Expressivity of ReLU-Networks under Convex Relaxations

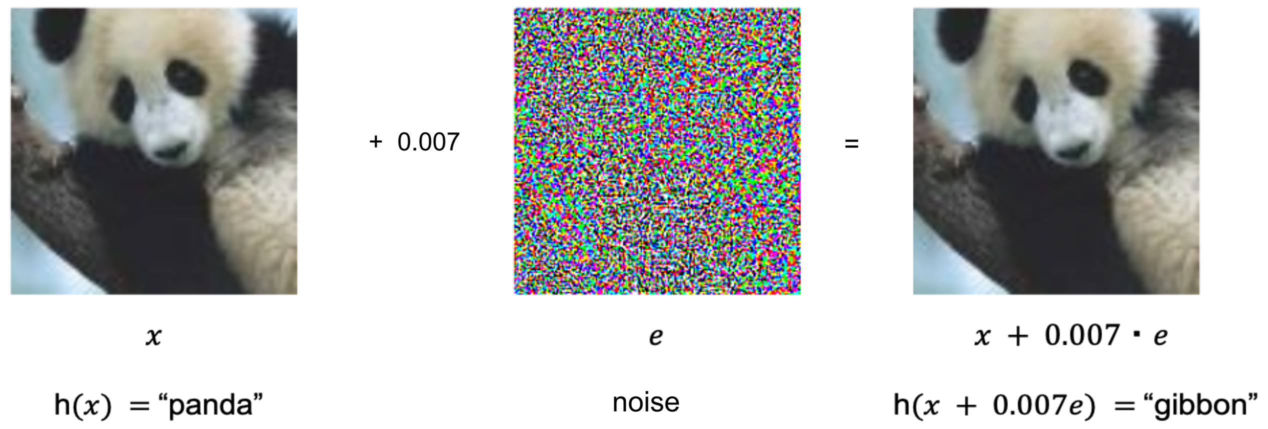
Maximilian Baader | Mark Müller | Yuhao Mao | Martin Vechev

ETH zürich SRILAB

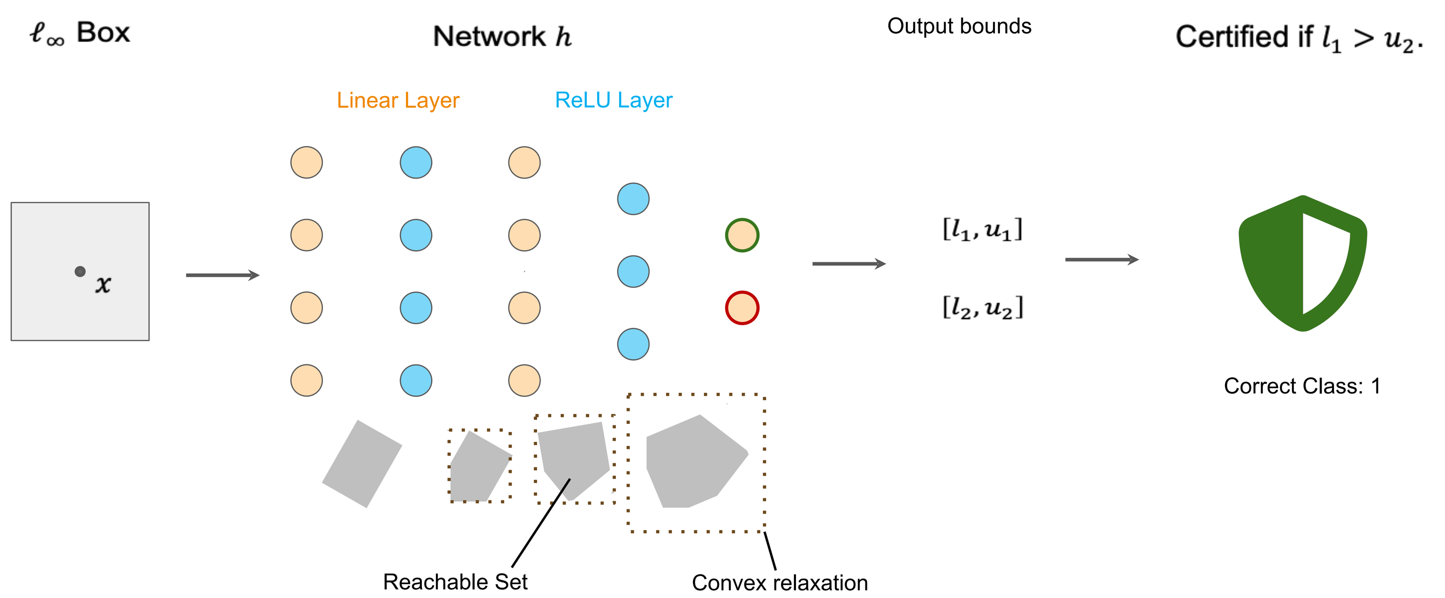
Paper & Code

Background: Robustness and Certification

Adversarial Examples: Neural networks can be fooled into misclassification by imperceptible input perturbations.

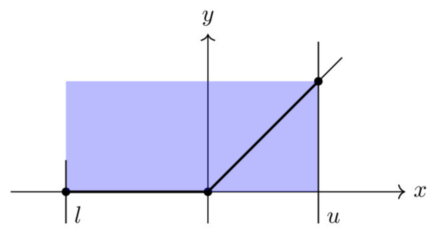


Certification: Local robustness to input perturbations of a network can be certified using convex relaxations.

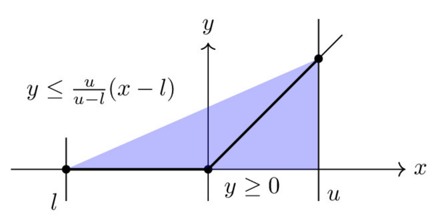


Convex Relaxations for ReLU:

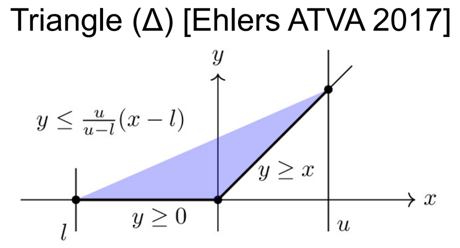
Box (IBP) [Gehr et al. S&P'18]



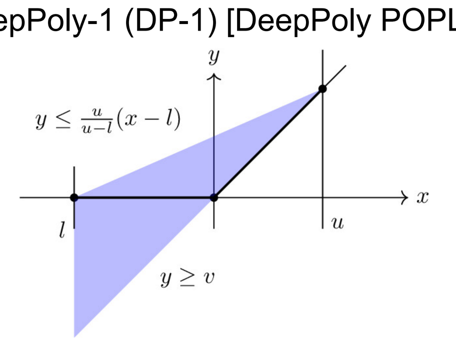
DeepPoly-0 (DP-0) [DeepPoly POPL'19]



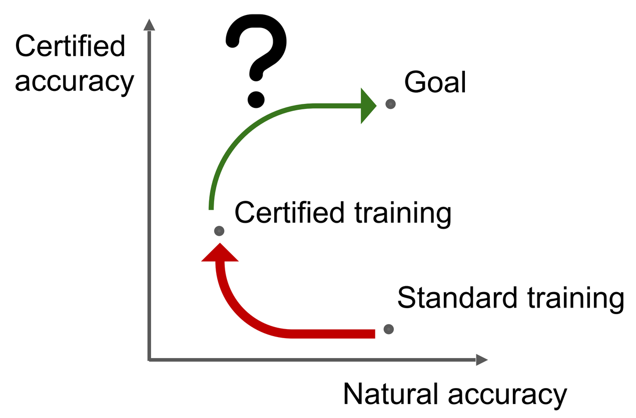
Triangle (Δ) [Ehlers ATVA 2017]



DeepPoly-1 (DP-1) [DeepPoly POPL'19]



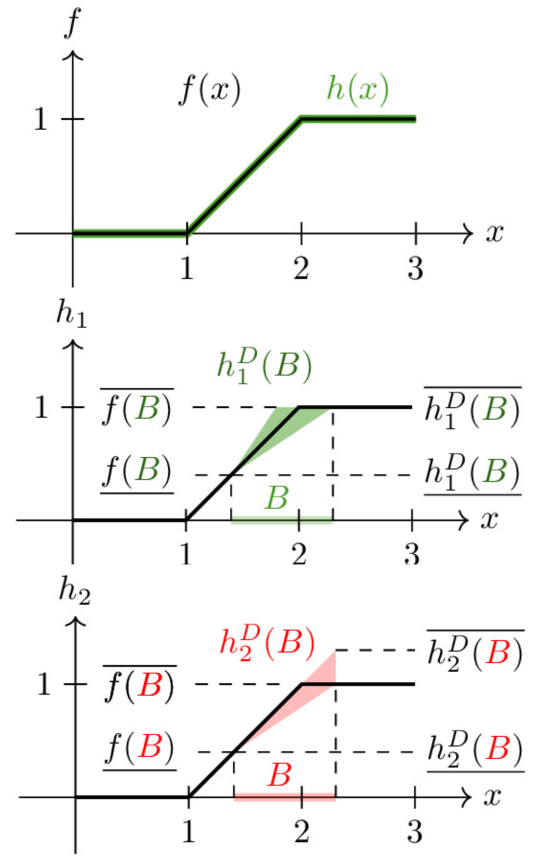
Fundamental Question:



Training for certifiability severely reduces accuracy, and thus real-world utility, despite best efforts [2].

What is the expressivity of certified neural networks?

Definitions



Encoding: Let $f: \mathcal{X} \to \mathcal{Y}$ be a function and $h: \mathcal{X} \to \mathcal{Y}$ be a neural network. We say h encodes f iff

$$h(x) = f(x) \quad \forall\, x \in \mathcal{X}.$$

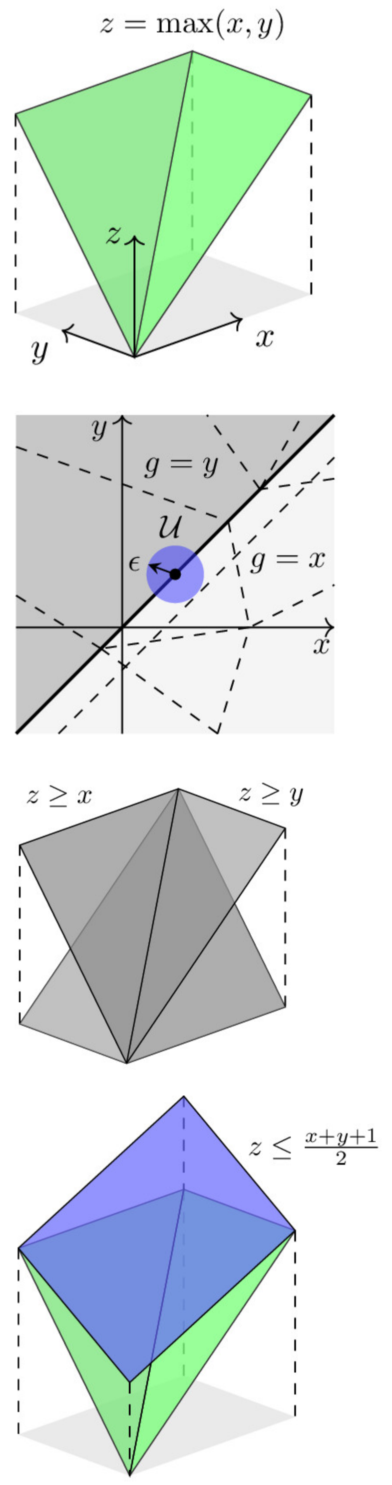
Analysis: $h^D(B)$ is the D-analysis of h on B, denoting the polytope in $\mathcal{X} \times \mathcal{Y}$ containing the graph $\{(x, h(x)) | x \in B\} \subseteq h^D(B)$ of h on B as obtained with D.

Precision: The D-analysis of h is precise if it yields precise lower and upper bounds, that is for all B

$$[\underline{h^D(B)}, \overline{h^D(B)}] = [\underline{f(B)}, \overline{f(B)}].$$

Expressivity: Let $\mathcal{F}$ be a set of functions and $\mathcal{N}$ a set of networks. $\mathcal{N}$ can D-express $\mathcal{F}$ iff $\forall f \in \mathcal{F}\ \exists h \in \mathcal{N}$ s.t. h encodes f and its D-analysis is precise

Theorem: Single Neuron Convex Relaxation Limit

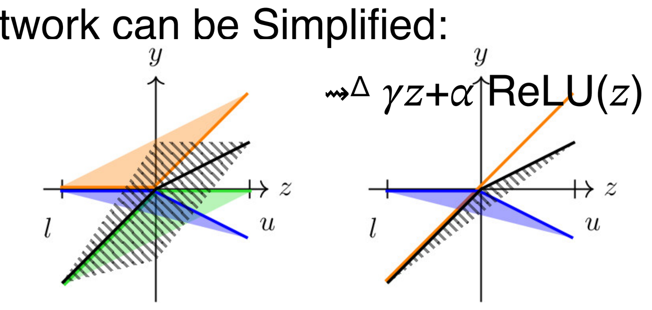


Theorem: Finite ReLU networks can not **Δ**-express convex, monotone, CPWL([0,1]²,ℝ) functions.

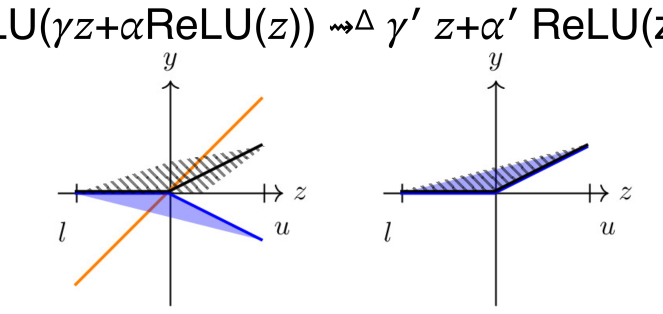
Proof: By contradiction. Let f=max: ℝ² → ℝ.

1. Locality: ∃ $\mathcal{U}$ s.t. all ReLUs are either stable or switch activation at $x{=}y$.
2. The network can be represented recursively as
$$h|_{\mathcal{U}} = h^i_{(R,L)} = h_L^{i-1} + W_i\ \mathrm{ReLU}(h_R^{i-1}),$$
with $h^0_{(R,L)} = b + W_0\, x$, s.t. all ReLUs switch at $x{=}y$.
1. This network can be Simplified:

$\rightsquigarrow^\Delta \gamma z + \alpha$ ReLU(z)



ReLU($\gamma z + \alpha$ReLU(z)) $\rightsquigarrow^\Delta \gamma' z + \alpha'$ ReLU(z).



This leads to $h(x) = b + w_x x + w_y y + \alpha$ ReLU(z).

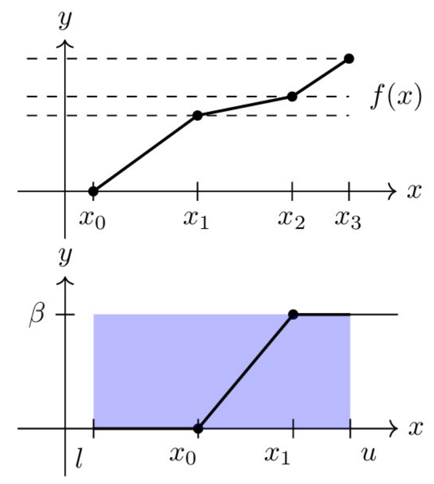
4. $h(x, y) = \max(x, y) \quad \Longrightarrow \quad b = 0,\ w_x = 0,\ w_y = 1,\ \alpha = 1.$

Analysis directly yields $h^\Delta(B) = 1.5 > 1 = \max(B)$.

ReLU networks can not Δ-express the set of MC-CPWL functions.

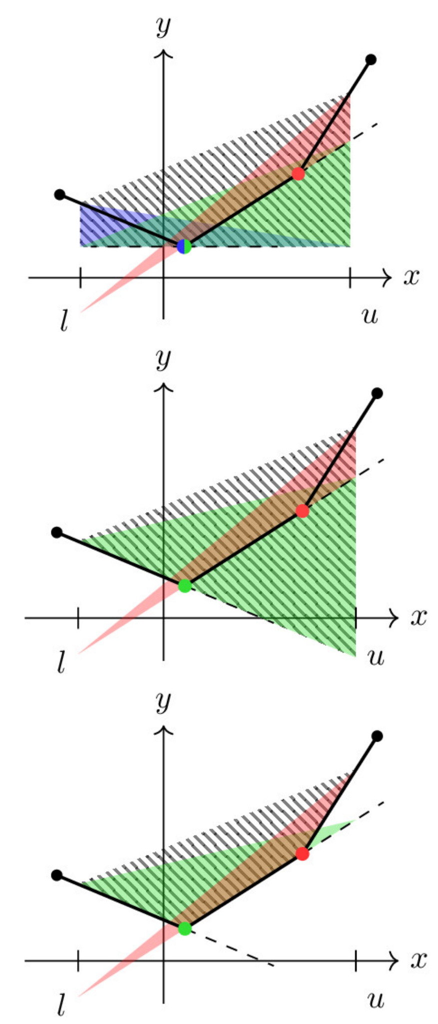
Separation

Prior Work:
No ReLU network can IBP-express convex CPWL(I, ℝ) functions.
No single-layer ReLU network can IBP-express monotone CPWL(I, ℝ) functions.



Theorem: Finite ReLU networks can IBP-express the set of monotone CPWL(I, ℝ) functions.

Depth increases expressivity of IBP-certified ReLU networks.



Theorem: For any convex CPWL function $f: I \to \mathbb{R}$, there exists exactly one network of the form

$$h(x) = b + \textstyle\sum_i \gamma_i\ \mathrm{ReLU}(\pm_i\,(x - x_i)),$$

with $\gamma_i > 0$ encoding f, with the minimum number of neurons such that its DP-0-analysis is precise.

DP-0 is more expressive than IBP.

Theorem: Let $f \in$ CPWL(I, ℝ) be convex. For any network h of the form

$$h(x) = b + c\,x + \textstyle\sum_i \gamma_i\ \mathrm{ReLU}(\pm_i (x - x_i)),$$

We have that its **Δ**-analysis is precise. In particular ±_i can be chosen freely.

Δ allows more parametrizations to express the same function compared to DP-0.

Theorem: For every network h, there exists a network g such that the DP-0 analysis of h and the DP-1 analysis of g are equivalent.

Results

Novel results are in red or green, previous results in **black**. M: monotone, C: convex, MC: monotone and convex.

$\mathcal{X}$	Relaxation	CPWL	M-CPWL	C-CPWL	MC-CPWL
ℝ	IBP	✗	✓	✗	✓
	DeepPoly-0	?	✓	✓	✓
	DeepPoly-1	?	✓	✓	✓
	Δ	?	✓	✓	✓
	Multi-Neuron$_\infty$	✓	✓	✓	✓
$\mathbb{R}^d$	Δ	✗	✗	✗	✗