UNIVERSITY OF COPENHAGEN

Sparse Polynomial Optimization

Theory and its application to deep neural networks

Tong Chen (toch@di.ku.dk)
Machine Learning Section, DIKU

March 21, 2024

Outline

1. Part I: Motivation and Background

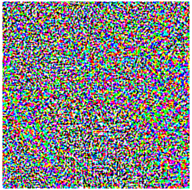
2. Part II: Polynomial Optimization

3. Part III: Experiments and Future work

Motivation: Adversarial Example

This is a panda!

+

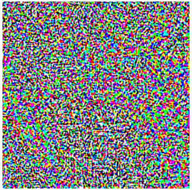


=

This is a gibbon!

Motivation: Adversarial Example

+

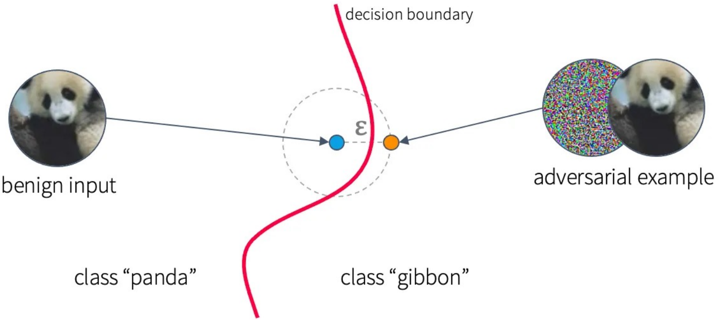


=

This is a panda!

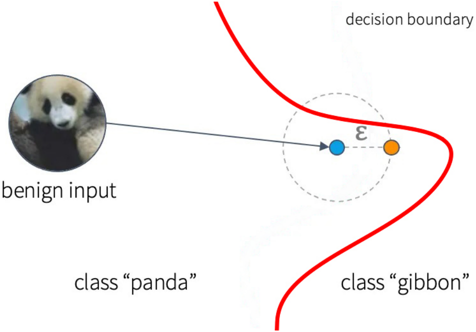
This is a gibbon!

Adversarial Example



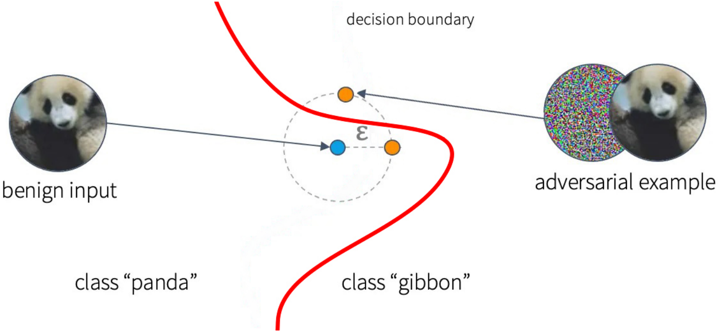
$$\delta_1 = \arg\max_{\|\delta\| \leq \varepsilon} \; l(\mathbf{x} + \delta, y; \theta)$$

Adversarial Training



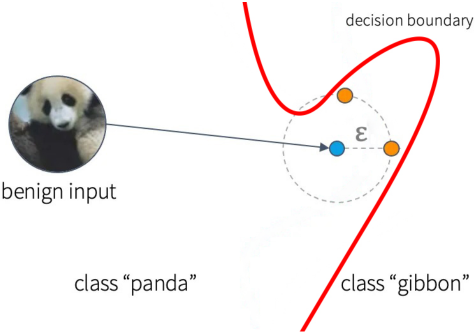
$$\theta_1 = \arg\min_\theta \ \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\max_{\|\delta\|\leq\varepsilon} \ l(\mathbf{x}+\delta, y; \theta)\right]$$

Adversarial Example



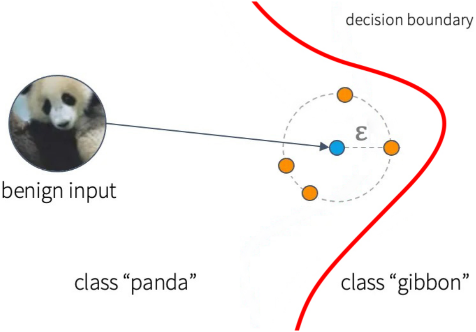
$$\delta_2 = \arg\max_{\|\delta\| \leq \varepsilon} \; l(\mathbf{x} + \delta, y; \theta_1)$$

Adversarial Training



$$\theta_2 = \arg\min_\theta \; \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\max_{\|\delta\|\leq\varepsilon} \; l(\mathbf{x}+\delta, y; \theta)\right]$$

Certified Training



$$\theta^* = \arg\min_{\theta} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\tilde{l}(\mathbf{x}, y, \varepsilon; \theta)\right]$$

- $\tilde{l}$ convex, and $\tilde{l}(\mathbf{x}, y, \varepsilon; \theta) \geq \max_{\|\delta\|\leq\varepsilon} l(\mathbf{x}+\delta, y; \theta)$.

Lipschitz Constant Controls Robustness

- Let $f : \mathcal{X} \to \mathbb{R}$:

$$L_f^p = \inf_{\mathbf{x},\mathbf{y} \in \mathcal{X}} \{L : |f(\mathbf{x}) - f(\mathbf{y})| \leq L \cdot \|\mathbf{x} - \mathbf{y}\|_p\}.$$

- Let $L(\theta)$ be the (global) Lipschitz constant of $l(\mathbf{x}, y; \theta)$, then

$$\max_{\|\delta\| \leq \varepsilon} l(\mathbf{x} + \delta, y; \theta) \leq l(\mathbf{x}, y; \theta) + L(\theta) \cdot \varepsilon =: \tilde{l}(\mathbf{x}, y, \varepsilon; \theta).$$

Lischitz Constants of Neural Networks

- Let $f : \mathcal{X} \to \mathbb{R}$,

$$L_f^p = \inf_{\mathbf{x},\mathbf{y} \in \mathcal{X}} \{L : |f(\mathbf{x}) - f(\mathbf{y})| \leq L \cdot \|\mathbf{x} - \mathbf{y}\|_p\}.$$

- If $\mathcal{X}$ is convex, f is smooth,

$$L_f^p = \sup_{\mathbf{x} \in \mathcal{X}} \|\nabla f(\mathbf{x})\|_p^* = \sup_{\mathbf{x} \in \mathcal{X}} \{\mathbf{t}^T \nabla f(\mathbf{x}) : \|\mathbf{t}\|_p \leq 1\}.$$

Outline

1. Part I: Motivation and Background
2. Part II: Polynomial Optimization
3. Part III: Experiments and Future work

Polynomial Optimization

Polynomial optimization problem:

$$\min_{\mathbf{x}} f(\mathbf{x}) \quad \text{s.t. } g_i(\mathbf{x}) \geq 0, \; i = 1, \ldots, p, \tag{POP}$$

where $f, \; g_i$ are polynomials.

- **Non-convex, NP-hard**.

From Hard to Easy:

$$\mathbf{K} := \{\mathbf{x} : g_i(\mathbf{x}) \geq 0,\ i = 1, \ldots, p\}$$

$$\min_{\mathbf{x}} \{f(\mathbf{x}) : \mathbf{x} \in \mathbf{K}\} \qquad \text{(non-convex)}$$

$$\downarrow$$

$$\max_{\rho} \{\rho : f - \rho \geq 0 \text{ over } \mathbf{K}\}$$

$$\vee|$$

$$\max_{\rho} \{\rho : f - \rho = \sigma^2 + \sum_{i=1}^{p} \lambda \cdot g_i,\ \lambda \geq 0\}$$

$$\downarrow$$

semidefinite program (SDP) (convex)

An Example:

$$\mathbf{K} := \{(x_1, x_2) : g(x_1, x_2) = 1 - x_1^2 - x_2^2 \geq 0\} \subseteq \mathbb{R}^2$$

$$\min_{x_1, x_2} \{x_1 x_2 : (x_1, x_2) \in \mathbf{K}\}$$

$$\downarrow$$

$$\max_{\rho} \{\rho : x_1 x_2 - \rho \geq 0 \text{ over } \mathbf{K}\}$$

$$\geq$$

$$\max_{\rho} \{\rho : x_1 x_2 - \rho = \sigma^2 + \lambda \cdot g, \ \lambda \geq 0\}$$

$$\downarrow$$

$$x_1 x_2 - \underbrace{\left(-\frac{1}{2}\right)}_{\rho} = \underbrace{\left(\frac{x_1 + x_2}{\sqrt{2}}\right)^2}_{\sigma^2 \geq 0} + \underbrace{\frac{1}{2}}_{\lambda \geq 0} \cdot \underbrace{(1 - x_1^2 - x_2^2)}_{g \geq 0}$$

Recall: Lischitz Constants of NN

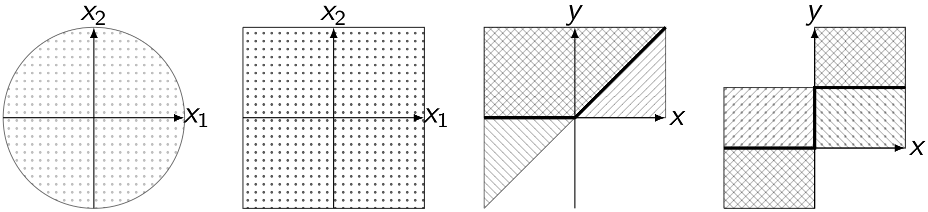
- Let $f : \mathcal{X} \to \mathbb{R}$,

$$L_f^p = \inf_{\mathbf{x},\mathbf{y}\in\mathcal{X}} \{L : |f(\mathbf{x}) - f(\mathbf{y})| \leq L \cdot \|\mathbf{x} - \mathbf{y}\|_p\}.$$

- If $\mathcal{X}$ is convex, f is smooth,

$$L_f^p = \sup_{\mathbf{x}\in\mathcal{X}} \|\nabla f(\mathbf{x})\|_p^* = \sup_{\mathbf{x}\in\mathcal{X}} \{\mathbf{t}^T \nabla f(\mathbf{x}) : \|\mathbf{t}\|_p \leq 1\}.$$

Semialgebraicity



L_2 norm L_∞ norm ReLU ∂ReLU

Outline

1. Part I: Motivation and Background
2. Part II: Polynomial Optimization
3. Part III: Experiments and Future work

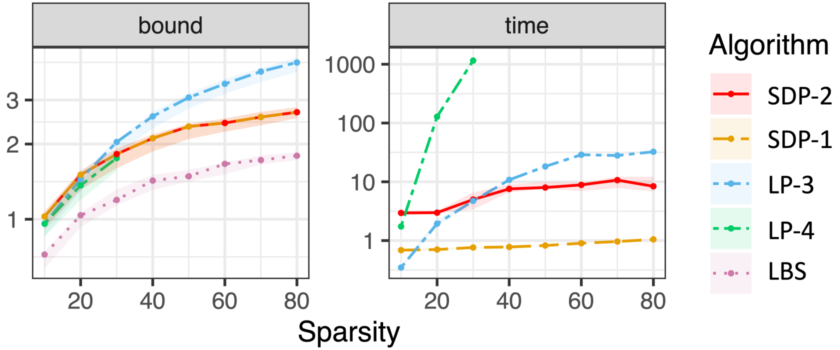
Algorithms

- **LP-3/4**: 3rd-/4th-degree Linear Programming (LP);
- **SDP-1/2**: 1st-/2nd-order Semidefinite Programming (SDP);
- **LBS**: lower bound by random sampling.

Random (80,80) MLP

$$\begin{bmatrix}
* & * & * & * & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 & 0 \\
* & * & * & * & * & 0 & \cdots & 0 & 0 & 0 & 0 & 0 & 0 \\
* & * & * & * & * & * & \cdots & 0 & 0 & 0 & 0 & 0 & 0 \\
* & * & * & * & * & * & \cdots & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & * & * & * & * & * & \cdots & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & * & * & * & * & \cdots & 0 & 0 & 0 & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & 0 & 0 & \cdots & * & * & * & * & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \cdots & * & * & * & * & * & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \cdots & * & * & * & * & * & * \\
0 & 0 & 0 & 0 & 0 & 0 & \cdots & * & * & * & * & * & * \\
0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & * & * & * & * & * \\
0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & * & * & * & *
\end{bmatrix}$$

Random (80,80) MLP



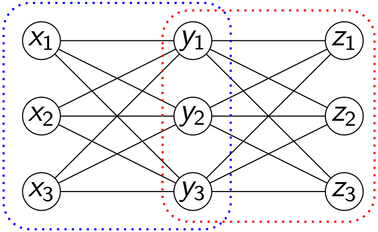
MNIST (784, 500) MLP

	SDP-2	SDP-1	LP-3	LBS
bound	14.56	17.85	OfM	9.69
time (s)	12246	2869	OfM	-

Future Work

Exploiting sparsity:

$$I = \{x_i, y_j, z_k\} = I_1 \cup I_2$$

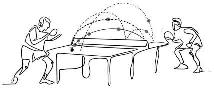


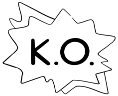
$$I_1 = \{x_i, y_j\} \qquad I_2 = \{y_j, z_k\}$$

$$81 = 9^2 = |I_1 \cup I_2|^2 \longrightarrow |I_1|^2 + |I_2|^2 = 6^2 + 6^2 = 72$$

Thank you!

Attack v.s. Defense





adversarial example

attack

defend

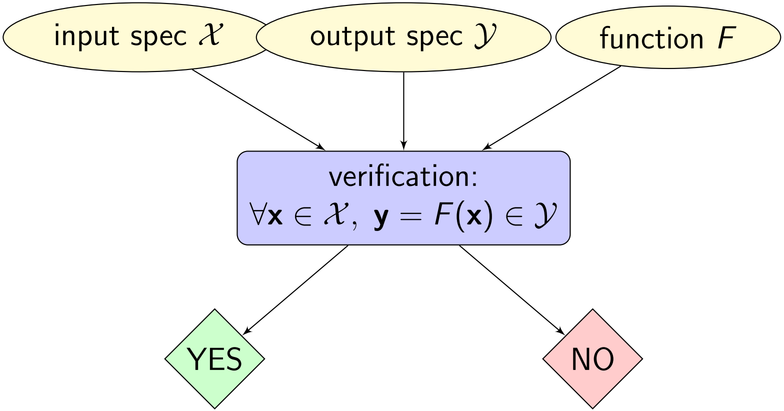
adversarial training

(sound) verification

train

certified training

NN Verification



Robustness Verification

- $F : \mathcal{X} \to \mathbb{R}^K$, classification;
- $F_k := F(\cdot)_k,\ y(\mathbf{x}_0) = \arg\max_k F_k(\mathbf{x}_0)$;
- Fix $\bar{\mathbf{x}}$, take $\mathcal{B} := \{\mathbf{x} : \|\mathbf{x} - \bar{\mathbf{x}}\|_p \leq \varepsilon\}$.

$$\forall \mathbf{x}_0 \in \mathcal{B},\ y_0 := y(\mathbf{x}_0) = y(\bar{\mathbf{x}}) =: \bar{y},$$

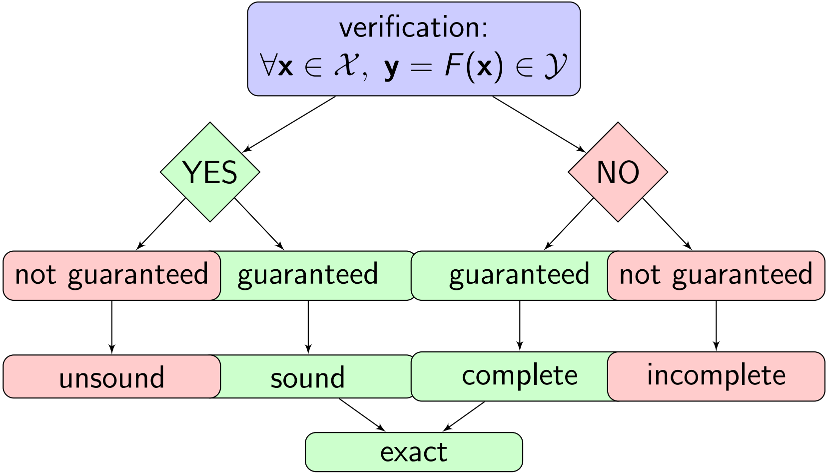
$$\Updownarrow$$

$$F_k(\mathbf{x}_0) < F_{\bar{y}}(\mathbf{x}_0),\ \forall k \neq \bar{y},$$

$$\Updownarrow$$

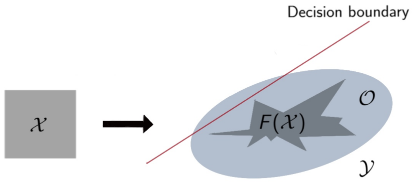
$$F_k(\mathbf{x}_0) - F_{\bar{y}}(\mathbf{x}_0) < 0,\ \forall k \neq \bar{y}.$$

Completeness and soundness

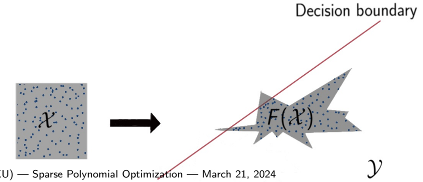


Examples

- sound (not complete) approach:



- complete (not sound) approach:



Sound Verification

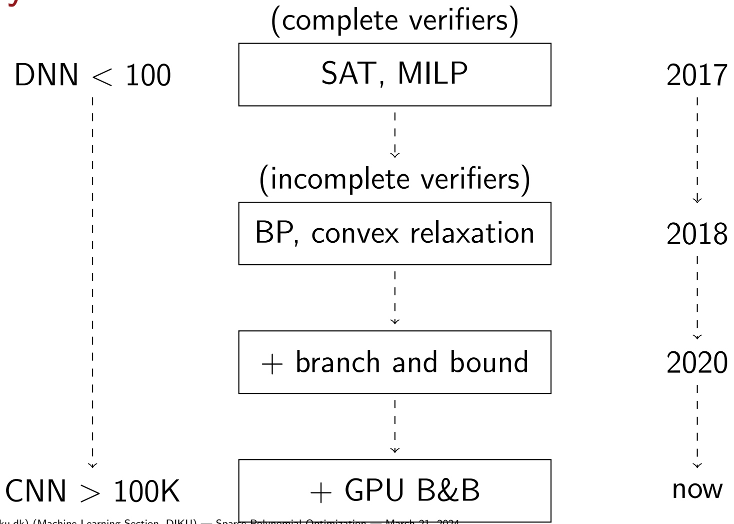
- Robustness verification: given input $\mathbf{x}_0$ and its prediction y_0,

$$\forall \mathbf{x} \in \mathcal{N}(\mathbf{x}_0),\ y(\mathbf{x}) = y_0?$$

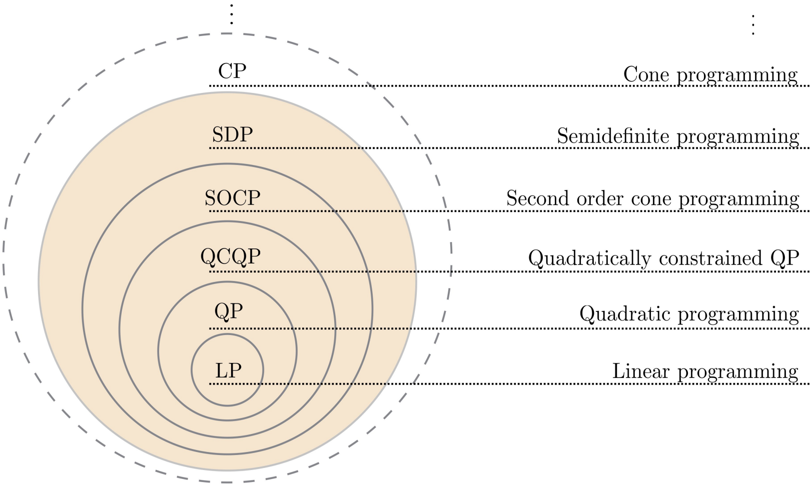
- Lipschitz constant estimation: given network F and input domain $\mathcal{X}$, find

$$L^F_{\mathcal{X}} \leq \tilde{L}^F_{\mathcal{X}}.$$

History of NN Verification



Inc


CP
Cone programming
SDP
Semidefinite programming
SOCP
Second order cone programming
QCQP
Quadratically constrained QP
QP
Quadratic programming
LP
Linear programming

Lasserre's Hierarchy [Lasserre01]

convexity	type	bound	complexity
non-convex	POP	f^*	NP-hard
$\uparrow$	$\uparrow$	$\parallel$	$\uparrow$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\uparrow$	$\uparrow$	$\geq$	$\uparrow$
convex	SDP_d	ρ_d	$O(n^d)$
$\uparrow$	$\uparrow$	$\geq$	$\uparrow$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\uparrow$	$\uparrow$	$\geq$	$\uparrow$
convex	SDP_2	ρ_2	$O(n^2)$
$\uparrow$	$\uparrow$	$\geq$	$\uparrow$
convex	SDP_1	ρ_1	$O(n)$

Future Work

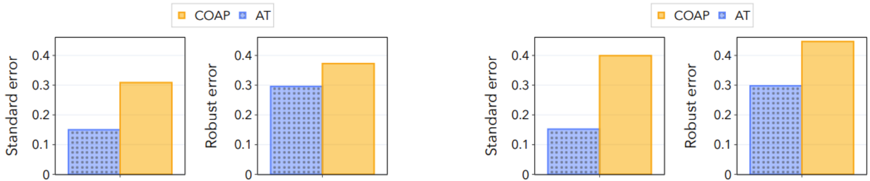
Paradox of certified training [Jovanovic22]:

Table 1: The Paradox of Certified Training: training with tighter relaxations leads to worse certified robustness, failing to outperform the loose IBP relaxation. Tightness formalization and further details given in Section 3.

Relaxation	Tightness	Certified (%)
IBP / Box	0.73	86.8
hBox / Symbolic Intervals	1.76	83.7
CROWN / DeepPoly	3.36	70.2
DeepZ / CAP / FastLin / Neurify	3.00	69.8
CROWN-IBP (R)	2.15	75.4

Future Work

Adversarial accuracy suffers from certified training [Bartolomeis23]:



(a) ℓ_∞-ball

(b) ℓ_2-ball

Figure 1: Standard and robust error of adversarial (dotted bars) and certified training (solid bars) on the CIFAR-10 test set. Models were trained for robustness against: (a) ℓ_∞-ball perturbations with radius $\epsilon_\infty = 1/255$, and (b) ℓ_2-ball perturbations with radius $\epsilon_2 = 36/255$. We report the best performing certified training method among many convex relaxations (FAST-IBP [32], IBP [9], CROWN-IBP [40, 43] and COAP [38, 39]). We refer the reader to Section 2 for further details on the models and robust evaluation.