Introduction to Machine and Deep Learning Theory

Certified Robustness I: Randomized Smoothing

Aleksandr Petiushko

Lomonosov MSU
Faculty of Mechanics and Mathematics

April 19th, 2023

AP

Content

1. Certified robustness definitions
2. Certified robustness via Lipschitzness
3. Randomized Smoothing and its variants

AP

Robustness in Machine Learning

Robustness [informally]

Ability for a machine learning algorithm a to provide similar outputs on the similar data (i.e. having the same class or other invariant features)

Two types of **Robustness** in ML:

Generalization

Dataset issue: algorithm needs to be robust if the dataset to evaluate it differs (sometimes significantly: we can treat it is a distribution shift) from the training dataset

Adversarial Robustness

Noise issue: algorithm needs to provide the similar output w.r.t. both clean and noisy images (where the model of noise is the topic to consider itself)

For now we'll consider the **Adversarial Robustness**.

AP

Adversarial Robustness topics

- Perturbations (also called 'adversarial *attacks*'): how to generate noise to fool the neural net

AP

Adversarial Robustness topics

- Perturbations (also called 'adversarial *attacks*'): how to generate noise to fool the neural net
- Defense: how to diminish the influence of adversarial perturbations

Adversarial Robustness topics

- Perturbations (also called 'adversarial *attacks*'): how to generate noise to fool the neural net
- Defense: how to diminish the influence of adversarial perturbations
- Certification (or verification): how to provide theoretical guarantees on the noise level not fooling the neural net

Adversarial Robustness topics

- Perturbations (also called 'adversarial *attacks*'): how to generate noise to fool the neural net
- Defense: how to diminish the influence of adversarial perturbations
- Certification (or verification): how to provide theoretical guarantees on the noise level not fooling the neural net

AP

Certified Robustness: for classification

- Let us NN function $f(x)$ is the classifier to K classes: $f : \mathbb{R}^d \to Y, Y = \{1, \dots, K\}$
- Usually we have NN $h(x) : \mathbb{R}^d \to \mathbb{R}^K$, and $f(x) = \arg\max_{i \in Y} h(x)_i$

Deterministic approach

Need to find the class of input perturbation $S(x, f)$ so as the classifier's output doesn't not change, or more formally:

$$f(x + \delta) = f(x) \quad \forall \delta \in S(x, f)$$

Probabilistic approach

Need to find the class of input perturbation $S(x, f, P)$ w.r.t. robustness probability P s.t.:

$$Prob_{\delta \in S(x,f,P)}(f(x + \delta) = f(x)) = P$$

Remark: Probabilistic approach coincides with Deterministic one when $P = 1$.

AP

Certified Robustness: for regression

- Let us NN function $f(x)$ NN f(x) is the regressor: $f : \mathbb{R}^d \rightarrow \mathbb{R}$

Deterministic approach

Need to find the class of input perturbation $S(x, f, f_{low}, f_{up})$ w.r.t. the upper and lower bounds on the output perturbation f_{low}, f_{up} s.t.:

$$f(x) - f_{low} \leq f(x+\delta) \leq f(x) + f_{up} \quad \forall \delta \in S(x, f, f_{low}, f_{up})$$

Probabilistic approach

Need to find the class of input perturbation $S(x, f, f_{low}, f_{up}, P)$ w.r.t. robustness probability P and the upper / lower bounds on the output perturbation f_{low}, f_{up} s.t.:

$$Prob_{\delta \in S(x, f, f_{low}, f_{up}, P)}(f(x) - f_{low} \leq f(x+\delta) \leq f(x) + f_{up}) = P$$

AP

Certified Robustness: inverse tasks for classification

- Suppose that we know the input perturbation class S
- For classification we have only probabilistic formulation

Classification

Need to measure the probability P of retaining the classifier's output under some class of input perturbations S:

$$Prob_{\delta \in S}(f(x+\delta) = f(x)) = P$$

AP

Certified Robustness: inverse tasks for regression

- Suppose that we know the input perturbation class S
- For regression we have both deterministic and probabilistic formulations

Regression (deterministic formulation)

Need to find the upper and lower bounds $f_{low}(f, x, S), f_{up}(f, x, S)$ of the output perturbation under some class of input perturbations S:

$$f(x) - f_{low}(f, x, S) \leq f(x + \delta) \leq f(x) + f_{up}(f, x, S)$$

Regression (probabilistic formulation)

Need to measure the probability P of keeping the classifier's output inside the lower / upper bounds f_{low}, f_{up} under some class of input perturbations S:

$$Prob_{\delta \in S}(f(x) - f_{low} \leq f(x + \delta) \leq f(x) + f_{up}) = P$$

Certified Robustness via Lipschitzness (1)

- NN classifier to K classes is $f(x)$: $f : \mathbb{R}^d \to Y, Y = \{1, \ldots, K\}$
- NN itself is $h(x) : \mathbb{R}^d \to \mathbb{R}^K$, and $f(x) = \arg\max_{i \in Y} h(x)_i$
- Consider binary case (other cases are treated similarly) $K = 2$ and probabilistic (SoftMax) output: $h(x)_1 + h(x)_2 = 1, \quad h(x)_i \geq 0 \quad \forall i$

Definition of Lipschitz function

Lipschitz function g: $g : \mathbb{R}^d \to \mathbb{R}$ with a Lipschitz constant L so as $\forall x_1, x_2$ it holds $|g(x_1) - g(x_2)| \leq L \|x_1 - x_2\|$

Definition of Local Lipschitz function

Local Lipschitz function g: $g : \mathbb{R}^d \to \mathbb{R}$ with a Lipschitz constant $L(x_0)$ so as $\forall x \in S(x_0)$ it holds $|g(x_0) - g(x)| \leq L(x_0) \|x_0 - x\|$

AP

Certified Robustness via Lipschitzness (2)

- Let $j = \arg\max_{i \in Y} h(x_0)_i$, and $h(x_0)_j - h(x_0)_{i \neq j} \geq \epsilon$
- Let $h(x_0)_j$ — local Lipschitz function with a Lipschitz constant $L(x_0)$
- Then if $S(x_0) = \{x : \|x_0 - x\| \leq \frac{\epsilon}{2L(x_0)}\}$ we have $|h(x_0)_j - h(x)_j| \leq L(x_0)\frac{\epsilon}{2L(x_0)} = \frac{\epsilon}{2}$
- Therefore $j = \arg\max_{i \in Y} h(x)_i$ and $f(x) = f(x_0) = j$ in the vicinity $S(x_0) = \{x : \|x_0 - x\| \leq \frac{\epsilon}{2L(x_0)}\}$
- $\Rightarrow$ Certified Robustness!

Certified Robustness via Lipschitzness (3)

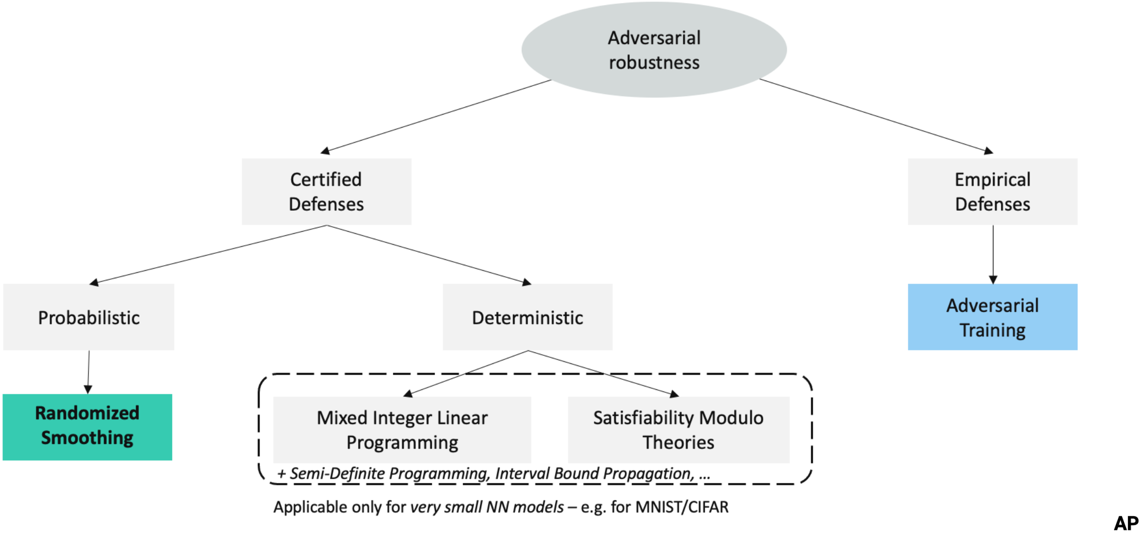
But:

Problems

- The certified radius can be much bigger than the local Lipschitz vicinity $S(x_0)$
- It is hard to provide the adequate (not tending to ∞) Lipschitz constant for any industrial Deep Neural Network

AP

Adversarial Robustness: overview



Adversarial Robustness: empirical vs certified

Empirical robustness

Bound

The upper bound on the true robust accuracy

Cons

Only valid *until* the *new* – and stronger – *attack* appears

Certified robustness

Bound

The lower bound on the true robust accuracy

Pros

It is what has been *theoretically proven*, and no one attack can beat it

AP

Empirical Robustness: Adversarial Training

- Let us have the training dataset $D = \{(x_i, y_i)_{i=1}^M\}$
- Parameters of the neural net f are denoted as θ
- Loss function is $L(f_\theta(x), y) \Rightarrow$ the training process is

$$\min_\theta \mathbf{E}_{(x,y)\in D} L(f_\theta(x), y)$$

Adversarial Training (AT)

Idea: train on the **hardest examples** using some class of perturbations $S(x)$ around training examples $\Rightarrow$ AT is

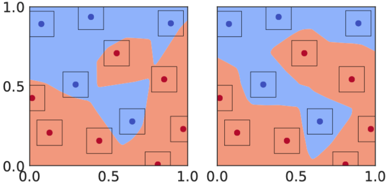
$$\min_\theta \mathbf{E}_{(x,y)\in D}[\max_{x+\delta\in S(x)} L(f_\theta(x+\delta), y)]$$

AP

Adversarial Training: pros and cons

AT pros

Very simple methodological principle of training

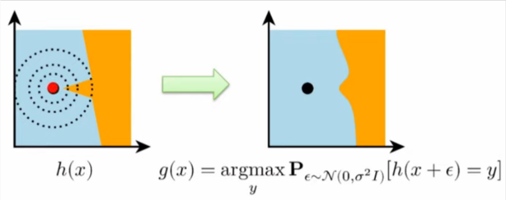


AT cons

- Quite *inefficient training* (longer than usual because need to find hard perturbation for **every training example** for **every iteration**)
- The *accuracy on clean samples is lower* than for usual training

Adversarial Examples: boundary curvature

- Very **curved boundary** leads to *adversarial examples* looking very similar to ones near the classification boundary
- So let's **diminish** this curvature **spike** influence!
- Different approaches exist e.g. by *Lecuyer et al.*[1] and *Li et al.*[2], but the most famous one is by *Cohen et al.*[3]



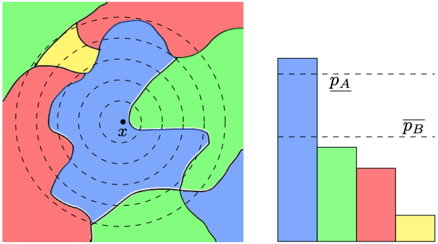
[1] Lecuyer, Mathias, et al. "Certified robustness to adversarial examples with differential privacy." 2018
[2] Li, Bai, et al. "Certified adversarial robustness with additive noise." 2018
[3] Cohen, Jeremy, et al. "Certified adversarial robustness via randomized smoothing." 2019

Randomized Smoothing

Idea of Randomized Smoothing (RS)

- Let's use the **T**est **T**ime **A**ugmentation (**TTA**) in order to mitigate the boundary effect
- The new classifier $g(x)$ is defined as:

$$g(x) = \underset{c \in Y}{\arg\max}\, P(f(x+\epsilon) = c), \epsilon \sim N(0, \sigma^2)$$



RS main result

- If the initial classifier $f(x)$ is robust under Gaussian noise,
- Then the new classifier $g(x)$ is robust under **ANY** noise

Randomized Smoothing: Theory overview

Theorem: Certification Radius

Suppose $c_A \in Y$ and $\underline{p_A}, \overline{p_B} \in [0,1]$ satisfy $\mathbb{P}(f(x+\epsilon) = c_A) \geq \underline{p_A} \geq \overline{p_B} \geq \max_{c_B \neq c_A} \mathbb{P}(f(x+\epsilon) = c_B)$. Then $g(x+\delta) = c_A \quad \forall \, \|\delta\|_2 < R$, where

$$R = \frac{\sigma}{2}(\Phi^{-1}(\underline{p_A}) - \Phi^{-1}(\overline{p_B}))$$

Tightness of Radius R

Assume $\underline{p_A} + \overline{p_B} \leq 1$. Then for any perturbation $\delta, \|\delta\|_2 > R$ there exist a base classifier f s.t. $\mathbb{P}(f(x+\epsilon) = c_A) \geq \underline{p_A} \geq \overline{p_B} \geq \max_{c_B \neq c_A} \mathbb{P}(f(x+\epsilon) = c_B)$ so as $g(x+\delta) \neq c_A$

Remark. Φ^{-1} is the inverse of the standard Gaussian CDF: $\Phi(x) = \frac{1}{2\pi} \int_{-\infty}^{x} e^{-\frac{t^2}{2}} dt$.

AP

Randomized Smoothing: Theory insights

Why $\underline{p_A}$ and $\overline{p_B}$ instead of p_A and p_B?

Because in most cases we cannot get exact probabilities for $P(f(x+\epsilon)=c), \epsilon \sim N(0,\sigma^2)$, and we need to estimate.

How to get the R?

Use so called Neyman-Pearson Lemma[4].

[4] Neyman, Jerzy, and Egon Sharpe Pearson. "On the problem of the most efficient tests of statistical hypotheses." 1933

Randomized Smoothing: Interesting cases

Let us consider the linear binary classifier $f(x) = \text{sign}(w^T x + b)$

It is a smoothed version of itself

If g is a smoothed version of f with any σ, then $f(x) = g(x)$.

Certified radius just a distance to the boundary

If g is a smoothed version of f with any σ, then using the previous Theorem for certification radius R with $\underline{p_A} = p_A$ and $\underline{p_B} = p_B$ will yield $R = \frac{|w^T x + b|}{\|w\|}$.

But sometimes the certification radius can be really big (for non-linear binary classifier):

Certified radius can be of any value

For any $\tau > 0$, there exists a base classifier f and an input x_0 for which the corresponding g is robust around x_0 at radius ∞, whereas the previous Theorem for certification radius R only certifies a radius $R = \tau$ around x_0.

Randomized Smoothing: Training

- To certify the classifiers, authors **trained the base models with Gaussian noise** from $N(0, \sigma^2 I)$ — actually, to make the classifier $f(x)$ to be more robust to Gaussian noise
- So no any other training-specific tricks aside from simple **augmentation**

AP

Randomized Smoothing: Inference

- Trained models are compared using "**approximate certified accuracy**":
  - $\forall$ test radius $\delta = r$ the fraction of examples is returned so as the procedure CERTIFY:
    - Provides the answer
    - Returns the correct class
    - Returns a radius R so as $r \leq R$

Procedure CERTIFY

- Can return ABSTAIN if confidence bounds are too loose (done by **Clopper-Pearson** confidence intervals for the Binomial distribution[5])
- If not ABSTAIN, then return the majority class $\hat{c}_A$ and certification radius $R = \sigma\Phi^{-1}(\underline{p_A})$

Remark1. Quantile of Gaussian distribution corresponding to the error rate is denoted as α (larger α, tighter the **c**onfidence **i**nterval (CI), but less reliable).

Remark2. Class estimation and CI of g are done by Monte Carlo sampling n times.

[5] Clopper, Charles J., and Egon S. Pearson. "The use of confidence or fiducial limits illustrated in the case of the binomial." 1934

AP

Randomized Smoothing: Results on ImageNet

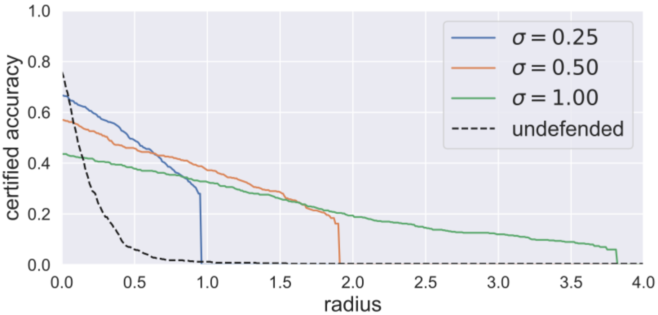


Table 1. Approximate certified accuracy on ImageNet. Each row shows a radius r, the best hyperparameter σ for that radius, the approximate certified accuracy at radius r of the corresponding smoothed classifier, and the standard accuracy of the corresponding smoothed classifier. To give a sense of scale, a perturbation with ℓ_2 radius 1.0 could change one pixel by 255, ten pixels by 80, 100 pixels by 25, or 1000 pixels by 8. Random guessing on ImageNet would attain 0.1% accuracy.

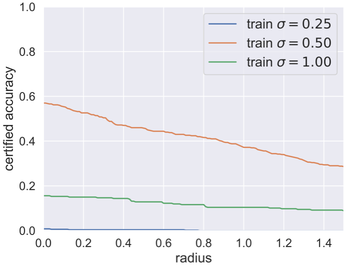
ℓ_2 RADIUS	BEST σ	CERT. ACC (%)	STD. ACC(%)
0.5	0.25	49	67
1.0	0.50	37	57
2.0	0.50	19	57
3.0	1.00	12	44

Remark1. Waterfall just because the trained model is robust usually under some $r \leq R$.
Remark2. "Certified accuracy" = approximate certified accuracy.
Remark3. The difference between "clean" and "certified" accuracy is not order of magnitude (it works! and can be useful).

AP

Randomized Smoothing: Influence of training noise parameter σ

- Main outcomes:
  - Best results are when the inference σ_I and training σ_T parameters of noise σ are exactly the same: $\sigma_T = \sigma_I = \sigma$
  - If not the same, better results are when the training noise is more severe than the inference one: $\sigma_T > \sigma_I$

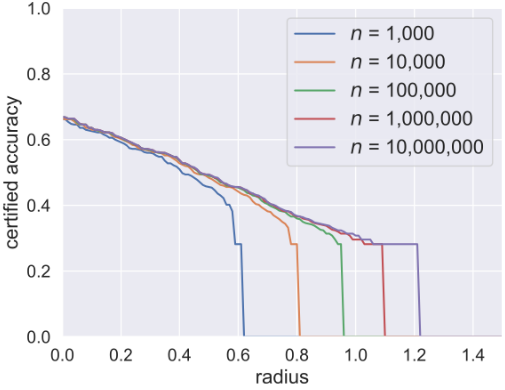


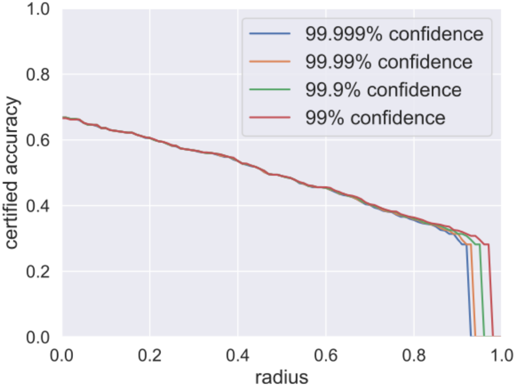
(b) ImageNet

Remark. Here the inference noise level is $\sigma_I = 0.5$.

Randomized Smoothing: Influence of Inference parameters n and α

- Main outcomes:
  - Larger number of Monte Carlo samples n, the larger the certified radius R (significantly; but veeery slow)
  - Larger confidence in results $(1-\alpha)$, the smaller the certified radius R (but not much)





Randomized Smoothing: Robustness radius in practice

- Actually, with NN classifier $f(x)$ we can have **larger real robustness radius** than R from Theorem
- Authors just tried to find the real adversaries under $r > R$ and measure the success of the attack (cf. the **inverse formulation** of Certified Robustness)
- (The lower the success rate the more robust the model):
  - $r = 1.5 \cdot R \Rightarrow 17\%$ of success rate
  - $r = 2 \cdot R$: $\Rightarrow 53\%$ of success rate

Certification: intermediate takeaway

- Randomized Smoothing = Smoothing distribution + norm l_p of perturbation
- Randomized Smoothing requires multiple inferences :(
- Certified robustness is better than empirical adversarial training in certification, but worse than clean performance (and too much time to train)

AP

Thank you!