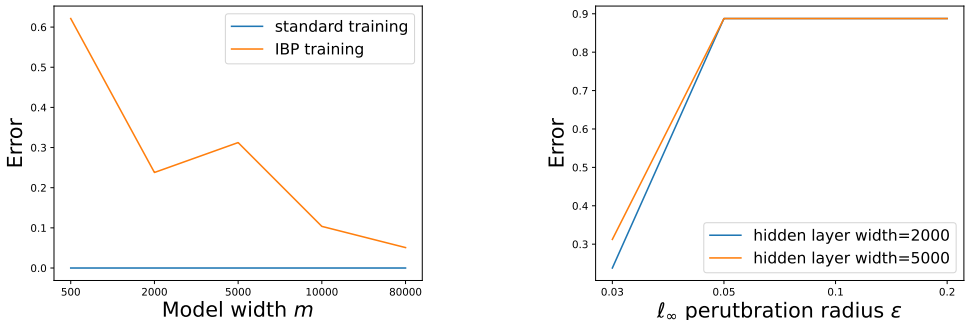
$$\overline{L}(0) \leq \frac{c_1 n\sqrt{m}d\epsilon}{\delta\lambda_0} + \frac{c_2 n}{\delta\lambda_0} + c_3. \tag{30}$$

And requirement in Eq. (29) on the upper bound of t can be relaxed to $\frac{c\delta\lambda_0\sqrt{m}}{d^{1.5}n^3} \geq \frac{8}{\lambda_0}(\overline{L}(0)) + c_4$, where c_4 is a constant. Plug in Eq. (30), we can make

$$\frac{c\delta\lambda_0\sqrt{m}}{d^{1.5}n^3} \geq \frac{c_1 n\sqrt{m}d\epsilon}{\delta\lambda_0} + \frac{c_2 n}{\delta\lambda_0} + c_3. \tag{31}$$

If Eq. (31) holds, Eq. (29) also holds, and thereby IBP training is guaranteed to converge to an optimal classification loss where the certified robust *error* is zero. We want to make Eq. (31) hold by sufficiently enlarging m. Also plug in the constraint on ϵ in Lemma 1, we require

$$\epsilon \leq \frac{\sqrt{2d}R}{\log(\sqrt{\frac{2\pi d}{R}}\xi)}, \frac{c\delta\lambda_0}{d^{1.5}n^3} \geq \frac{c_1 nd\epsilon}{\delta\lambda_0} \quad \Rightarrow \epsilon \leq \min\bigg(\frac{c\delta^2\lambda_0^2}{c_1 d^{2.5}n^3}, \frac{\sqrt{2d}R}{\log(\sqrt{\frac{2\pi d}{R}}\xi)}\bigg). \tag{32}$$



(a) Final training error of standard training and IBP training with different hidden layer width m.

(b) Final training error of IBP training with different perturbation radius ϵ.

Figure 1: Experimental results.

And our requirement on width m is

$$m \geq \left(\frac{\frac{c_2 n}{\delta \lambda_0} + c_3}{\frac{c\delta\lambda_0}{d^{1.5}n^3} - \frac{c_1 n d \epsilon}{\delta \lambda_0}} \right)^2 = \Omega\left(\left(\frac{d^{1.5} n^4 \delta \lambda_0}{\epsilon d^{2.5} n^4 - \delta^2 \lambda_0^2} \right)^2 \right). \tag{33}$$

This completes the proof of the main theorem.

In our analysis, we focus on the IBP training with $\epsilon > 0$. But IBP training with $\epsilon = 0$ can also be viewed as standard training. By setting $\epsilon = 0$, if $m = O(\frac{n^8 d^3}{\lambda_0^4 \delta^4})$, our result implies that standard training with logistic loss also converges to zero training error with probability at least $1 - \delta$. And as ϵ gets larger, the required m for convergence also becomes larger.

5 Experiments

We conduct experiments to compare the convergence of networks with different width m for natural training and IBP training respectively. We use MNIST (LeCun et al., 2010) dataset and a two-layer fully-connected ReLU network with a variable hidden layer width. In each experiment, we train the model for 70 epochs. For IBP training, we keep ϵ fixed throughout the whole training process. In Figure 1 we show that: **Figure (a)**: The training loss curve of standard training and IBP training with different m. For IBP training, we choose $\epsilon = 0.03$. As we can observe from the figure, IBP training needs larger m for small final training accuracy. And even if we enlarge the m to 80000 which is limited by the GPU memory, IBP certified robust error still cannot decrease to zero. We believe that this is due to a large ϵ. When ϵ is large, the required m for convergence is large. And by our main theorem, when ϵ exceeds the given upper bound, enlarging m can not guarantee convergence with a high probability. **Figure (b)**: When ϵ grows larger, it can be difficult to initiate training although standard training is possible. Therefore in IBP training, many works use ϵ scheduling to gradually increase ϵ as a warm-up. We believe that this is because λ_0 can be very small after perturbation, which leads to difficulty in initial training. And this difficulty in initial training cannot be alleviated by simply enlarging the network width m.

6 Conclusion

In this paper, we present the first theoretical analysis of IBP-based certified robust training, and we have shown that IBP training can converge to zero training error with high probability, under certain conditions on perturbation radius and network width. Meanwhile, since the certified robust accuracy by IBP which converges to 100% is a lower bound of the true robust accuracy (see Section 3.2), the true robust accuracy also converges to 100%, and thus our results also imply that upon convergence, the certification by IBP accurately reflects the true robustness of the network. However, our results has a condition requiring a small upper bound on ϵ. As a future work, it will be interesting to study how to relax this condition, or extend the analysis to deeper networks.

ETHICS STATEMENT

We don't see any potential ethical issues in our work.

REPRODUCIBILITY STATEMENT

For all our theoretical results, the assumptions are stated clearly, and the complete proof can be found either in the paper or the appendix. We only have a small toy experiment by running IBP training with the hyperparameters specified in the paper and the open-source code from Shi et al. (2021), so the results are reproducible.

REFERENCES

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

Sanjeev Arora, Nadav Cohen, and Elad Hazan. On the optimization of deep networks: Implicit acceleration by overparameterization. *CoRR*, abs/1802.06509, 2018. URL http://arxiv.org/abs/1802.06509.

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

Maximilian Baader, Matthew Mirman, and Martin Vechev. Universal approximation with certified networks. In *International Conference on Learning Representations*, 2019.

Mislav Balunovic and Martin Vechev. Adversarial training and provable defenses: Bridging the gap. In *International Conference on Learning Representations*, 2020.

Rudy R Bunel, Ilker Turkaslan, Philip Torr, Pushmeet Kohli, and Pawan K Mudigonda. A unified view of piecewise linear neural network verification. In *Advances in Neural Information Processing Systems*, pp. 4790–4799, 2018.

Nicholas Carlini and David Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pp. 3–14. ACM, 2017.

Zixiang Chen, Yuan Cao, Difan Zou, and Quanquan Gu. How much over-parameterization is sufficient to learn deep relu networks? *CoRR*, abs/1911.12360, 2019. URL http://arxiv.org/abs/1911.12360.

Jeremy M Cohen, Elan Rosenfeld, and J Zico Kolter. Certified adversarial robustness via randomized smoothing. In *ICML*, 2019.

Simon S Du, Jason D Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. *arXiv preprint arXiv:1811.03804*, 2018a.

Simon S Du, Xiyu Zhai, Barnabas Poczos, and Aarti Singh. Gradient descent provably optimizes over-parameterized neural networks. *arXiv preprint arXiv:1810.02054*, 2018b.

Krishnamurthy Dvijotham, Sven Gowal, Robert Stanforth, Relja Arandjelovic, Brendan O'Donoghue, Jonathan Uesato, and Pushmeet Kohli. Training verified learners with learned verifiers. *arXiv preprint arXiv:1805.10265*, 2018.

Ruiqi Gao, Tianle Cai, Haochuan Li, Liwei Wang, Cho-Jui Hsieh, and Jason D. Lee. Convergence of adversarial training in overparameterized networks. *arXiv preprint arXiv:1906.07916*, 2019.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *ICLR*, 2015.

Sven Gowal, Krishnamurthy Dvijotham, Robert Stanforth, Rudy Bunel, Chongli Qin, Jonathan Uesato, Timothy Mann, and Pushmeet Kohli. On the effectiveness of interval bound propagation for training verifiably robust models. *arXiv preprint arXiv:1810.12715*, 2018.

Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens van der Maaten. Countering adversarial images using input transformations. In *ICLR*, 2018.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *NIPS*, 2018.

Ziwei Ji and Matus Telgarsky. Polylogarithmic width suffices for gradient descent to achieve arbitrarily small test error with shallow relu networks. *CoRR*, abs/1909.12292, 2019. URL `http://arxiv.org/abs/1909.12292`.

Nikola Jovanović, Mislav Balunović, Maximilian Baader, and Martin Vechev. Certified defenses: Why tighter relaxations may hurt training? *arXiv preprint arXiv:2102.06700*, 2021.

Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016.

Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2, 2010.

Sungyoon Lee, Woojin Lee, Jinseong Park, and Jaewook Lee. Loss landscape matters: Training certifiably robust models with favorable loss landscape, 2021. URL `https://openreview.net/forum?id=lvXLfNeCQdK`.

Bai Li, Changyou Chen, Wenlin Wang, and Lawrence Carin. Certified adversarial robustness with additive noise. In *Advances in Neural Information Processing Systems*, pp. 9464–9474, 2019.

Yuanzhi Li and Yingyu Liang. Learning overparameterized neural networks via stochastic gradient descent on structured data. *arXiv preprint arXiv:1808.01204*, 2018.

Zhaoyang Lyu, Minghao Guo, Tong Wu, Guodong Xu, Kehuan Zhang, and Dahua Lin. Towards evaluating and training verifiably robust neural networks, 2021.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018.

Matthew Mirman, Timon Gehr, and Martin Vechev. Differentiable abstract interpretation for provably robust neural networks. In *International Conference on Machine Learning*, pp. 3575–3583, 2018.

Aditi Raghunathan, Jacob Steinhardt, and Percy Liang. Certified defenses against adversarial examples. *International Conference on Learning Representations (ICLR), arXiv preprint arXiv:1801.09344*, 2018a.

Aditi Raghunathan, Jacob Steinhardt, and Percy S Liang. Semidefinite relaxations for certifying robustness to adversarial examples. In *Advances in Neural Information Processing Systems*, pp. 10877–10887, 2018b.

Hadi Salman, Jerry Li, Ilya Razenshteyn, Pengchuan Zhang, Huan Zhang, Sebastien Bubeck, and Greg Yang. Provably robust deep learning via adversarially trained smoothed classifiers. In *Advances in Neural Information Processing Systems*, pp. 11289–11300, 2019.

Zhouxing Shi, Yihan Wang, Huan Zhang, Jinfeng Yi, and Cho-Jui Hsieh. Fast certified robust training via better initialization and shorter warmup. *arXiv preprint arXiv:2103.17268*, 2021.

Gagandeep Singh, Timon Gehr, Matthew Mirman, Markus Püschel, and Martin Vechev. Fast and effective robustness certification. In *Advances in Neural Information Processing Systems*, pp. 10825–10836, 2018.

Gagandeep Singh, Timon Gehr, Markus Püschel, and Martin Vechev. An abstract domain for certifying neural networks. *Proceedings of the ACM on Programming Languages*, 3(POPL):41, 2019.

Yang Song, Taesup Kim, Sebastian Nowozin, Stefano Ermon, and Nate Kushman. Pixeldefend: Leveraging generative models to understand and defend against adversarial examples. In *ICLR*, 2018.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *ICLR*, 2013.

Florian Tramer, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. *arXiv preprint arXiv:2002.08347*, 2020.

Shiqi Wang, Yizheng Chen, Ahmed Abdou, and Suman Jana. Mixtrain: Scalable training of formally robust neural networks. *arXiv preprint arXiv:1811.02625*, 2018a.

Shiqi Wang, Kexin Pei, Justin Whitehouse, Junfeng Yang, and Suman Jana. Efficient formal safety analysis of neural networks. In *Advances in Neural Information Processing Systems*, pp. 6367–6377, 2018b.

Shiqi Wang, Kexin Pei, Justin Whitehouse, Junfeng Yang, and Suman Jana. Formal security analysis of neural networks using symbolic intervals. In *27th {USENIX} Security Symposium ({USENIX} Security 18)*, pp. 1599–1614, 2018c.

Yisen Wang, Xingjun Ma, James Bailey, Jinfeng Yi, Bowen Zhou, and Quanquan Gu. On the convergence and robustness of adversarial training. In *ICML*, volume 1, pp. 2, 2019.

Zi Wang, Aws Albarghouthi, Gautam Prakriya, and Somesh Jha. Interval universal approximation for neural networks. *arXiv preprint arXiv:2007.06093*, 2020.

Tsui-Wei Weng, Huan Zhang, Hongge Chen, Zhao Song, Cho-Jui Hsieh, Luca Daniel, Duane Boning, and Inderjit Dhillon. Towards fast computation of certified robustness for relu networks. In *International Conference on Machine Learning*, pp. 5273–5282, 2018.

Eric Wong and Zico Kolter. Provable defenses against adversarial examples via the convex outer adversarial polytope. In *International Conference on Machine Learning*, pp. 5283–5292, 2018.

Eric Wong, Frank Schmidt, Jan Hendrik Metzen, and J Zico Kolter. Scaling provable adversarial defenses. In *NIPS*, 2018.

Chang Xiao, Peilin Zhong, and Changxi Zheng. Enhancing adversarial defense by k-winners-take-all. In *8th International Conference on Learning Representations*, 2020.

Kai Y Xiao, Vincent Tjeng, Nur Muhammad Shafiullah, and Aleksander Madry. Training for faster adversarial robustness verification via inducing relu stability. In *ICLR*, 2019.

Kaidi Xu, Zhouxing Shi, Huan Zhang, Yihan Wang, Kai-Wei Chang, Minlie Huang, Bhavya Kailkhura, Xue Lin, and Cho-Jui Hsieh. Automatic perturbation analysis for scalable certified robustness and beyond. *Advances in Neural Information Processing Systems*, 33, 2020.

Bohang Zhang, Tianle Cai, Zhou Lu, Di He, and Liwei Wang. Towards certifying ℓ_∞ robustness using neural networks with ℓ_∞-dist neurons. *arXiv preprint arXiv:2102.05363*, 2021.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In *International Conference on Machine Learning*, pp. 7472–7482. PMLR, 2019.

Huan Zhang, Tsui-Wei Weng, Pin-Yu Chen, Cho-Jui Hsieh, and Luca Daniel. Efficient neural network robustness certification with general activation functions. In *Advances in neural information processing systems*, pp. 4939–4948, 2018.

Huan Zhang, Hongge Chen, Chaowei Xiao, Bo Li, Duane Boning, and Cho-Jui Hsieh. Towards stable and efficient training of verifiably robust neural networks. In *International Conference on Learning Representations*, 2020a.

Yi Zhang, Orestis Plevrakis, Simon S Du, Xingguo Li, Zhao Song, and Sanjeev Arora. Overparameterized adversarial training: An analysis overcoming the curse of dimensionality. *arXiv preprint arXiv:2002.06668*, 2020b.

Difan Zou, Yuan Cao, Dongruo Zhou, and Quanquan Gu. Stochastic gradient descent optimizes over-parameterized deep relu networks. arxiv e-prints, art. *arXiv preprint arXiv:1811.08888*, 2018.

Difan Zou, Spencer Frei, and Quanquan Gu. Provable robustness of adversarial training for learning halfspaces with noise. *arXiv preprint arXiv:2104.09437*, 2021.

A PROOF OF LEMMAS

A.1 PROOF OF LEMMA 1

Proof. We first consider the change of indicator $\mathbb{1}(\mathbf{w}_r(t)^\top \mathbf{x}_i \pm \epsilon \|\mathbf{w}_r(t)\|_1 > 0)$ $(\forall 1 \leq i \leq n)$ during training compared to the value at $t = 0$ (the notation $\pm$ here means the analysis is consistent for both $+$ and $-$ cases). Under the constraint that $\|\mathbf{w}_r(t) \pm \mathbf{w}_r(0)\|_2 \leq R$ and $\|\mathbf{x}_i\|_\infty \in [0, 1]^d$, we have (see Appendix B.2 for details):

$$\left|\mathbf{w}_r(t)^\top \mathbf{x}_i \pm \epsilon \|\mathbf{w}_r(t)\|_1 - (\mathbf{w}_r(0)^\top \mathbf{x}_i \pm \epsilon \|\mathbf{w}_r(0)\|_1)\right| \leq (1+\epsilon)\sqrt{d}R. \tag{34}$$

Thereby, if $\text{sign}(\mathbf{w}_r(t)^\top \mathbf{x}_i \pm \epsilon \|\mathbf{w}_r(t)\|_1) \neq \text{sign}(\mathbf{w}_r(0)^\top \mathbf{x}_i \pm \epsilon \|\mathbf{w}_r(0)\|_1)$, then at initialization, we must have

$$|\mathbf{w}_r(0)^\top \mathbf{x}_i \pm \epsilon \|\mathbf{w}_r(0)\|_1| \leq (1+\epsilon)\sqrt{d}R. \tag{35}$$

We want to upper bound the probability that Eq. (35) holds. It is easy to show that if the following two inequalities hold, then Eq. (35) does not hold for sure:

$$|\mathbf{w}_r(0)^\top \mathbf{x}_i| \geq 2(1+\epsilon)\sqrt{d}R, \tag{36}$$

$$\epsilon \|\mathbf{w}_r(0)\|_1 \leq (1+\epsilon)\sqrt{d}R. \tag{37}$$

Therefore, the probability that Eq. (35) holds is at most 1 minus the probability that Eq. (36) and Eq. (37) hold. For Eq. (36), by anti-concentration inequality for Gaussian, we have

$$\Pr(|\mathbf{w}_r(0)^\top \mathbf{x}_i| \leq 2(1+\epsilon)\sqrt{d}R) \leq \frac{4(1+\epsilon)\sqrt{d}R}{\sqrt{2\pi}\xi}. \tag{38}$$

In other words, with probability at least $1 - 4(1+\epsilon)\sqrt{d}R/(\sqrt{2\pi}\xi)$, Eq. (36) holds. And for Eq. (37), by the tail bound of standard Gaussian and union bound, we have

$$\Pr(\epsilon \|\mathbf{w}_r(0)\|_1 \leq (1+\epsilon)\sqrt{d}R) \geq 1 - 2d \exp\left(-\frac{2(1+\epsilon)^2 dR^2}{\epsilon^2}\right). \tag{39}$$

Combining Eq. (38) and Eq. (39), Eq. (35) holds with at most the following probability

$$\frac{4(1+\epsilon)\sqrt{d}R}{\sqrt{2\pi}\xi} + 2d \exp\left(-\frac{2(1+\epsilon)^2 dR^2}{\epsilon^2}\right). \tag{40}$$

Here we require ϵ to be sufficiently small such that

$$\frac{(1+\epsilon)\sqrt{d}R}{\sqrt{2\pi}\xi} \geq d \exp\left(-\frac{2(1+\epsilon)^2 dR^2}{\epsilon^2}\right) \tag{41}$$

and we can solve the inequality to obtain an upper bound for ϵ (detailed in Appendix B.3):

$$\epsilon \leq \frac{\sqrt{2d}R}{\log(\sqrt{\frac{2\pi d}{R}}\xi)}, \tag{42}$$

and in this case Eq. (40) holds with probability at least

$$\frac{6}{\sqrt{2\pi}\xi}(1+\epsilon)\sqrt{d}R.$$

Therefore, we upper bound the probability:

$$\Pr\left(\text{sign}(\mathbf{w}_r(t)^\top \mathbf{x}_i \pm \epsilon \|\mathbf{w}_r(t)\|_1) \neq \text{sign}(\mathbf{w}_r(0)^\top \mathbf{x}_i \pm \epsilon \|\mathbf{w}_r(0)\|_1)\right) \leq \frac{6}{\sqrt{2\pi}\xi}(1+\epsilon)\sqrt{d}R.$$

Thereby

$$\Pr(A_{ri}^+(t) \neq A_{ri}^+(0)), \Pr(A_{ri}^-(t) \neq A_{ri}^-(0)) \leq \frac{6}{\sqrt{2\pi}\xi}(1+\epsilon)\sqrt{d}R.$$

Note that at least one of $A_{ri}^{+}(t)$ and $A_{ri}^{-}(t)$ always remains zero during training, because condition $y_i a_r = 1$ in $A_{ri}^{+}(t)$ and condition $y_i a_r = -1$ in $A_{ri}^{-}(t)$ are mutually exclusive. Then

$$\Pr(A_{ri}^{+}(t) + A_{ri}^{-}(t) \neq A_{ri}^{+}(0) + A_{ri}^{-}(0)) \leq \frac{6}{\sqrt{2\pi}\xi}(1+\epsilon)\sqrt{d}R,$$
$$\Pr(A_{ri}^{+}(t) - A_{ri}^{-}(t) \neq A_{ri}^{+}(0) - A_{ri}^{-}(0)) \leq \frac{6}{\sqrt{2\pi}\xi}(1+\epsilon)\sqrt{d}R.$$

Next we can upper bound the probability that each of $\alpha_{rij}(t), \beta_{rij}(t), \gamma_{rij}(t)$ changes respectively:

$$\Pr(\alpha_{rij}(t) \neq \alpha_{rij}(0)), \Pr(\beta_{rij}(t) \neq \beta_{rij}(0)), \Pr(\gamma_{rij}(t) \neq \gamma_{rij}(0)) \leq \frac{12}{\sqrt{2\pi}\xi}(1+\epsilon)\sqrt{d}R.$$

□

A.2 Proof of Lemma 2

Proof. With the probability that each of $\alpha_{rij}(t), \beta_{rij}(t), \gamma_{rij}(t)$ changes respectively as shown in Lemma 1, we can bound the expectation of the change for each element in $\mathbf{H}(t)$ (Eq. (17)) as:

$$\begin{aligned}
&\mathbb{E}[|\mathbf{H}_{ij}(t) - \mathbf{H}_{ij}(0)|] \\
\leq\ & \frac{1}{m}\mathbb{E}\Big(m\tilde{R}\|\mathbf{x}_i\|_2\|\mathbf{x}_j\|_2 + \epsilon m\tilde{R}\big((\|\mathbf{x}_i\|_2 + \|\mathbf{x}_j\|_2)\|\operatorname{sign}(\mathbf{w}_r(t))\|_2\big) + \epsilon^2 dm\tilde{R}\Big) \\
\leq\ & \tilde{R}d(1+2\epsilon+\epsilon^2) \\
=\ & \frac{12(1+\epsilon)(1+2\epsilon+\epsilon^2)d^{1.5}}{\sqrt{2\pi}}
\end{aligned}$$

Then by Markov's inequality, we have that with probability at least $1-\delta$,

$$\|\mathbf{H}(t) - \mathbf{H}(0)\|_2 \leq \sum_{(i,j)} |\mathbf{H}_{ij}(t) - \mathbf{H}_{ij}(0)| \leq \frac{12(1+\epsilon)(1+2\epsilon+\epsilon^2)d^{1.5}n^2}{\sqrt{2\pi}\xi\delta}R.$$

□

A.3 Proof of Lemma 3

Proof. First for simplicity, we define $\phi(\mathbf{x}_i)(\mathbf{w}_r(0)) = y_i(\mathbf{x}_i + \kappa_i\epsilon\operatorname{sign}(\mathbf{w}_r(0))\mathbb{1}(\mathbf{x}_i + \kappa_i\epsilon\operatorname{sign}(\mathbf{w}_r(0)) > 0)$, where $(\kappa_i = -A_{ri}^{+}(0) + A_{ri}^{-}(0)) \in \{1, -1\}$. To apply Theorem 3.1 in Du et al. (2018b) which proved $\mathbf{H}^{\infty}$ is positive definite in their standard training setting, here we need to prove that $\{\phi(\mathbf{x}_i) : 1 \leq i \leq n\}$ are linearly independent. By Theorem 3.1 in Du et al. (2018b), $\{\phi(\mathbf{x}_i) : 1 \leq i \leq n\}$ are linearly independent based on Assumption 2, which indicates that $\mathbf{H}^{\infty}$ is positive definite and $\lambda_0 > 0$. □

A.4 Proof of Lemma 5

The lemma can be proved by solving inequality Eq. (25).

Proof. According to Lemma 4, $\lambda_{\min}(\mathbf{H}(0)) \geq \frac{3}{4}\lambda_0$. And with Eq. (24), in order to ensure $\lambda_{\min}(\mathbf{H}(t)) \geq \frac{\lambda_0}{2}$, we can make

$$\frac{12(1+\epsilon)(1+2\epsilon+\epsilon^2)d^{1.5}n^2}{\sqrt{2\pi}\xi\delta}R \leq \frac{\lambda_0}{4}.$$

This yields

$$R \leq \frac{\sqrt{2\pi}\xi\delta\lambda_0}{48(1+\epsilon)(1+2\epsilon+\epsilon^2)d^{1.5}n^2}.$$

Note that $0 \leq \epsilon \leq 1$, and thus $1+\epsilon \leq 2$ and $1+2\epsilon+\epsilon^2 \leq 4$ can be upper bounded by constants respectively. Then we can take

$$R \leq \frac{\sqrt{2\pi}\xi\delta\lambda_0}{384d^{1.5}n^2} = \frac{c\delta\lambda_0}{d^{1.5}n^2}, \quad \text{where } c = \frac{\sqrt{2\pi}\xi}{384},$$

and in this case $\lambda_{\min}(\mathbf{H}(t)) \geq \frac{\lambda_0}{2}$ w.p. at least $1-\delta$ probability. □

A.5 PROOF OF LEMMA 6

Proof. The proof of this lemma is inspired by the proof of Lemma 5.4 in Zou et al. (2018). When $\lambda_{\min}(\mathbf{H})(s) \geq \frac{\lambda_0}{2}$ holds for $0 \leq s \leq t$, we can bound the derivative of $\overline{L}(t)$:

$$
\begin{aligned}
\frac{d\overline{L}(\underline{\mathbf{u}})}{dt} &= \sum_{i=1}^{n} l'(\underline{u}_i)\frac{\partial \underline{u}_i}{\partial t} \\
&= -\sum_{i=1}^{n} l'(\underline{u}_i)\sum_{j=1}^{n} l'(\underline{u}_j)\mathbf{H}_{ij} \\
&= -l'(\underline{\mathbf{u}})^\top \mathbf{H} l'(\underline{\mathbf{u}}) \\
&\overset{(i)}{\leq} -\frac{\lambda_0}{2}\sum_{i=1}^{n} l'(u_i)^2 \\
&\overset{(ii)}{\leq} \frac{\lambda_0}{2}\sum_{i=1}^{n} l'(u_i) \\
&\overset{(iii)}{\leq} -\frac{\lambda_0}{2}\sum_{i=1}^{n} \min(1/2, \frac{l(u_i)}{2}) \\
&\overset{(iv)}{\leq} -\frac{\lambda_0}{2}\min\left(1/2, \sum_{i=1}^{n}\frac{l(u_i)}{2}\right) \\
&= -\frac{\lambda_0}{2}\min(1/2, \frac{\overline{L}(\underline{\mathbf{u}})}{2}) \\
&\overset{(v)}{\leq} -\frac{\lambda_0}{2}\frac{1}{2+2/\overline{L}(\underline{\mathbf{u}})}
\end{aligned}
$$

where (i) is due to $\lambda_{\min}(\underline{\mathbf{H}}) \geq \lambda_0$, (ii) is due to $-l'(u) \leq 1$, (iii) holds due to the following property of cross entropy loss $-l'(u) \geq \min(1/2, \frac{l(u)}{2})$, (iv) holds due to the function $\min(1/2, x)$ is a concave function and Jenson's inequality, (v) holds due to $\min(a, b) \geq \frac{1}{1/a+1/b}$

Therefore, we have

$$
2\frac{d\overline{L}(\underline{\mathbf{u}})}{dt} + \frac{2}{\overline{L}(\underline{\mathbf{u}})}\frac{d\overline{L}(\underline{\mathbf{u}})}{dt} \leq -\frac{\lambda_0}{2}.
$$

By integration on both sides from 0 to t, we have

$$
\overline{L}(\underline{\mathbf{u}}(t)) - \overline{L}(\underline{\mathbf{u}}(0)) + \log(\overline{L}(\underline{\mathbf{u}}(t))) - \log\left(\overline{L}(\underline{\mathbf{u}}(0))\right) \leq -\frac{\lambda_0 t}{4}.
$$

Therefore, we have

$$
\log(\overline{L}(\underline{\mathbf{u}}(t))) \leq -\frac{\lambda_0 t}{4} + \overline{L}(\underline{\mathbf{u}}(0)) + \log(\overline{L}(\underline{\mathbf{u}}(0))),
$$

which yields

$$
\overline{L}(\underline{\mathbf{u}}(t)) \leq \exp(-\frac{\lambda_0 t}{4})\exp(\overline{L}(\underline{\mathbf{u}}(0)))\overline{L}(\underline{\mathbf{u}}(0)).
$$

And we can bound the change of $\mathbf{w}_r$.

$$\begin{aligned}
\|\frac{d\mathbf{w}_r(t)}{dt}\|_2 &= \|\frac{d\overline{L}(t)}{d\mathbf{w}_r}\|_2 \\
&= \|\sum_{i=1}^{n} l'(\underline{u}_i)\frac{1}{\sqrt{m}}a_r y_i \sigma'(\langle \mathbf{w}_r, \mathbf{x}_i \pm \epsilon\|\mathbf{w}_r\|_1\rangle)(\mathbf{x}_i \pm \epsilon\|\mathbf{w}_r\|_1)\|_2 \\
&\leq \frac{1}{\sqrt{m}}\sum_{i=1}^{n}\|l'(\underline{u}_i)\|_2 \\
&\leq \frac{n}{\sqrt{m}},
\end{aligned}$$

where $\sigma'(\cdot)$ stands for the derivative of the ReLU activation. Thus

$$\|\mathbf{w}_r(t) - \mathbf{w}_r(0)\|_2 \leq \frac{nt}{\sqrt{m}}.$$

□

A.6 Proof of Lemma 7

Proof. We first prove the standard training part. As we have defined previously that

$$L(0) = \sum_{i=1}^{n}\log(1+\exp(-u_i(0))),$$

where

$$u_i(0) = y_i\frac{1}{\sqrt{m}}\sum_{r=1}^{m}a_r\sigma(\mathbf{w}_r(0)^\top\mathbf{x}_i).$$

For each $a_r\sigma(\mathbf{w}_r(0)^\top\mathbf{x}_i)$, note that the randomness only comes from random initialization for $\mathbf{w}_r$, there is $\frac{1}{2}$ possibility that it is equal to 0, and another $\frac{1}{2}$ possibility that it follows a normal distribution $\mathcal{N}(0,\sigma_i^2)$, where $\sigma_i = \|\mathbf{x}_i\|_2^2$. Therefore, we have

$$\begin{aligned}
\mathbb{E}(a_r\sigma(\mathbf{w}_r(0)^\top\mathbf{x}_i)) &= 0, \\
\text{Var}(a_r\sigma(\mathbf{w}_r(0)^\top\mathbf{x}_i)) &= \frac{\sigma_i^2}{2}, \\
\mathbb{E}(\frac{1}{\sqrt{m}}\sum_{r=1}^{m}a_r\sigma(\mathbf{w}_r(0)^\top\mathbf{x}_i)) &= 0, \\
\text{Var}(\frac{1}{\sqrt{m}}\sum_{r=1}^{m}a_r\sigma(\mathbf{w}_r(0)^\top\mathbf{x}_i)) &= \frac{\sigma_i^2}{2}.
\end{aligned}$$

Therefore, by Chebyshev's inequality, we can bound

$$\Pr(|u_i(0)| \leq \frac{\sigma_i^2}{2\delta}) \geq 1-\delta.$$

And we can bound $L(0) = O(\frac{n\max_{i=1}^{n}\sigma_i^2}{2\delta}) = O(\frac{n}{\delta})$ with probability at least $1-\delta$.

For IBP training,

$$\underline{u}_i(0) = \frac{1}{\sqrt{m}}\sum_{r=1}^{m}\mathbb{1}(y_i a_r = 1)\sigma\big(\mathbf{w}_r(0)^\top\mathbf{x}_i - \epsilon\|\mathbf{w}_r(0)\|_1\big) + \mathbb{1}(y_i a_r = -1)\sigma\big(\mathbf{w}_r(0)^\top\mathbf{x}_i + \epsilon\|\mathbf{w}_r(0)\|_1\big).$$

Thus

$$|\underline{u}_i(0) - u_i(0)| \leq \frac{1}{\sqrt{m}}m\epsilon\|\mathbf{w}_r(0)\|_1 = \sqrt{m}\epsilon\|\mathbf{w}_r(0)\|_1.$$

By $\mathbb{E}(\|\mathbf{w}_r\|_1) = O(d)$ and Markov's inequality, with probability at least $1-\delta$,

$$|\underline{u}_i(0) - u_i(0)| \leq O(\frac{\sqrt{m}d\epsilon}{\delta}).$$

And we can bound $\overline{L}(0) = O(\frac{n\sqrt{m}d\epsilon}{\delta} + \frac{n}{\delta})$ with probability at least $1-\delta$. □

B Detailed Derivation for Other Equations or Inequalities

B.1 Derivation on the Dynamics of $\underline{u}_i(t)$

We provide a detailed derivation on the dynamics of $\underline{u}_i(t)$ presented in Eq. (12), which we use $\mathbf{H}_i(t)$ to describe $\frac{d}{dt}\underline{u}_i(t)$:

$$\begin{aligned}
\frac{d}{dt}\underline{u}_i(t) &= \sum_{r=1}^{m}\langle \frac{\partial \underline{u}_i(t)}{\partial \mathbf{w}_r(t)}, \frac{d\mathbf{w}_r(t)}{dt}\rangle \\
&= \sum_{r=1}^{m}\langle \frac{\partial \underline{u}_i(t)}{\partial \mathbf{w}_r(t)}, -\frac{\partial \overline{L}(\mathbf{W}(t),\mathbf{a})}{\partial \mathbf{w}_r(t)}\rangle \\
&= \sum_{r=1}^{m}\langle \frac{\partial \underline{u}_i(t)}{\partial \mathbf{w}_r(t)}, -\sum_{j=1}^{n} l'(\underline{u}_j)\frac{\partial \underline{u}_j(t)}{\partial \mathbf{w}_r(t)}\rangle \\
&= \sum_{j=1}^{n} -l'(\underline{u}_j)\sum_{r=1}^{m}\langle \frac{\partial \underline{u}_i(t)}{\partial \mathbf{w}_r(t)}, \frac{\partial \underline{u}_j(t)}{\partial \mathbf{w}_r(t)}\rangle \\
&= \sum_{j=1}^{n} -l'(\underline{u}_j)\mathbf{H}_{ij}(t),
\end{aligned}$$

B.2 Derivation for Eq. (34)

Eq. (34) basically comes by triangle inequality:

$$\begin{aligned}
& \left|\mathbf{w}_r(t)^\top \mathbf{x}_i - \epsilon\|\mathbf{w}_r(t)\|_1 - (\mathbf{w}_r(0)^\top \mathbf{x}_i - \epsilon\|\mathbf{w}_r(0)\|_1)\right| \\
= \; & \left|(\mathbf{w}_r(t)-\mathbf{w}_r(0))^\top \mathbf{x}_i - \epsilon\|\mathbf{w}_r(t)\|_1 + \epsilon\|\mathbf{w}_r(0)\|_1\right| \\
\leq \; & |(\mathbf{w}_r(t)-\mathbf{w}_r(0))^\top \mathbf{x}_i| + \epsilon\left|\|\mathbf{w}_r(t)\|_1 - \|\mathbf{w}_r(0)\|_1\right| \\
\leq \; & |(\mathbf{w}_r(t)-\mathbf{w}_r(0))^\top \mathbf{x}_i| + \epsilon\left\|\mathbf{w}_r(t)-\mathbf{w}_r(0)\right\|_1 \\
\leq \; & (1+\epsilon)\sqrt{d}R.
\end{aligned}$$

B.3 Derivation for Eq. (42)

We solve the inequality in Eq. (41) to derive an upper bound for ϵ in Eq. (42):

$$\frac{1}{\sqrt{2\pi}\xi}(1+\epsilon)\sqrt{d}R \geq \frac{1}{\sqrt{2\pi}\xi}\sqrt{d}R \geq d\exp\Big(-\frac{2(1+\epsilon)^2 dR^2}{\epsilon^2}\Big),$$

$$\frac{R}{\sqrt{2\pi d}\xi} \geq \exp\Big(-\frac{2(1+\epsilon)^2 dR^2}{\epsilon^2}\Big),$$

$$\log(\frac{R}{\sqrt{2\pi d}\xi}) \geq -\frac{2(1+\epsilon)^2 dR^2}{\epsilon^2},$$

and then we can require

$$\log(\frac{R}{\sqrt{2\pi d}\xi}) \geq -\frac{2dR^2}{\epsilon^2} \geq -\frac{2(1+\epsilon)^2 dR^2}{\epsilon^2} \quad \Rightarrow \epsilon \leq \frac{\sqrt{2d}R}{\log(\sqrt{\frac{2\pi d}{R}}\xi)}.$$