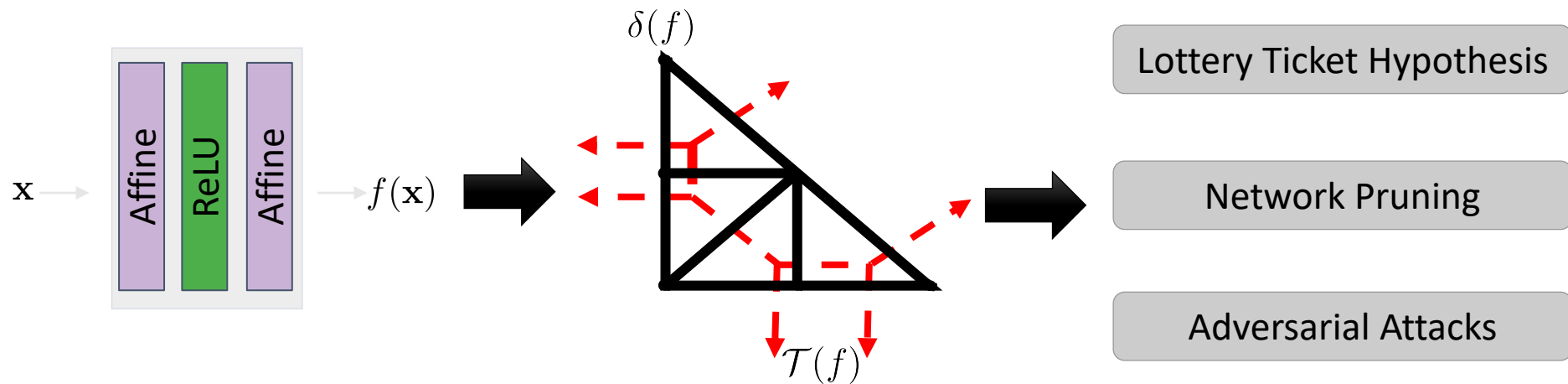


On the Decision Boundaries of Neural Networks. A Tropical Geometry Perspective

Motivation

Applications of Tropical Geometry in DNNs:



Outline

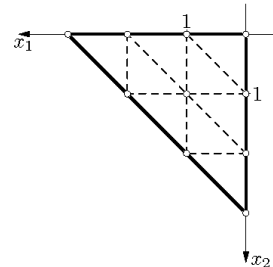
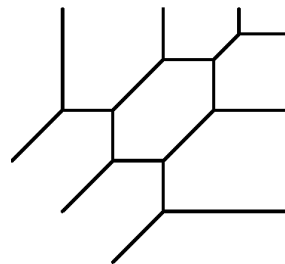
- Overview on Tropical Geometry
- DNNs in Tropical Geometry
- Lottery Ticket Hypothesis in TG
- Tropical Network Pruning
- Tropical Adversarial Attacks

Overview on Tropical Geometry

Tropical Semiring: $\mathcal{T} = (\mathbb{R} \cup \{-\infty\}, \oplus, \odot)$

• Tropical Addition: $a \oplus b = \max\{a, b\}$

• Tropical Multiplication: $a \odot b = a + b$



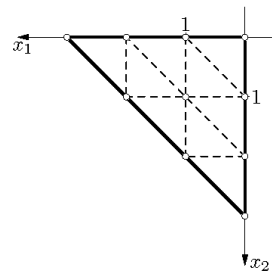
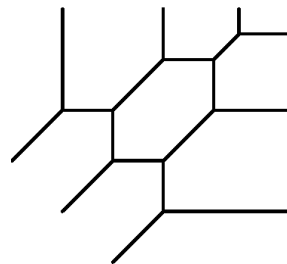
Overview on Tropical Geometry

Tropical Semiring: $\mathcal{T} = (\mathbb{R} \cup \{-\infty\}, \oplus, \odot)$.

• Tropical Addition: $a \oplus b = \max\{a, b\}$

• Tropical Multiplication: $a \odot b = a + b$

Tropical Power: $n \in \mathbb{N}$. $a^n = \underbrace{a \odot a \odot \cdots \odot a}_{n\text{-times}} = na$



Overview on Tropical Geometry

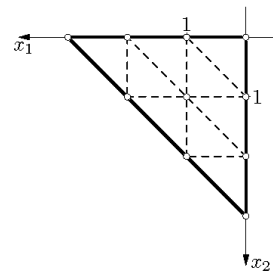
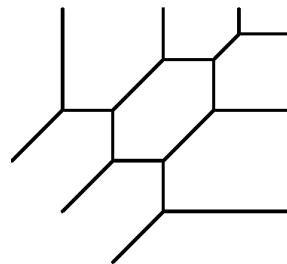
Tropical Semiring: $\mathcal{T} = (\mathbb{R} \cup \{-\infty\}, \oplus, \odot)$.

• Tropical Addition: $a \oplus b = \max\{a, b\}$

• Tropical Multiplication: $a \odot b = a + b$

Tropical Power: $n \in \mathbb{N}$. $a^n = \underbrace{a \odot a \odot \cdots \odot a}_{n\text{-times}} = na$

Tropical Division: $a \oslash b = a - b$



Overview on Tropical Geometry

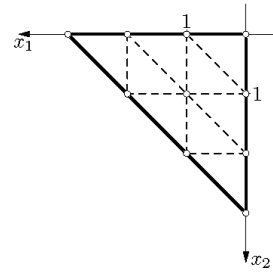
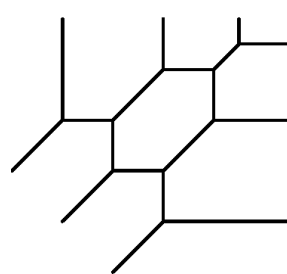
Tropical Semiring: $\mathcal{T} = (\mathbb{R} \cup \{-\infty\}, \oplus, \odot)$.

• Tropical Addition: $a \oplus b = \max\{a, b\}$

• Tropical Multiplication: $a \odot b = a + b$

Tropical Power: $n \in \mathbb{N}$. $a^n = \underbrace{a \odot a \odot \dots \odot a}_{n\text{-times}} = na$

Tropical Division: $a \oslash b = a - b$



No Tropical Subtraction: $\max\{5, 2\} = \max\{5, 3\} = \max\{5, x\} \quad \forall x \leq 5$

Overview on Tropical Geometry

Tropical Semiring: $\mathcal{T} = (\mathbb{R} \cup \{-\infty\}, \oplus, \odot)$.

- Tropical Addition: $a \oplus b = \max\{a, b\}$
- Tropical Multiplication: $a \odot b = a + b$

Tropical Polynomials: $c_i \in \mathbb{R}, \mathbf{a}_i \in \mathbb{N}^d$. $f(\mathbf{x}) = c_1 \mathbf{x}^{\mathbf{a}_1} \oplus \dots \oplus c_r \mathbf{x}^{\mathbf{a}_r}$

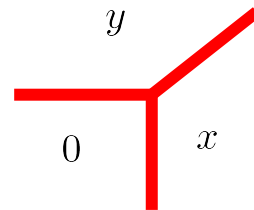
Overview on Tropical Geometry

Tropical Semiring: $\mathcal{T} = (\mathbb{R} \cup \{-\infty\}, \oplus, \odot)$.

- Tropical Addition: $a \oplus b = \max\{a, b\}$
- Tropical Multiplication: $a \odot b = a + b$

Tropical Polynomials: $c_i \in \mathbb{R}, \mathbf{a}_i \in \mathbb{N}^d$. $f(\mathbf{x}) = c_1 \mathbf{x}^{\mathbf{a}_1} \oplus \dots \oplus c_r \mathbf{x}^{\mathbf{a}_r}$

$$f(x, y) = x \oplus y \oplus 0$$



Overview on Tropical Geometry

Tropical Semiring: $\mathcal{T} = (\mathbb{R} \cup \{-\infty\}, \oplus, \odot)$.

• Tropical Addition: $a \oplus b = \max\{a, b\}$

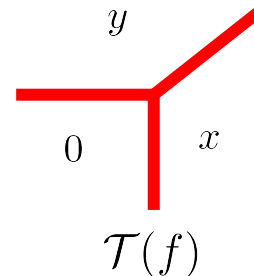
• Tropical Multiplication: $a \odot b = a + b$

Tropical Polynomials: $c_i \in \mathbb{R}, \mathbf{a}_i \in \mathbb{N}^d$. $f(\mathbf{x}) = c_1 \mathbf{x}^{\mathbf{a}_1} \oplus \dots \oplus c_r \mathbf{x}^{\mathbf{a}_r}$

Definition. (Tropical Hypersurfaces) For a tropical polynomial $f(\mathbf{x})$, the tropical hypersurface is the set of points \mathbf{x} where f is attained by two or more monomials in f .

$$\mathcal{T}(f) := \{\mathbf{x} \in \mathbb{R}^d : c_i \mathbf{x}^{\mathbf{a}_i} = c_j \mathbf{x}^{\mathbf{a}_j} = f(\mathbf{x}) \text{ for some } \mathbf{a}_i \neq \mathbf{a}_j\}.$$

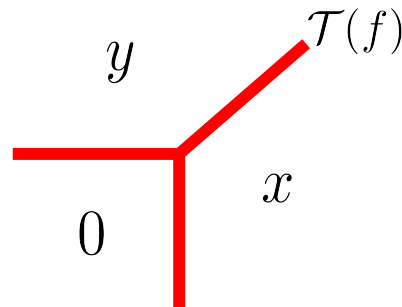
$$f(x, y) = x \oplus y \oplus 0$$



Overview on Tropical Geometry

Tropical Polynomials: $c_i \in \mathbb{R}$, $\mathbf{a}_i \in \mathbb{N}^d$. $f(\mathbf{x}) = c_1 \mathbf{x}^{\mathbf{a}_1} \oplus \dots \oplus c_r \mathbf{x}^{\mathbf{a}_r}$

$$f(x, y) = x \oplus y \oplus 0$$



Overview on Tropical Geometry

Tropical Polynomials: $c_i \in \mathbb{R}$, $\mathbf{a}_i \in \mathbb{N}^d$. $f(\mathbf{x}) = c_1 \mathbf{x}^{\mathbf{a}_1} \oplus \cdots \oplus c_r \mathbf{x}^{\mathbf{a}_r}$

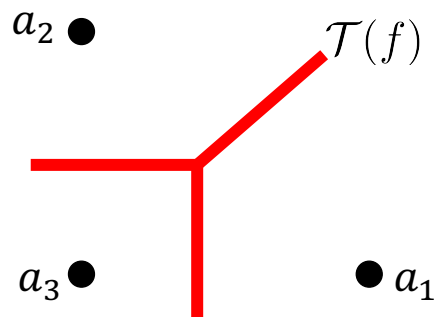
$$f(x, y) = x \oplus y \oplus 0$$

$$x^1 y^0 \rightarrow a_1 = (1, 0)$$

$$x^0 y^1 \rightarrow a_2 = (0, 1)$$

$$x^0 y^0 \rightarrow a_3 = (0, 0)$$

$$f(x, y) = x \oplus y \oplus 0$$



Overview on Tropical Geometry

Tropical Polynomials: $c_i \in \mathbb{R}$, $\mathbf{a}_i \in \mathbb{N}^d$. $f(\mathbf{x}) = c_1 \mathbf{x}^{\mathbf{a}_1} \oplus \dots \oplus c_r \mathbf{x}^{\mathbf{a}_r}$

$$f(x, y) = x \oplus y \oplus 0$$

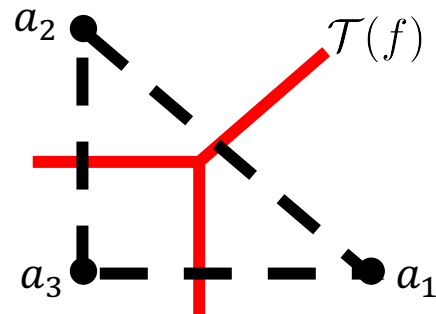
$$x^1 y^0 \rightarrow a_1 = (1, 0)$$

$$x^0 y^1 \rightarrow a_2 = (0, 1)$$

$$x^0 y^0 \rightarrow a_3 = (0, 0)$$

Dual Subdivision of f : $\delta(f) = \text{ConvHull}\{a_1, a_2, a_3\}$

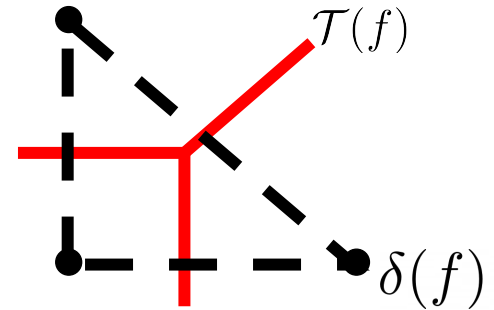
$$f(x, y) = x \oplus y \oplus 0$$



Overview on Tropical Geometry

What is special about the dual subdivision?

$$f(x, y) = x \oplus y \oplus 0$$

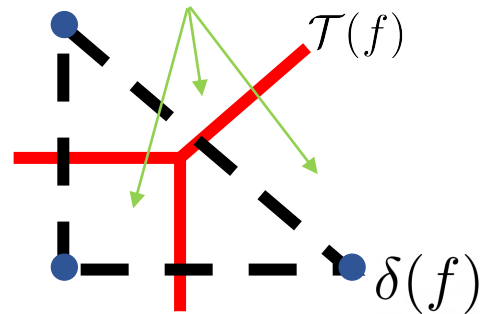


Overview on Tropical Geometry

What is special about the dual subdivision?

- The number of **vertices** in $\delta(f)$ is in one to one with the number of **regions** where f is linear.
- One can study the capacity of a tropical polynomial by counting the number of vertices in the dual subdivision.

$$f(x, y) = x \oplus y \oplus 0$$



Overview on Tropical Geometry

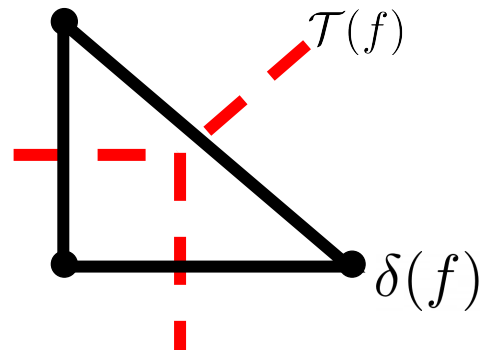
What is special about the dual subdivision?

- The number of vertices in $\delta(f)$ is in one to one with the number of regions where f is linear.

→ One can study the capacity of a tropical polynomial by counting the number of vertices in the dual subdivision.

- The normal to the edges of the dual subdivision $\delta(f)$ are parallel to the tropical hypersurface $\mathcal{T}(f)$.

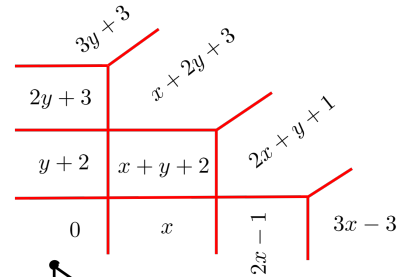
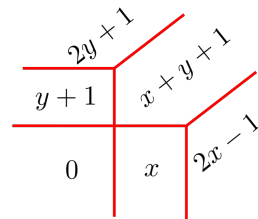
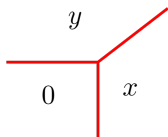
$$f(x, y) = x \oplus y \oplus 0$$



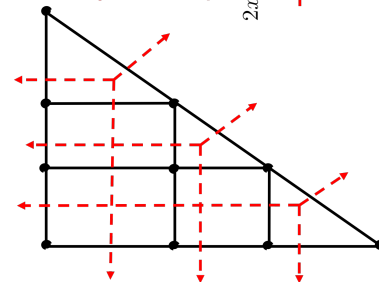
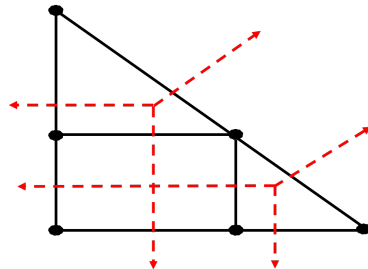
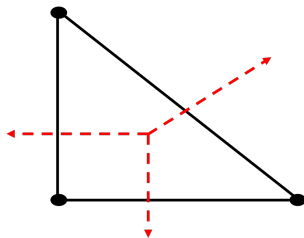
Overview on Tropical Geometry

$$f(x, y) \quad x \oplus y \oplus 0 \quad (x \oplus y \oplus 0) \odot (x - 1 \oplus y + 1 \oplus 0) \quad (x \oplus y \oplus 0) \odot (x - 1 \oplus y + 1 \oplus 0) \odot (x - 2 \oplus y + 2 \oplus 0)$$

$\mathcal{T}(f)$



$\delta(f)$



Overview on Tropical Geometry

- How useful can this be?

Outline

- Overview on Tropical Geometry
- DNNs in Tropical Geometry
- Lottery Ticket Hypothesis in TG
- Tropical Network Pruning
- Tropical Adversarial Attacks

DNNs in Tropical Geometry

Functional Characterization of DNNs in Tropical Geometry:

Zhang et al. Developed the following Equivalency.

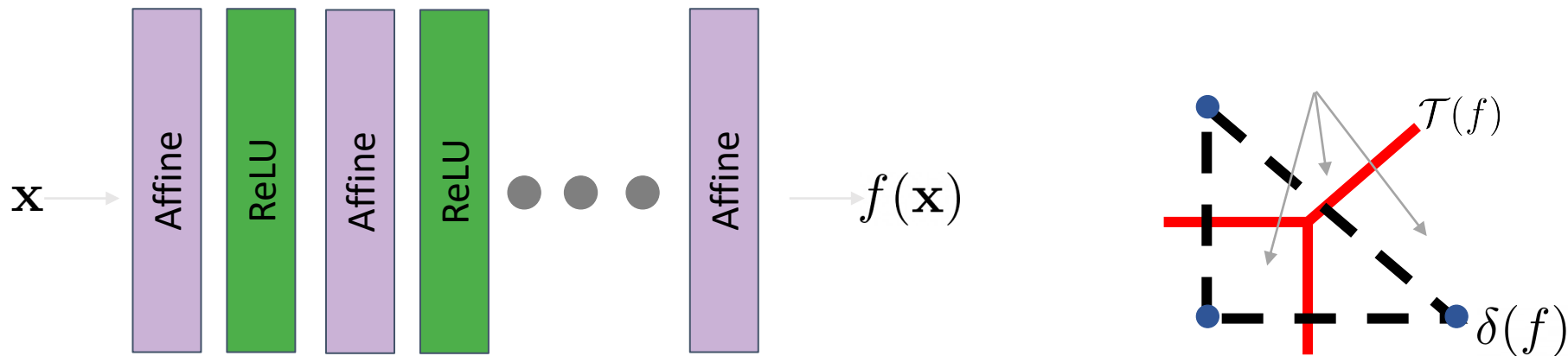
Theorem. (*Tropical Characterization of Neural Networks*). A feedforward neural network with integer weights and real biases with piecewise linear activation functions is a function $f : \mathbb{R}^n \rightarrow \mathbb{R}^k$, whose coordinates are tropical rational functions of the input, i.e.,

$$f(\mathbf{x}) = H(\mathbf{x}) \oslash Q(\mathbf{x}) = H(\mathbf{x}) - Q(\mathbf{x}),$$

where H and Q are tropical polynomials.

"Tropical Geometry of Deep Neural Networks", Liwen Zhang, Gregory Naitzat, Lek-Heng Lim, ICML18

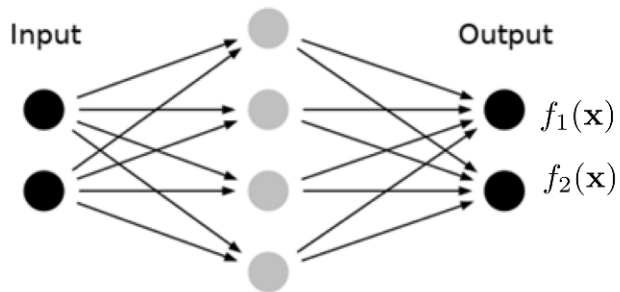
DNNs in Tropical Geometry



Studying the capacity of a network can be done by counting the number of **vertices** of a polytope

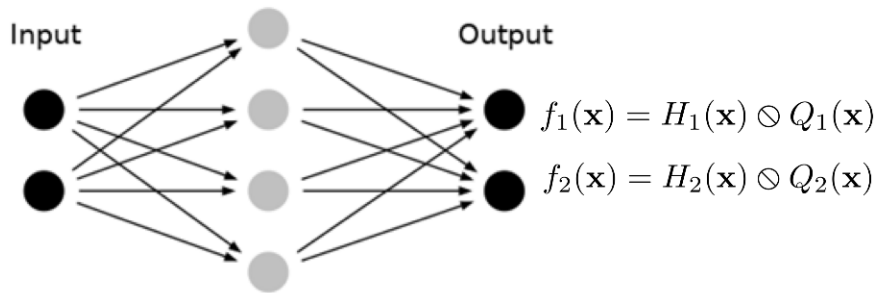
DNNs in Tropical Geometry

Consider the Binary Classification Problem:



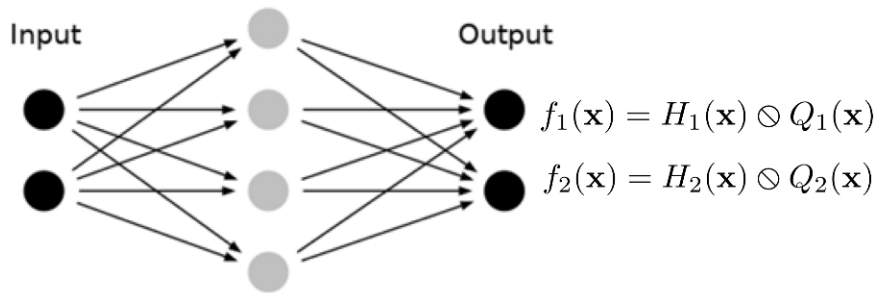
DNNs in Tropical Geometry

Consider the Binary Classification Problem:



DNNs in Tropical Geometry

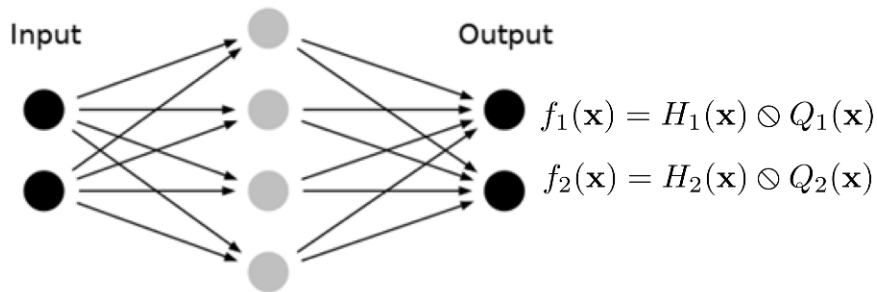
Consider the Binary Classification Problem:



$$f_1(\mathbf{x}) = f_2(\mathbf{x})$$
$$H_1(\mathbf{x}) + Q_2(\mathbf{x}) = H_2(\mathbf{x}) + Q_1(\mathbf{x})$$
$$H_1(\mathbf{x}) \odot Q_2(\mathbf{x}) = H_2(\mathbf{x}) \odot Q_1(\mathbf{x})$$

DNNs in Tropical Geometry

Consider the Binary Classification Problem:



$$f_1(\mathbf{x}) = f_2(\mathbf{x})$$

$$H_1(\mathbf{x}) + Q_2(\mathbf{x}) = H_2(\mathbf{x}) + Q_1(\mathbf{x})$$

$$H_1(\mathbf{x}) \odot Q_2(\mathbf{x}) = H_2(\mathbf{x}) \odot Q_1(\mathbf{x})$$

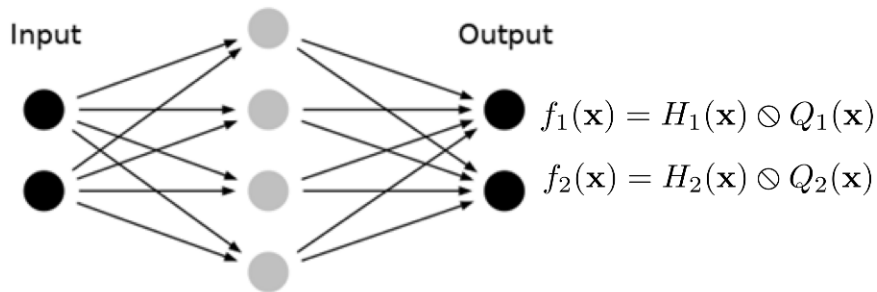


$$R(\mathbf{x}) = H_1(\mathbf{x}) \odot Q_2(\mathbf{x}) \oplus H_2(\mathbf{x}) \odot Q_1(\mathbf{x})$$

$$\mathcal{T}(R(\mathbf{x})) = \mathcal{T}(H_1 \odot Q_2) \cup \mathcal{T}(H_2 \odot Q_1) \cup \{\mathbf{x} : f_1(\mathbf{x}) = f_2(\mathbf{x})\}$$

DNNs in Tropical Geometry

Consider the Binary Classification Problem:



$$f_1(\mathbf{x}) = f_2(\mathbf{x})$$

$$H_1(\mathbf{x}) + Q_2(\mathbf{x}) = H_2(\mathbf{x}) + Q_1(\mathbf{x})$$

$$H_1(\mathbf{x}) \odot Q_2(\mathbf{x}) = H_2(\mathbf{x}) \odot Q_1(\mathbf{x})$$

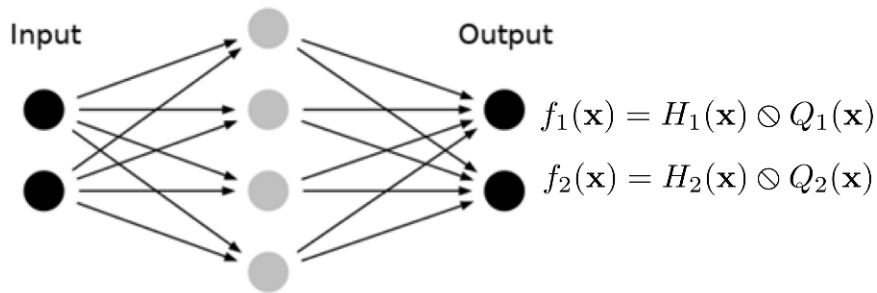


$$R(\mathbf{x}) = H_1(\mathbf{x}) \odot Q_2(\mathbf{x}) \oplus H_2(\mathbf{x}) \odot Q_1(\mathbf{x})$$

$$\mathcal{T}(R(\mathbf{x})) = \mathcal{T}(H_1 \odot Q_2) \cup \mathcal{T}(H_2 \odot Q_1) \cup \mathcal{B}$$

DNNs in Tropical Geometry

Consider the Binary Classification Problem:



$$f_1(\mathbf{x}) = f_2(\mathbf{x})$$
$$H_1(\mathbf{x}) + Q_2(\mathbf{x}) = H_2(\mathbf{x}) + Q_1(\mathbf{x})$$
$$H_1(\mathbf{x}) \odot Q_2(\mathbf{x}) = H_2(\mathbf{x}) \odot Q_1(\mathbf{x})$$



$$R(\mathbf{x}) = H_1(\mathbf{x}) \odot Q_2(\mathbf{x}) \oplus H_2(\mathbf{x}) \odot Q_1(\mathbf{x})$$
$$\mathcal{T}(R(\mathbf{x})) = \mathcal{T}(H_1 \odot Q_2) \cup \mathcal{T}(H_2 \odot Q_1) \cup \mathcal{B}$$

Analyze the decision boundaries by analyzing $\mathcal{T}(R(\mathbf{x}))$

Leverage the duality between $\mathcal{T}(R(\mathbf{x}))$ and $\delta(R(\mathbf{x}))$

DNNs in Tropical Geometry

Geometrical Characterization of DNNs in Tropical Geometry:

Theorem. *For a bias-free neural network in the form $f(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}^2$, where $\mathbf{A} \in \mathbb{Z}^{p \times n}$ and $\mathbf{B} \in \mathbb{Z}^{2 \times p}$, let $R(\mathbf{x}) = H_1(\mathbf{x}) \odot Q_2(\mathbf{x}) \oplus H_2(\mathbf{x}) \odot Q_1(\mathbf{x})$ be a tropical polynomial. Then:*

- Let $\mathcal{B} = \{\mathbf{x} \in \mathbb{R}^n : f_1(\mathbf{x}) = f_2(\mathbf{x})\}$ define the decision boundaries of f , then $\mathcal{B} \subseteq \mathcal{T}(R(\mathbf{x}))$.

DNNs in Tropical Geometry

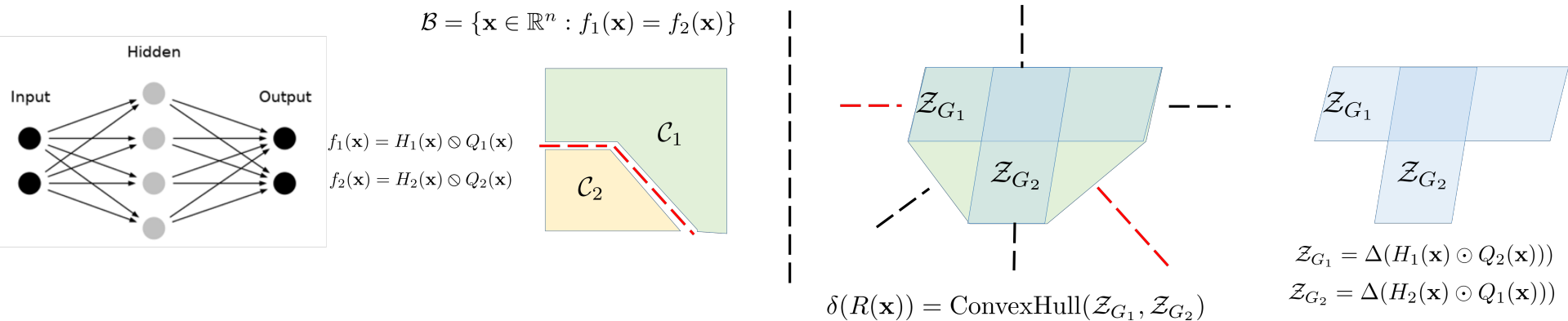
Geometrical Characterization of DNNs in Tropical Geometry:

Theorem. For a bias-free neural network in the form $f(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}^2$, where $\mathbf{A} \in \mathbb{Z}^{p \times n}$ and $\mathbf{B} \in \mathbb{Z}^{2 \times p}$, let $R(\mathbf{x}) = H_1(\mathbf{x}) \odot Q_2(\mathbf{x}) \oplus H_2(\mathbf{x}) \odot Q_1(\mathbf{x})$ be a tropical polynomial. Then:

- Let $\mathcal{B} = \{\mathbf{x} \in \mathbb{R}^n : f_1(\mathbf{x}) = f_2(\mathbf{x})\}$ define the decision boundaries of f , then $\mathcal{B} \subseteq \mathcal{T}(R(\mathbf{x}))$.

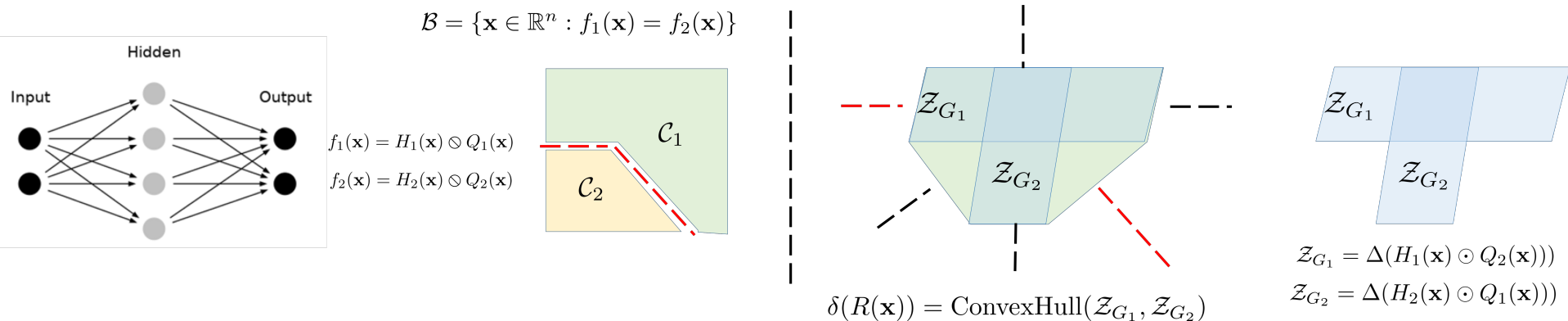
- $\delta(R(\mathbf{x})) = \text{ConvHull}(\mathcal{Z}_{\mathbf{G}_1}, \mathcal{Z}_{\mathbf{G}_2})$. $\mathcal{Z}_{\mathbf{G}_1}$ is a zonotope in \mathbb{R}^n with line segments $\{(\mathbf{B}^+(1, j) + \mathbf{B}^-(2, j))[\mathbf{A}^+(j, :), \mathbf{A}^-(j, :)]\}_{j=1}^p$ and shift $(\mathbf{B}^-(1, :) + \mathbf{B}^+(2, :))\mathbf{A}^-$. $\mathcal{Z}_{\mathbf{G}_2}$ is a zonotope in \mathbb{R}^n with line segments $\{(\mathbf{B}^-(1, j) + \mathbf{B}^+(2, j))[\mathbf{A}^+(j, :), \mathbf{A}^-(j, :)]\}_{j=1}^p$ and shift $(\mathbf{B}^+(1, :) + \mathbf{B}^-(2, :))\mathbf{A}^-$. Note that $\mathbf{A}^+ = \max(\mathbf{A}, 0)$ and $\mathbf{A}^- = \max(-\mathbf{A}, 0)$. The line segment $(\mathbf{B}^+(1, j) + \mathbf{B}^-(2, j))[\mathbf{A}^+(j, :), \mathbf{A}^-(j, :)]$ has end points $\mathbf{A}^+(j, :)$ and $\mathbf{A}^-(j, :)$ in \mathbb{R}^n and scaled by $(\mathbf{B}^+(1, j) + \mathbf{B}^-(2, j))$.

DNNs in Tropical Geometry



• $\delta(R(\mathbf{x})) = \text{ConvHull}(\mathcal{Z}_{G_1}, \mathcal{Z}_{G_2})$. \mathcal{Z}_{G_1} is a zonotope in \mathbb{R}^n with line segments $\{(\mathbf{B}^+(1, j) + \mathbf{B}^-(2, j))[\mathbf{A}^+(j, :), \mathbf{A}^-(j, :)]\}_{j=1}^p$ and shift $(\mathbf{B}^-(1, :) + \mathbf{B}^+(2, :))\mathbf{A}^-$. \mathcal{Z}_{G_2} is a zonotope in \mathbb{R}^n with line segments $\{(\mathbf{B}^-(1, j) + \mathbf{B}^+(2, j))[\mathbf{A}^+(j, :), \mathbf{A}^-(j, :)]\}_{j=1}^p$ and shift $(\mathbf{B}^+(1, :) + \mathbf{B}^-(2, :))\mathbf{A}^-$. Note that $\mathbf{A}^+ = \max(\mathbf{A}, 0)$ and $\mathbf{A}^- = \max(-\mathbf{A}, 0)$. The line segment $(\mathbf{B}^+(1, j) + \mathbf{B}^-(2, j))[\mathbf{A}^+(j, :), \mathbf{A}^-(j, :)]$ has end points $\mathbf{A}^+(j, :)$ and $\mathbf{A}^-(j, :)$ in \mathbb{R}^n and scaled by $(\mathbf{B}^+(1, j) + \mathbf{B}^-(2, j))$.

DNNs in Tropical Geometry

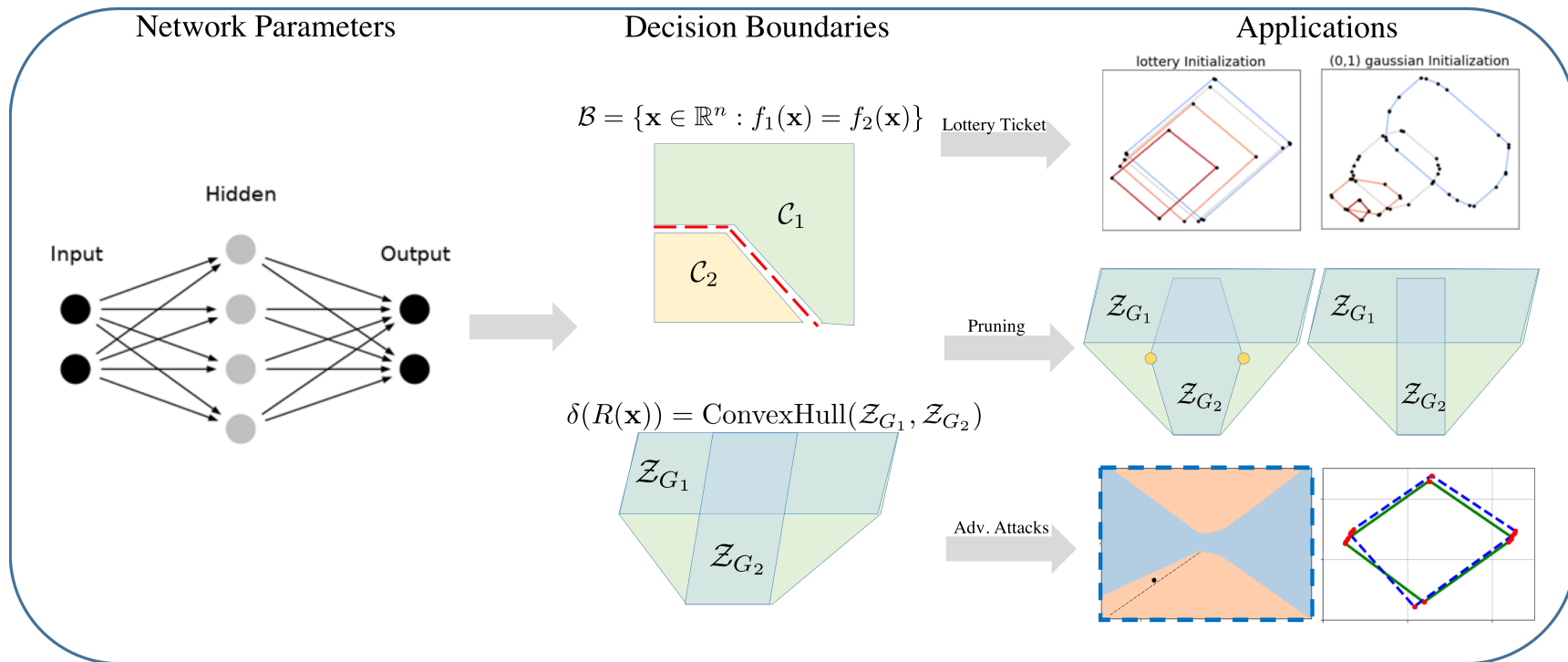


Instead of doing exponential number of forward passes to get the actual decision boundaries, one can study $\delta(R(\mathbf{x}))$.

DNNs in Tropical Geometry

Can we leverage this somewhere?

DNNs in Tropical Geometry



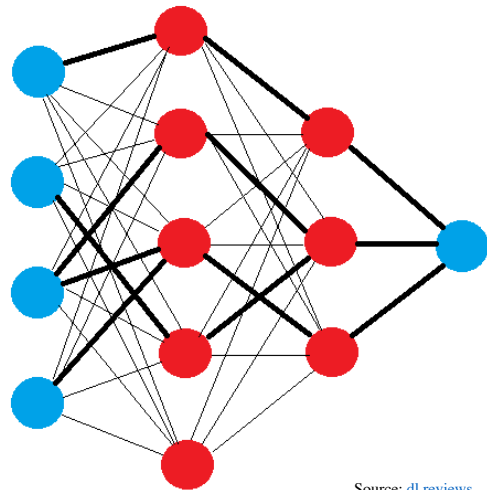
Outline

- Overview on Tropical Geometry
- DNNs in Tropical Geometry
- Lottery Ticket Hypothesis in TG
- Tropical Network Pruning
- Tropical Adversarial Attacks

Lottery Ticket Hypothesis in TG.

What is “Lottery Ticket Hypothesis”?

“There exist **sparse** trainable sub-networks of dense, randomly-initialized, feed-forward networks that when trained in isolation perform as well as the original network in a similar number of iterations.”

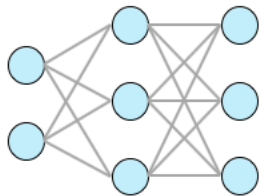


Source: [dl.reviews](#).

Lottery Ticket Hypothesis in TG.

What is “Lottery Ticket Hypothesis”?

1) Initialize

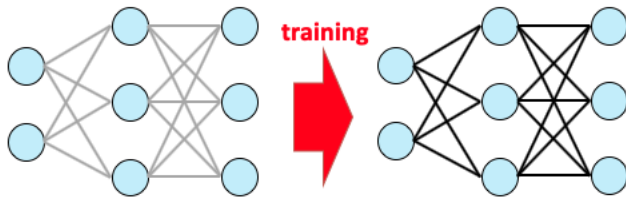


Lottery Ticket Hypothesis in TG.

What is “Lottery Ticket Hypothesis”?

1) Initialize

2) Train



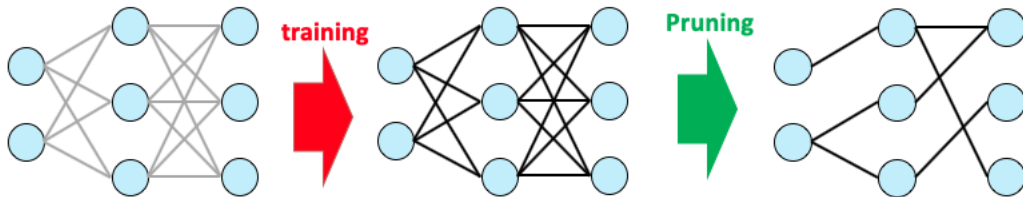
Lottery Ticket Hypothesis in TG.

What is “Lottery Ticket Hypothesis”?

1) Initialize

2) Train

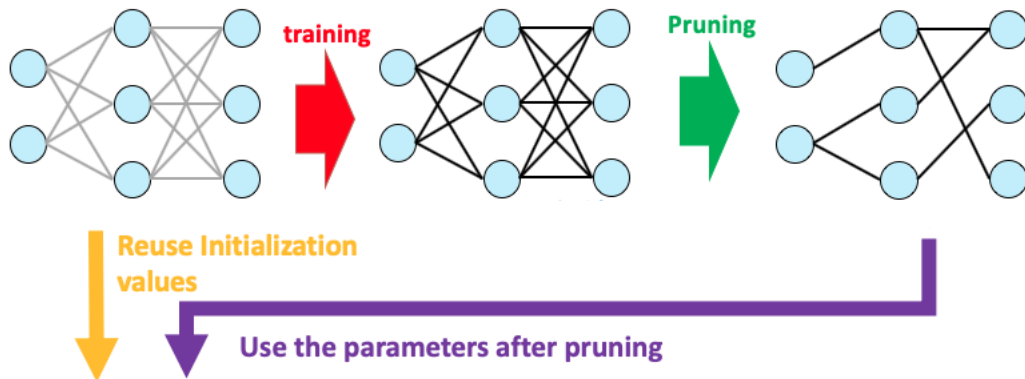
3) Prune



Lottery Ticket Hypothesis in TG.

What is “Lottery Ticket Hypothesis”?

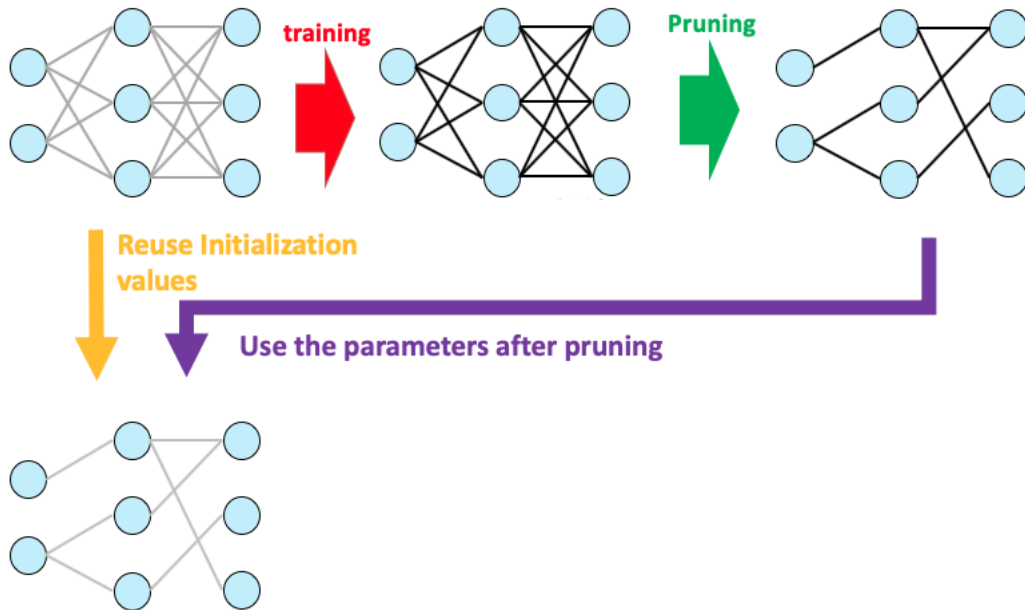
- 1) Initialize
- 2) Train
- 3) Prune



Lottery Ticket Hypothesis in TG.

What is “Lottery Ticket Hypothesis”?

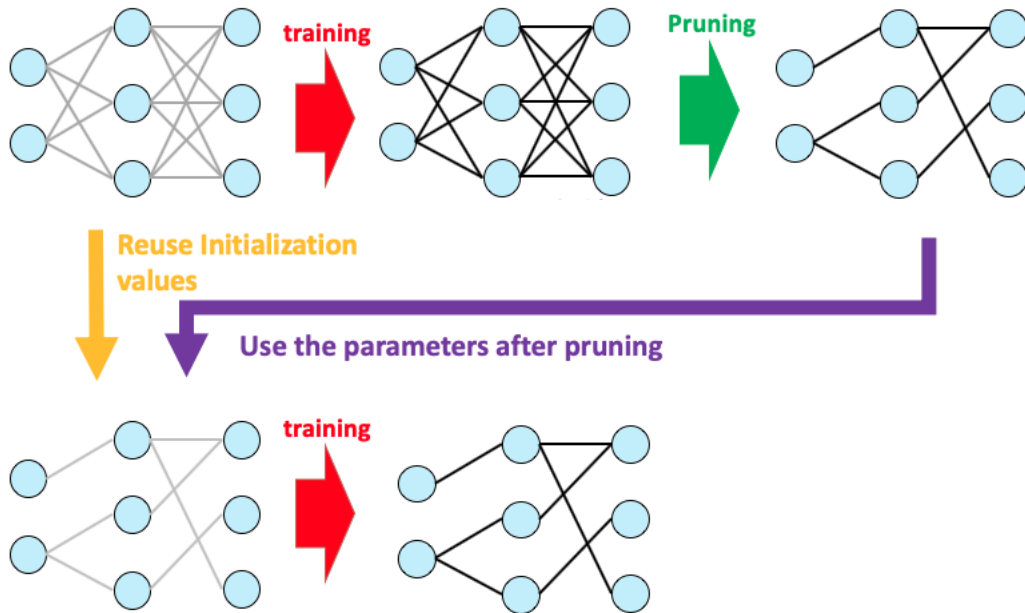
- 1) Initialize
- 2) Train
- 3) Prune
- 4) Initialize



Lottery Ticket Hypothesis in TG.

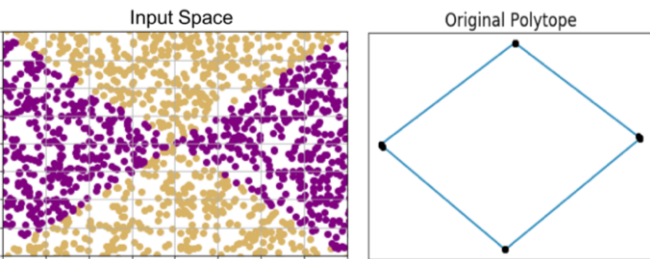
What is “Lottery Ticket Hypothesis”?

- 1) Initialize
- 2) Train
- 3) Prune
- 4) Initialize
- 5) Train



Lottery Ticket Hypothesis in TG.

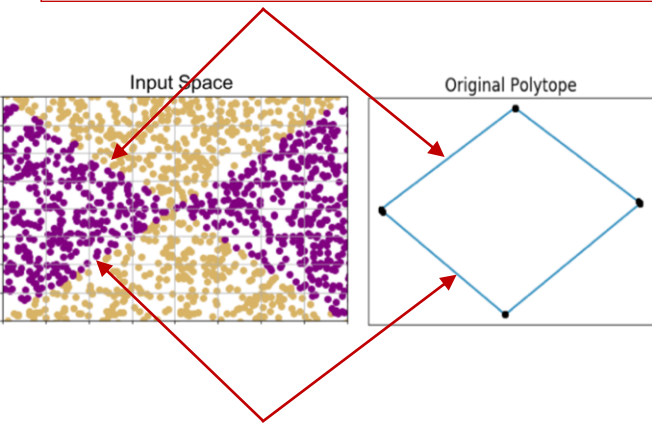
Experiments:



Lottery Ticket Hypothesis in TG.

Experiments:

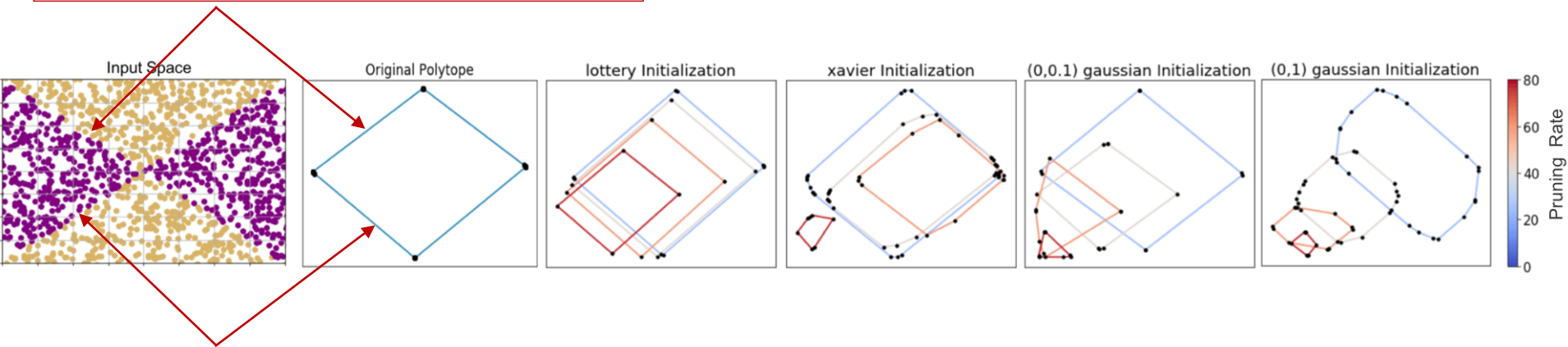
Note that the normal of the decision boundaries polytope are parallel to the decision boundaries



Lottery Ticket Hypothesis in TG.

Experiments:

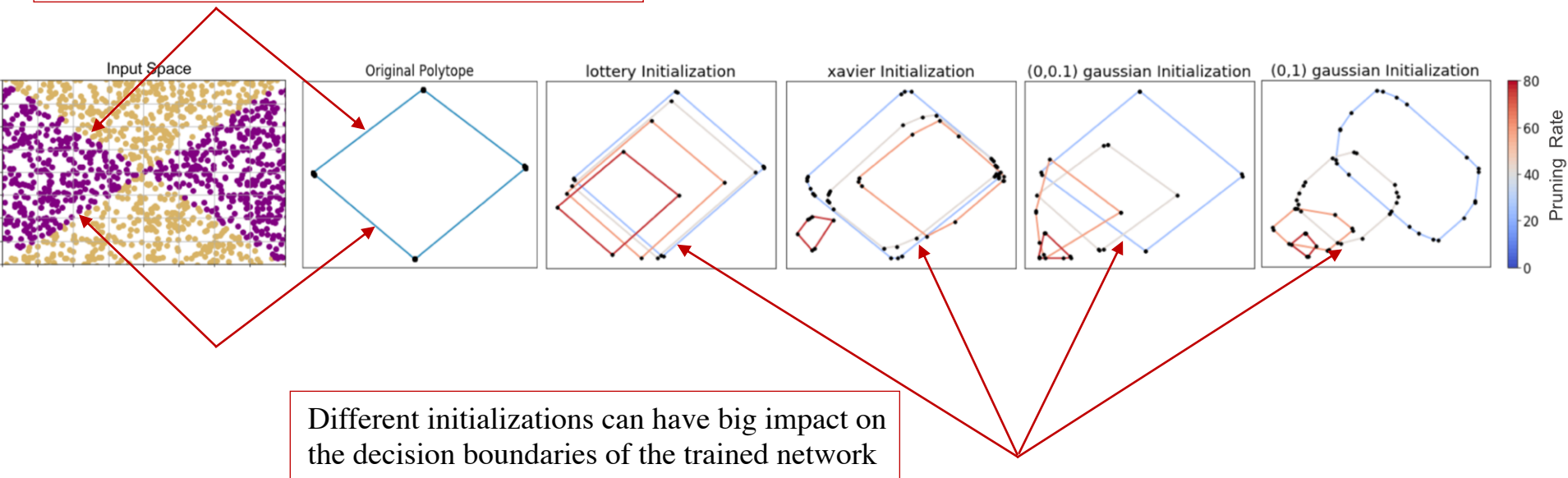
Note that the normal of the decision boundaries polytope are parallel to the decision boundaries



Lottery Ticket Hypothesis in TG.

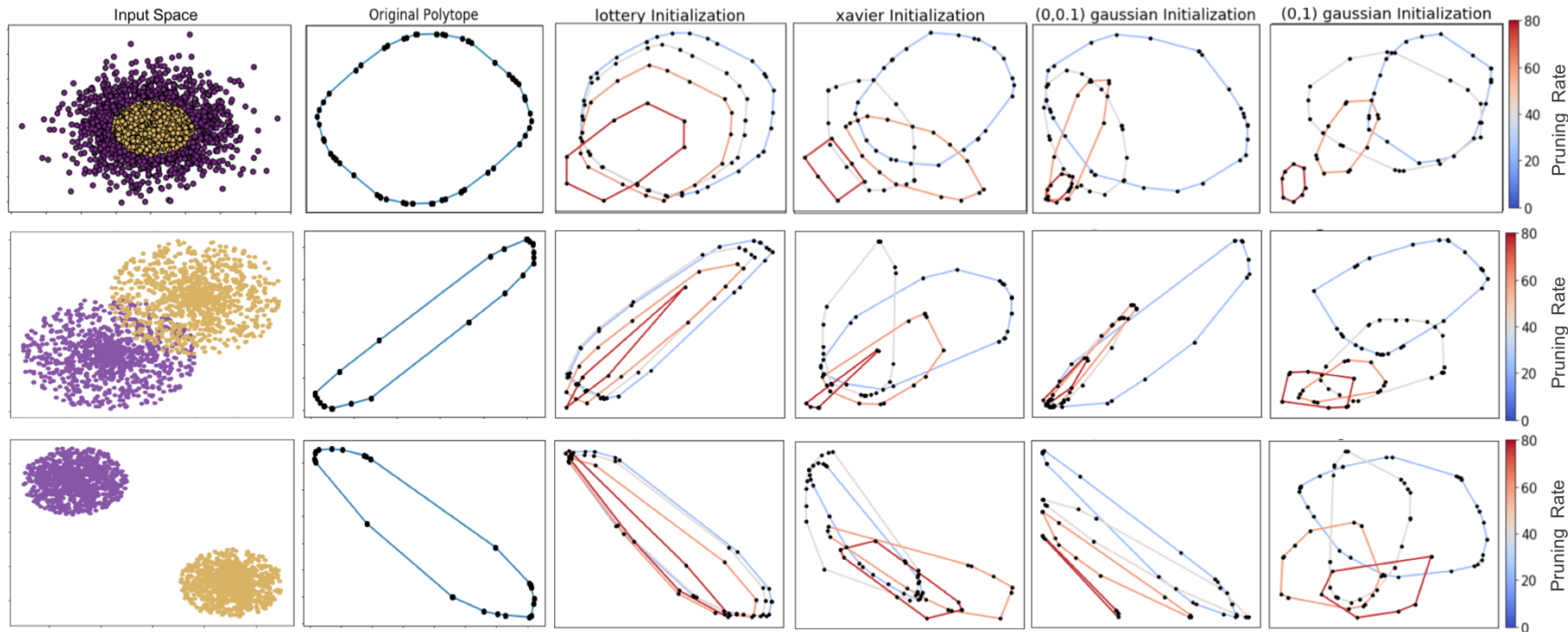
Experiments:

Note that the normal of the decision boundaries polytope are parallel to the decision boundaries



Lottery Ticket Hypothesis in TG.

Experiments:



Lottery Ticket Hypothesis in TG.

Remarks:

Lottery Ticket Hypothesis in TG.

Remarks:

- We do not claim that the lottery initialization is the best in terms of performance, however we investigate its effect on the decision boundaries at multiple pruning iterations.

Lottery Ticket Hypothesis in TG.

Remarks:

- We do not claim that the lottery initialization is the best in terms of performance, however we investigate its effect on the decision boundaries at multiple pruning iterations.
- While extracting the actual decision boundaries might be super expensive, tropical geometry delivers its promises about giving vital information about the decision boundaries.

Outline

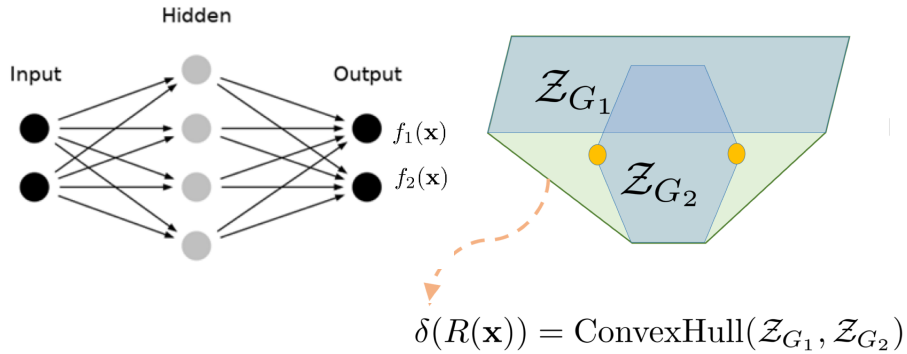
- Overview on Tropical Geometry
- DNNs in Tropical Geometry
- Lottery Ticket Hypothesis in TG
- Tropical Network Pruning
- Tropical Adversarial Attacks

Tropical Network Pruning

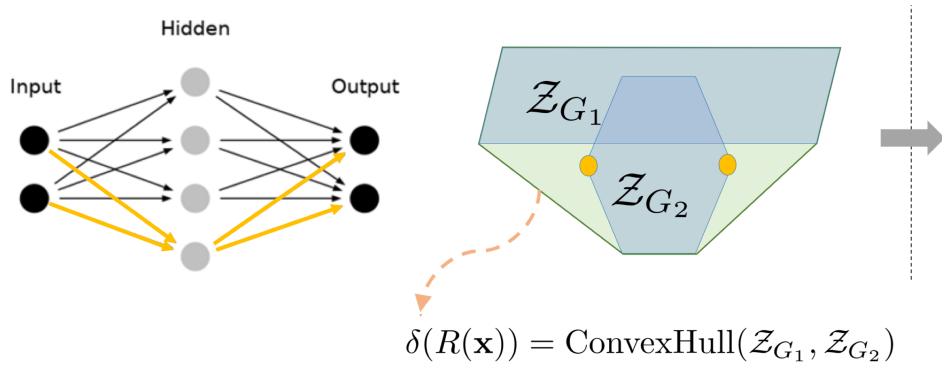
Question

If the shape of the decision boundaries polytope is what matters, can we prune parts of the network that do not contribute to the shape of the decision boundaries polytope ?

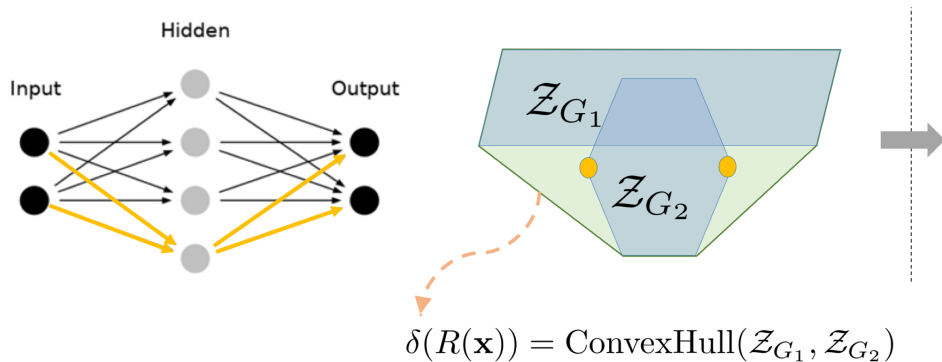
Tropical Network Pruning



Tropical Network Pruning

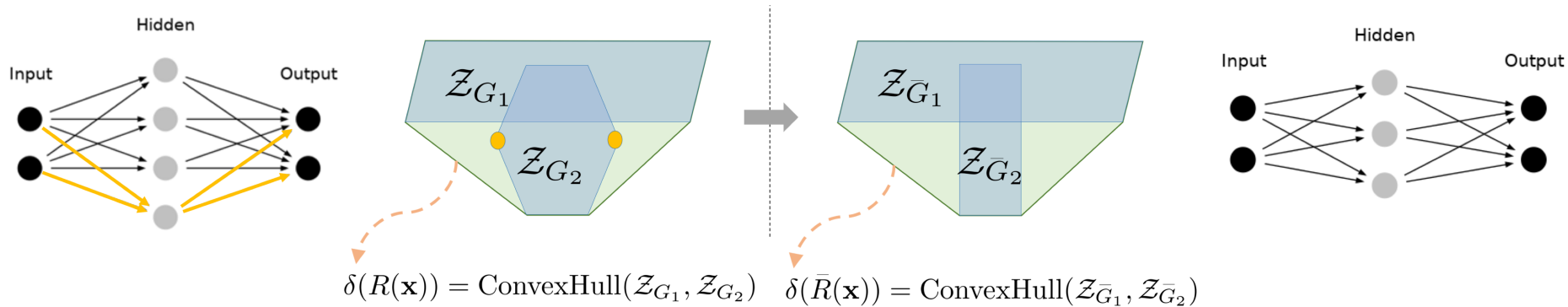


Tropical Network Pruning



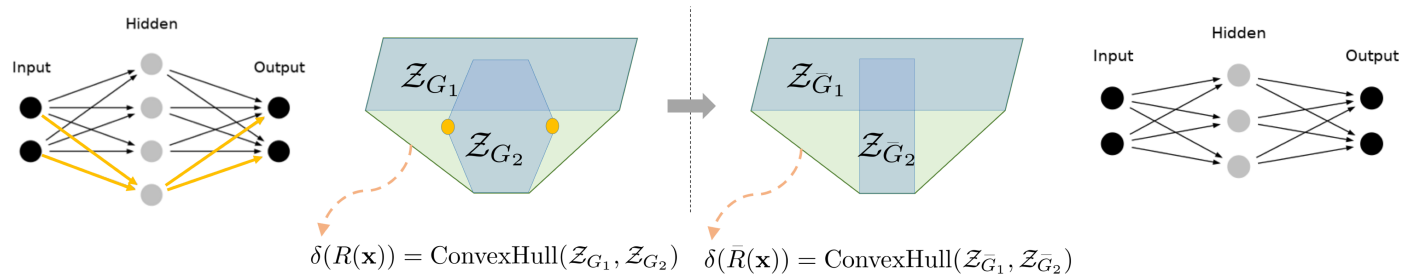
Removing yellow vertices will not affect the shape of the decision boundaries polytope

Tropical Network Pruning



Removing yellow vertices will not affect the shape of the decision boundaries polytope

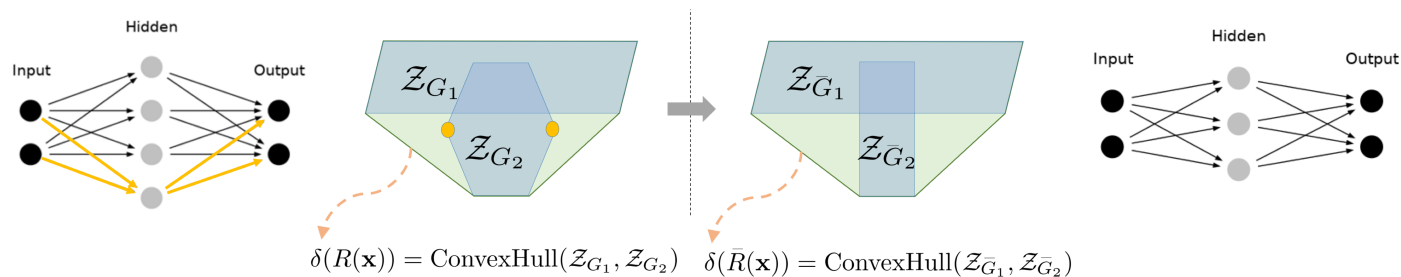
Tropical Network Pruning



For the binary classification problem:

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} d\left(\delta(\tilde{R}(\mathbf{x})), \delta(R(\mathbf{x}))\right)$$

Tropical Network Pruning

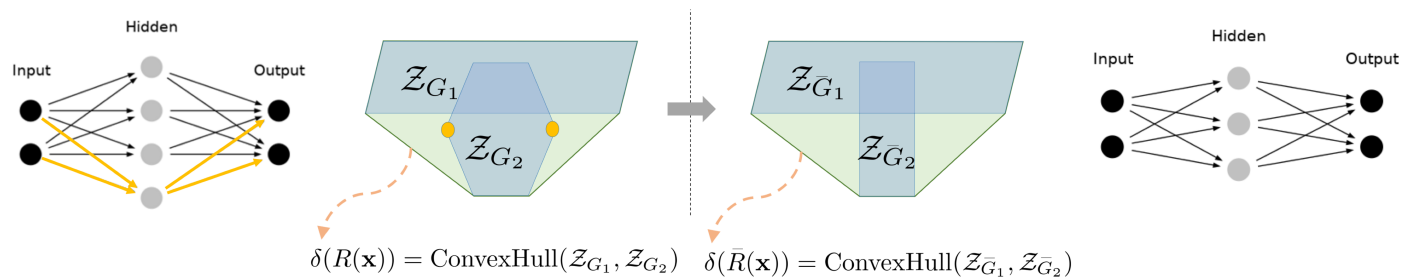


For the binary classification problem:

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} d\left(\delta(\tilde{R}(\mathbf{x})), \delta(R(\mathbf{x}))\right)$$

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} d\left(\text{ConvHull}(Z_{\tilde{G}_1}, Z_{\tilde{G}_2}), \text{ConvHull}(Z_{G_1}, Z_{G_2})\right)$$

Tropical Network Pruning



For the binary classification problem:

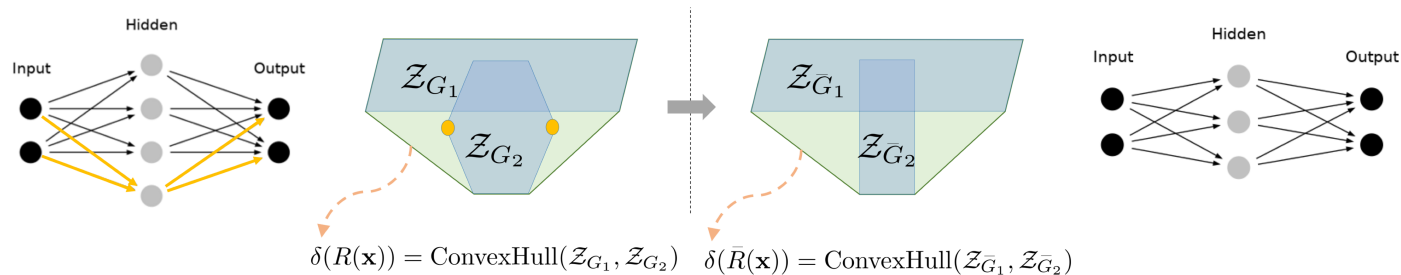
$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} d\left(\delta(\tilde{R}(\mathbf{x})), \delta(R(\mathbf{x}))\right)$$

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} d\left(\text{ConvHull}(Z_{\tilde{G}_1}, Z_{\tilde{G}_2}), \text{ConvHull}(Z_{G_1}, Z_{G_2})\right)$$

Optimizing a ConvHull is challenging

How can we choose a distance function

Tropical Network Pruning



For the binary classification problem:

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} d\left(\delta(\tilde{R}(\mathbf{x})), \delta(R(\mathbf{x}))\right)$$

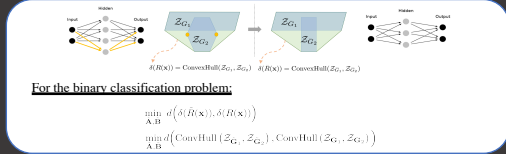
$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} d\left(\text{ConvHull}(Z_{\tilde{G}_1}, Z_{\tilde{G}_2}), \text{ConvHull}(Z_{G_1}, Z_{G_2})\right)$$

Optimizing a ConvHull is challenging

How can we choose a distance function

Instead of preserving the ConvHull, try to preserve each zonotope

Tropical Network Pruning

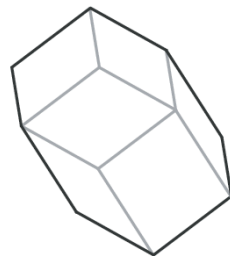


Zonotopes:

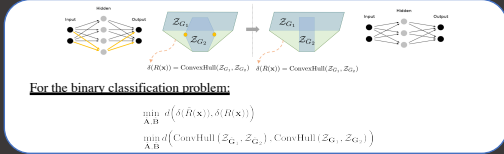
Minkowski sum of finite set of line segment.

Let $\mathbf{g}^1, \dots, \mathbf{g}^p \in \mathbb{R}^n$, then $\mathcal{Z}(\mathbf{g}^1, \dots, \mathbf{g}^p) := \left\{ \sum_{i=1}^p x_i \mathbf{g}^i : 0 \leq x_i \leq 1 \right\}$

Equivalently, $\mathcal{Z}_{\mathbf{G}} := \{ \mathbf{G}^\top \mathbf{x} : \forall \mathbf{x} \in [0, 1]^p \}$, where $\mathbf{G}(i, :) = \mathbf{g}^i^\top$



Tropical Network Pruning

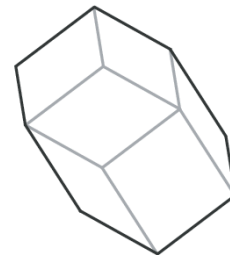


Zonotopes:

Minkowski sum of finite set of line segment.

Let $\mathbf{g}^1, \dots, \mathbf{g}^p \in \mathbb{R}^n$, then $\mathcal{Z}(\mathbf{g}^1, \dots, \mathbf{g}^p) := \left\{ \sum_{i=1}^p x_i \mathbf{g}^i : 0 \leq x_i \leq 1 \right\}$

Equivalently, $\mathcal{Z}_{\mathbf{G}} := \{ \mathbf{G}^\top \mathbf{x} : \forall \mathbf{x} \in [0, 1]^p \}$, where $\mathbf{G}(i, :) = \mathbf{g}^i^\top$

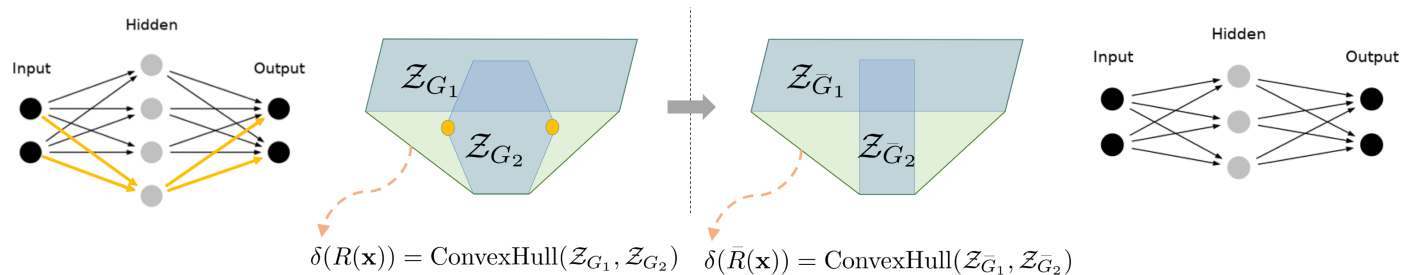


From Theorem 2

$$\tilde{\mathbf{G}}_1 = \text{Diag}[\text{ReLU}(\tilde{\mathbf{B}}(1, :)) + \text{ReLU}(-\tilde{\mathbf{B}}(2, :))] \tilde{\mathbf{A}}$$

$$\tilde{\mathbf{G}}_2 = \text{Diag}[\text{ReLU}(\tilde{\mathbf{B}}(2, :)) + \text{ReLU}(-\tilde{\mathbf{B}}(1, :))] \tilde{\mathbf{A}}$$

Tropical Network Pruning

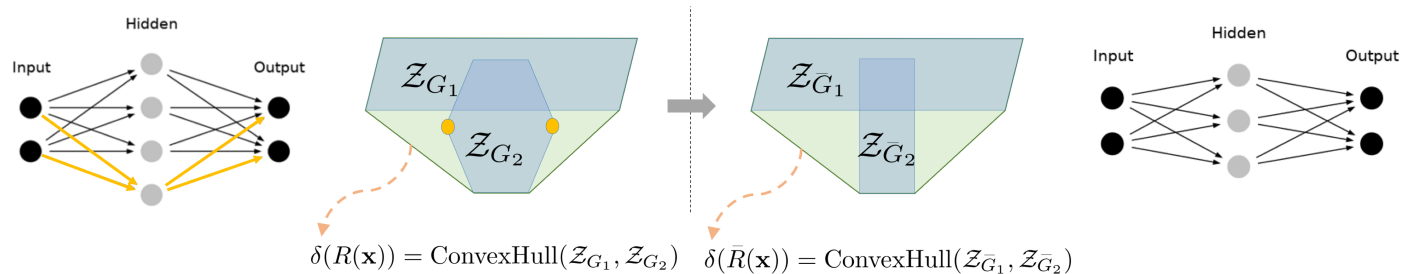


For the binary classification problem:

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} d\left(\delta(\tilde{R}(\mathbf{x})), \delta(R(\mathbf{x}))\right)$$

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} d\left(\text{ConvHull}(Z_{\tilde{G}_1}, Z_{\tilde{G}_2}), \text{ConvHull}(Z_{G_1}, Z_{G_2})\right)$$

Tropical Network Pruning



For the binary classification problem:

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} d\left(\delta(\tilde{R}(\mathbf{x})), \delta(R(\mathbf{x}))\right)$$

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} d\left(\text{ConvHull}(\tilde{Z}_{\tilde{G}_1}, \tilde{Z}_{\tilde{G}_2}), \text{ConvHull}(Z_{G_1}, Z_{G_2})\right)$$

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \frac{1}{2} \left(\left\| \tilde{\mathbf{G}}_1 - \mathbf{G}_1 \right\|_F^2 + \left\| \tilde{\mathbf{G}}_2 - \mathbf{G}_2 \right\|_F^2 \right) + \lambda_1 \left\| \tilde{\mathbf{G}}_1 \right\|_{2,1} + \lambda_2 \left\| \tilde{\mathbf{G}}_2 \right\|_{2,1}$$

Enhances Sparse rows

From Theorem 2

$$\begin{aligned} \tilde{\mathbf{G}}_1 &= \text{Diag}[\text{ReLU}(\tilde{\mathbf{B}}(1, :)) + \text{ReLU}(-\tilde{\mathbf{B}}(2, :))] \tilde{\mathbf{A}} \\ \tilde{\mathbf{G}}_2 &= \text{Diag}[\text{ReLU}(\tilde{\mathbf{B}}(2, :)) + \text{ReLU}(-\tilde{\mathbf{B}}(1, :))] \tilde{\mathbf{A}} \end{aligned}$$

Tropical Network Pruning

For the multi-class problem:

- Study the decision boundaries between all possible pairwise combinations of different classes

Tropical Network Pruning

For the multi-class problem:

- Study the decision boundaries between all possible pairwise combinations of different classes

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{\forall [i,j] \in S} d\left(\text{ConvexHull}\left(\mathbf{z}_{\tilde{\mathbf{G}}_{(i+,j-)}}, \mathbf{z}_{\tilde{\mathbf{G}}_{(j+,i-)}}\right), \text{ConvexHull}\left(\mathbf{z}_{\mathbf{G}_{(i+,j-)}}, \mathbf{z}_{\mathbf{G}_{(j+,i-)}}\right)\right)$$

Tropical Network Pruning

For the multi-class problem:

- Study the decision boundaries between all possible pairwise combinations of different classes

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{\forall [i,j] \in S} d\left(\text{ConvexHull}\left(\mathbf{Z}_{\tilde{\mathbf{G}}_{(i+,j-)}}, \mathbf{Z}_{\tilde{\mathbf{G}}_{(j+,i-)}}\right), \text{ConvexHull}\left(\mathbf{Z}_{\mathbf{G}_{(i+,j-)}}, \mathbf{Z}_{\mathbf{G}_{(j+,i-)}}\right)\right)$$

$$\begin{aligned}\tilde{\mathbf{G}}_{(i+,j-)} &= \text{Diag}\left[\text{ReLU}(\tilde{\mathbf{B}}(i,:)) + \text{ReLU}(-\tilde{\mathbf{B}}(j,:))\right] \tilde{\mathbf{A}} \\ &= \text{Diag}\left[\text{ReLU}(\tilde{\mathbf{B}}(i,:))\right] \tilde{\mathbf{A}} + \text{Diag}\left[\text{ReLU}(-\tilde{\mathbf{B}}(j,:))\right] \tilde{\mathbf{A}}\end{aligned}$$

Tropical Network Pruning

For the multi-class problem:

- Study the decision boundaries between all possible pairwise combinations of different classes

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{\forall [i,j] \in S} d\left(\text{ConvexHull}\left(\mathbf{z}_{\tilde{\mathbf{G}}_{(i+,j-)}}, \mathbf{z}_{\tilde{\mathbf{G}}_{(j+,i-)}}\right), \text{ConvexHull}\left(\mathbf{z}_{\mathbf{G}_{(i+,j-)}}, \mathbf{z}_{\mathbf{G}_{(j+,i-)}}\right)\right)$$

$$\begin{aligned}\tilde{\mathbf{G}}_{(i+,j-)} &= \text{Diag}\left[\text{ReLU}(\tilde{\mathbf{B}}(i,:)) + \text{ReLU}(-\tilde{\mathbf{B}}(j,:))\right] \tilde{\mathbf{A}} \\ &= \underbrace{\text{Diag}\left[\text{ReLU}(\tilde{\mathbf{B}}(i,:))\right] \tilde{\mathbf{A}}}_{\mathbf{G}_{i+}} + \underbrace{\text{Diag}\left[\text{ReLU}(-\tilde{\mathbf{B}}(j,:))\right] \tilde{\mathbf{A}}}_{\mathbf{G}_{j-}}\end{aligned}$$

Tropical Network Pruning

For the multi-class problem:

- Study the decision boundaries between all possible pairwise combinations of different classes

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{\forall [i,j] \in S} d\left(\text{ConvexHull}\left(\mathbf{Z}_{\tilde{\mathbf{G}}_{(i+,j-)}}, \mathbf{Z}_{\tilde{\mathbf{G}}_{(j+,i-)}}\right), \text{ConvexHull}\left(\mathbf{Z}_{\mathbf{G}_{(i+,j-)}}, \mathbf{Z}_{\mathbf{G}_{(j+,i-)}}\right)\right)$$

$$\begin{aligned} \tilde{\mathbf{G}}_{(i+,j-)} &= \text{Diag}\left[\text{ReLU}(\tilde{\mathbf{B}}(i,:)) + \text{ReLU}(-\tilde{\mathbf{B}}(j,:))\right] \tilde{\mathbf{A}} \\ &= \underbrace{\text{Diag}\left[\text{ReLU}(\tilde{\mathbf{B}}(i,:))\right] \tilde{\mathbf{A}}}_{\mathbf{G}_{i+}} + \underbrace{\text{Diag}\left[\text{ReLU}(-\tilde{\mathbf{B}}(j,:))\right] \tilde{\mathbf{A}}}_{\mathbf{G}_{j-}} \end{aligned}$$

Therefore

$$\mathbf{Z}_{\tilde{\mathbf{G}}_{(i+,j-)}} = \mathbf{Z}_{\tilde{\mathbf{G}}_{i+}} \tilde{+} \mathbf{Z}_{\tilde{\mathbf{G}}_{j-}}$$

Minkowski sum

Tropical Network Pruning

For the multi-class problem:

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{\forall [i,j] \in S} d\left(\text{ConvexHull}\left(\mathbf{z}_{\tilde{\mathbf{G}}_{(i+,j-)}}, \mathbf{z}_{\tilde{\mathbf{G}}_{(j+,i-)}}\right), \text{ConvexHull}\left(\mathbf{z}_{\mathbf{G}_{(i+,j-)}}, \mathbf{z}_{\mathbf{G}_{(j+,i-)}}\right)\right)$$

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{\forall [i,j] \in S} d\left(\text{ConvexHull}\left(\mathbf{z}_{\tilde{\mathbf{G}}_{i+}} \tilde{+} \mathbf{z}_{\tilde{\mathbf{G}}_{j-}}, \mathbf{z}_{\tilde{\mathbf{G}}_j^+} \tilde{+} \mathbf{z}_{\tilde{\mathbf{G}}_{i-}}\right), \text{ConvexHull}\left(\mathbf{z}_{\mathbf{G}_{i+}} \tilde{+} \mathbf{z}_{\mathbf{G}_{j-}}, \mathbf{z}_{\mathbf{G}_j^+} \tilde{+} \mathbf{z}_{\mathbf{G}_{i-}}\right)\right)$$

Tropical Network Pruning

For the multi-class problem:

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{\forall [i,j] \in S} d\left(\text{ConvexHull}\left(\mathbf{Z}_{\tilde{\mathbf{G}}_{(i+,j-)}}, \mathbf{Z}_{\tilde{\mathbf{G}}_{(j+,i-)}}\right), \text{ConvexHull}\left(\mathbf{Z}_{\mathbf{G}_{(i+,j-)}}, \mathbf{Z}_{\mathbf{G}_{(j+,i-)}}\right)\right)$$

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{\forall [i,j] \in S} d\left(\text{ConvexHull}\left(\mathbf{Z}_{\tilde{\mathbf{G}}_{i+}} \tilde{+} \mathbf{Z}_{\tilde{\mathbf{G}}_{j-}}, \mathbf{Z}_{\tilde{\mathbf{G}}_j^+} \tilde{+} \mathbf{Z}_{\tilde{\mathbf{G}}_{i-}}\right), \text{ConvexHull}\left(\mathbf{Z}_{\mathbf{G}_{i+}} \tilde{+} \mathbf{Z}_{\mathbf{G}_{j-}}, \mathbf{Z}_{\mathbf{G}_j^+} \tilde{+} \mathbf{Z}_{\mathbf{G}_{i-}}\right)\right)$$

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{\forall [i,j] \in S} \left\| \begin{pmatrix} \tilde{\mathbf{G}}_{i+} \\ \tilde{\mathbf{G}}_{j-} \end{pmatrix} - \begin{pmatrix} \mathbf{G}_{i+} \\ \mathbf{G}_{j-} \end{pmatrix} \right\|_F^2 + \left\| \begin{pmatrix} \tilde{\mathbf{G}}_{i-} \\ \tilde{\mathbf{G}}_{j+} \end{pmatrix} - \begin{pmatrix} \mathbf{G}_{i-} \\ \mathbf{G}_{j+} \end{pmatrix} \right\|_F^2$$

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{\forall [i,j] \in S} \frac{1}{2} \left\| \tilde{\mathbf{G}}_{i+} - \mathbf{G}_{i+} \right\|_F^2 + \frac{1}{2} \left\| \tilde{\mathbf{G}}_{i-} - \mathbf{G}_{i-} \right\|_F^2 + \frac{1}{2} \left\| \tilde{\mathbf{G}}_{j+} - \mathbf{G}_{j+} \right\|_F^2 + \frac{1}{2} \left\| \tilde{\mathbf{G}}_{j-} - \mathbf{G}_{j-} \right\|_F^2$$

Tropical Network Pruning

For the multi-class problem:

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{\forall [i,j] \in S} d\left(\text{ConvexHull}\left(\mathbf{Z}_{\tilde{\mathbf{G}}_{(i+,j-)}}, \mathbf{Z}_{\tilde{\mathbf{G}}_{(j+,i-)}}\right), \text{ConvexHull}\left(\mathbf{Z}_{\mathbf{G}_{(i+,j-)}}, \mathbf{Z}_{\mathbf{G}_{(j+,i-)}}\right)\right)$$

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{\forall [i,j] \in S} d\left(\text{ConvexHull}\left(\mathbf{Z}_{\tilde{\mathbf{G}}_{i+}} \tilde{+} \mathbf{Z}_{\tilde{\mathbf{G}}_{j-}}, \mathbf{Z}_{\tilde{\mathbf{G}}_j^+} \tilde{+} \mathbf{Z}_{\tilde{\mathbf{G}}_{i-}}\right), \text{ConvexHull}\left(\mathbf{Z}_{\mathbf{G}_{i+}} \tilde{+} \mathbf{Z}_{\mathbf{G}_{j-}}, \mathbf{Z}_{\mathbf{G}_j^+} \tilde{+} \mathbf{Z}_{\mathbf{G}_{i-}}\right)\right)$$

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{\forall [i,j] \in S} \left\| \begin{pmatrix} \tilde{\mathbf{G}}_{i+} \\ \tilde{\mathbf{G}}_{j-} \end{pmatrix} - \begin{pmatrix} \mathbf{G}_{i+} \\ \mathbf{G}_{j-} \end{pmatrix} \right\|_F^2 + \left\| \begin{pmatrix} \tilde{\mathbf{G}}_{i-} \\ \tilde{\mathbf{G}}_{j+} \end{pmatrix} - \begin{pmatrix} \mathbf{G}_{i-} \\ \mathbf{G}_{j+} \end{pmatrix} \right\|_F^2$$

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{\forall [i,j] \in S} \frac{1}{2} \left\| \tilde{\mathbf{G}}_{i+} - \mathbf{G}_{i+} \right\|_F^2 + \frac{1}{2} \left\| \tilde{\mathbf{G}}_{i-} - \mathbf{G}_{i-} \right\|_F^2 + \frac{1}{2} \left\| \tilde{\mathbf{G}}_{j+} - \mathbf{G}_{j+} \right\|_F^2 + \frac{1}{2} \left\| \tilde{\mathbf{G}}_{j-} - \mathbf{G}_{j-} \right\|_F^2$$

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{i=1}^k \frac{1}{2} (k-1) \left(\left\| \tilde{\mathbf{G}}_{i+} - \mathbf{G}_{i+} \right\|_F^2 + \left\| \tilde{\mathbf{G}}_{i-} - \mathbf{G}_{i-} \right\|_F^2 \right).$$

Tropical Network Pruning

For the multi-class problem:

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{\forall [i,j] \in S} d\left(\text{ConvexHull}\left(\mathbf{z}_{\tilde{\mathbf{G}}_{(i+,j-)}}, \mathbf{z}_{\tilde{\mathbf{G}}_{(j+,i-)}}\right), \text{ConvexHull}\left(\mathbf{z}_{\mathbf{G}_{(i+,j-)}}, \mathbf{z}_{\mathbf{G}_{(j+,i-)}}\right)\right)$$

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{i=1}^k \frac{1}{2} (k-1) \left(\left\| \tilde{\mathbf{G}}_{i+} - \mathbf{G}_{i+} \right\|_F^2 + \left\| \tilde{\mathbf{G}}_{i-} - \mathbf{G}_{i-} \right\|_F^2 \right).$$

Tropical Network Pruning

For the multi-class problem:

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{\forall [i,j] \in S} d\left(\text{ConvexHull}\left(\mathbf{Z}_{\tilde{\mathbf{G}}_{(i+,j-)}}, \mathbf{Z}_{\tilde{\mathbf{G}}_{(j+,i-)}}\right), \text{ConvexHull}\left(\mathbf{Z}_{\mathbf{G}_{(i+,j-)}}, \mathbf{Z}_{\mathbf{G}_{(j+,i-)}}\right)\right)$$

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{i=1}^k \frac{1}{2} (k-1) \left(\left\| \tilde{\mathbf{G}}_{i+} - \mathbf{G}_{i+} \right\|_F^2 + \left\| \tilde{\mathbf{G}}_{i-} - \mathbf{G}_{i-} \right\|_F^2 \right).$$

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{i=1}^k \frac{1}{2} \left(\left\| \tilde{\mathbf{G}}_{i+} - \mathbf{G}_{i+} \right\|_F^2 + \left\| \tilde{\mathbf{G}}_{i-} - \mathbf{G}_{i-} \right\|_F^2 \right) + \lambda \left(\left\| \tilde{\mathbf{G}}_{i+} \right\|_{2,1} + \left\| \tilde{\mathbf{G}}_{i-} \right\|_{2,1} \right)$$

Tropical Network Pruning

For the multi-class problem:

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{\forall [i,j] \in S} d\left(\text{ConvexHull}\left(\mathbf{Z}_{\tilde{\mathbf{G}}_{(i+,j-)}}, \mathbf{Z}_{\tilde{\mathbf{G}}_{(j+,i-)}}\right), \text{ConvexHull}\left(\mathbf{Z}_{\mathbf{G}_{(i+,j-)}}, \mathbf{Z}_{\mathbf{G}_{(j+,i-)}}\right)\right)$$

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{i=1}^k \frac{1}{2} (k-1) \left(\left\| \tilde{\mathbf{G}}_{i+} - \mathbf{G}_{i+} \right\|_F^2 + \left\| \tilde{\mathbf{G}}_{i-} - \mathbf{G}_{i-} \right\|_F^2 \right).$$

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{i=1}^k \frac{1}{2} \left(\left\| \tilde{\mathbf{G}}_{i+} - \mathbf{G}_{i+} \right\|_F^2 + \left\| \tilde{\mathbf{G}}_{i-} - \mathbf{G}_{i-} \right\|_F^2 \right) + \lambda \left(\left\| \tilde{\mathbf{G}}_{i+} \right\|_{2,1} + \left\| \tilde{\mathbf{G}}_{i-} \right\|_{2,1} \right)$$

Enhances Sparse rows

Tropical Network Pruning

For the multi-class problem:

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{\forall [i,j] \in S} d\left(\text{ConvexHull}\left(\mathbf{z}_{\tilde{\mathbf{G}}_{(i+,j-)}}, \mathbf{z}_{\tilde{\mathbf{G}}_{(j+,i-)}}\right), \text{ConvexHull}\left(\mathbf{z}_{\mathbf{G}_{(i+,j-)}}, \mathbf{z}_{\mathbf{G}_{(j+,i-)}}\right)\right)$$

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{i=1}^k \frac{1}{2} (k-1) \left(\left\| \tilde{\mathbf{G}}_{i+} - \mathbf{G}_{i+} \right\|_F^2 + \left\| \tilde{\mathbf{G}}_{i-} - \mathbf{G}_{i-} \right\|_F^2 \right).$$

$$\min_{\tilde{\mathbf{A}}, \tilde{\mathbf{B}}} \sum_{i=1}^k \frac{1}{2} \left(\left\| \tilde{\mathbf{G}}_{i+} - \mathbf{G}_{i+} \right\|_F^2 + \left\| \tilde{\mathbf{G}}_{i-} - \mathbf{G}_{i-} \right\|_F^2 \right) + \lambda \left(\left\| \tilde{\mathbf{G}}_{i+} \right\|_{2,1} + \left\| \tilde{\mathbf{G}}_{i-} \right\|_{2,1} \right)$$

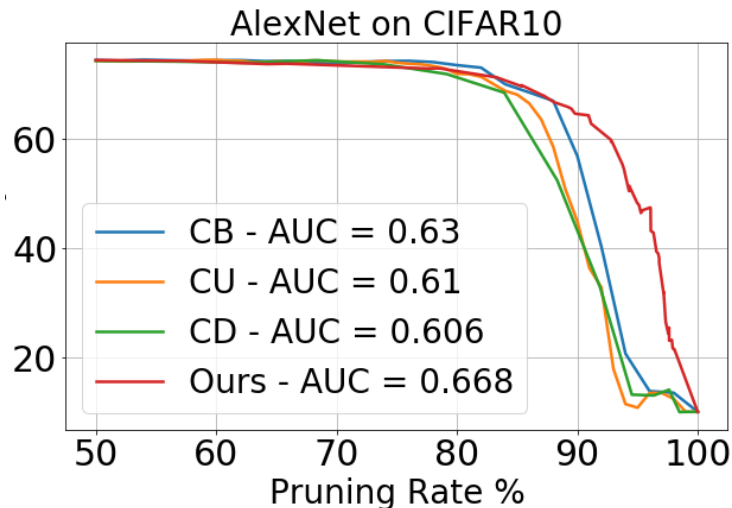
Enhances Sparse rows

Solved by alternating optimization

Tropical Network Pruning

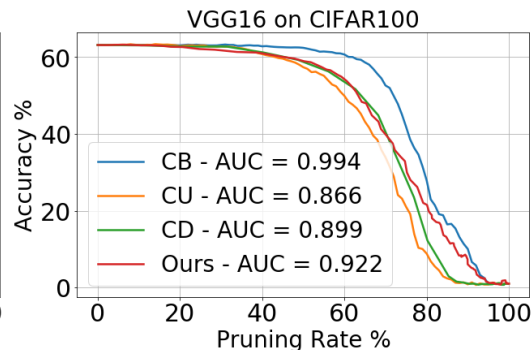
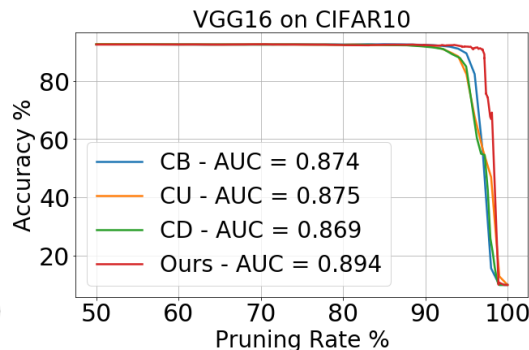
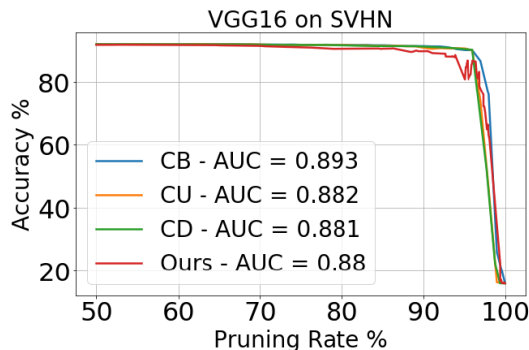
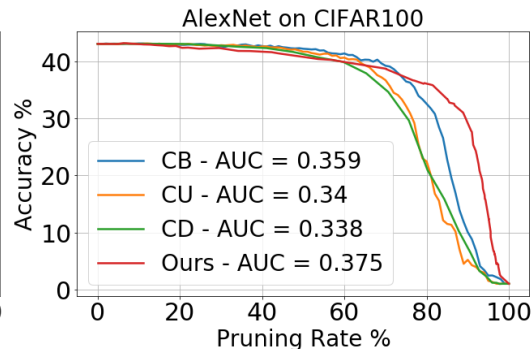
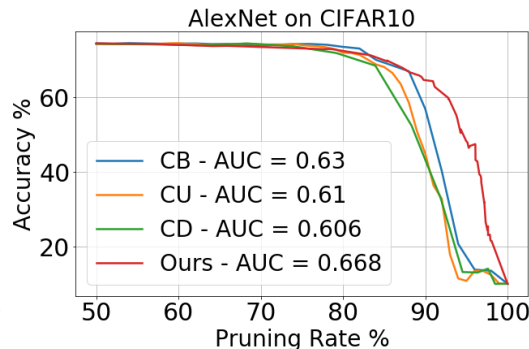
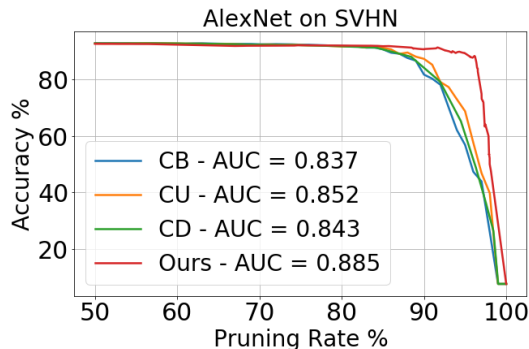
Experiments.

- Although of doing a simple approximation of the original objective from preserving the actual decision boundaries polytope to preserve the generator matrices of the zonotopes, our methods performs well.
- One can do a better approximation to the main objective and thus get much better performance; pruning more and lose less in terms of accuracy.



Tropical Network Pruning

Experiments.

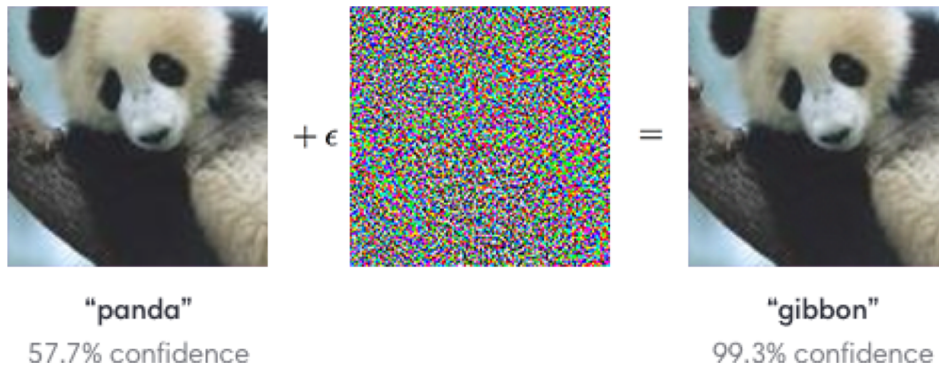


Outline

- Overview on Tropical Geometry
- DNNs in Tropical Geometry
- Lottery Ticket Hypothesis in TG
- Tropical Network Pruning
- Tropical Adversarial Attacks

Tropical Adversarial Attacks

Overview on Adversarial Attacks



Tropical Adversarial Attacks

Overview on Adversarial Attacks

- Generating an adversary can be as easy as doing a single gradient ascent on an objective function.

Tropical Adversarial Attacks

Overview on Adversarial Attacks

- Generating an adversary can be as easy as doing a single gradient ascent on an objective function.

$$\min_{\eta} \mathcal{D}(\eta) \quad \text{s.t.} \quad \underset{i}{\operatorname{argmax}} f_i(\mathbf{x}_0 + \eta) = t \neq c$$

Tropical Adversarial Attacks

Overview on Adversarial Attacks

- Generating an adversary can be as easy as doing a single gradient ascent on an objective function.

$$\min_{\eta} \mathcal{D}(\eta) \quad \text{s.t.} \quad \operatorname{argmax}_i f_i(\mathbf{x}_0 + \eta) = t \neq c$$

Limits the injected noise

Perturbed input fools the network

Tropical Adversarial Attacks

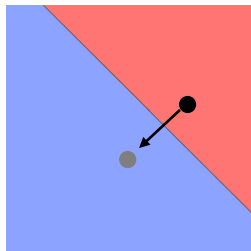
Overview on Adversarial Attacks

- Generating an adversary can be as easy as doing a single gradient ascent on an objective function.

$$\min_{\eta} \mathcal{D}(\eta)$$

Limits the injected noise

$$\text{s.t. } \underset{i}{\operatorname{argmax}} f_i(\mathbf{x}_0 + \eta) = t \neq c$$

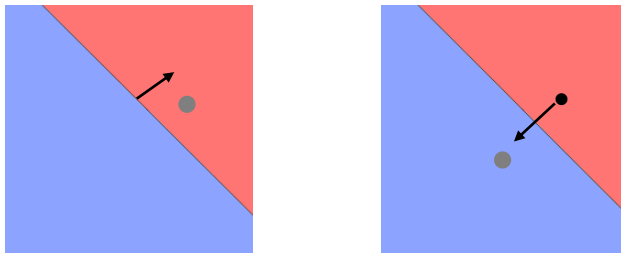


Perturbed input fools the network

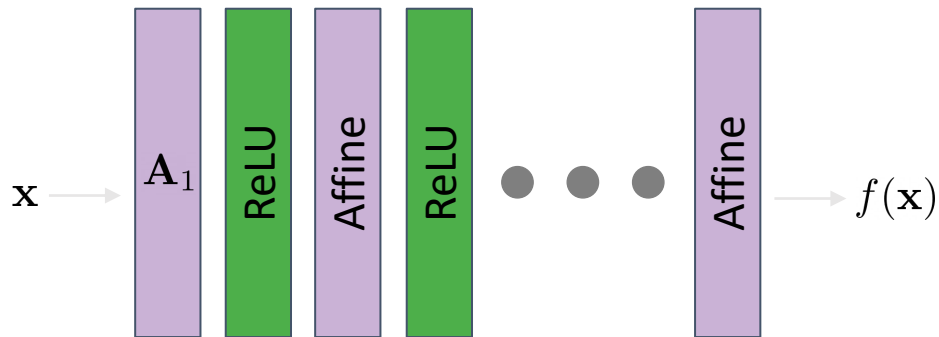
Tropical Adversarial Attacks

Question

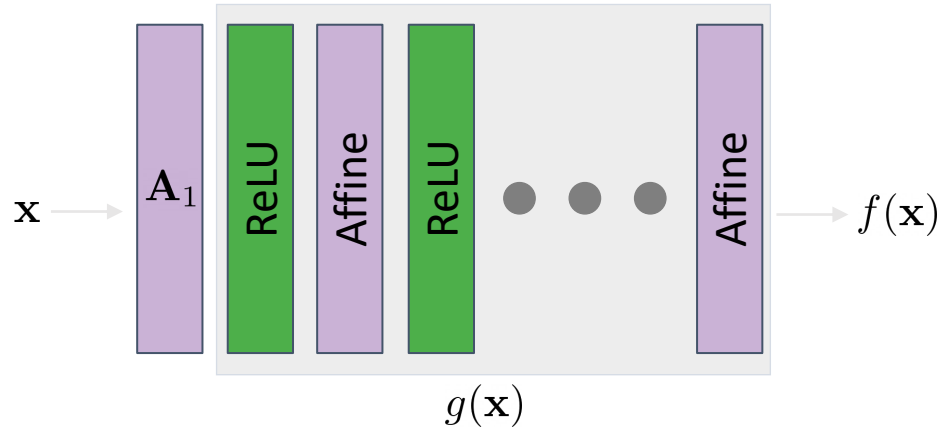
Can we generate input perturbation that is equivalent to perturb the decision boundaries of network through perturbing its parameters?



Tropical Adversarial Attacks

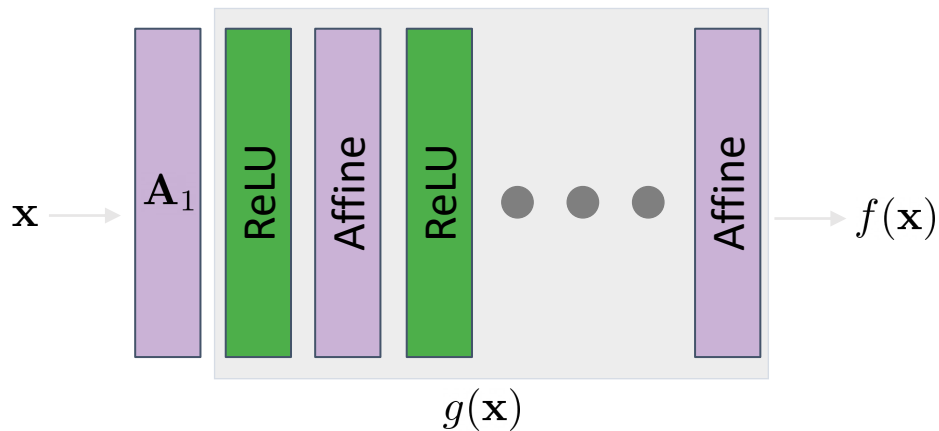


Tropical Adversarial Attacks



$$f(\mathbf{x}) = g(\mathbf{A}_1 \mathbf{x})$$

Tropical Adversarial Attacks

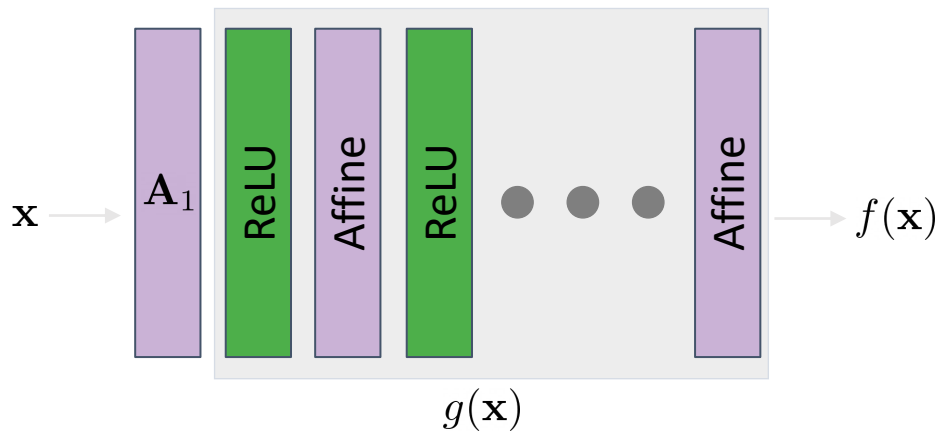


Perturbing network parameters

$$f(\mathbf{x}) = g(\mathbf{A}_1 \mathbf{x})$$

$$g((\mathbf{A}_1 + \xi_{\mathbf{A}_1})\mathbf{x}_0) = g(\mathbf{A}_1 \mathbf{x}_0 + \xi_{\mathbf{A}_1} \mathbf{x}_0)$$

Tropical Adversarial Attacks



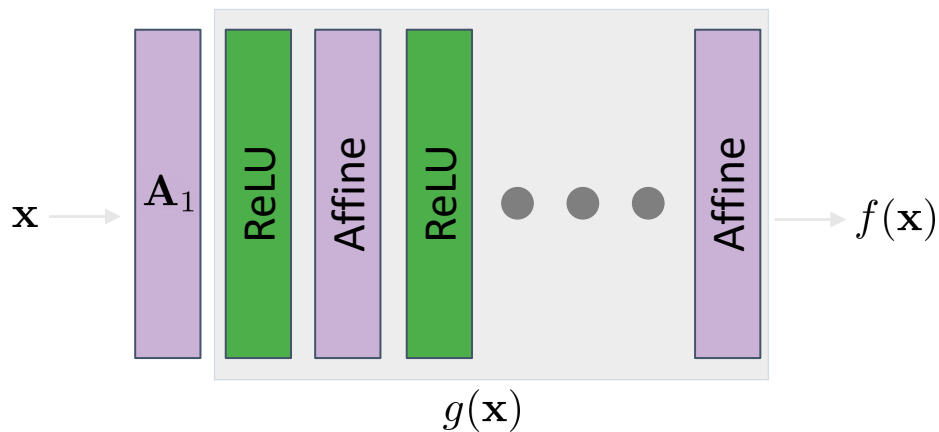
Perturbing network parameters

Perturbing Input pixels

$$f(\mathbf{x}) = g(\mathbf{A}_1 \mathbf{x})$$

$$g((\mathbf{A}_1 + \xi_{\mathbf{A}_1}) \mathbf{x}_0) = g(\mathbf{A}_1 \mathbf{x}_0 + \xi_{\mathbf{A}_1} \mathbf{x}_0) \quad g(\mathbf{A}_1 \mathbf{x}_0 + \mathbf{A}_1 \eta) = f(\mathbf{x}_0 + \eta).$$

Tropical Adversarial Attacks



Perturbing network parameters

Perturbing Input pixels

$$f(\mathbf{x}) = g(\mathbf{A}_1 \mathbf{x})$$

$$g((\mathbf{A}_1 + \xi_{\mathbf{A}_1})\mathbf{x}_0) = g(\mathbf{A}_1 \mathbf{x}_0 + \xi_{\mathbf{A}_1} \mathbf{x}_0) = g(\mathbf{A}_1 \mathbf{x}_0 + \mathbf{A}_1 \boldsymbol{\eta}) = f(\mathbf{x}_0 + \boldsymbol{\eta}).$$

$$\mathbf{A}_1 \boldsymbol{\eta} = \xi_{\mathbf{A}_1} \mathbf{x}$$

Equivalency between input perturbation and parameter perturbation

Tropical Adversarial Attacks

Problem formulation:

$$\mathcal{D}_2(\xi_{\mathbf{A}_1}) = \frac{1}{2} \sum_{j=1}^2 \left\| \text{Diag}(\mathbf{B}^+(j, :)) \xi_{\mathbf{A}_1} \right\|_F^2 + \left\| \text{Diag}(\mathbf{B}^-(j, :)) \xi_{\mathbf{A}_1} \right\|_F^2.$$

$\min_{\eta, \xi_{\mathbf{A}_1}}$

$$\mathcal{D}_1(\eta) + \mathcal{D}_2(\xi_{\mathbf{A}_1})$$

s.t.

$$\begin{aligned} & -\text{loss}(g(\mathbf{A}_1(\mathbf{x}_0 + \eta)), t) \leq -1; & -\text{loss}(g(\mathbf{A}_1 + \xi_{\mathbf{A}_1})\mathbf{x}_0, t) \leq -1; \\ & (\mathbf{x}_0 + \eta) \in [0, 1]^n, \quad \|\eta\|_\infty \leq \epsilon_1; & \|\xi_{\mathbf{A}_1}\|_{\infty, \infty} \leq \epsilon_2, \quad \mathbf{A}_1 \eta = \xi_{\mathbf{A}_1} \mathbf{x}_0. \end{aligned}$$

Tropical Adversarial Attacks

Problem formulation:

$\min_{\eta, \xi_{\mathbf{A}_1}}$

s.t.

Input perturbation fools the network

$$\begin{aligned} & -\text{loss}(g(\mathbf{A}_1(\mathbf{x}_0 + \eta)), t) \leq -1; \\ & (\mathbf{x}_0 + \eta) \in [0, 1]^n, \quad \|\eta\|_\infty \leq \epsilon_1; \end{aligned}$$

$$\mathcal{D}_1(\eta) + \mathcal{D}_2(\xi_{\mathbf{A}_1})$$

Parameter perturbation fools the network

$$\begin{aligned} & -\text{loss}(g(\mathbf{A}_1 + \xi_{\mathbf{A}_1})\mathbf{x}_0, t) \leq -1; \\ & \|\xi_{\mathbf{A}_1}\|_{\infty, \infty} \leq \epsilon_2, \quad \mathbf{A}_1\eta = \xi_{\mathbf{A}_1}\mathbf{x}_0. \end{aligned}$$

$$\mathcal{D}_2(\xi_{\mathbf{A}_1}) = \frac{1}{2} \sum_{j=1}^2 \|\text{Diag}(\mathbf{B}^+(j, :))\xi_{\mathbf{A}_1}\|_F^2 + \|\text{Diag}(\mathbf{B}^-(j, :))\xi_{\mathbf{A}_1}\|_F^2.$$

Tropical Adversarial Attacks

Problem formulation:

$$\min_{\eta, \xi_{\mathbf{A}_1}}$$

s.t.

$$-\text{loss}(g(\mathbf{A}_1(\mathbf{x}_0 + \eta)), t) \leq -1;$$

$$(\mathbf{x}_0 + \eta) \in [0, 1]^n, \quad \|\eta\|_\infty \leq \epsilon_1;$$

$$-\text{loss}(g(\mathbf{A}_1 + \xi_{\mathbf{A}_1})\mathbf{x}_0, t) \leq -1;$$

$$\|\xi_{\mathbf{A}_1}\|_{\infty, \infty} \leq \epsilon_2, \quad \mathbf{A}_1 \eta = \xi_{\mathbf{A}_1} \mathbf{x}_0.$$

$$\mathcal{D}_1(\eta) + \mathcal{D}_2(\xi_{\mathbf{A}_1})$$

$$\mathcal{D}_2(\xi_{\mathbf{A}_1}) = \frac{1}{2} \sum_{j=1}^2 \|\text{Diag}(\mathbf{B}^+(j, :)) \xi_{\mathbf{A}_1}\|_F^2 + \|\text{Diag}(\mathbf{B}^-(j, :)) \xi_{\mathbf{A}_1}\|_F^2.$$

Input perturbation fools the network

Parameter perturbation fools the network

Feasible perturbed input

Bounded perturbation

Equivalency between perturbations

Tropical Adversarial Attacks

Problem formulation:

$$\min_{\eta, \xi_{\mathbf{A}_1}}$$

s.t.

Input perturbation fools the network

Parameter perturbation fools the network

$$\mathcal{D}_1(\eta) + \mathcal{D}_2(\xi_{\mathbf{A}_1})$$

$$\begin{aligned} & -\text{loss}(g(\mathbf{A}_1(\mathbf{x}_0 + \eta)), t) \leq -1; & -\text{loss}(g(\mathbf{A}_1 + \xi_{\mathbf{A}_1})\mathbf{x}_0, t) \leq -1; \\ & (\mathbf{x}_0 + \eta) \in [0, 1]^n, \quad \|\eta\|_\infty \leq \epsilon_1; & \|\xi_{\mathbf{A}_1}\|_{\infty, \infty} \leq \epsilon_2, \quad \mathbf{A}_1\eta = \xi_{\mathbf{A}_1}\mathbf{x}_0. \end{aligned}$$

Feasible perturbed input

Bounded perturbation

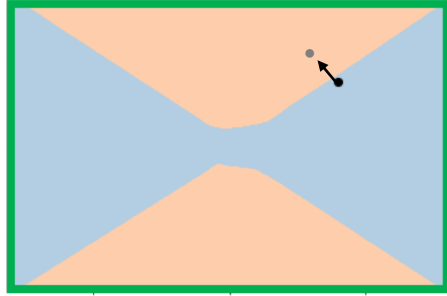
Equivalency between perturbations

$$\mathcal{D}_2(\xi_{\mathbf{A}_1}) = \frac{1}{2} \sum_{j=1}^2 \|\text{Diag}(\mathbf{B}^+(j, :))\xi_{\mathbf{A}_1}\|_F^2 + \|\text{Diag}(\mathbf{B}^-(j, :))\xi_{\mathbf{A}_1}\|_F^2.$$

Solved using penalty method on the last constraint, and each iteration is solved using ADMM

Tropical Adversarial Attacks

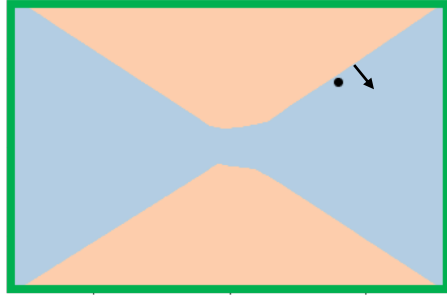
Experiments:



Input perturbation fools the network

Tropical Adversarial Attacks

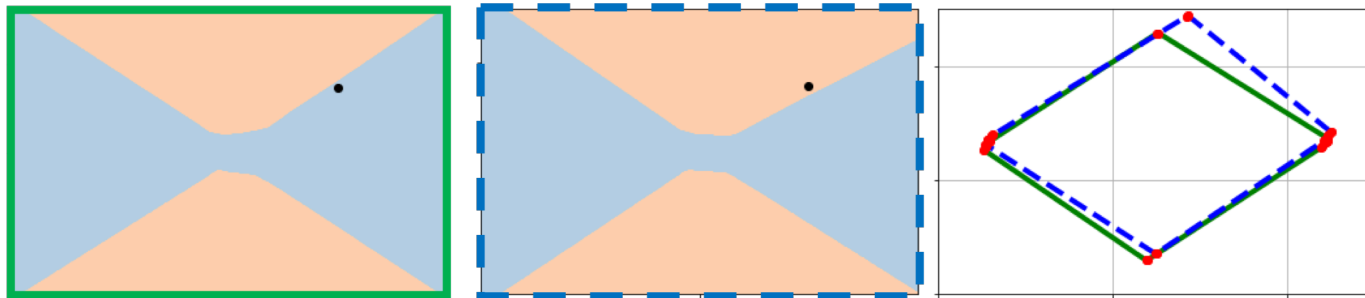
Experiments:



Decision boundaries perturbation fools the network

Tropical Adversarial Attacks

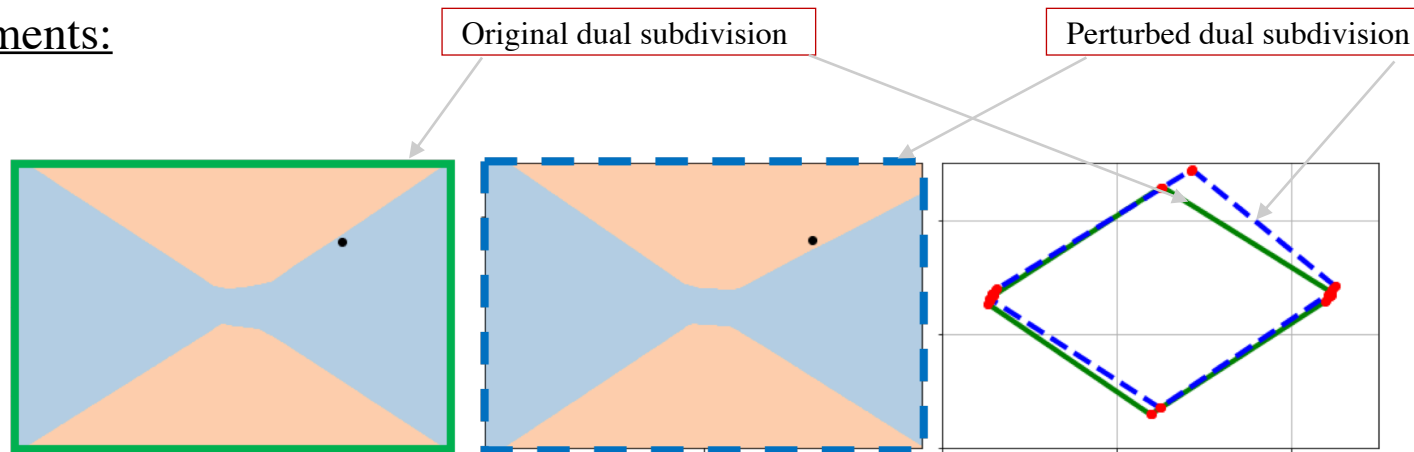
Experiments:



Decision boundaries perturbation fools the network

Tropical Adversarial Attacks

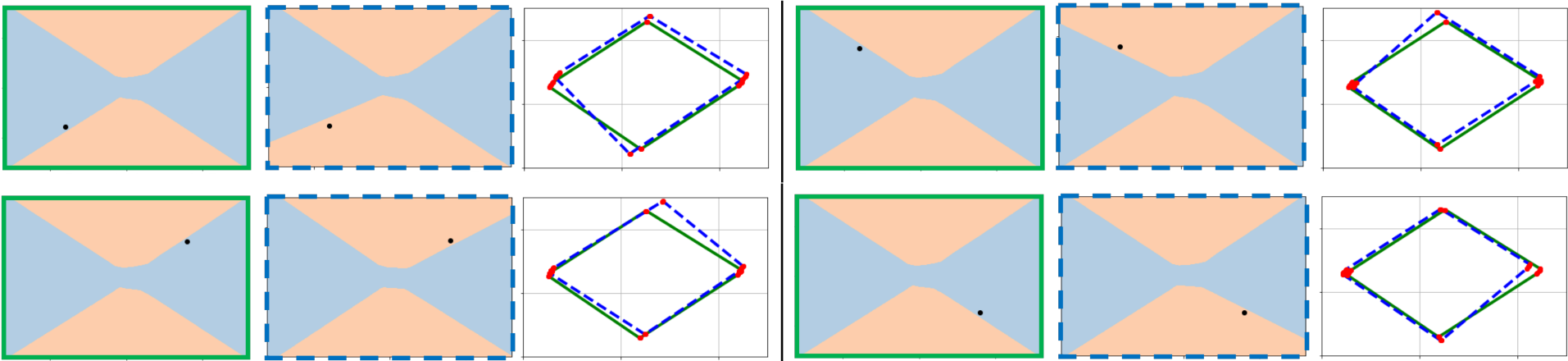
Experiments:



Decision boundaries perturbation fools the network

Tropical Adversarial Attacks

Experiments: Different scenarios of attacking the decision boundaries.



Tropical Adversarial Attacks

Experiments on MNIST:

