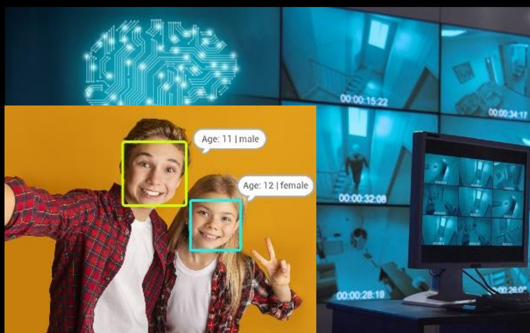
Certified Adversarial Robustness

Mengdie Huang

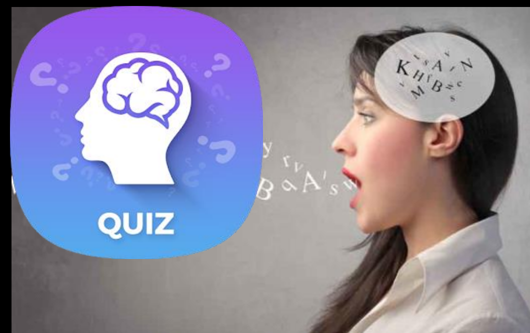
April 1, 2022

Overview

AI Applications



Computer Vision



Speech Recognition

Natural Language Processing

Expert system

Smart Robot

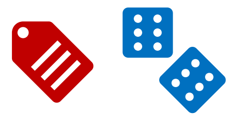
Chess Game

AI Error

- Traffic Light & Stop Sign Must Reads by Autopilot

Transform

DNN

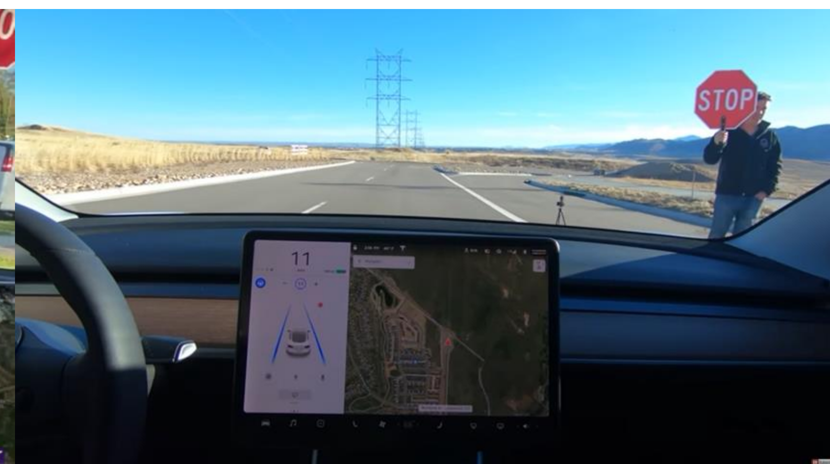


Stop 91.53%

Response

- Stop Sign can be Ignored by Autopilot

Autopilot action: Stop



Autopilot action: Keep going

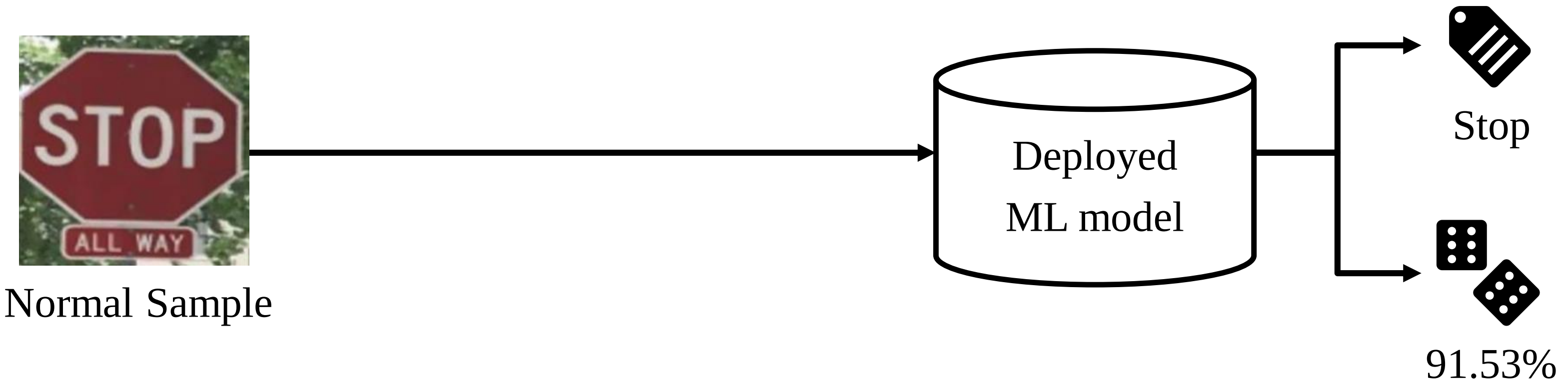
Background

Attack

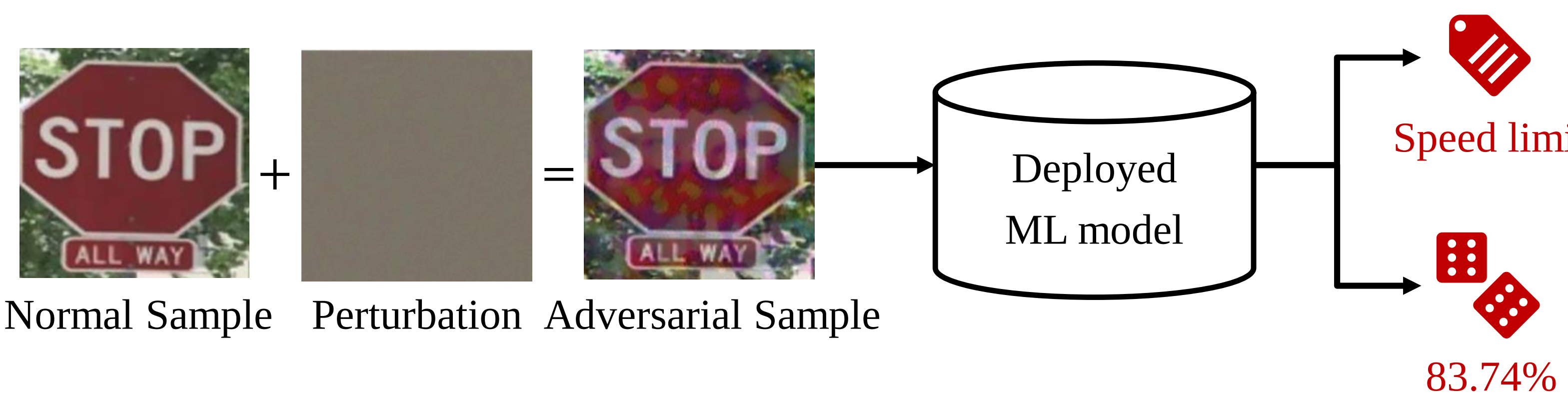
German Traffic Sign Recognition Benchmark (**GTSRB**) dataset

- Adversarial attack

➢ Normal Prediction



➢ Adversarial Attack

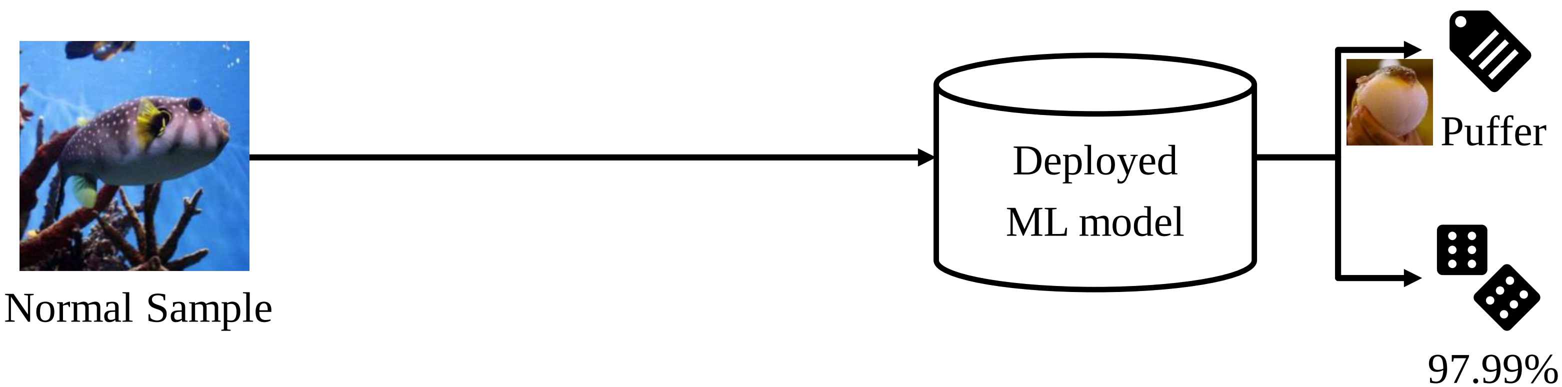


Attack

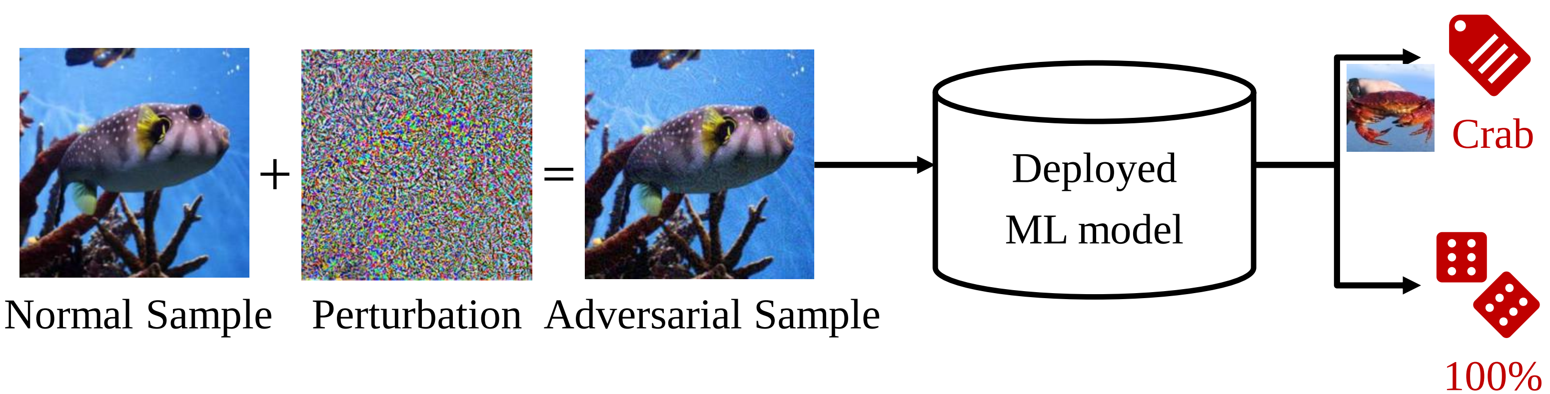
ImageNet dataset

- Adversarial attack

➤ Normal Prediction



➤ Adversarial Attack

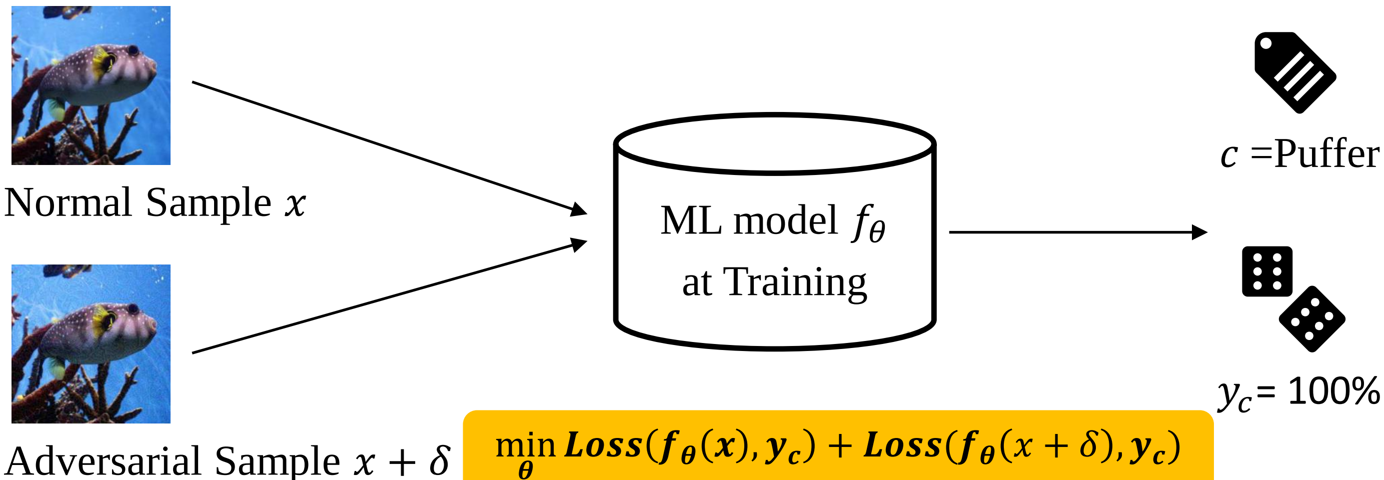


Dong Yinpeng. Boosting adversarial attacks with momentum. CVPR. 2018.

Defense Category

- Empirical Defense

➢ Typical method: Adversarial training



$$\min_{\theta} \boldsymbol{Loss}(\boldsymbol{f_\theta}(\boldsymbol{x}), \boldsymbol{y_c}) + \boldsymbol{Loss}(\boldsymbol{f_\theta}(x+\delta), \boldsymbol{y_c})$$

➢ Drawback:

most of heuristics defenses have been shown to fail against suitably powerful adversaries (cat-and-mouse game).

➢ New requirement:

rigorous, theory-backed defensive approaches to stop this arms race.

Defense Category

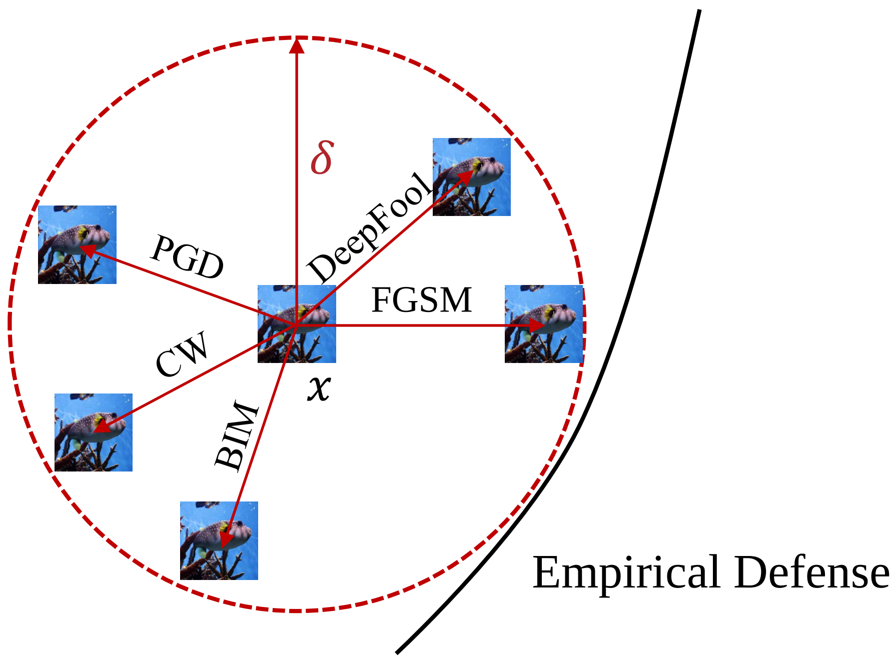
- Empirical Defense

➢ Typical method: Adversarial training

Defense Category

- Empirical Defense

➢ Typical method: Adversarial training

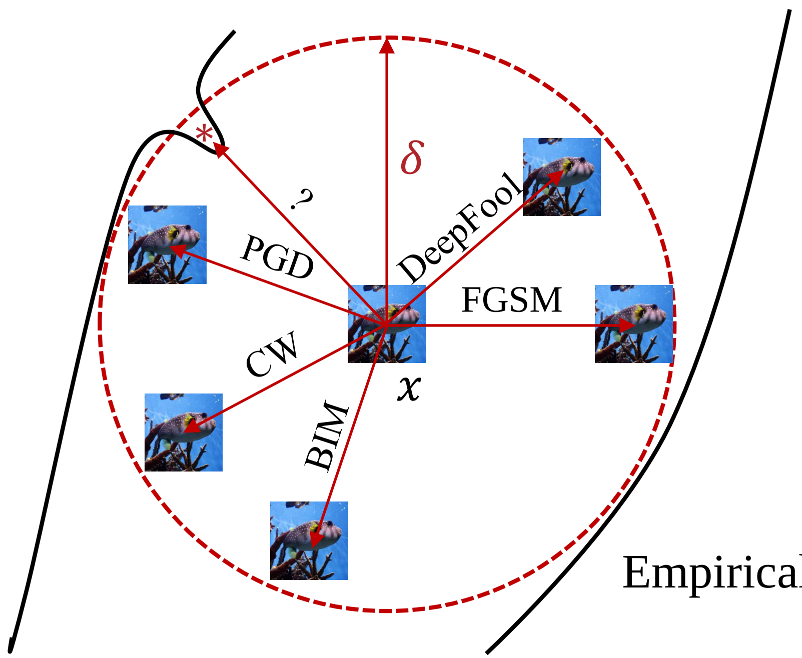


Goodfellow. Explaining and harnessing adversarial examples. ICLR. 2015.

Defense Category

- Empirical Defense

➢ Typical method: Adversarial training

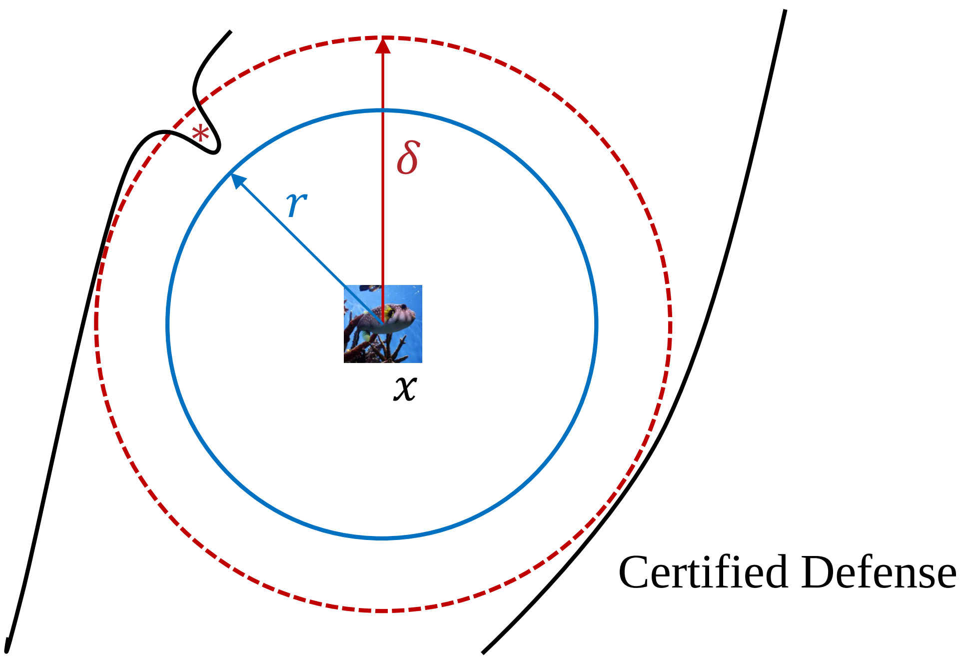


Empirical Defense

Defense Category

- Certified Defense
- ➢ Provide a certificate for adversarial robustness
- ➢ Certificate (x, f, r)

  For any input x, the prediction output by the classifier on some set around x are guaranteed to be constant.



Goodfellow. Explaining and harnessing adversarial examples. ICLR. 2015.

Defense Category

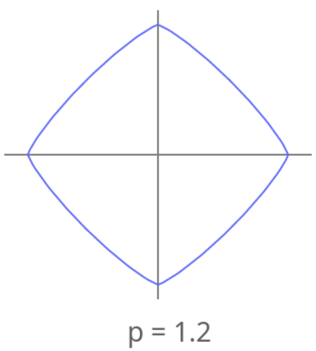
- **Certified Defense**

➢ Provide a certificate for adversarial robustness

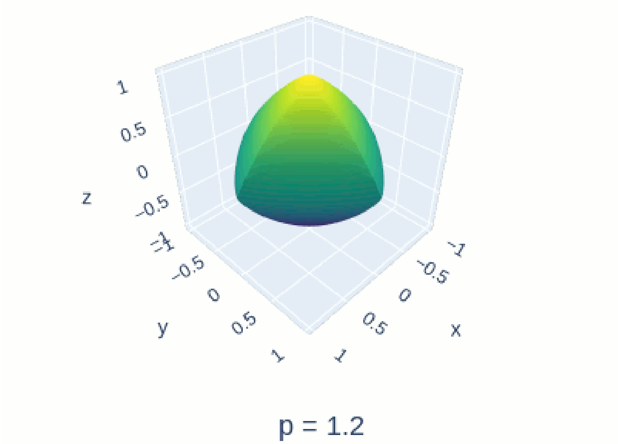
➢ Certificate (x, f, r)

For any input x, the prediction output by the classifier on some set around x are guaranteed to be constant.

➢ Some set: L_p ball with radius r



2D L_p Balls



3D L_p Balls

L_1 ball with $r = 1$

L_2 ball with $r = 1$ 球

L_3 ball with $r = 1$

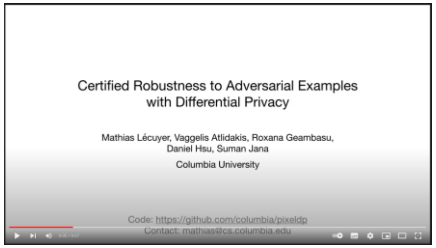
L_4 ball with $r = 1$

…

L_∞ ball with $r = 1$ 立方体

Goodfellow. Explaining and harnessing adversarial examples. ICLR. 2015.

Certified Robustness to Adversarial Examples with Differential Privacy



Mathias Lecuyer, … , Suman Jana
Columbia University
IEEE S&P 2019

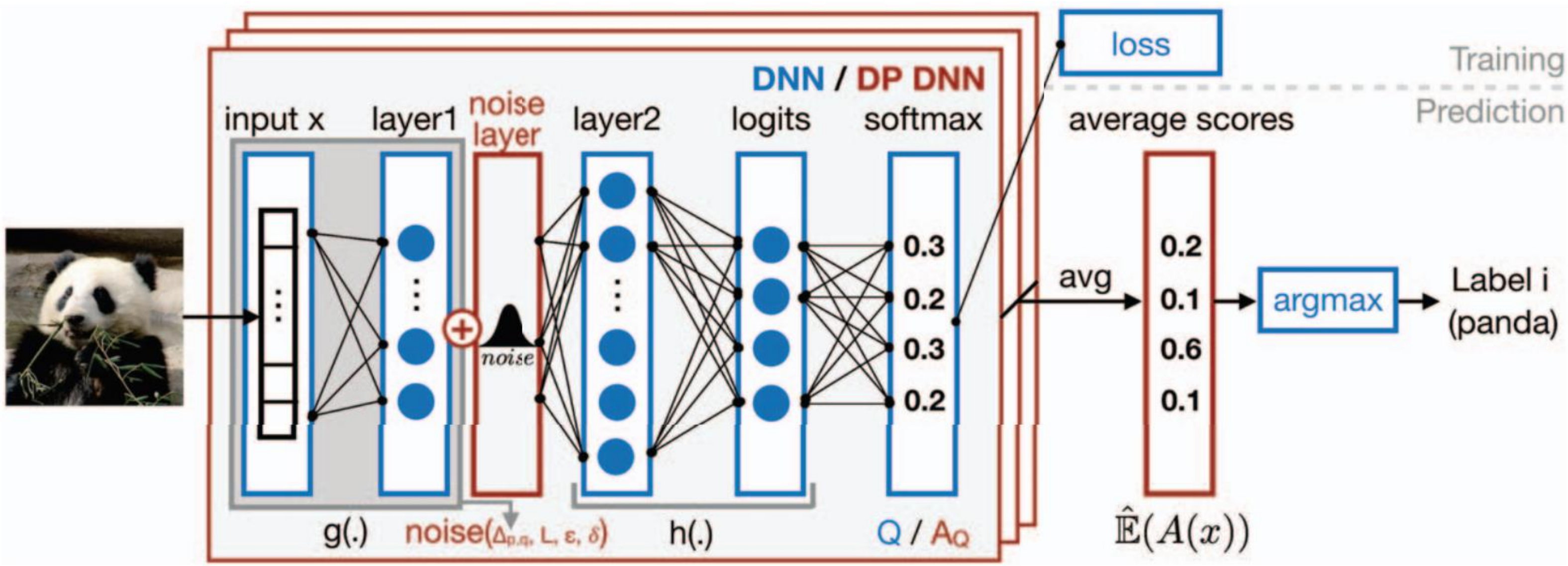
Overview

- Design a DNN classifier with differential privacy properties

➢ Essence: Introduce randomness into the prediction of the classifier.

➢ Way: Add a noise layer to the network.

➢ Purpose: Guarantee that the output of the model is insensitive to small changes in the input.

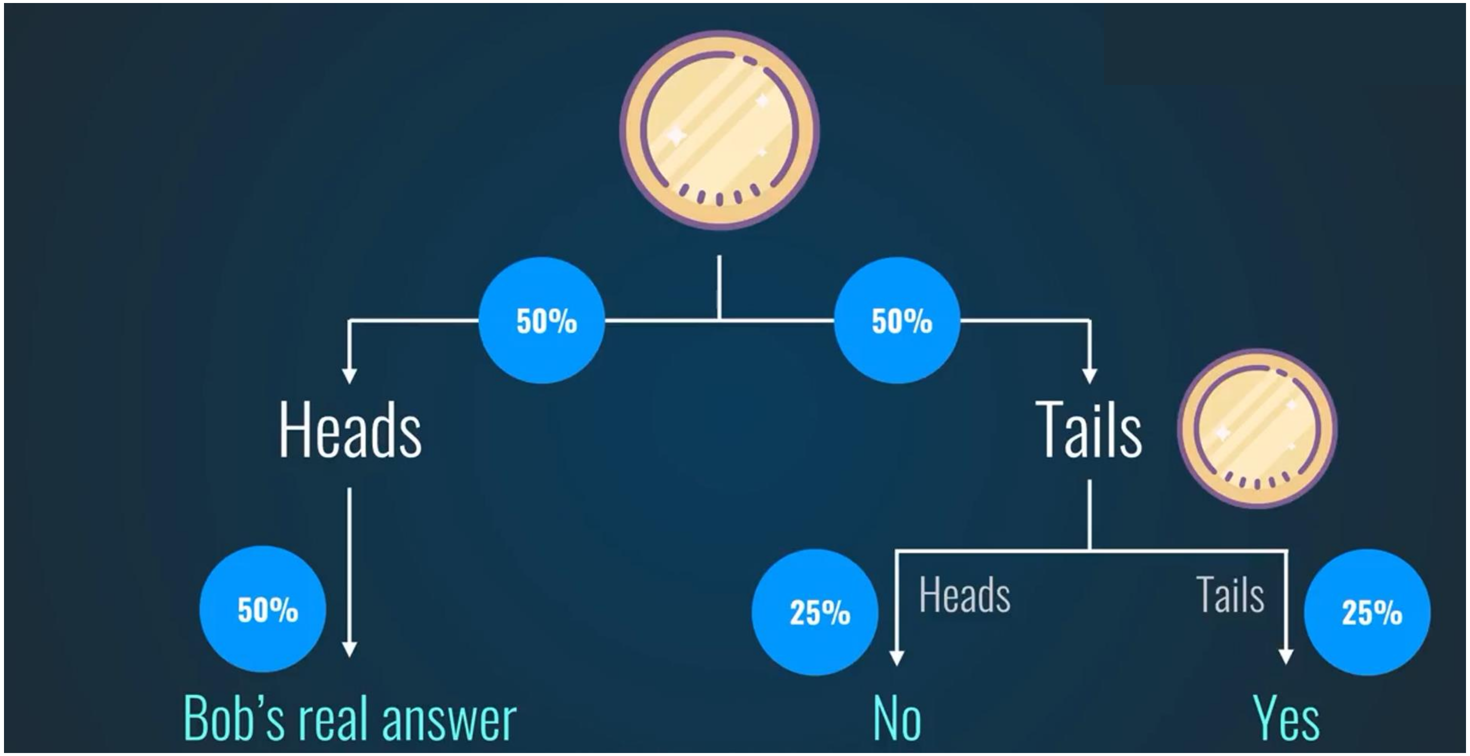


Lecuyer, Mathias. Certified robustness to adversarial examples with differential privacy. IEEE S&P 2019.

Preliminary

- Differential Privacy

➢ Randomize responses

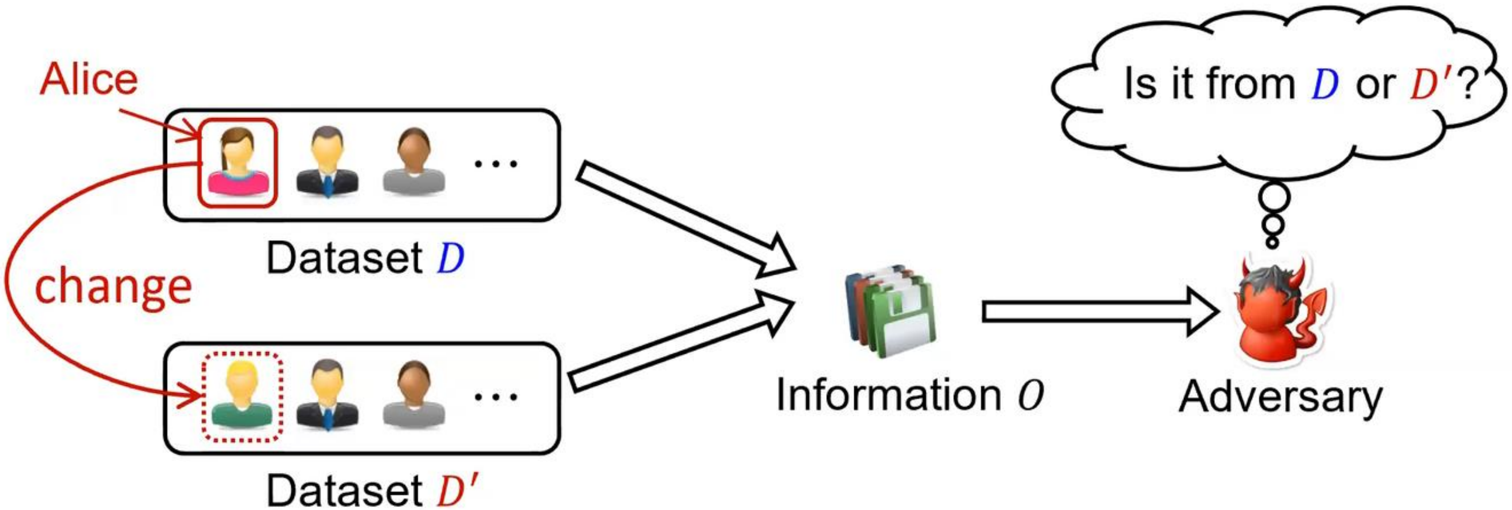


Lecuyer, Mathias. Certified robustness to adversarial examples with differential privacy. IEEE S&P 2019.

Preliminary

- ϵ-DP (Differential Privacy)

➢ Intuitive understanding



- The only difference between D and D' is Alice.
- If the attacker cannot tell whether the information O comes from D or D', it can be considered that the privacy of Alice is protected.
- DP requires information O to be randomized before output.

Lecuyer, Mathias. Certified robustness to adversarial examples with differential privacy. IEEE S&P 2019.

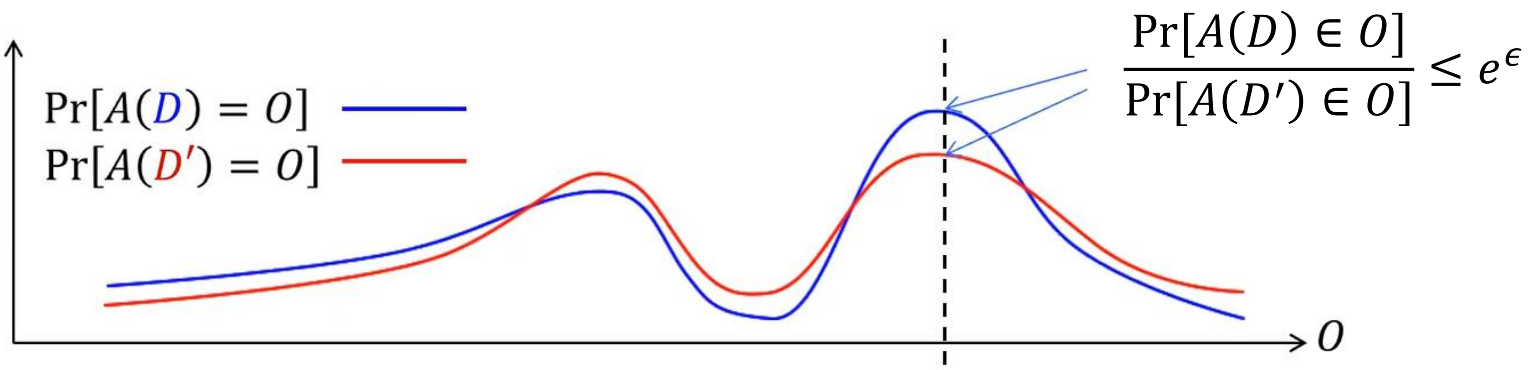
Preliminary

- ϵ-DP (Differential Privacy)

➤ Formalizing

Randomized algorithm A is ϵ-differentially private if for any $O \subseteq Range(A)$ and for any neighboring dataset D, D' ($||D - D'||_1 \leq 1$):

$$\Pr[A(D) \in O] \leq e^{\epsilon} \Pr[A(D') \in O]$$



Lecuyer, Mathias. Certified robustness to adversarial examples with differential privacy. IEEE S&P 2019.

Preliminary

- (ϵ, δ)-DP (Differential Privacy)

➢ Formalizing

Randomized algorithm A is (ϵ, δ)-differentially private if for any $O \subseteq Range(A)$ and for any neighboring dataset D, D' ($||D - D'||_1 \leq 1$):

$$\Pr[A(D) \in O] \leq e^{\epsilon} \Pr[(D') \in O] + \delta$$

$$\frac{\Pr[A(D) \in O] - \delta}{\Pr[A(D') \in O]} \leq e^{\epsilon}$$

- when an event is more likely under D than under D', δ is positive(+).
- when an event is more likely under D' than under D, δ is negative(-).
- $||D - D'||_1 \leq 1$ can be generalized to $||D - D'||_p \leq L$ by applying group privacy.

C. Dwork. The algorithmic foundations of differential privacy. Foundations and Trends in Theoretical Computer Science, 2014.

Preliminary

- Properties of DP

➢ Post-processing

If $A(x)$ satisfies (ϵ,δ)-DP, h is a x-independent mapping algorithm，
Then, the composition $h\circ A = h(A(x))$ satisfies (ϵ,δ)-DP.

➢ Expected output stability

The expected value $\mathbb{E}\big(A(x)\big)$ of an (ϵ,δ)-DP algorithm A with bounded output $A(x)\in[0,b]$ is not sensitive to small changes in the input.

$\forall\,\alpha\le Ball_p(r=1),\quad \mathbb{E}\big(A(x)\big)\le e^{\epsilon}\mathbb{E}\big(A(x+\alpha)\big)+b\delta$

✓ *Proof*

- $\Pr[A(x)\in O]\le e^{\epsilon}\Pr[(x+\alpha)\in O]+\delta$ (ϵ,δ)-DP定义
- $\int_0^b \Pr[A(x)\in O]\,dt\le \int_0^b e^{\epsilon}\Pr[(x+\alpha)\in O]+\delta\,dt$ 积分
- $\int_0^b \Pr[A(x)<t]\,dt\le e^{\epsilon}\int_0^b \Pr[(x+\alpha)>t]\,dt+\int_0^b \delta\,dt$ PDF定义
- $\mathbb{E}\big(A(x)\big)\le e^{\epsilon}\mathbb{E}\big(A(x+\alpha)\big)+b\delta$ 期望定义

Lecuyer, Mathias. Certified robustness to adversarial examples with differential privacy. IEEE S&P 2019.

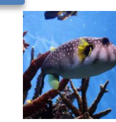
DP-robustness Connection

- One-to-one correspondence

Differential Privacy	Adversarial Robustness

Sex	Blood	...	HIV
F	B	...	Y
M	A	...	N
M	O	...	N
M	O	...	Y
F	A	...	N
M	B	...	Y

- Databases
- Input images



$A(d)$

- Randomized query algorithm
- Randomized classify algorithm

$A(x)$

M	A	...	N

- Records in a database
- Pixels in an images

M	O	...	N

- Remove/alter on records
- Increase or decrease pixels

$\Delta A(d) \leq 1$

- Result in a bounded change in the output
- Result in a bounded change in the prediction

$A(x + \alpha) = A(x)$

Lecuyer, Mathias. Certified robustness to adversarial examples with differential privacy. IEEE S&P 2019.

Notation

Symbol	Definition
$\mathcal{K} = \{1, \ldots, K\}$	The set of all labels
$x = (x_1, \ldots, x_n) \in \mathbb{R}^n$	n-dimensional input (n pixels of a image) x_i is the ith pixel in the image
$y = (y_1, \ldots, y_K)$	A vector of scores $y_k(x) \in [0,1], \sum_{k=1}^{K} y_k(x) = 1$
$Q(x) = y = (y_1(x), \ldots, y_k(x))$	Scoring function
$f(x) = \arg\max_{k \in \mathcal{K}} y_k(x)$	Prediction procedure
$\alpha = (\alpha_1, \ldots, \alpha_n)$	Perturbation (or called the change in the input) α_i is the change to the ith pixel in the image
$x + \alpha$	Adversarial example
$\lvert\lvert\alpha\rvert\rvert_p = \lvert\lvert(\alpha_1, \ldots, \alpha_n)\rvert\rvert_p$	p-norm of the perturbation (change) for $1 \leq p < \infty$, $\lvert\lvert\alpha\rvert\rvert_p = (\sum_{i=1}^{n} \lvert\alpha_i\rvert^p)^{1/p}$ for $p = \infty$, $\lvert\lvert\alpha\rvert\rvert_p = \max \lvert\alpha_i\rvert$ for $p = 0$, $\lvert\lvert\alpha\rvert\rvert_p = \lvert\{i: \alpha_i \neq 0\}\rvert$
$B_p(r) \coloneqq \{\alpha \in \mathbb{R}^n: \lvert\lvert\alpha\rvert\rvert_p \leq r\}$	p-norm ball of radius r
L	Radius of the $\alpha \in B_p(L)$ where α is such that $f(x + \alpha) \neq f(x)$

Lecuyer, Mathias. Certified robustness to adversarial examples with differential privacy. IEEE S&P 2019.

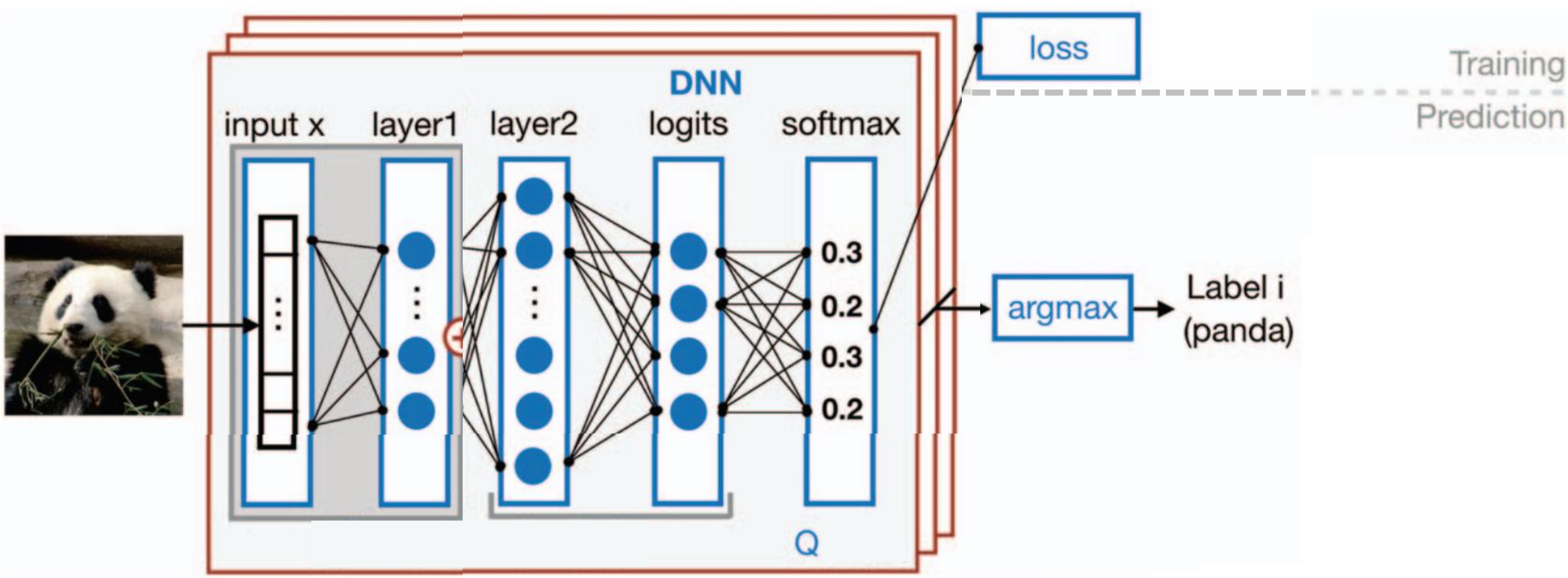
PixelDP classifier

- PixelDP DNN

➢ Deterministic scoring function Q: $x = (x_1, \dots, x_n) \in \mathbb{R}^n \to y = (y_1, \dots, y_K)$

$Q(x) = \left(y_1(x), \dots, y_K(x)\right)$

➢ The vulnerability of DNN to adversarial example $(x + \alpha)$ stems from the unbounded sensitivity of Q with respect to l_p changes in x.

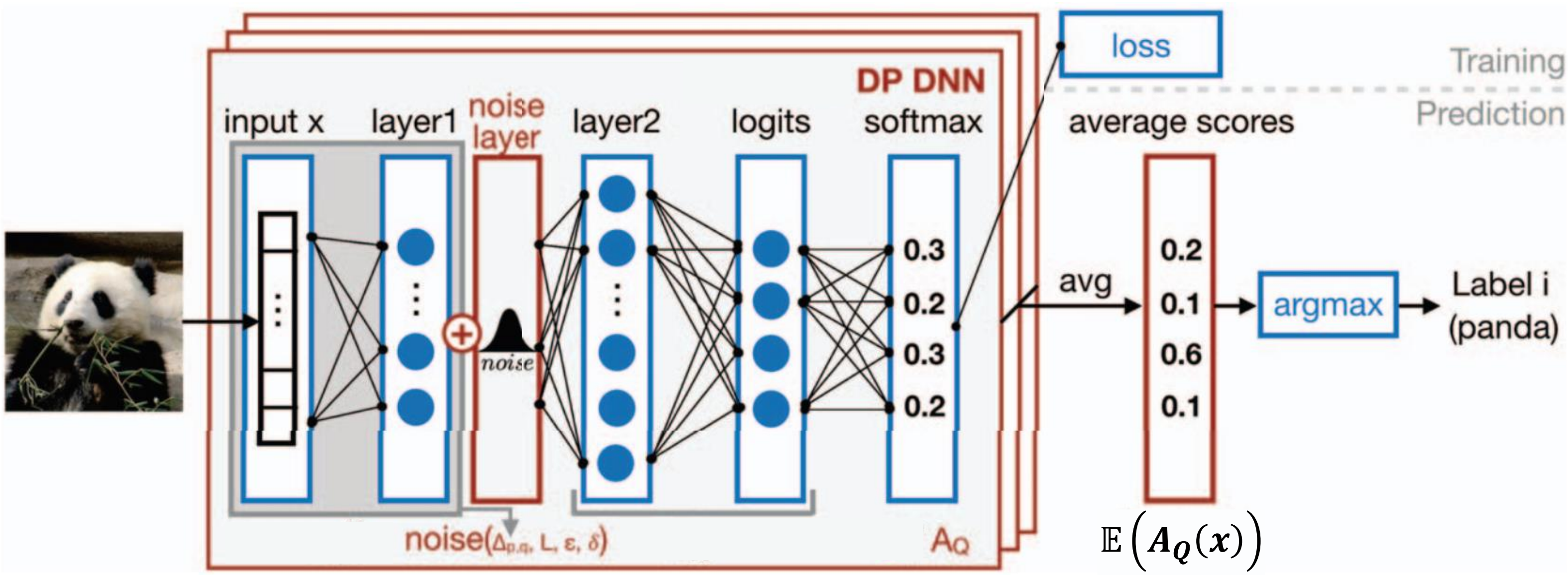


PixelDP classifier

- PixelDP DNN

➢ Randomized scoring function of the network that satisfy (ϵ, δ)-PixelDP: $A_Q(x) = \left(y_1(x), \dots, y_K(x)\right)$

➢ The expected output $\mathbb{E}\left(A_Q(x)\right)$ of $A_Q(x)$ will have bounded sensitivity to l_p changes in x.

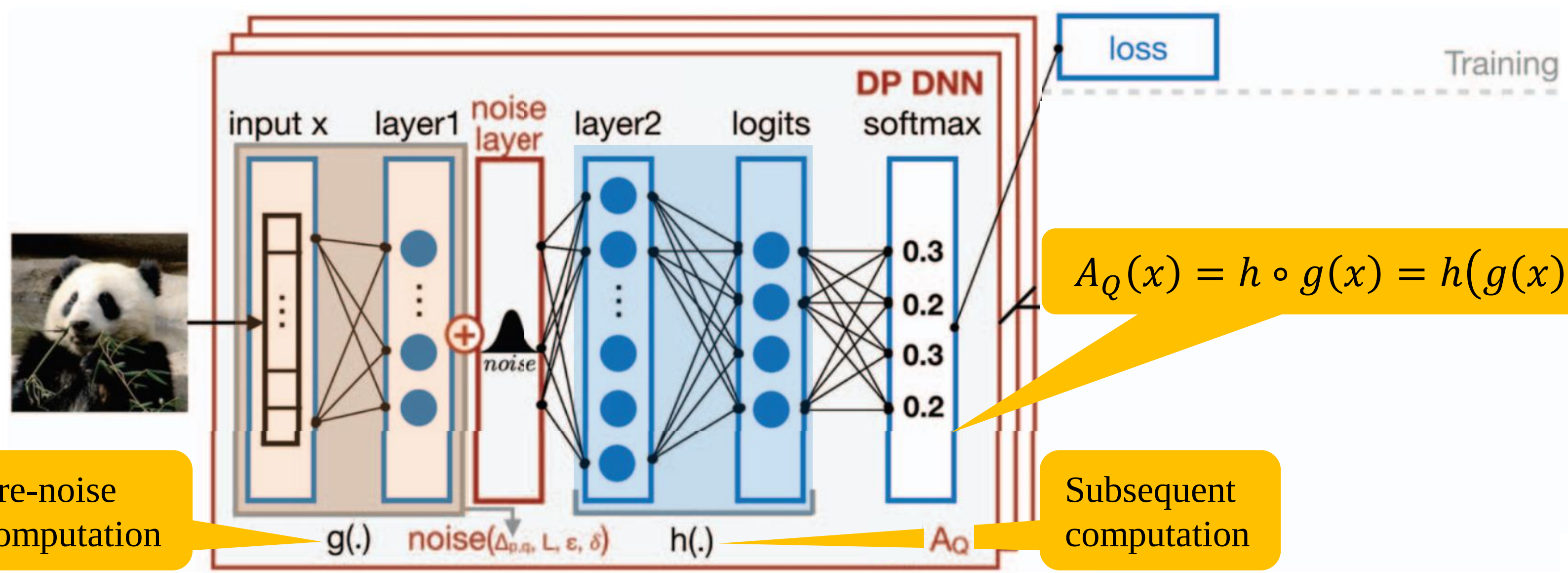


Lecuyer, Mathias. Certified robustness to adversarial examples with differential privacy. IEEE S&P 2019.

PixelDP classifier

- Training Procedure

➢ **Step 1** Transform g into another function $\tilde{g}$ that has a fixed sensitivity $\Delta \leq 1$ to l_p changes in x.



$A_Q(x) = h \circ g(x) = h(g(x))$

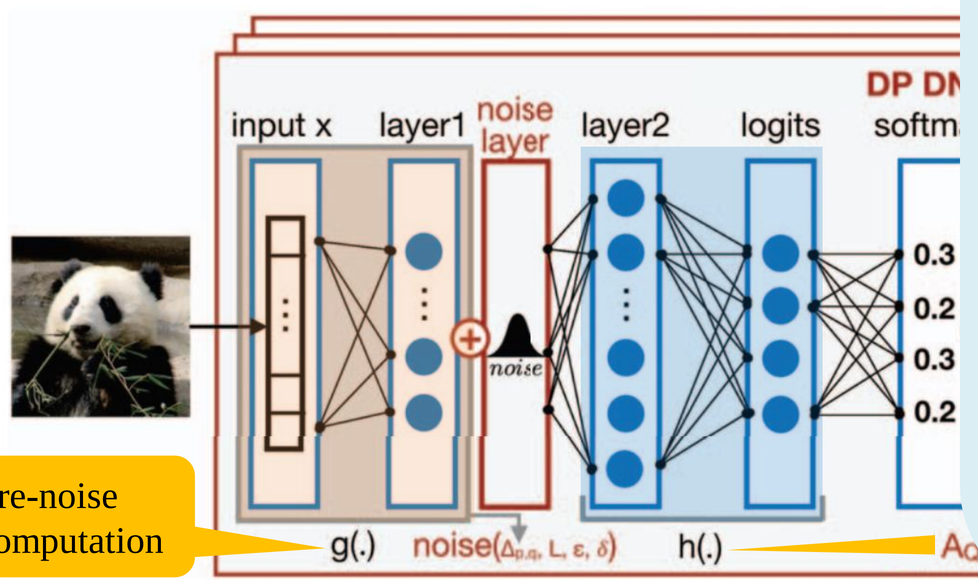
Pre-noise computation

Subsequent computation

PixelDP classifier

- **Training Procedure**

➢ **Step 1** Transform g into another function $\tilde{g}$ that has a fixed sensitivity $\Delta \leq 1$ to l_p changes in x.



Pre-noise computation

✓ Sensitivity : the maximum change in output that can be produced by a change in the input.

$$\Delta_{p,q} = \max_{Sx,x':x\neq x'} \frac{||g(x) - g(x')||_q}{||x - x'||_p}$$

✓ Transform Reason:

Training procedure will enlarge the sensitivity $\Delta_{p,q}$ of the g, voiding the DP guarantees.

✓ Transform Purpose:

Keep g's sensitivity $\Delta_{p,q}$ constant (eg. $\Delta \leq 1$) during training.

✓ Transform Ways:

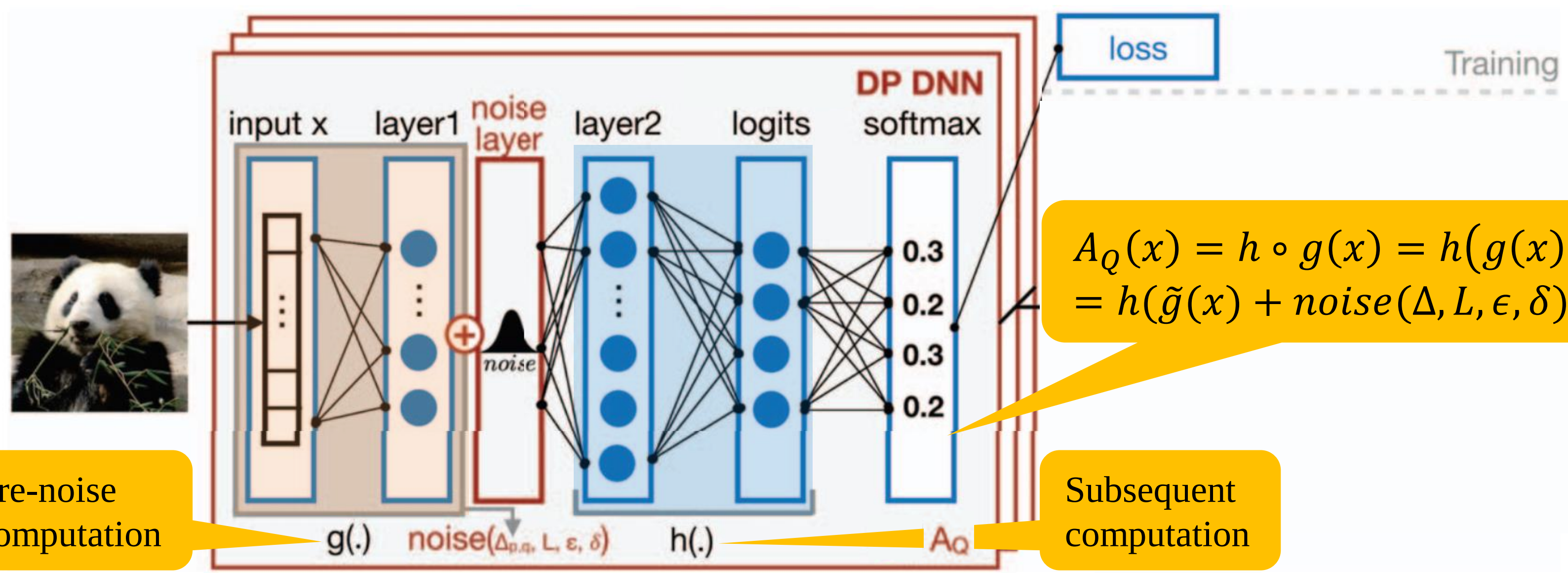
- For $\Delta_{1,1}, \Delta_{1,2}, \Delta_{\infty,\infty}$: SGD.
- For $\Delta_{2,2}$: projection after SGD.

PixelDP classifier

- Training Procedure

➢ **Step 1** Transform g into another function $\tilde{g}$ that has a fixed sensitivity $\Delta \leq 1$ to l_p changes in x.

➢ **Step 2** Add the noise layer to the output of $\tilde{g}$ with a standard deviation scaled by Δ and L to ensure (ϵ, δ)-PixelDP for l_p changes of size L.



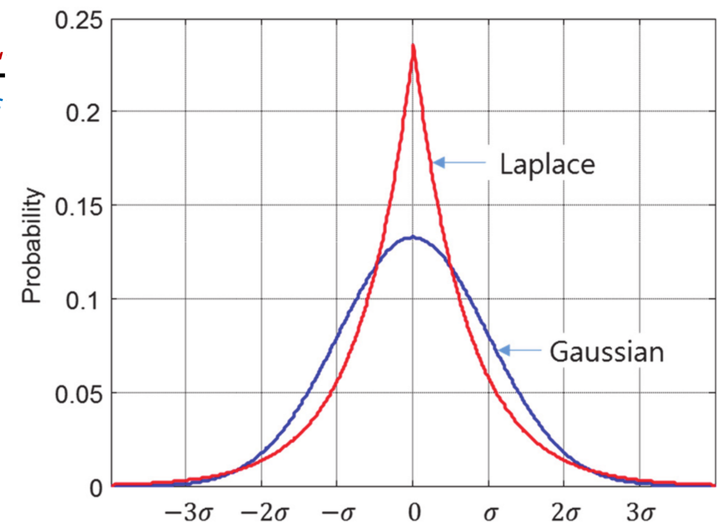
$$A_Q(x) = h \circ g(x) = h\big(g(x)\big) = h\big(\tilde{g}(x) + noise(\Delta, L, \epsilon, \delta)\big)$$

Pre-noise computation

Subsequent computation

Architecture

- Sample a noise sample Z from noise distribution $noise(\Delta, L, \epsilon, \delta)$
- Mean: μ=0
- Standard deviation: σ (b) is proportional to L and $\Delta_{p,q}$.
- If Gaussian :
  - PDF of $N(\mu, \sigma^2)$: $f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}$
  - $Z \sim N(\mu, \sigma^2) = N(0, \sqrt{2\ln\left(\frac{1.25}{\delta}\right)} \cdot \Delta_{p,2} \cdot \frac{L}{\epsilon}$
- If Laplace :
  - PDF of $L(\mu, b)$: $f(x) = \frac{1}{2b} \cdot e^{-\frac{|x-\mu|}{b}}$
  - $Z \sim L(\mu, b) = L(0, \sqrt{2} \cdot \Delta_{p,1} \cdot \frac{L}{\epsilon})$

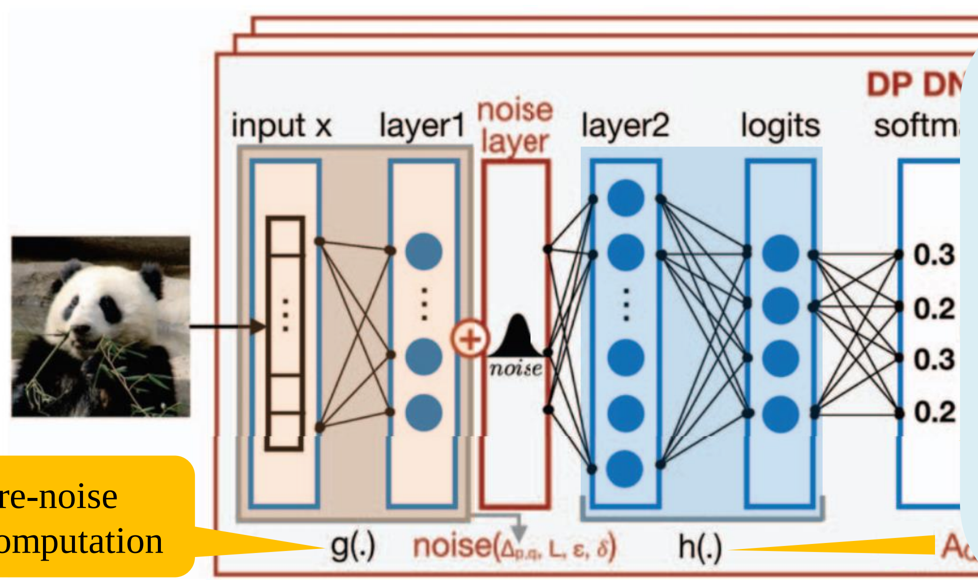


PixelDP classifier

- Training Procedure

➢ **Step 1** Transform g into another function $\tilde{g}$ that has a fixed sensitivity $\Delta \leq 1$ to l_p changes in x.

➢ **Step 2** Add the noise layer to the output of $\tilde{g}$ with a standard deviation scaled by Δ and L to ensure (ϵ, δ)-PixelDP for l_p changes of size L.



Pre-noise computation

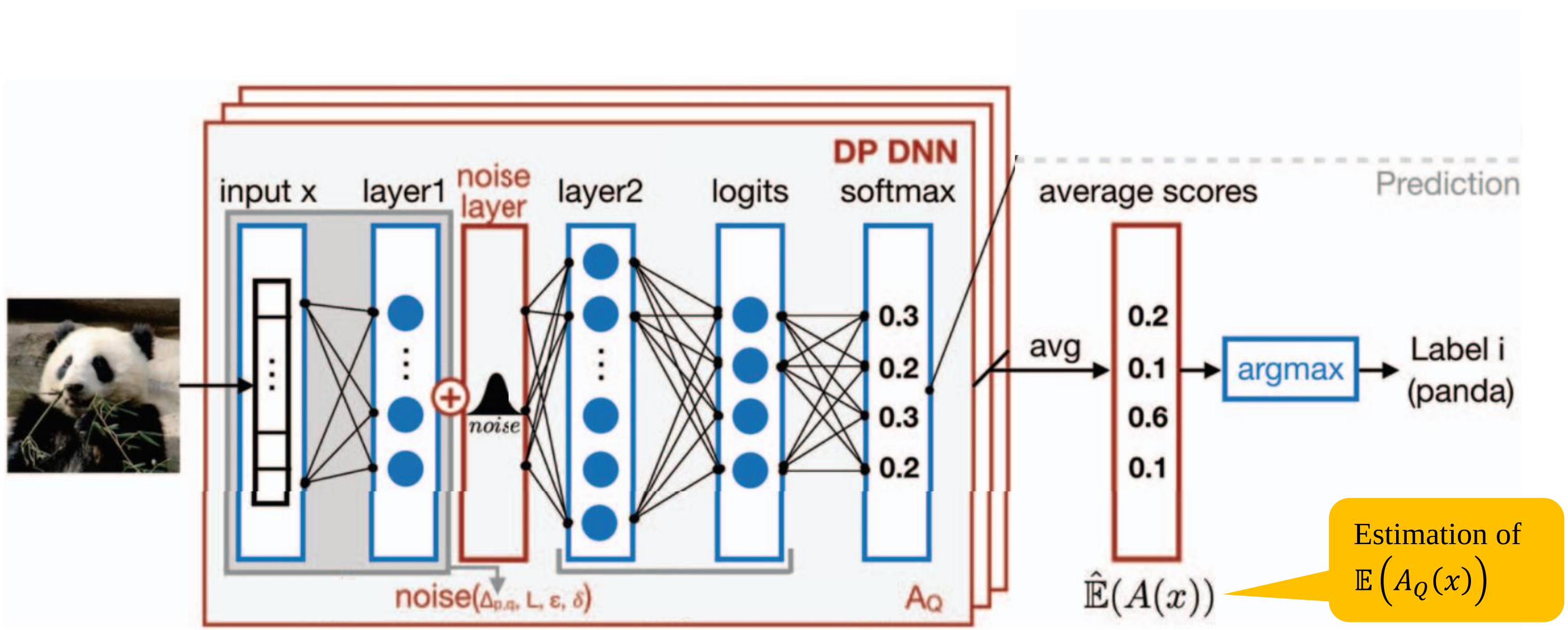
- ✓ Set L, ϵ, δ
- ✓ Compute fixed sensity Δ
- ✓ Input sample $(x, y_{true} = c_k)$
- ✓ Sample a noise sample Z from noise distribution $noise(\Delta, L, \epsilon, \delta)$
- • $Gaussian(\Delta, L, \epsilon, \delta) \rightarrow \epsilon$-DP
- • $Laplace(\Delta, L, \epsilon, \delta) \rightarrow (\epsilon, \delta)$-DP
- ✓ Optimization

$$\min_{\theta 1, \theta 2} \boldsymbol{Loss}(h_{\theta 2}(\tilde{g}_{\theta 1}(x) + noise(\Delta, L, \epsilon, \delta), y_{true})$$

PixelDP classifier

- Prediction Procedure

✓ Prediction on the $A_Q(x)$ affords the robustness certification in **Proposition 1** if the prediction procedure uses $\mathbb{E}\left(A_Q(x)\right)$.



PixelDP classifier

- Proposition 1

  If randomized algorithm A satisfies (ϵ,δ)-PixelDP to l_p 1 in x ,

  If for some $k \in K$, $\mathbb{E}\big(A_k(x)\big) > e^{2\epsilon} \max_{i:i\neq k} \mathbb{E}\big(A_i(x)\big) + (1+e^{\epsilon})\delta$

  Then, the classifier is robust to any attack $\alpha \in B_p(1)$ on input x.

➢ *Proof*

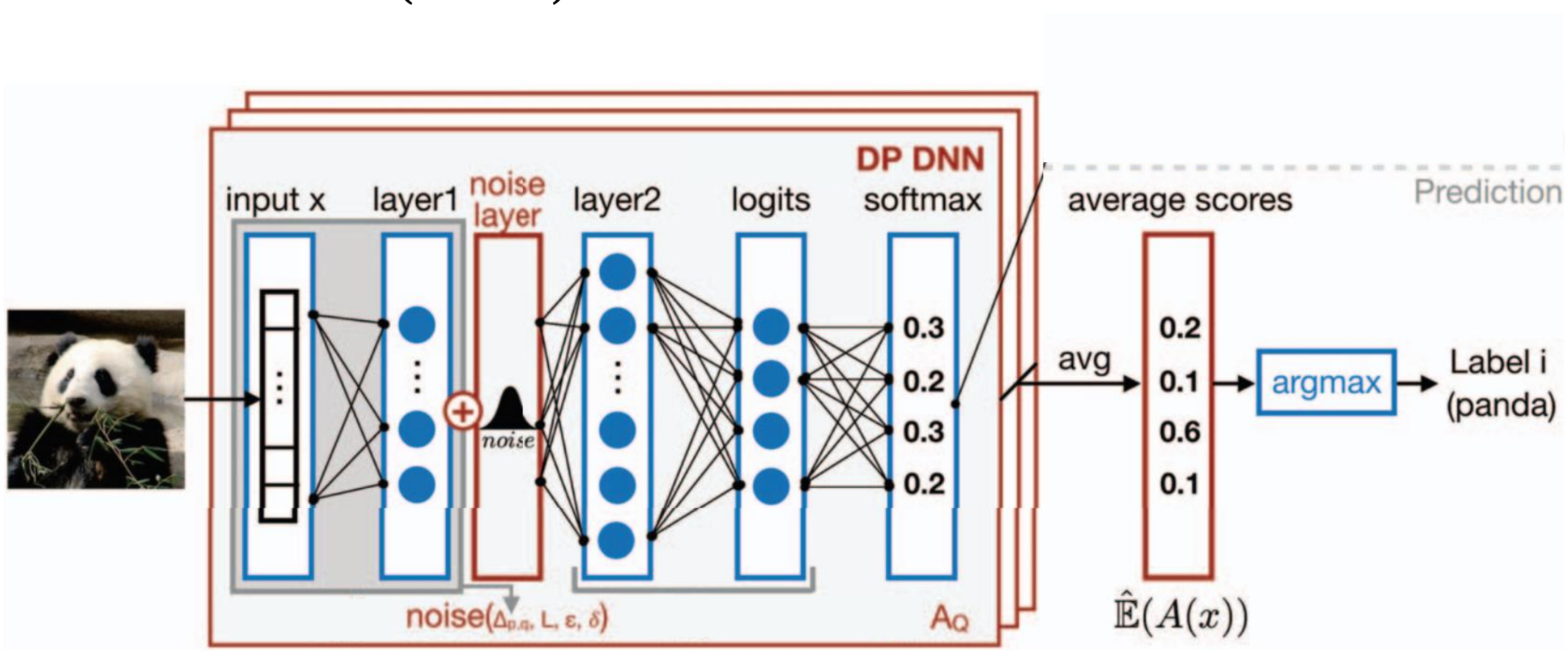
1. $\mathbb{E}\big(A_k(x)\big) \le e^{\epsilon}\mathbb{E}\big(A_k(x')\big) + \delta$ — (ϵ,δ)-DP定义
2. $\mathbb{E}\big(A_k(x')\big) \ge \frac{\mathbb{E}(A_k(x))-\delta}{e^{\epsilon}}$ — 移项
3. $\mathbb{E}\big(A_k(x)\big) > e^{2\epsilon} \max_{i:i\neq k} \mathbb{E}\big(A_i(x)\big) + (1+e^{\epsilon})\delta$ — 假设条件
4. $\mathbb{E}\big(A_k(x')\big) \ge \frac{e^{2\epsilon} \max_{i:i\neq k} \mathbb{E}(A_i(x)) + (1+e^{\epsilon})\delta - \delta}{e^{\epsilon}} = e^{\epsilon} \max_{i:i\neq k} \mathbb{E}\big(A_i(x)\big) + \delta$ — 代入第二行
5. $\mathbb{E}\big(A_i(x')\big) \le e^{\epsilon}\mathbb{E}\big(A_i(x)\big) + \delta, i \neq k$ — (ϵ,δ)-DP定义
6. $\max_{i:i\neq k} \mathbb{E}\big(A_i(x')\big) \le e^{\epsilon} \max_{i:i\neq k} \mathbb{E}\big(A_i(x)\big) + \delta, i \neq k$ — 两边求最大值
7. $\mathbb{E}\big(A_k(x')\big) \ge \max_{i:i\neq k} \mathbb{E}\big(A_i(x')\big)$ — 代入第四行

PixelDP classifier

- Prediction Procedure

✓ Prediction on the $A_Q(x)$ affords the robustness certification in **Proposition 1** if the prediction procedure uses $\mathbb{E}\left(A_Q(x)\right)$.

✓ Unfortunately, $\mathbb{E}\left(A_Q(x)\right)$ cannot be computed exactly.

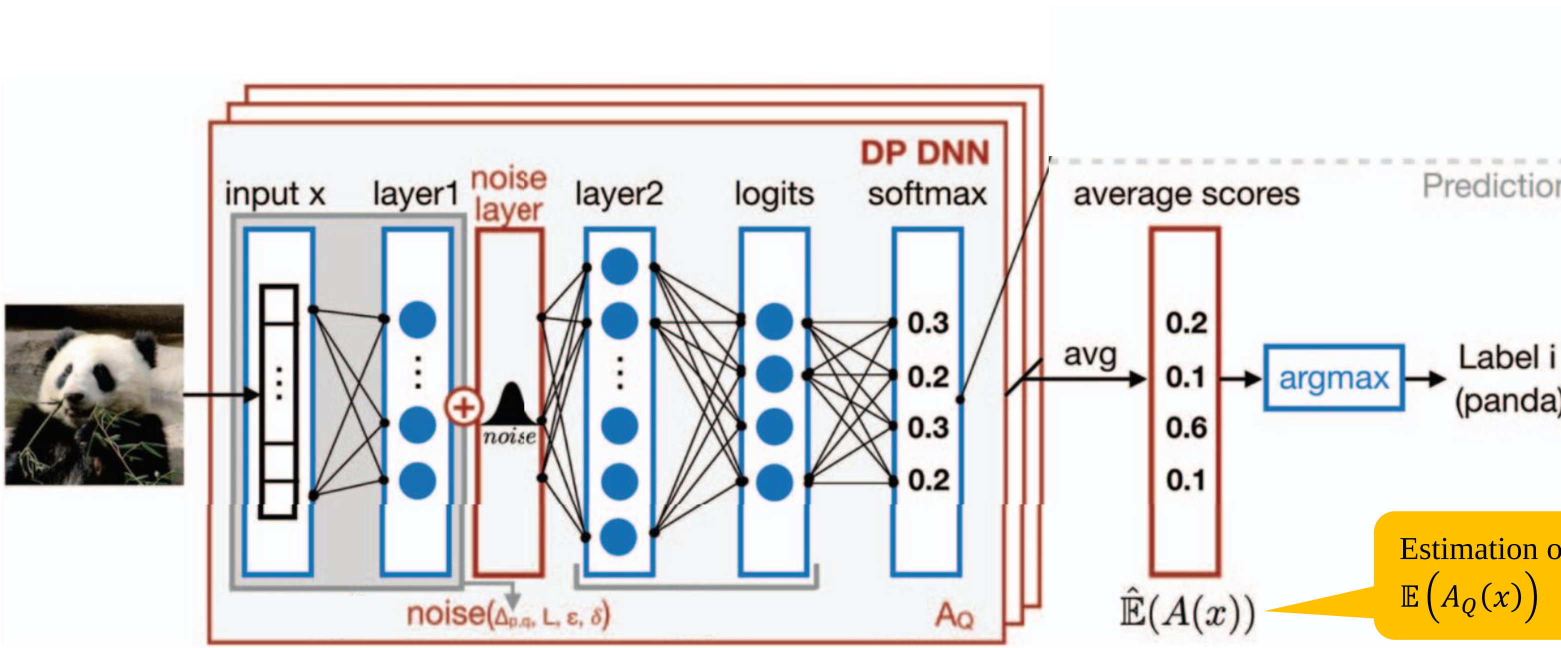


Lecuyer, Mathias. Certified robustness to adversarial examples with differential privacy. IEEE S&P 2019.

PixelDP classifier

- Prediction Procedure

➤ **Step 1** Use Monte Carlo methods to estimate approximate value $\hat{\mathbb{E}}\left(A_Q(x)\right)$ of $\mathbb{E}\left(A_Q(x)\right)$.



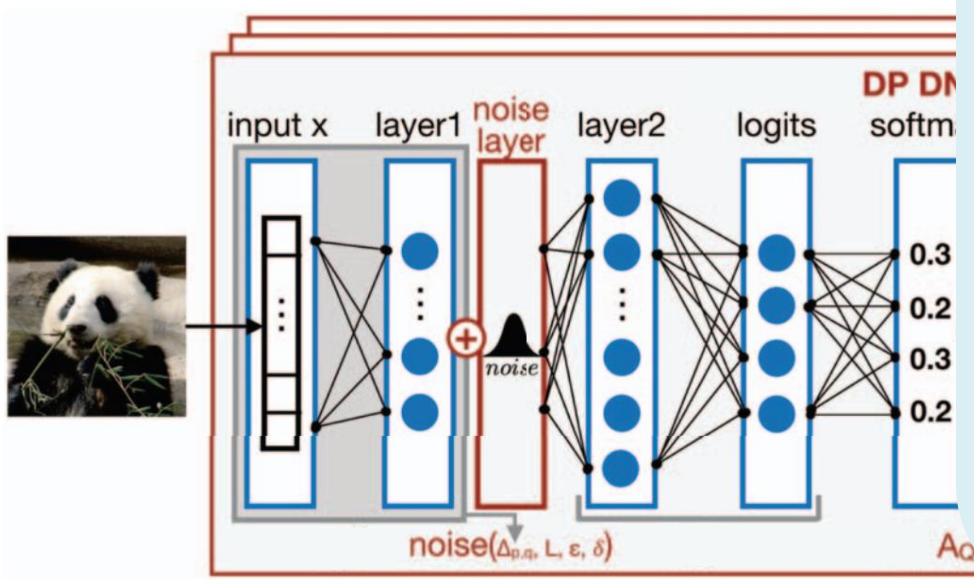
Estimation of $\mathbb{E}\left(A_Q(x)\right)$

Lecuyer, Mathias. Certified robustness to adversarial examples with differential privacy. IEEE S&P 2019.

PixelDP classifier

- **Prediction Procedure**

➢ **Step 1** Use Monte Carlo methods to estimate approximate value $\hat{\mathbb{E}}\left(A_Q(x)\right)$ of $\mathbb{E}\left(A_Q(x)\right)$.



✓ Invoke $A_Q(x)$ n times with independent draws in the noise layer.

✓ For $i = 1$ to n

- Input sample x without label
- Sample ith noise sample from noise distribution $noise^{i}(\Delta, L, \epsilon, \delta)$
- Score: $A_Q^i(x) = h(\tilde{g}(x) + noise^{i}(\Delta, L, \epsilon, \delta))$
- The ith draw from the distribution of the randomized function A_Q on the kth label: $A_{Q,k}^i(x)$
- $\hat{\mathbb{E}}\left(A_{Q,k}(x)\right) = \frac{1}{n}\sum_{i=1}^{n} A_{Q,k}^i(x)$

Lecuyer, Mathias. Certified robustness to adversarial examples with differential privacy. IEEE S&P 2019.

PixelDP classifier

- Proposition 1

  If randomized algorithm A satisfies (ϵ, δ)-PixelDP to l_p of size 1 in x ,

  If for some $k \in K$, $\mathbb{E}\big(A_k(x)\big) > e^{2\epsilon} \max_{i:i \neq k} \mathbb{E}\big(A_i(x)\big) + (1 + e^{\epsilon})\delta$

  Then, the classifier is robust to any attack $\alpha \in B_p(1)$ on input x.

- Proposition 2

  If randomized algorithm A satisfies (ϵ, δ)-PixelDP to l_p of size L in x ,

  Let $\hat{\mathbb{E}}^{ub} A_i(x)$ and $\hat{\mathbb{E}}^{lb} A_i(x)$ be the η-confidence upper and lower bound for $\hat{\mathbb{E}}(A_i(x))$.

  If for some $k \in K$, $\hat{\mathbb{E}}^{lb}\big(A_k(x)\big) > e^{2\epsilon} \max_{i:i \neq k} \hat{\mathbb{E}}^{ub}\big(A_i(x)\big) + (1 + e^{\epsilon})\delta)$

  Then, the classifier is robust to any attack $\alpha \in B_p(L)$ on input x with probability $\geq \eta$.

PixelDP classifier

- Proposition 2

If for some $k \in K$, $\hat{\mathbb{E}}^{lb}\big(A_k(x)\big) > e^{2\epsilon} \max_{i:i\neq k} \hat{\mathbb{E}}^{ub}\big(A_i(x)\big) + (1+e^{\epsilon})\delta)$

Then, the classifier is robust to $\alpha \in B_p(L)$ on x with probability $\geq \eta$.

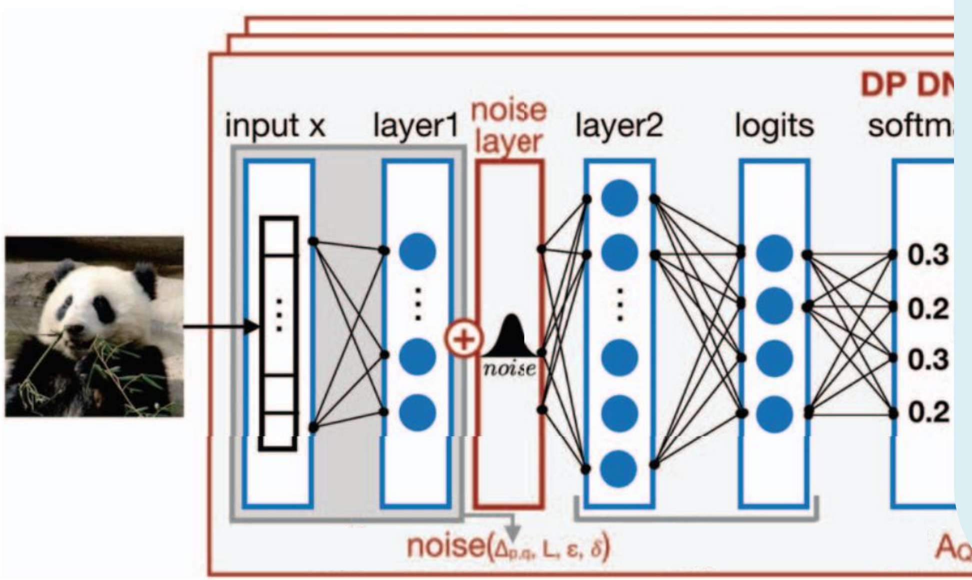
➢ *Proof*

1. $\hat{\mathbb{E}}\big(A_k(x)\big) \leq e^{\epsilon}\hat{\mathbb{E}}\big(A_k(x')\big) + \delta$ (ϵ,δ)-DP定义
2. $\hat{\mathbb{E}}\big(A_k(x')\big) \geq \frac{\hat{\mathbb{E}}(A_k(x))-\delta}{e^{\epsilon}} \geq \frac{\hat{\mathbb{E}}^{lb}(A_k(x))-\delta}{e^{\epsilon}}$ 移项，取下届
3. $\hat{\mathbb{E}}^{lb}\big(A_k(x)\big) > e^{2\epsilon} \max_{i:i\neq k} \hat{\mathbb{E}}^{ub}\big(A_i(x)\big) + (1+e^{\epsilon})\delta)$ 假设条件
4. $\hat{\mathbb{E}}\big(A_k(x')\big) \geq \frac{\hat{\mathbb{E}}^{lb}(A_k(x))-\delta}{e^{\epsilon}} > \frac{e^{2\epsilon}\max_{i:i\neq k}\hat{\mathbb{E}}^{ub}(A_i(x))+(1+e^{\epsilon})\delta)\ -\delta}{e^{\epsilon}} =$ 代入第二行
$e^{\epsilon} \max_{i:i\neq k} \hat{\mathbb{E}}^{ub}\big(A_i(x)\big) + \delta$
5. $\hat{\mathbb{E}}\big(A_{i:i\neq k}(x')\big) \leq e^{\epsilon} \max_{i:i\neq k} \hat{\mathbb{E}}\big(A_i(x)\big) + \delta$ (ϵ,δ)-DP定义
6. $\max \hat{\mathbb{E}}\big(A_{i:i\neq k}(x')\big) \leq e^{\epsilon} \max_{i:i\neq k} \hat{\mathbb{E}}\big(A_i(x)\big) + \delta \leq e^{\epsilon} \max_{i:i\neq k} \hat{\mathbb{E}}^{ub}\big(A_i(x)\big) + \delta$ 最大值
7. $\hat{\mathbb{E}}\big(A_k(x')\big) > \max \hat{\mathbb{E}}\big(A_{i:i\neq k}(x')\big)$ 代入第四行

PixelDP classifier

- Prediction Procedure

➢ **Step 1** Use Monte Carlo methods to estimate approximate value $\hat{\mathbb{E}}\left(A_Q(x)\right)$ of $\mathbb{E}\left(A_Q(x)\right)$.



✓ Compute a interval $[\hat{\mathbb{E}}^{lb}\left(A_Q(x)\right), \hat{\mathbb{E}}^{ub}\left(A_Q(x)\right)]$ for $\hat{\mathbb{E}}\left(A_Q(x)\right)$ holds with probability η.

- Use Hoeffding's inequality (霍夫丁不等式) to bound error in $\hat{\mathbb{E}}\left(A_Q(x)\right)$.

$$\hat{\mathbb{E}}\left(A_Q(x)\right) - \sqrt{\frac{1}{2n}\ln\left(\frac{2k}{1-n}\right)} \le \mathbb{E}\left(A_Q(x)\right) \le \hat{\mathbb{E}}\left(A_Q(x)\right) + \sqrt{\frac{1}{2n}\ln\left(\frac{2k}{1-n}\right)}$$

$$\hat{\mathbb{E}}^{lb}\left(A_Q(x)\right) \triangleq \hat{\mathbb{E}}\left(A_Q(x)\right) - \sqrt{\frac{1}{2n}\ln\left(\frac{2k}{1-n}\right)}$$

$$\hat{\mathbb{E}}^{ub}\left(A_Q(x)\right) \triangleq \hat{\mathbb{E}}\left(A_Q(x)\right) + \sqrt{\frac{1}{2n}\ln\left(\frac{2k}{1-n}\right)}$$

✓ Integrate this interval into the stability bound for $\mathbb{E}\left(A_Q(x)\right)$.

$$\mathbb{E}\left(A_Q(x)\right) \le e^{\epsilon}\mathbb{E}\left(A_Q(x+\alpha)\right) + b\delta$$

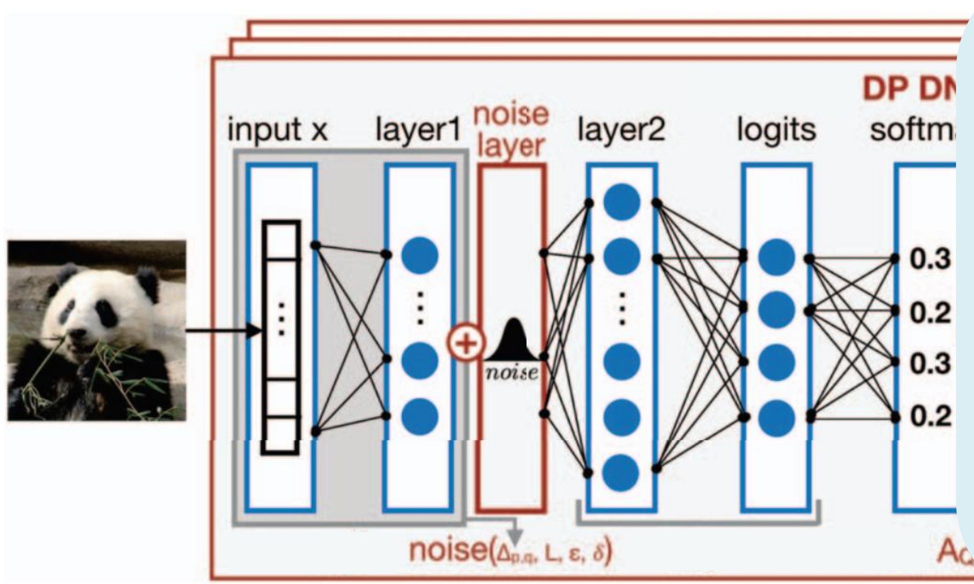
Lecuyer, Mathias. Certified robustness to adversarial examples with differential privacy. IEEE S&P 2019.

PixelDP classifier

- Prediction Procedure

➢ **Step 1** Use Monte Carlo methods to estimate approximate value $\hat{\mathbb{E}}\left(A_Q(x)\right)$ of $\mathbb{E}\left(A_Q(x)\right)$.

➢ **Step 2** PixelDP returns a prediction for x ($\arg\max \hat{\mathbb{E}}\left(A_Q(x)\right)$) and a *robustness size certificate* for that prediction.



✓ Obtain upper and lower bounds on the change to $\hat{\mathbb{E}}\left(A_{Q,i}(x)\right)$ with input change of size L with probability η.

✓ Compute *robustness size certificate*:

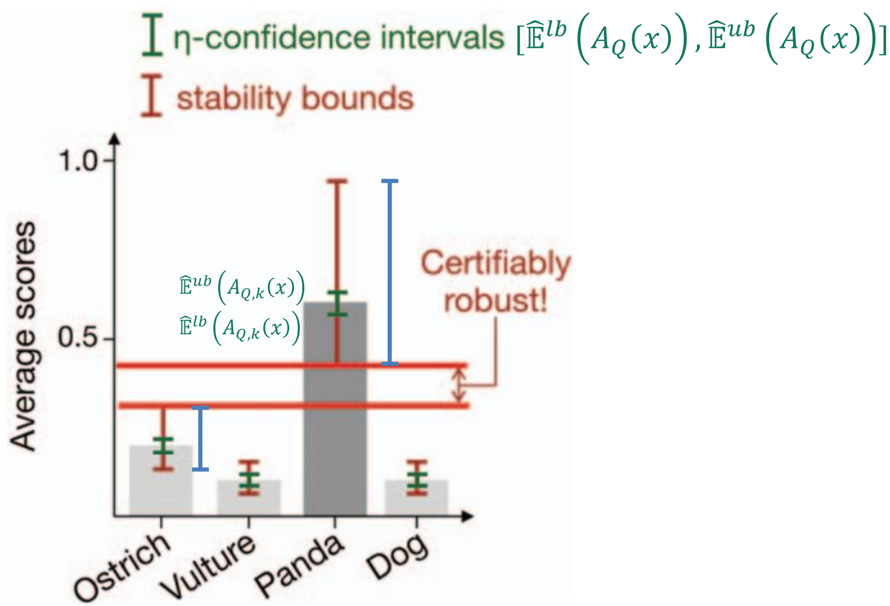
- If, $\hat{\mathbb{E}}^{lb}\left(A_{Q,k}(x)\right) > e^{2\epsilon} \max_{i:i\neq k} \hat{\mathbb{E}}^{ub}\left(A_{Q,i}(x)\right) + (1+e^{\epsilon})\delta$
- Then, the classifier is robust to any $\alpha \in B_p(L)$ around x with probability $\geq \eta$.
- Else, x not meet robustness check for L.

PixelDP classifier

- $\hat{\mathbb{E}}^{up}\left(A_{Q,i}(x)\right)$ and $\hat{\mathbb{E}}^{lp}\left(A_{Q,i}(x)\right)$

➢ If the lower bound for the label with the top average score $\hat{\mathbb{E}}^{lb}\left(A_{Q,k}(x)\right)$ is strictly greater than the upper bound for every other label $\max_{i:i\neq k} \hat{\mathbb{E}}^{ub}\left(A_{Q,i}(x)\right)$,

➢ Then, with probability η, the prediction for input x is robust to arbitrary attacks of l_p size L.



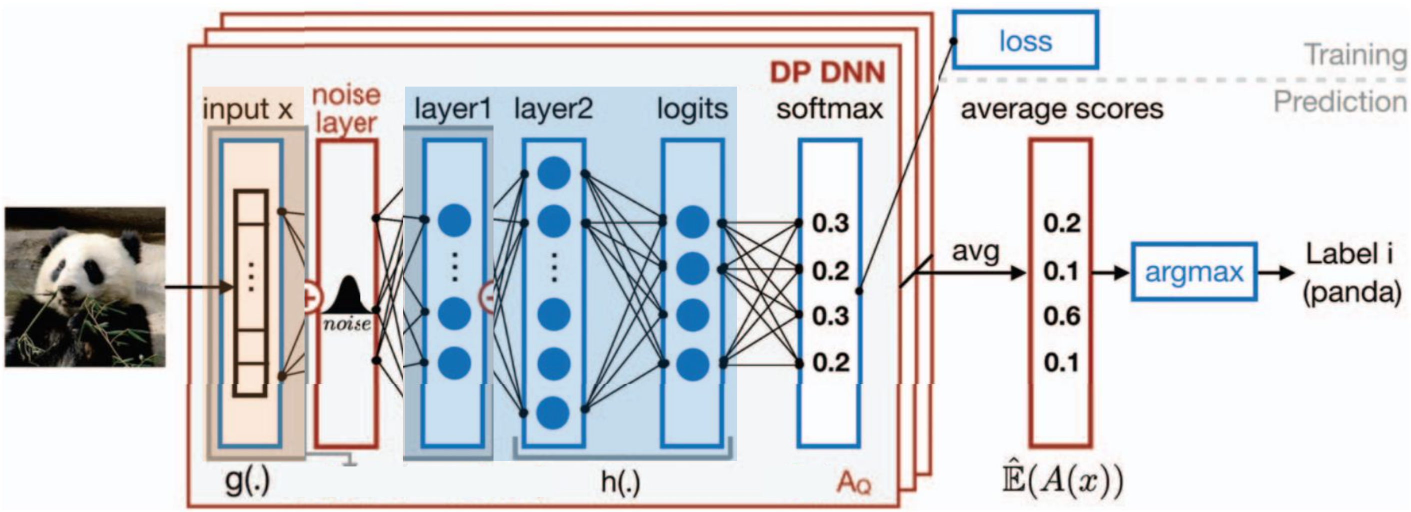
鸵鸟　秃鹫

PixelDP classifier

- **Noise layer place**

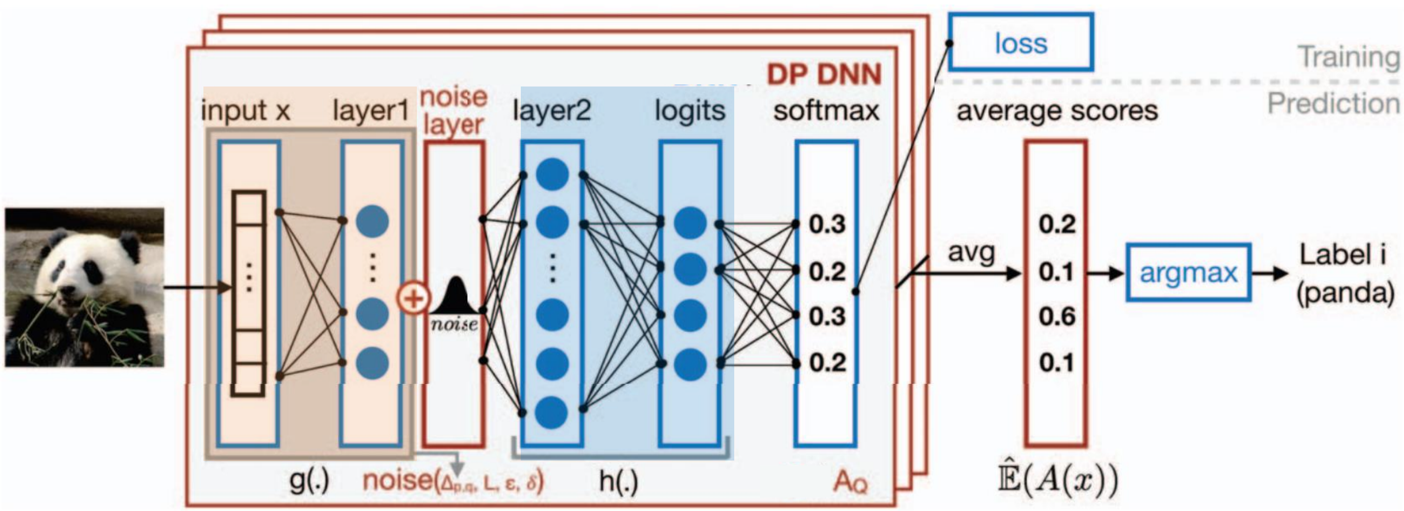
➢ Noise in the image

- $\Delta_{p,q} = \Delta_{1,1}^{g} = \max_{x,x':x\neq x'} \frac{\|g(x)-g(x')\|_q}{\|x-x'\|_p} = \max_{x,x':x\neq x'} \frac{\|g(x)-g(x')\|_1}{\|x-x'\|_1}$
- $\Delta_{p,q} = \Delta_{2,2}^{g} = \max_{x,x':x\neq x'} \frac{\|g(x)-g(x')\|_q}{\|x-x'\|_p} = \max_{x,x':x\neq x'} \frac{\|g(x)-g(x')\|_2}{\|x-x'\|_2}$



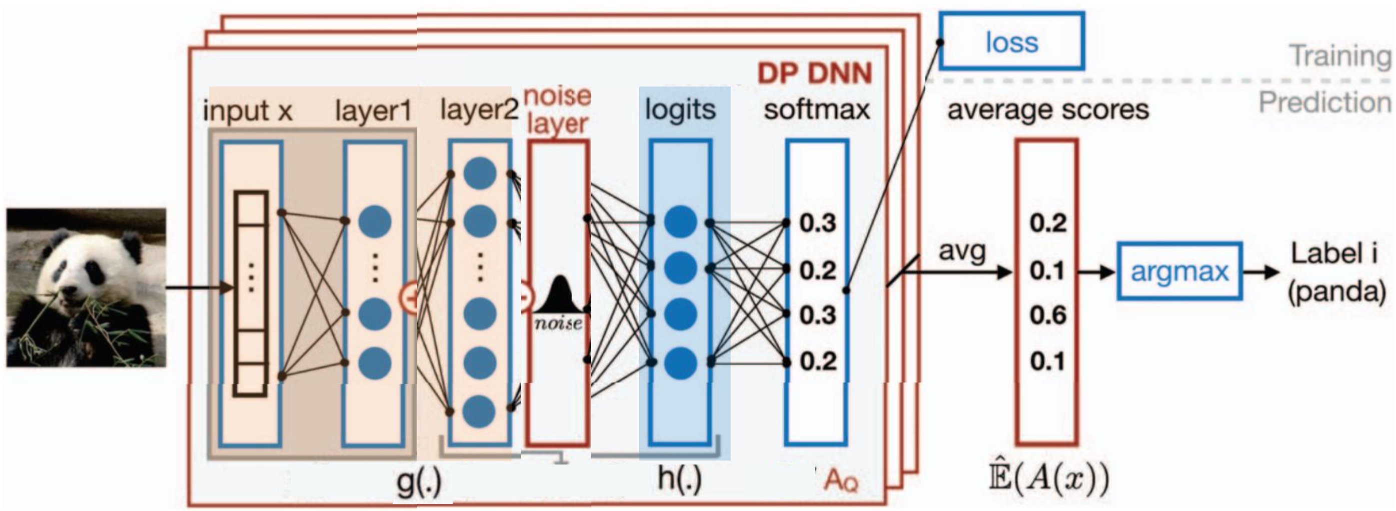
PixelDP classifier

- Noise layer place
- ➢ Noise in the image
- ➢ Noise after first hidden layer
- $g(x) = f_1(x)$
- $\Delta_{p,q} = \Delta_{p,q}^{g} = \Delta_{p,q}^{f_1}$



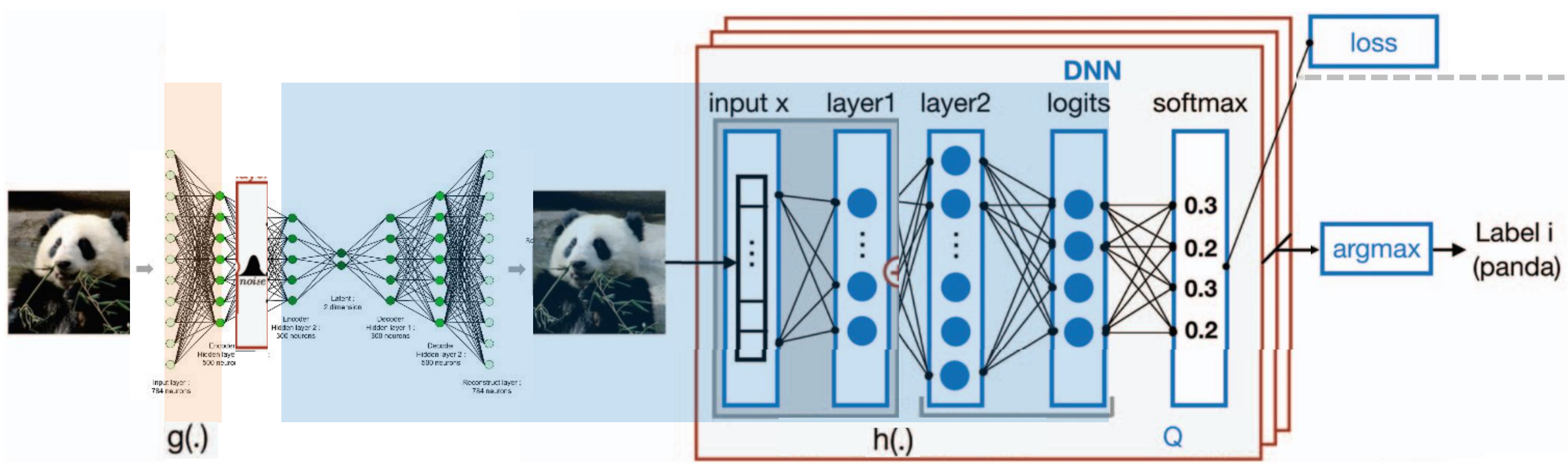
PixelDP classifier

- Noise layer place
  - Noise in the image
  - Noise after first hidden layer
  - Noise after deeper hidden layer
- $g(x) = f_1(f_2(x))$
- $\Delta_{p,q} = \Delta_{p,q}^{g} = \Delta_{p,q}^{(f_1 \circ f_2)}$



PixelDP classifier

- Noise layer place
  - Noise in the image
  - Noise after first hidden layer
  - Noise after deeper hidden layer
  - Noise in Auto-encoder
- Auto-encoders are smaller than DNN, much faster to train. (ImageNet)



Lecuyer, Mathias. Certified robustness to adversarial examples with differential privacy. IEEE S&P 2019.

Contribution

- Establish a connection between adversarial robustness and differential privacy.
- Develop the first certified defense that scales to large networks (Google's Inception network) and datasets (ImageNet).
- Develop the certified defense that applies broadly to arbitrary model types.

➢ Datasets:

- ImageNet,
- CIFAR10, CIFAR100, SVHN
- MNIST

➢ Networks:

- Inception
- ResNet

Lecuyer, Mathias. Certified robustness to adversarial examples with differential privacy. IEEE S&P 2019.

Authors

Mathias Lecuyer
PhD
Columbia University

Vaggelis Atlidakis
PhD
Columbia University

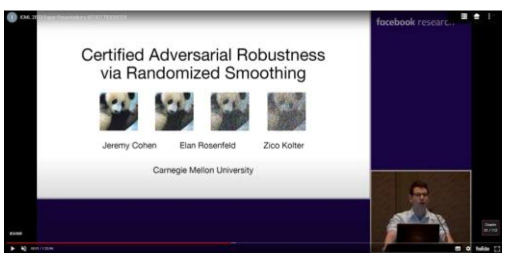
Roxana Geambasu
Associate Professor
Columbia University

Daniel Hsu
associate professor
Columbia University

Suman Jana
associate professor
Columbia University

Lecuyer, Mathias. Certified robustness to adversarial examples with differential privacy. IEEE S&P 2019.

Certified Adversarial Robustness via Randomized Smoothing

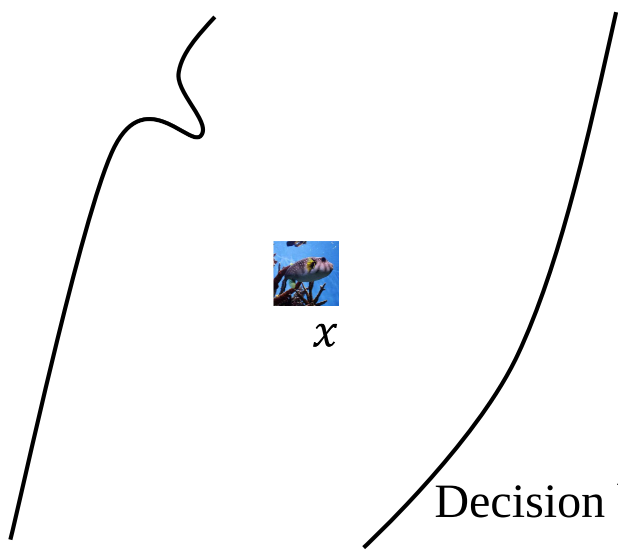


Jeremy Cohen, Elan Rosenfeld, J. Zico Kolter
Carnegie Mellon University
ICML 2019

Randomized Smoothing

Overview

- Applying Gaussian noise and taking majority class label.
- Essence: Smoothing the decision boundary of the base classifier f.
- Way: Sampling Gaussian noise to perturb x multiple times, then vote on the labels most frequently given by the base classifier f.
- Purpose: Output the majority class label as the prediction of x.

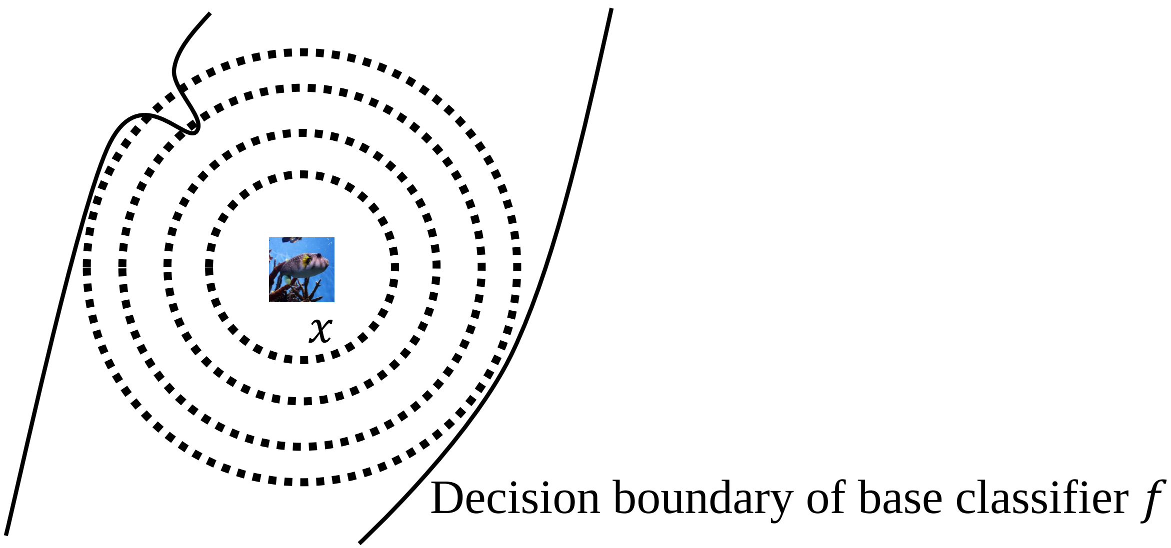


Decision boundary of base classifier f

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

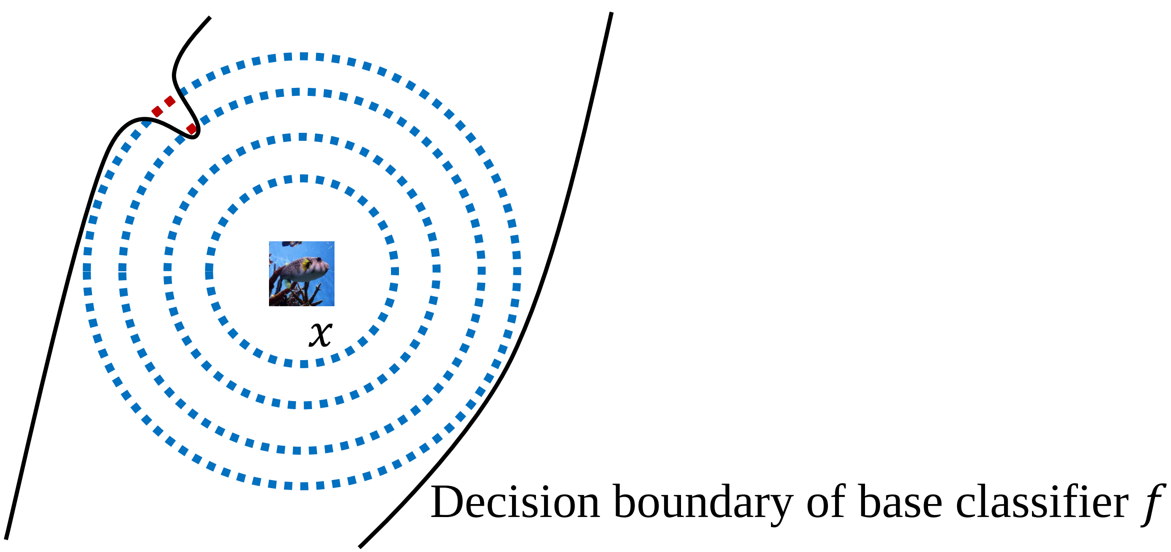
Overview

- Applying Gaussian noise and taking majority class label.
- ➢ Essence: Smoothing the decision boundary of the base classifier f.
- ➢ Way: Sampling Gaussian noise to perturb x multiple times, then vote on the labels most frequently given by the base classifier f.
- ➢ Purpose: Output the majority class label as the prediction of x.



Overview

- Applying Gaussian noise and taking majority class label.
- ➢ Essence: Smoothing the decision boundary of the base classifier f.
- ➢ Way: Sampling Gaussian noise to perturb x multiple times, then vote on the labels most frequently given by the base classifier f.
- ➢ Purpose: Output the majority class label as the prediction of x.



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

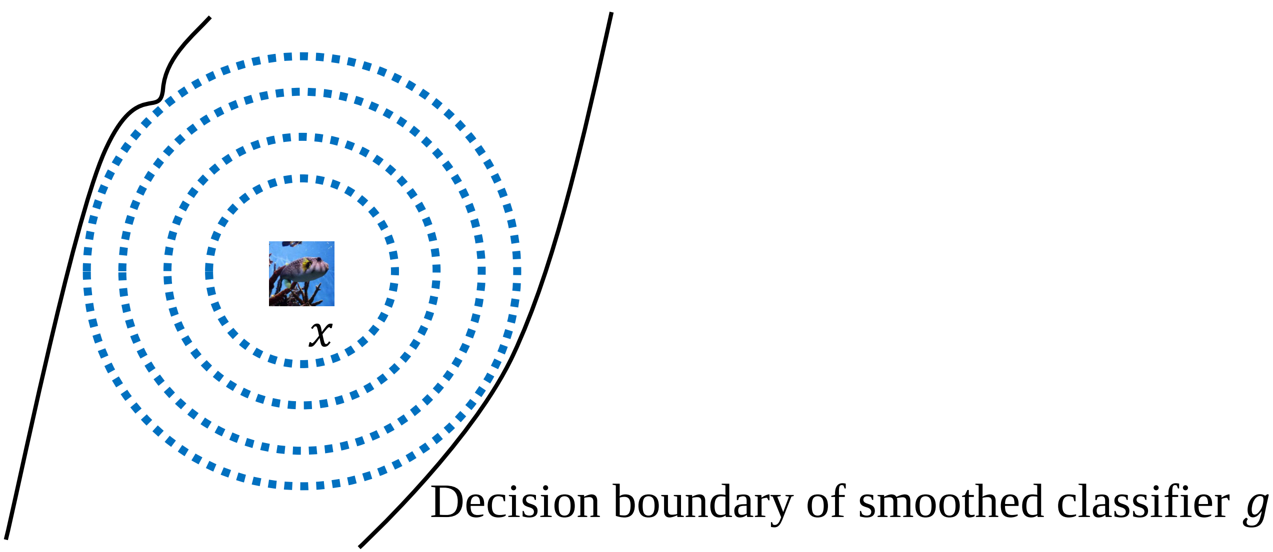
Overview

- Applying Gaussian noise and taking majority class label.

➢ Essence: Smoothing the decision boundary of the base classifier f.

➢ Way: Sampling Gaussian noise to perturb x multiple times, then vote on the labels most frequently given by the base classifier f.

➢ Purpose: Output the majority class label as the prediction of x.



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

Overview

- Evaluating the smoothed classifier at an input x.
  - Smoothed classifier g: a virtual classifier with smoother boundary than f
  - Here, $g(x) = C_A$.

Left:

- Colors: decision regions of f.
- Dotted lines: level sets of $N\ (x, \sigma^2 I)$.

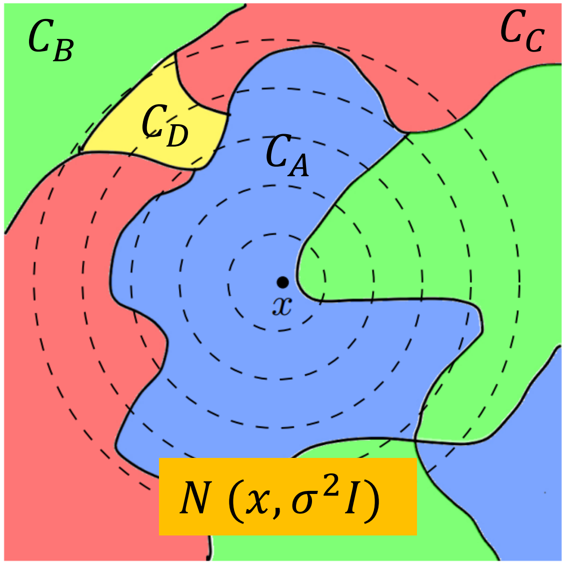


Figure 1

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

Randomized Smoothing

Problem

- Randomized smoothing-based heuristic defense
  - Method
    - Insert noise layer before each convolution layer. (Liu. ECCV 2018)
    - Ensemble information in a region centered at x to predict.(Cao. ACSAC2017)
  - Problem: did not prove any guarantees.
- Randomized smoothing-based certified defense
  - Methods
    - Smooth the classifier with noise and use DP inequalities to prove robustness guarantee. (Lecuyer. IEEE S&P 2019)
    - Use Renyi divergence to prove a stronger guarantee. (Li. arXiv2018)
  - Problem: existing robustness guarantees are loose.

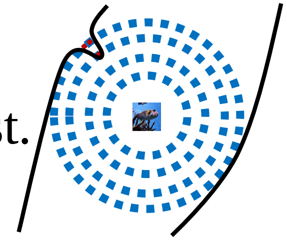
Liu, Xuanqing. Towards robust neural networks via random self-ensemble. ECCV 2018.
Cao, Xiaoyu. Mitigating evasion attacks to deep neural networks via region-based classification. ACSAC2017.
Lecuyer, Mathias. Certified robustness to adversarial examples with differential privacy. IEEE S&P 2019.
Li, Bai, et al. "Second-order adversarial attack and certifiable robustness." arXiv2018.

Randomized Smoothing

Problem

- **Advantage of randomized smoothing**
- Makes no assumptions about classifier architecture.
- Permits to use arbitrarily large neural networks. (others do not support)
- The only certified defense been shown feasible on ImageNet task. (before this work)

- **Challenge of randomized smoothing**
- Not possible to exactly compute the probabilities with which f classifies $\mathcal{N}(x, \sigma^2 I)$ as each class.
- Not possible to exactly evaluate the smoothed classifier g
- Not possible to exactly compute the radius in which g is robust.



- **Solution**
- Use Monte Carlo method to evaluate prediction of the classifier around x.

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

Notation

Symbol	Definition
$\mathcal{Y}$	Classes set
g	Random smoothed classifier
$f: \mathbb{R}^d \to \mathcal{Y}$	Base classifier
$x \in \mathbb{R}^d$	Input space
$\varepsilon \sim N(0, \sigma^2 I)$	Isotropic Gaussian noise
$C_A \in \mathcal{Y}$	most probable class returned by $f(x+\varepsilon)$
$p_A \in [0,1]$	$\mathbb{P}(f(x+\epsilon) = C_A)$.
C_B	"runner-up" class returned by $f(x+\varepsilon)$
$p_B \in [0,1]$	$\mathbb{P}(f(x+\epsilon) = C_B)$.
$\underline{p_A} \in [0,1]$	lower bound of p_A
$\overline{p_B} \in [0,1]$	upper bound of p_B
$\delta = (\delta_1, \delta_2, \dots, \delta_n)$	n-dimensional perturbation
$\|\|\delta\|\|_2$	2 norm of the vector δ , $\sqrt{(\delta_1{}^2 + \delta_2{}^2 + \cdots + \delta_n{}^2)}$

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

Randomized Smoothing

Algorithm

- Smoothed Classifier g

➢ Definition:

Smoothed classifier g returns whichever class the base classifier f is most likely to return when x is perturbed by isotropic Gaussian noise:

$$g(x) = \arg\max_{c \in \mathcal{Y}} \mathbb{P}(f(x+\varepsilon) = c) \qquad (1)$$

Where $\varepsilon \sim N(0, \sigma^2)$

➢ Interpretation:

Randomized smoothing in high dimension is that these large random perturbations ε drown out small adversarial perturbations δ.

random Gaussian corruptions of x ($\sigma = 0.5$)

Algorithm

- Smoothed Classifier g

➢ Definition:

Smoothed classifier g returns whichever class the base classifier f is most likely to return when x is perturbed by isotropic Gaussian noise:

$$g(x) = \arg\max_{c \in \mathcal{Y}} \mathbb{P}(f(x+\varepsilon) = c) \qquad (1)$$

Where $\varepsilon \sim N(0, \sigma^2)$

- Base classifier f

➢ In order for g to classify the labeled example (x, c) correctly and robustly, f needs to consistently classify $N(x, \sigma^2 I)$ as c at training.

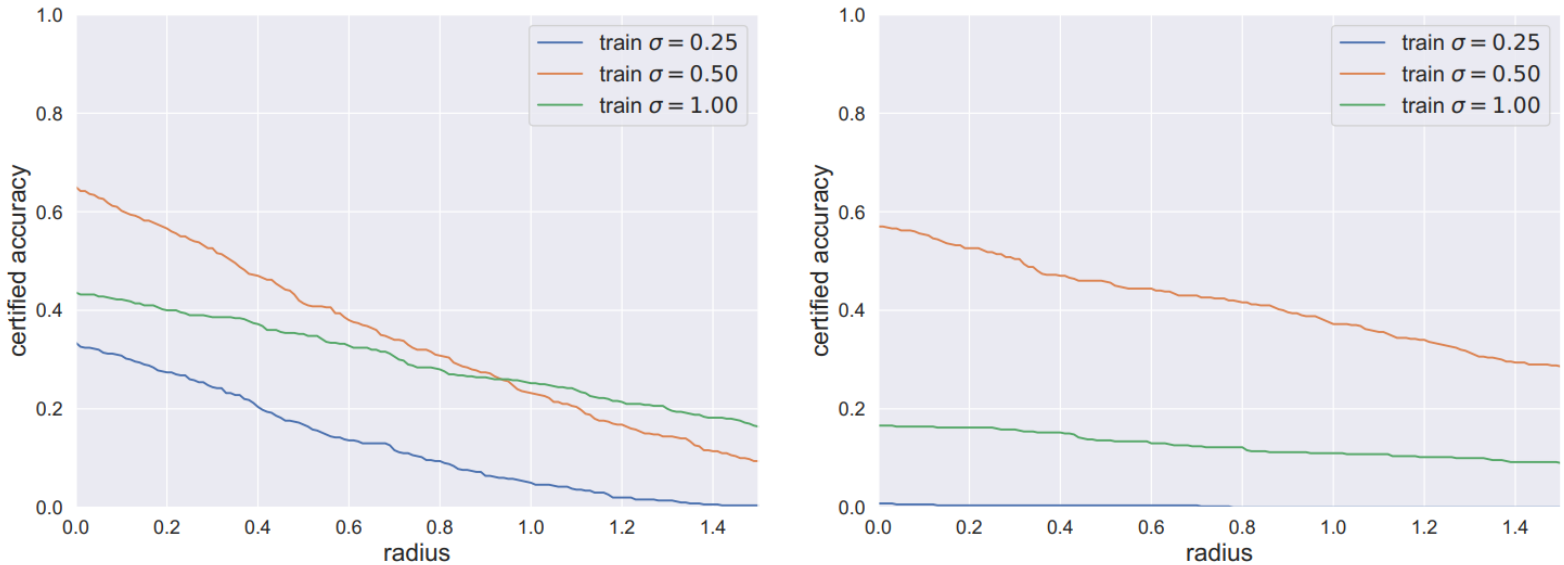
➢ Train f with Gaussian data augmentation at variance σ^2, with training noise level $\sigma_{train} \geq \sigma_{pred}$ prediction noise level .

Algorithm

- How much noise to use when training the base classifier ?

➢ Holding prediction noise level fixed at $\sigma_{pred} = 0.5$.

- If f was trained with a different noise level ($\sigma_{train} \neq 0.5$), g has lower certified accuracy. (blue and green)
- It is better to train with $\sigma_{train} > \sigma_{pred}$ than to train with $\sigma_{train} < \sigma_{pred}$.



(a) CIFAR-10 Figure 15 (b) ImageNet

Algorithm

- PREDICT $(f, \sigma, x, n, \alpha)$

 # evaluate g at x

function $\textsc{Predict}(f, \sigma, x, n, \alpha)$

1 $\texttt{counts} \leftarrow \textsc{SampleUnderNoise}(f, x, n, \sigma)$

$\hat{c}_A, \hat{c}_B \leftarrow$ top two indices in $\texttt{counts}$

$n_A, n_B \leftarrow \texttt{counts}[\hat{c}_A], \texttt{counts}[\hat{c}_B]$

2 **if** $\textsc{BinomPValue}(n_A, n_A + n_B, 0.5) \leq \alpha$ **return** $\hat{c}_A$

else return ABSTAIN

- CERTIFY $(f, \sigma, x, n_0, n, \alpha)$

 # certify the robustness of g around x

function $\textsc{Certify}(f, \sigma, x, n_0, n, \alpha)$

$\texttt{counts0} \leftarrow \textsc{SampleUnderNoise}(f, x, n_0, \sigma)$

$\hat{c}_A \leftarrow$ top index in $\texttt{counts0}$

$\texttt{counts} \leftarrow \textsc{SampleUnderNoise}(f, x, n, \sigma^2)$

3 $\underline{p_A} \leftarrow \textsc{LowerConfBound}(\texttt{counts}[\hat{c}_A], n, 1 - \alpha)$

if $\underline{p_A} > \frac{1}{2}$ **return** prediction $\hat{c}_A$ and radius $\sigma\, \Phi^{-1}(\underline{p_A})$

else return ABSTAIN

Independent algorithms for the two tasks:
(A) Evaluating $g(x)$
(B) Evaluating and certifying $g(x)$

Algorithm

- PREDICT $(f, \sigma, x, n, \alpha)$

 # evaluate g at x

function PREDICT$(f, \sigma, x, n, \alpha)$

1 `counts` $\leftarrow$ SAMPLEUNDERNOISE(f, x, n, σ)

$\hat{c}_A, \hat{c}_B \leftarrow$ top two indices in `counts`

$n_A, n_B \leftarrow$ `counts`$[\hat{c}_A]$, `counts`$[\hat{c}_B]$

2 **if** BINOMPVALUE$(n_A, n_A + n_B, 0.5) \leq \alpha$ **return** $\hat{c}_A$

else return ABSTAIN

- Requirement:

Only need to identify the class c_A with maximal weight in $f(x + \varepsilon)$

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

Algorithm

- PREDICT $(f, \sigma, x, n, \alpha)$

 # evaluate g at x
function PREDICT$(f, \sigma, x, n, \alpha)$
1 counts ← SAMPLEUNDERNOISE(f, x, n, σ)
$\hat{c}_A, \hat{c}_B$ ← top two indices in counts
n_A, n_B ← counts$[\hat{c}_A]$, counts$[\hat{c}_B]$
2 **if** BINOMPVALUE$(n_A, n_A + n_B, 0.5) \leq \alpha$ **retu**
else return ABSTAIN

SampleUnderNoise (f, x, n, σ)

- Draw n samples of noise:
  $\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n \sim N(0;\ \sigma^2 I)$.
- Run noisy images through the base classifier f to obtain the predictions:
  $f(x+\varepsilon_1), f(x+\varepsilon_2), \dots, f(x+\varepsilon_n)$

Return the *counts* for each class, where the *count* for class c is defined as:

$$\sum_{i=1}^{n} 1[if\ f(x+\varepsilon_i) = c]$$

$$n_A = \sum_{i=1}^{n} 1[if\ f(x+\varepsilon_i) = c_A]$$

$$n_B = \sum_{i=1}^{n} 1[if\ f(x+\varepsilon_i) = c_B]$$

Algorithm

- PREDICT $(f, \sigma, x, n, \alpha)$

 # evaluate g at x

function PREDICT$(f, \sigma, x, n, \alpha)$

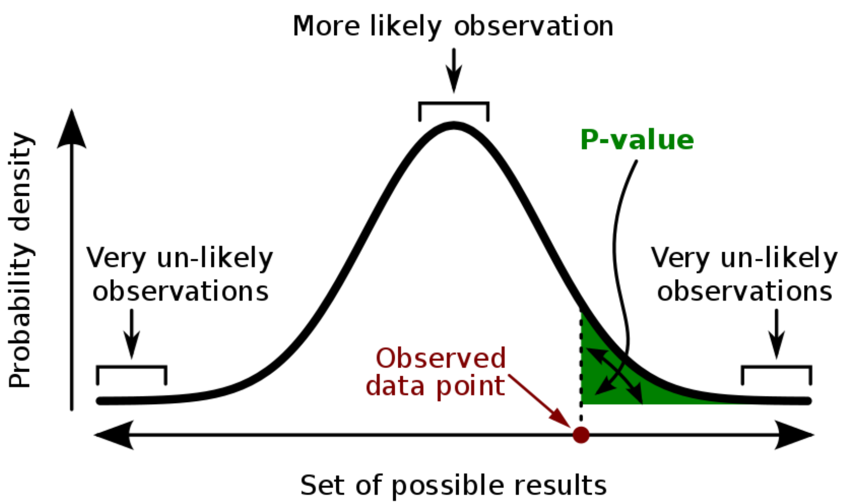
1 counts $\leftarrow$ SAMPLEUNDERNOISE(f, x, n, σ)

$\hat{c}_A, \hat{c}_B \leftarrow$ top two indices in counts

$n_A, n_B \leftarrow$ counts$[\hat{c}_A]$, counts$[\hat{c}_B]$

2 **if** BINOMPVALUE$(n_A, n_A + n_B, 0.5) \leq \alpha$ r

else return ABSTAIN



BinomPValue $(n_A, n_A + n_B, p = 0.5)$

- Return the p-value of the two-sided hypothesis test that $n_A \sim Binomial(n_A + n_B, p = 0.5)$.
- **P-value:** probability of obtaining very unlikely observations when assuming null hypothesis is correct.
- Two-sided hypothesis test: $\text{P-}value = 2 * \text{P-}value_{right}$
- When α is small, abstains frequently, but rarely returns wrong class.
- When α is large, seldom abstains, but often return the wrong class.

Algorithm

- CERTIFY$(f, \sigma, x, n_0, n, \alpha)$

➢ Requirement:

✓ Identify the class c_A with maximal weight in $f(x+\varepsilon)$

✓ Estimate a lower bound $\underline{p_A}$ on $p_A := P(f(x+\varepsilon) = c_A)$

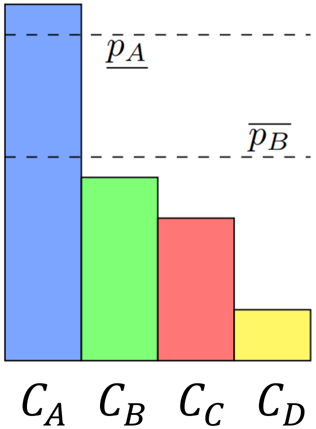
✓ Estimate an upper bound $\overline{p_B}$ on $p_B := \max_{c \neq c_A} P(f(x+\varepsilon) = c)$

➢ Problem

Statistically speaking, estimating $\underline{p_A}$ and $\overline{p_B}$ while simultaneously identifying the top class c_A is a little bit tricky.

➢ Solution

Two-step procedure



Algorithm

- CERTIFY$(f, \sigma, x, n_0, n, \alpha)$

➢ Requirement:

✓ Identify the class c_A with maximal weight in $f(x+\varepsilon)$

✓ Estimate a lower bound $\underline{p_A}$ on $p_A \coloneqq P(f(x+\varepsilon) = c_A)$

✓ Estimate an upper bound $\overline{p_B}$ on $p_B \coloneqq \max_{c \neq c_A} P(f(x+\varepsilon) = c)$

➢ Procedure:

 # certify the robustness of g around x

function CERTIFY$(f, \sigma, x, n_0, n, \alpha)$

    counts0 $\leftarrow$ SAMPLEUNDERNOISE(f, x, n_0, σ)

    $\hat{c}_A \leftarrow$ top index in counts0

    counts $\leftarrow$ SAMPLEUNDERNOISE(f, x, n, σ^2)

3 $\underline{p_A} \leftarrow$ LOWERCONFBOUND(counts$[\hat{c}_A]$, n, $1-\alpha$)

    if $\underline{p_A} > \frac{1}{2}$ **return** prediction $\hat{c}_A$ and radius $\sigma\, \Phi^{-1}(\underline{p_A})$

    else return ABSTAIN

Algorithm

- CERTIFY$(f, \sigma, x, n_0, n, \alpha)$

➢ Requirement:

✓ Identify the class c_A with maximal weight in $f(x+\varepsilon)$

✓ Estimate a lower bound $\underline{p_A}$ on $p_A := P(f(x+\varepsilon) = c_A)$

✓ Estimate an upper bound $\overline{p_B}$ on $p_B := \max_{c \neq c_A}$

➢ Procedure:

 # certify the robustness of g around x

function CERTIFY$(f, \sigma, x, n_0, n, \alpha)$

counts0 $\leftarrow$ SAMPLEUNDERNOISE(f, x, n_0, σ)

$\hat{c}_A \leftarrow$ top index in counts0

counts $\leftarrow$ SAMPLEUNDERNOISE(f, x, n, σ^2)

3 $\underline{p_A} \leftarrow$ LOWERCONFBOUND(counts$[\hat{c}_A]$, n, $1-\alpha$)

if $\underline{p_A} > \frac{1}{2}$ **return** prediction $\hat{c}_A$ and radius $\sigma\,\Phi^{-1}(\underline{p_A})$

else return ABSTAIN

➢ Use a small number n_0 of samples from $f(x+\varepsilon)$ to take a guess at c_A, because $f(x+\varepsilon)$ tends to put most of its weight on the top class,

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

Algorithm

- CERTIFY$(f, \sigma, x, n_0, n, \alpha)$

➢ Requirement:

✓ Identify the class c_A with maximal weight i

✓ Estimate a lower bound $\underline{p_A}$ on $p_A := P(f(x$

✓ Estimate an upper bound $\overline{p_B}$ on $p_B := \max_{c \neq c_A}$

➢ Procedure:

 # certify the robustness of g around x
function CERTIFY$(f, \sigma, x, n_0, n, \alpha)$
counts0 ← SAMPLEUNDERNOISE$(f, x, n_0, \sigma$
$\hat{c}_A$ ← top index in counts0
counts ← SAMPLEUNDERNOISE(f, x, n, σ^2)
3 $\underline{p_A}$ ← LOWERCONFBOUND(counts$[\hat{c}_A]$, n, $1-\alpha$)
if $\underline{p_A} > \frac{1}{2}$ **return** prediction $\hat{c}_A$ and radius $\sigma\,\Phi^{-1}(\underline{p_A})$
else return ABSTAIN

➢ Use a larger number n of samples to estimate $\underline{p_A}$ and $\overline{p_B}$

✓ Use LowerConfBound $(k, n, 1-\alpha)$ to return $\underline{p_A}$ of $[\underline{p_A}, \overline{p_A}]$ that holds with probability at least 1−α over the sampling of $k \sim Binomial(n, p_A)$.

■ Clopper-Person confidence interval (二项式比例置信区间): 根据一系列伯努利成功(c_A)-失败(c_B)实验的结果计算出的成功(c_A)概率(p_A)的置信区间$[\underline{p_A}, \overline{p_A}]$

➢ Take $\overline{p_B} = 1 - \underline{p_A}$.

Algorithm

- CERTIFY$(f, \sigma, x, n_0, n, \alpha)$

➢ Requirement:

✓ Identify the class c_A with maximal weight i

✓ Estimate a lower bound $\underline{p_A}$ on $p_A \coloneqq P(f(x$

✓ Estimate an upper bound $\overline{p_B}$ on $p_B \coloneqq \max_{c \neq c_A}$

➢ Procedure:

```
# certify the robustness of g around x
function CERTIFY(f, σ, x, n_0, n, α)
    counts0 ← SAMPLEUNDERNOISE(f, x, n_0, σ)
    ĉ_A ← top index in counts0
    counts ← SAMPLEUNDERNOISE(f, x, n, σ^2)
3   p_A ← LOWERCONFBOUND(counts[ĉ_A], n, 1 − α)
    if p_A > 1/2 return prediction ĉ_A and radius σ Φ^{-1}(p_A)
    else return ABSTAIN
```

➢ If $\underline{p_A} > 0.5$ (即 $\overline{\boldsymbol{p_B}} < 0.5$)

- Compute robustness guarantee

$$\boldsymbol{R} = \frac{\boldsymbol{\sigma}}{\boldsymbol{2}}\left(\phi^{-1}\left(\underline{\boldsymbol{p_A}}\right) - \phi^{-1}(\overline{\boldsymbol{p_B}})\right)$$
$$= \frac{\sigma}{2}\left(\phi^{-1}\left(\underline{p_A}\right) - \phi^{-1}(1 - \underline{p_A})\right)$$
$$= \frac{\sigma}{2} \cdot 2\phi^{-1}\left(\underline{p_A}\right) = \sigma\phi^{-1}(\underline{p_A})$$

- Return $\hat{c}_A$ and $\boldsymbol{R}$

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

Radius

- **Theorem 1.** *Let $f : \mathbb{R}^d \rightarrow \mathcal{Y}$ be any deterministic or random function, and let $\varepsilon \sim \mathcal{N}(0, \sigma^2 I)$. Let g be defined as in (1). Suppose $c_A \in \mathcal{Y}$ and $\underline{p_A}, \overline{p_B} \in [0,1]$ satisfy:*

$$\mathbb{P}(f(x+\varepsilon) = c_A) \geq \underline{p_A} \geq \overline{p_B} \geq \max_{c \neq c_A} \mathbb{P}(f(x+\varepsilon) = c) \quad (2)$$

Then $g(x+\delta) = c_A$ for all $\|\delta\|_2 < R$, where

$$R = \frac{\sigma}{2}\left(\Phi^{-1}(\underline{p_A}) - \Phi^{-1}(\overline{p_B})\right) \quad (3)$$

- 证明

➢ 要证 $g(x+\delta) = \arg\max_{c \in \mathcal{Y}} \mathbb{P}(f(x+\varepsilon+\delta) = c) = c_A$

➢ 就要证 $\mathbb{P}(f(x+\varepsilon+\delta) = c_A) > \max_{c \neq c_A} \mathbb{P}(f(x+\varepsilon+\delta) = c)$

➢ 对任意$c_B \neq c_A$，证 $\mathbb{P}(f(x+\varepsilon+\delta) = c_A) > \mathbb{P}(f(x+\varepsilon+\delta) = c_B)$

Radius

- Illustration of the proof of Theorem 1.

➢ 当且仅当 $\mathbb{P}\big(f(Y \in A)\big) > \mathbb{P}\big(f(Y \in B)\big)$成立，有$\mathbb{P}\big(f(x+\varepsilon+\delta) = c_A\big) > \mathbb{P}\big(f(x+\varepsilon+\delta) = c_B\big)$

- 定义：$A \coloneqq \{z: \delta^T(z-x) \le \sigma||\delta||\phi^{-1}(\underline{p_A})\}$　　$B \coloneqq \{z: \delta^T(z-x) \le \sigma||\delta||\phi^{-1}(\overline{p_B})\}$
- $\mathbb{P}\big(f(Y \in A)\big) = \phi\left(\phi^{-1}\left(\underline{p_A}\right) - \frac{||\delta||}{\sigma}\right)$　　$\mathbb{P}\big(f(Y \in B)\big) = \phi\left(\phi^{-1}(\overline{p_B}) + \frac{||\delta||}{\sigma}\right)$

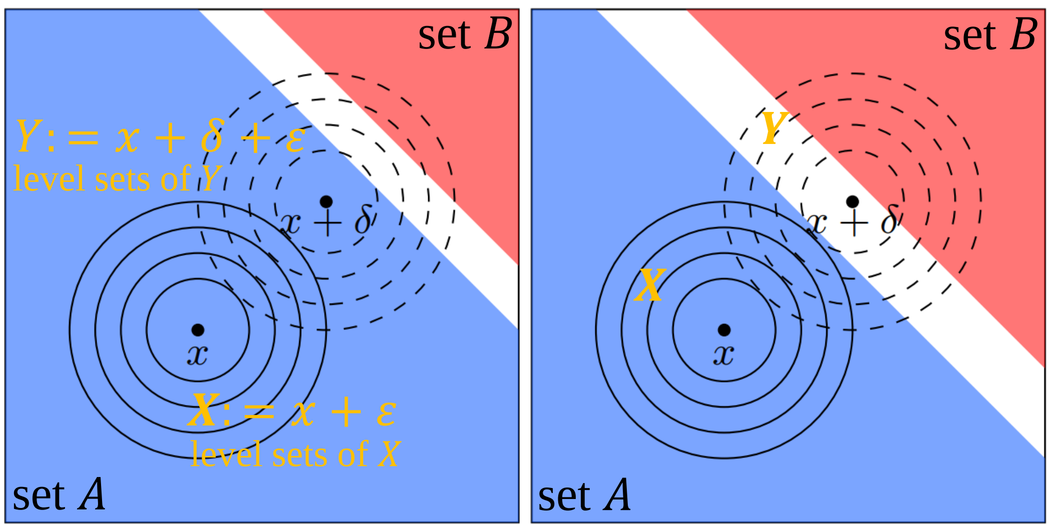


Figure 9

- The figure on the left depicts a situation where $\mathbb{P}(Y \in A) > \mathbb{P}(Y \in B)$, and hence $g(x+\delta) = g(Y)$ may equal c_A.
- The figure on the right depicts a situation where where $\mathbb{P}(Y \in A) < \mathbb{P}(Y \in B)$ and hence $g(x+\delta) = g(Y) \neq c_A$.

Evaluation

- Metric of normal classification

➢ Standard test set accuracy: $STD.ACC = \frac{N_{g-Correctly}}{N_{g-Total}}$

- Metric of certified classification

➢ Certified test set accuracy at radius r:

$$CERT.ACC = \frac{N_{g-Correctly-Robust}}{N_{g-Total}}$$

- g classifies correctly (without abstaining) and certifies robust with a radius $R \geq r$.
- r is similar toThreshold size T defined in Lecuyer. IEEE S&P 2019.

Evaluation

- Experiments with randomized smoothing on ImageNet with σ = 0.25.
- ➢ Left: certified accuracies obtained using CERTIFY VS. those obtained using the guarantees derived in prior works (SP2019).
- ➢ Middle: certified accuracy if the number of samples n used by CERTIFY had been larger or smaller.
- ➢ Right: certified accuracy as the failure probability α of CERTIFY is varied.

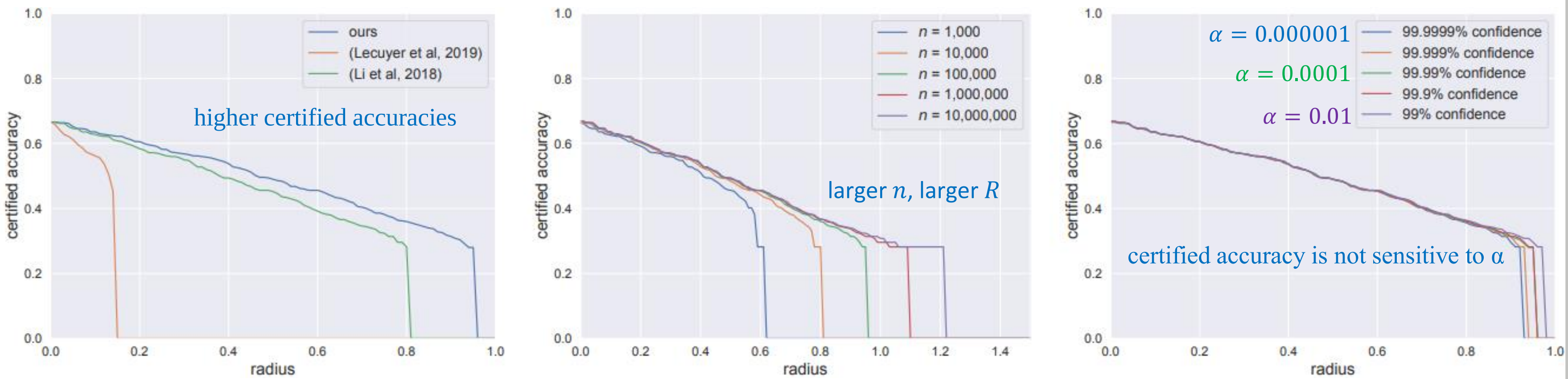


Figure 8

Contribution

- Prove the first tight robustness guarantee for randomized smoothing.
- Analysis reveals that smoothing with Gaussian noise naturally induces certifiable robustness under the l_2 norm.
- Suspect that other noise distributions might induce robustness to other perturbation sets such as general l_p norm balls.
- Enables the use of large networks on large scale datasets and does not constrain the architecture of the classifier.

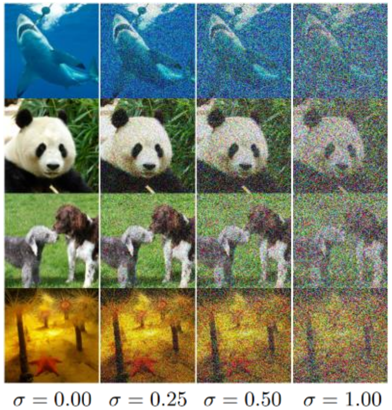
➢ Datasets:

- ImageNet
- CIFAR10

➢ Networks:

- ResNet

CIFAR10



ImageNet

Lecuyer, Mathias. Certified robustness to adversarial examples with differential privacy. IEEE S&P 2019.

Randomized Smoothing

Authors

Jeremy Cohen
PhD at CMU

Elan Rosenfeld
PhD at CMU

J. Zico Kolter
Associate
Professor at CMU

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

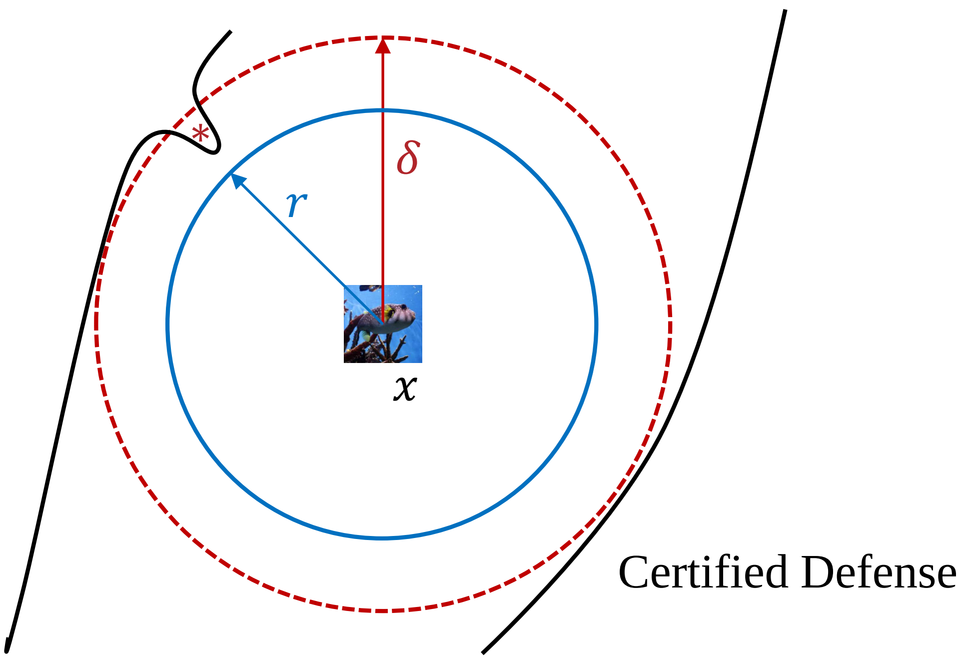
Overview

Defense Category

- Certified Defense

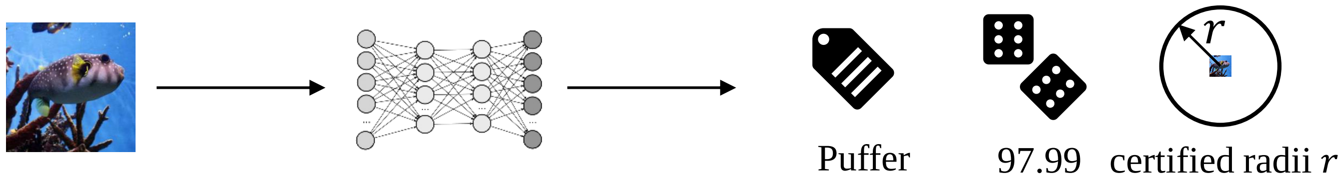
➢ Provide a certificate (r) for adversarial robustness

For any input x, the prediction output by the classifier f on samples in l_p ball centered at x are guaranteed to be constant.



Certified Defense Category

- Prediction



- Certification

➢ Exact Certified Defense

$(x, f, r) \longrightarrow$ Certification Procedure $\longrightarrow$

Does not exist δ within $||\delta|| \leq r$ for which $f(x) \neq f(x+\delta)$.

Exists δ within $||\delta|| \leq r$ for which $f(x) \neq f(x+\delta)$.

➢ Conservative Certified Defense

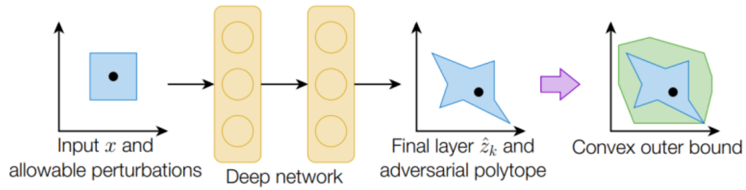
$(x, f, r) \longrightarrow$ Certification Procedure $\longrightarrow$

Does not exist δ within $||\delta|| \leq r$ for which $f(x) \neq f(x+\delta)$.

Abstain.

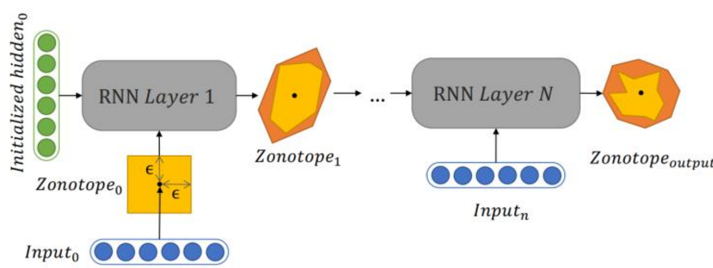
Future

- CNN Certified Defense



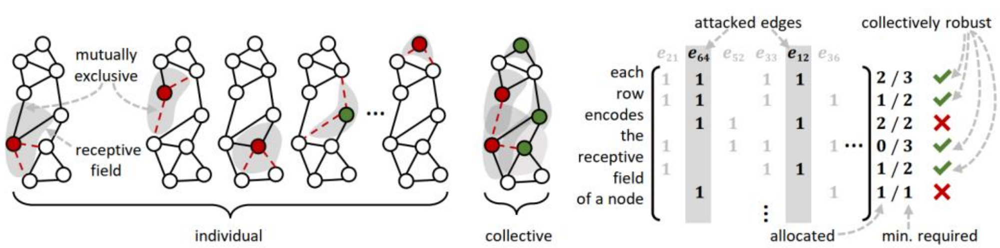
Zico Kolter.
Provable defenses against adversarial examples via the convex outer adversarial polytope.
ICML2018.

- RNN Certified Defense



Du Tianyu.
Cert-RNN: Towards Certifying the Robustness of Recurrent Neural Networks.
CCS2021.

- GNN Certified Defense



Schuchardt.
Collective robustness certificates: Exploiting interdependence in graph neural networks.
ICLR2021.

Thank You

Mengdie Huang

AI Security

Lab Ruiyun

2022-04-01