Non-Euclidean Monotone Operator Theory for Robustness of Implicit Neural Networks

Saber Jafarpour

University of Colorado Boulder

February 14, 2025

An increase in deployment of learning-based algorithms

Extremely fragile wrt input perturbations

Adversarial Perturbations[1]

Small changes in the input

$\downarrow$

Large changes in the output

[1] C. Szegedy, et al. Intriguing properties of neural networks, ICLR, 2014

An increase in deployment of learning-based algorithms

Extremely fragile wrt input perturbations

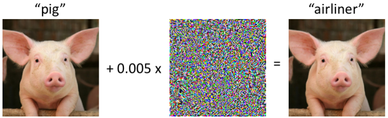


Image credit: MIT CSAIL

[1] C. Szegedy, et al. Intriguing properties of neural networks, ICLR, 2014

An increase in deployment of learning-based algorithms

Extremely fragile wrt input perturbations

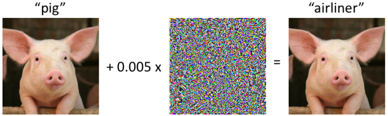


Image credit: MIT CSAIL

- Guaranteeing robustness of learning algorithms is critical in their real-world applications

[1] C. Szegedy, et al. Intriguing properties of neural networks, ICLR, 2014

An increase in deployment of learning-based algorithms

Extremely fragile wrt input perturbations

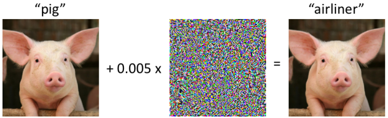


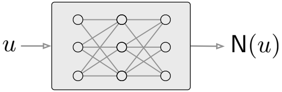
Image credit: MIT CSAIL

- Guaranteeing robustness of learning algorithms is critical in their real-world applications

Robustness of learning algorithm

Input perturbation set $\mathcal{U}$ and unsafe output domain $\mathcal{S}_{\text{unsafe}}$:

$$\mathsf{N}(\mathcal{U}) \cap \mathcal{S}_{\text{unsafe}} = \emptyset.$$



[1] C. Szegedy, et al. Intriguing properties of neural networks, ICLR, 2014

An increase in deployment of learning-based algorithms

Extremely fragile wrt input perturbations

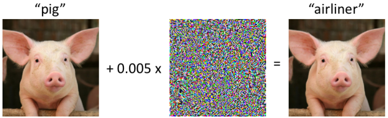


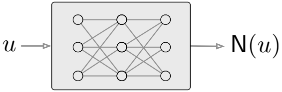
Image credit: MIT CSAIL

- Guaranteeing robustness of learning algorithms is critical in their real-world applications

Robustness of learning algorithm

Input perturbation set $\mathcal{U}$ and unsafe output domain $\mathcal{S}_{\text{unsafe}}$:

$$\mathsf{N}(\mathcal{U}) \cap \mathcal{S}_{\text{unsafe}} = \emptyset.$$



1. **Verification**: For a given learning algorithm can we check its robustness?
2. **Training**: how to design robust learning algorithms?

[1] C. Szegedy, et al. Intriguing properties of neural networks, ICLR, 2014

Goal: over-approximate $\mathsf{N}(\mathcal{U})$ with $\overline{\mathsf{N}}(\mathcal{U})$ and check if $\overline{\mathsf{N}}(\mathcal{U}) \cap \mathcal{S}_{\text{unsafe}} = \emptyset$.

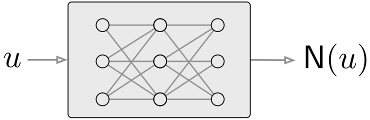
Input-output Lipschitz Bounds

A framework for robustness analysis

Goal: over-approximate $\mathsf{N}(\mathcal{U})$ with $\overline{\mathsf{N}}(\mathcal{U})$ and check if $\overline{\mathsf{N}}(\mathcal{U}) \cap \mathcal{S}_{\mathrm{unsafe}} = \emptyset$.

Lipschitz bound

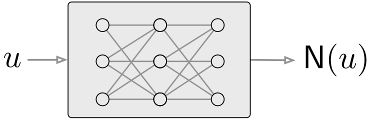
$\|\mathsf{N}(u) - \mathsf{N}(v)\| \leq \mathrm{Lip}\mathsf{N}\|u - v\|$, for every $u, v \in \mathcal{U}$



Goal: over-approximate $\mathsf{N}(\mathcal{U})$ with $\overline{\mathsf{N}}(\mathcal{U})$ and check if $\overline{\mathsf{N}}(\mathcal{U}) \cap \mathcal{S}_{\text{unsafe}} = \emptyset$.

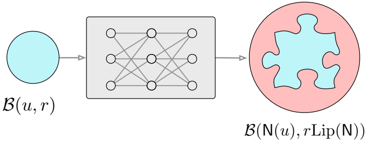
Lipschitz bound

$$\|\mathsf{N}(u) - \mathsf{N}(v)\| \leq \mathrm{Lip}\mathsf{N}\|u - v\|, \text{ for every } u, v \in \mathcal{U}$$



Robustness via Lipschitz bounds

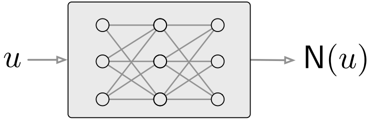
$$\mathsf{N}(\mathcal{B}(u, r)) \subseteq \mathcal{B}(\mathsf{N}(u), r\mathrm{Lip}\mathsf{N}) := \overline{\mathsf{N}}(\mathcal{U})$$



Goal: over-approximate $\mathsf{N}(\mathcal{U})$ with $\overline{\mathsf{N}}(\mathcal{U})$ and check if $\overline{\mathsf{N}}(\mathcal{U}) \cap \mathcal{S}_{\text{unsafe}} = \emptyset$.

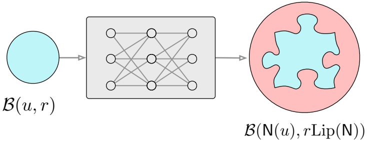
Lipschitz bound

$$\|\mathsf{N}(u) - \mathsf{N}(v)\| \leq \mathrm{Lip}\mathsf{N}\|u - v\|, \text{ for every } u, v \in \mathcal{U}$$



Robustness via Lipschitz bounds

$$\mathsf{N}(\mathcal{B}(u, r)) \subseteq \mathcal{B}(\mathsf{N}(u), r\mathrm{Lip}\mathsf{N}) := \overline{\mathsf{N}}(\mathcal{U})$$



- A. Virmaux and K. Scaman. Lipschitz regularity of deep neural networks: analysis and efficient estimation. *NeurIPS*, 2018
- Mahyar Fazlyab, et al, Efficient and Accurate Estimation of Lipschitz Constants for Deep Neural Networks. *NeurIPS*, 2019
- Alexandre Araujo, et al, A Unified Algebraic Perspective on Lipschitz Neural Networks, *ICLR*, 2023

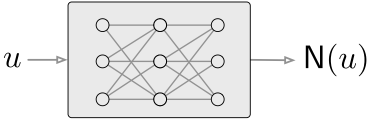
Input-output Lipschitz Bounds

A framework for robustness analysis

Goal: over-approximate $\mathsf{N}(\mathcal{U})$ with $\overline{\mathsf{N}}(\mathcal{U})$ and check if $\overline{\mathsf{N}}(\mathcal{U}) \cap \mathcal{S}_{\mathrm{unsafe}} = \emptyset$.

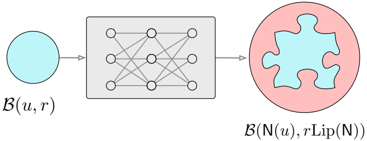
Lipschitz bound

$$\|\mathsf{N}(u) - \mathsf{N}(v)\| \leq \mathrm{Lip}\mathsf{N}\|u - v\|, \text{ for every } u, v \in \mathcal{U}$$



Robustness via Lipschitz bounds

$$\mathsf{N}(\mathcal{B}(u,r)) \subseteq \mathcal{B}(\mathsf{N}(u), r\mathrm{Lip}\mathsf{N}) := \overline{\mathsf{N}}(\mathcal{U})$$



In this talk: use Monotone Operator Theory to estimate Lipschitz bound of learning algorithms

Let $\mathcal{H}$ be a Hilbert space with inner product $\langle\cdot,\cdot\rangle_{\mathcal{H}}$. Then $\mathsf{F}:\mathcal{H}\to\mathcal{H}$ is a monotone operator if

$$\langle \mathsf{F}(x)-\mathsf{F}(y), x-y\rangle_{\mathcal{H}} \geq 0, \qquad \text{for all } x,y$$

and is strongly monotone with parameter $m\geq 0$ if

$$\langle \mathsf{F}(x)-\mathsf{F}(y), x-y\rangle_{\mathcal{H}} \geq m\|x-y\|_{\mathcal{H}}^2, \qquad \text{for all } x,y$$

Let $\mathcal{H}$ be a Hilbert space with inner product $\langle\cdot,\cdot\rangle_{\mathcal{H}}$. Then $\mathsf{F}:\mathcal{H}\to\mathcal{H}$ is a monotone operator if

$$\langle \mathsf{F}(x)-\mathsf{F}(y), x-y\rangle_{\mathcal{H}} \geq 0, \qquad \text{for all } x,y$$

and is strongly monotone with parameter $m\geq 0$ if

$$\langle \mathsf{F}(x)-\mathsf{F}(y), x-y\rangle_{\mathcal{H}} \geq m\|x-y\|^2_{\mathcal{H}}, \qquad \text{for all } x,y$$

- Minty (1962), Browder (1967), Rockafellar (1966)
- Bauschke and Combettes (2017), Ryu and and Boyd (2016)

Monotone Operator Theory

A classical framework in functional analysis

Let $\mathcal{H}$ be a Hilbert space with inner product $\langle\cdot,\cdot\rangle_{\mathcal{H}}$. Then $\mathsf{F}:\mathcal{H}\to\mathcal{H}$ is a monotone operator if

$$\langle \mathsf{F}(x)-\mathsf{F}(y), x-y\rangle_{\mathcal{H}} \geq 0, \qquad \text{for all } x,y$$

and is strongly monotone with parameter $m\geq 0$ if

$$\langle \mathsf{F}(x)-\mathsf{F}(y), x-y\rangle_{\mathcal{H}} \geq m\|x-y\|_{\mathcal{H}}^2, \qquad \text{for all } x,y$$

- Minty (1962), Browder (1967), Rockafellar (1966)
- Bauschke and Combettes (2017), Ryu and and Boyd (2016)

Optimization

$$\min_{x\in\mathbb{R}^n} \quad f(x)$$

f is convex iff ∇f is monotone

Dynamical systems

$$\dot{x} = f(x)$$

f is contracting wrt ℓ_2-norm iff $-f$ is strongly monotone

Let $\mathcal{H}$ be a Hilbert space with inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$. Then $\mathsf{F} : \mathcal{H} \to \mathcal{H}$ is a monotone operator if

$$\langle \mathsf{F}(x) - \mathsf{F}(y), x - y \rangle_{\mathcal{H}} \geq 0, \qquad \text{for all } x, y$$

and is strongly monotone with parameter $m \geq 0$ if

$$\langle \mathsf{F}(x) - \mathsf{F}(y), x - y \rangle_{\mathcal{H}} \geq m \|x - y\|_{\mathcal{H}}^2, \qquad \text{for all } x, y$$

- Minty (1962), Browder (1967), Rockafellar (1966)
- Bauschke and Combettes (2017), Ryu and and Boyd (2016)

Theorem (classical)

Let $\mathsf{F} : \mathbb{R}^n \to \mathbb{R}^n$ be a strongly monotone operator wrt to $\langle \cdot, \cdot \rangle_{\mathbb{R}^n}$, then

1. $\mathsf{F}(x) = 0$ has a unique solution x^*, and
2. x^* can be computed using the average iteration $x_{k+1} = (1 - \theta) x_k + \theta \mathsf{F}(x_k)$.

Definition: Logarithmic norm

Given a matrix $A \in \mathbb{R}^{n\times n}$ and a norm $\|\cdot\|$:

$$\mu_{\|\cdot\|}(A) := \lim_{h\to 0^+} \frac{\|I_n + hA\| - 1}{h}$$

$$\mu_2(A) = \tfrac{1}{2}\lambda_{\mathsf{max}}(A + A^\top)$$

$$\mu_1(A) = \max_j \left(a_{jj} + \sum\nolimits_{i\neq j} |a_{ij}|\right)$$

$$\mu_\infty(A) = \max_i \left(a_{ii} + \sum\nolimits_{j\neq i} |a_{ij}|\right)$$

- directional derivative of matrix norm $\|\cdot\|$ in direction of A at point I_n.

[3] A. Davydov, **SJ**, A. V. Proskurnikov, and F. Bullo. Non-Euclidean monotone operator theory and applications. JMLR, 2024

Definition: Logarithmic norm

Given a matrix $A \in \mathbb{R}^{n\times n}$ and a norm $\|\cdot\|$:

$$\mu_{\|\cdot\|}(A) := \lim_{h\to 0^+} \frac{\|I_n + hA\| - 1}{h}$$

$$\mu_2(A) = \tfrac{1}{2}\lambda_{\mathsf{max}}(A + A^\top)$$

$$\mu_1(A) = \max_j \left(a_{jj} + \sum\nolimits_{i\neq j} |a_{ij}|\right)$$

$$\mu_\infty(A) = \max_i \left(a_{ii} + \sum\nolimits_{j\neq i} |a_{ij}|\right)$$

- directional derivative of matrix norm $\|\cdot\|$ in direction of A at point I_n.
- **In the literature**: one-sided Lipschitz constant, matrix measure

[3] A. Davydov, **SJ**, A. V. Proskurnikov, and F. Bullo. Non-Euclidean monotone operator theory and applications. JMLR, 2024

Definition: Logarithmic norm

Given a matrix $A \in \mathbb{R}^{n\times n}$ and a norm $\|\cdot\|$:

$$\mu_{\|\cdot\|}(A) := \lim_{h\to 0^+} \frac{\|I_n + hA\| - 1}{h}$$

$$\mu_2(A) = \tfrac{1}{2}\lambda_{\mathsf{max}}(A + A^\top)$$

$$\mu_1(A) = \max_j \left(a_{jj} + \sum\nolimits_{i\neq j} |a_{ij}|\right)$$

$$\mu_\infty(A) = \max_i \left(a_{ii} + \sum\nolimits_{j\neq i} |a_{ij}|\right)$$

- directional derivative of matrix norm $\|\cdot\|$ in direction of A at point I_n.
- **In the literature**: one-sided Lipschitz constant, matrix measure

Theorem[2](Logarithmic norms and monotone operators)

$\mathsf{F} : \mathbb{R}^n \to \mathbb{R}^n$ is a monotone operator if and only if $\mu_2(-D_x\mathsf{F}(x)) \leq 0$, for every $x \in \mathbb{R}^n$

[3] A. Davydov, **SJ**, A. V. Proskurnikov, and F. Bullo. Non-Euclidean monotone operator theory and applications. JMLR, 2024

Definition: Logarithmic norm

Given a matrix $A \in \mathbb{R}^{n\times n}$ and a norm $\|\cdot\|$:

$$\mu_{\|\cdot\|}(A) := \lim_{h\to 0^+} \frac{\|I_n + hA\| - 1}{h}$$

$$\mu_2(A) = \tfrac{1}{2}\lambda_{\max}(A + A^\top)$$

$$\mu_1(A) = \max_j \left(a_{jj} + \sum\nolimits_{i\neq j} |a_{ij}|\right)$$

$$\mu_\infty(A) = \max_i \left(a_{ii} + \sum\nolimits_{j\neq i} |a_{ij}|\right)$$

- directional derivative of matrix norm $\|\cdot\|$ in direction of A at point I_n.
- **In the literature**: one-sided Lipschitz constant, matrix measure

Theorem[2] (Logarithmic norms and monotone operators)

$\mathsf{F} : \mathbb{R}^n \to \mathbb{R}^n$ is a monotone operator if and only if $\mu_2(-D_x\mathsf{F}(x)) \leq 0$, for every $x \in \mathbb{R}^n$

Extend monotone operator theory to non-Euclidean norm spaces

[3] A. Davydov, **SJ**, A. V. Proskurnikov, and F. Bullo. Non-Euclidean monotone operator theory and applications. JMLR, 2024

Let $\|\cdot\|$ be a norm on $\mathbb{R}^n$. Then $\mathsf{F} : \mathbb{R}^n \to \mathbb{R}^n$ is a monotone wrt to $\|\cdot\|$ if

$$\mu_{\|\cdot\|}(-D\mathsf{F}(x)) \leq 0, \qquad \text{for all } x$$

and is strongly monotone with parameter $m \geq 0$ if

$$\mu_{\|\cdot\|}(-D\mathsf{F}(x)) \leq -m, \qquad \text{for all } x$$

[3] A. Davydov, **SJ**, A. V. Proskurnikov, and F. Bullo. Non-Euclidean monotone operator theory and applications. JMLR, 2024

Let $\|\cdot\|$ be a norm on $\mathbb{R}^n$. Then $\mathsf{F} : \mathbb{R}^n \to \mathbb{R}^n$ is a monotone wrt to $\|\cdot\|$ if

$$\mu_{\|\cdot\|}(-D\mathsf{F}(x)) \leq 0, \qquad \text{for all } x$$

and is strongly monotone with parameter $m \geq 0$ if

$$\mu_{\|\cdot\|}(-D\mathsf{F}(x)) \leq -m, \qquad \text{for all } x$$

Theorem (Non-Euclidean version)[3]

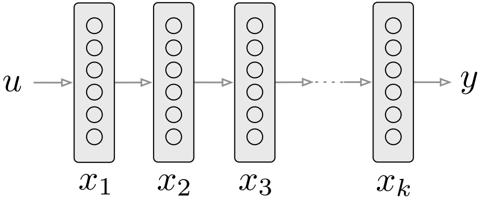
Let $\mathsf{F} : \mathbb{R}^n \to \mathbb{R}^n$ be a strongly monotone operator wrt to a norm $\|\cdot\|$, then

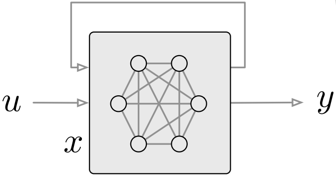
1. $\mathsf{F}(x) = 0$ has a unique solution x^*, and
2. x^* can be computed using the average iteration $x_{k+1} = (1-\theta)x_k + \theta\mathsf{F}(x_k)$.

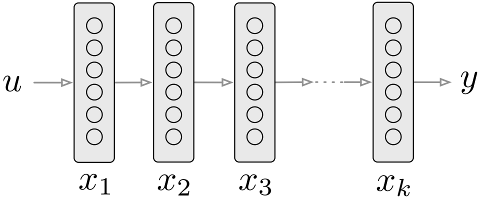
[3] A. Davydov, **SJ**, A. V. Proskurnikov, and F. Bullo. Non-Euclidean monotone operator theory and applications. JMLR, 2024

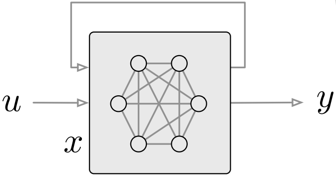
Implicit Neural Networks

Definition via fixed-point equations







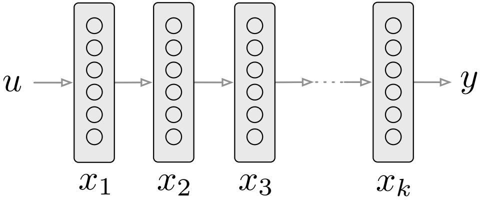


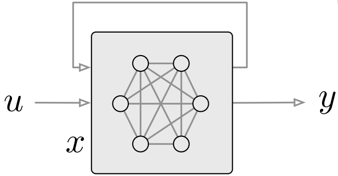
- Feedforward neural networks:

$$x^{i+1} = \Phi(A_i x^i + b_i), \quad x^0 = u$$
$$y = A_k x^k + b_k$$

- Implicit neural networks:

$$x = \Phi(Ax + Bu + b)$$
$$y = Cx + c$$





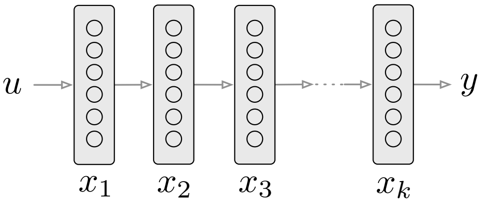
- Feedforward neural networks:

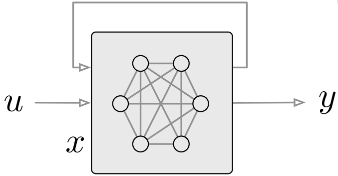
$$x^{i+1} = \Phi(A_i x^i + b_i), \quad x^0 = u$$
$$y = A_k x^k + b_k$$

- Implicit neural networks:

$$x = \Phi(Ax + Bu + b)$$
$$y = Cx + c$$

- $\Phi(y_1, \dots, y_n) = (\phi_1(y_1), \dots, \phi_n(y_n))^\top$ is a diagonal activation function
- activation functions are slope-restricted in $[0, 1]$, i.e., $0 \le \frac{\phi_i(x) - \phi_i(y)}{x - y} \le 1$ for all $x, y \in \mathbb{R}$





- Feedforward neural networks:

$$x^{i+1} = \Phi(A_i x^i + b_i), \quad x^0 = u$$
$$y = A_k x^k + b_k$$

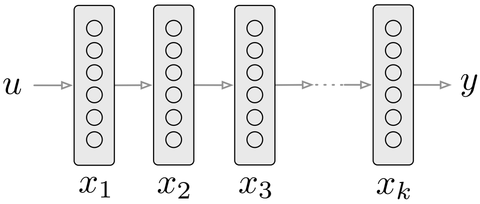
- Implicit neural networks:

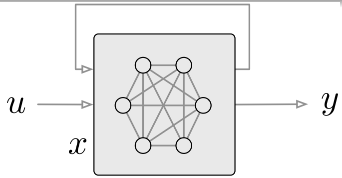
$$x = \Phi(Ax + Bu + b)$$
$$y = Cx + c$$

- $\Phi(y_1, \dots, y_n) = (\phi_1(y_1), \dots, \phi_n(y_n))^\top$ is a diagonal activation function
- activation functions are slope-restricted in $[0, 1]$, i.e., $0 \leq \frac{\phi_i(x)-\phi_i(y)}{x-y} \leq 1$ for all $x, y \in \mathbb{R}$

Notion of Layer: output is defined **implicitly** as a function of input

e.g., fixed-point equation, differential equations, optimization problem





- Feedforward neural networks:

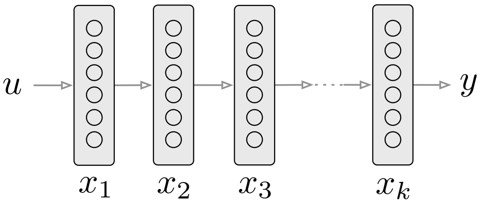
$$x^{i+1} = \Phi(A_i x^i + b_i), \quad x^0 = u$$
$$y = A_k x^k + b_k$$

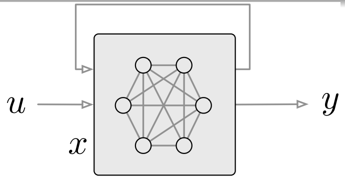
- Implicit neural networks:

$$x = \Phi(Ax + Bu + b)$$
$$y = Cx + c$$

- $\Phi(y_1, \dots, y_n) = (\phi_1(y_1), \dots, \phi_n(y_n))^\top$ is a diagonal activation function
- activation functions are slope-restricted in $[0, 1]$, i.e., $0 \le \frac{\phi_i(x) - \phi_i(y)}{x - y} \le 1$ for all $x, y \in \mathbb{R}$

1. S. Bai, J. Z. Kolter, and V. Koltun. Deep equilibrium models, NeurIPS, 2019
2. L. El Ghaoui, F. Gu, B. Travacca, A. Askari, and A. Y. Tsai. Implicit deep learning. SIMODS, 2019





- Feedforward neural networks:

$$x^{i+1} = \Phi(A_i x^i + b_i), \quad x^0 = u$$
$$y = A_k x^k + b_k$$

- Implicit neural networks:

$$x = \Phi(Ax + Bu + b)$$
$$y = Cx + c$$

- $\Phi(y_1, \dots, y_n) = (\phi_1(y_1), \dots, \phi_n(y_n))^\top$ is a diagonal activation function
- activation functions are slope-restricted in $[0,1]$, i.e., $0 \le \frac{\phi_i(x)-\phi_i(y)}{x-y} \le 1$ for all $x, y \in \mathbb{R}$

Advantages: Representation, Performance, Memory

Main Questions

$$x = \Phi(Ax + Bu + b)$$
$$y = Cx + c$$

1. Existence and computation of solutions?
2. How to estimate the input-output $x \mapsto u$ robustness?

Main Questions

$$x = \Phi(Ax + Bu + b)$$
$$y = Cx + c$$

1. Existence and computation of solutions?
2. How to estimate the input-output $x \mapsto u$ robustness?

Key insight

Fixed-point equation	$\Longleftrightarrow$	Operator theory
$x = \Phi(Ax + Bu + b)$		$\mathsf{N}_u(x) = x - \Phi(Ax + Bu + b)$
fixed-points	$\Longleftrightarrow$	**zeros of** $\mathsf{N}_u(x)$
robustness	$\Longleftrightarrow$	**sensitivity wrt** u

- We can use tools from monotone operator theory to study implicit neural networks

Main observation

If $\mu_\infty(A) < 1$ then N_u is a monotone operator wrt $\|\cdot\|_\infty$.

[4] **SJ**, A. Davydov, A. V. Proskurnikov, and F. Bullo. Robust implicit networks via non-Euclidean contractions. In NeurIPs 2021

Main observation

If $\mu_\infty(A) < 1$ then N_u is a monotone operator wrt $\|\cdot\|_\infty$.

Theorem[4]

If $\mu_\infty(A) < 1$ then

1. $x = \Phi(Ax + Bu + b)$ has a unique solution x_u^*
2. x_u^* can be computed using average iterations for $x = \Phi(Ax + Bu + b)$
3. $\|x_u^* - x_v^*\|_\infty \leq \frac{\|B\|_\infty}{1-[\mu_\infty(A)]_+}\|u - v\|_\infty$

[4] **SJ**, A. Davydov, A. V. Proskurnikov, and F. Bullo. Robust implicit networks via non-Euclidean contractions. In NeurIPs 2021

Main observation

If $\mu_\infty(A) < 1$ then N_u is a monotone operator wrt $\|\cdot\|_\infty$.

Theorem[4]

If $\mu_\infty(A) < 1$ then

1. $x = \Phi(Ax + Bu + b)$ has a unique solution x_u^*
2. x_u^* can be computed using average iterations for $x = \Phi(Ax + Bu + b)$
3. $\|x_u^* - x_v^*\|_\infty \leq \frac{\|B\|_\infty}{1 - [\mu_\infty(A)]_+} \|u - v\|_\infty$

$$u \underbrace{\mapsto}_{\mathrm{Lip}_{u \to x_u^*}} x^* \underbrace{\mapsto}_{\mathrm{Lip}_{x_u^* \to y}} y \quad \Longrightarrow \quad \mathrm{Lip}_{u \to y} = \mathrm{Lip}_{u \to x_u^*} \mathrm{Lip}_{x_u^* \to y}$$

$$\Longrightarrow \quad \mathrm{Lip}_{u \to y} = \frac{\|C\|_\infty \|B\|_\infty}{1 - [\mu_\infty(A)]_+}$$

[4] **SJ**, A. Davydov, A. V. Proskurnikov, and F. Bullo. Robust implicit networks via non-Euclidean contractions. In NeurIPs 2021

1. loss function $\mathcal{L}$
2. training data $(\widehat{u}_i, \widehat{y}_i)_{i=1}^N$

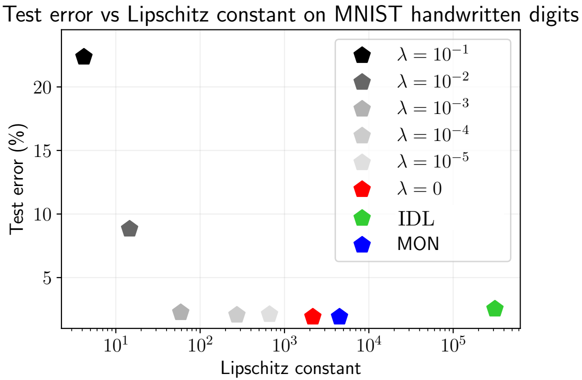
$$\min_{A,B,C,b,c} \quad \sum_{i=1}^{N} \mathcal{L}(\widehat{y}_i, Cx_i + c) \quad +\lambda \quad \mathrm{Lip}_{u\to y}$$

$$x_i = \Phi(Ax_i + B\widehat{u}_i + b)$$

$$\mu_\infty(A) \le \gamma,$$

- $\gamma < 1$ is a hyperparameter and $\lambda \ge 0$ is a regularization parameter
- training optimization problem is solved via SGD
- at each step of SGD, $x_i = \Phi(Ax_i + B\widehat{u}_i + b)$ is solved using the average-iterations

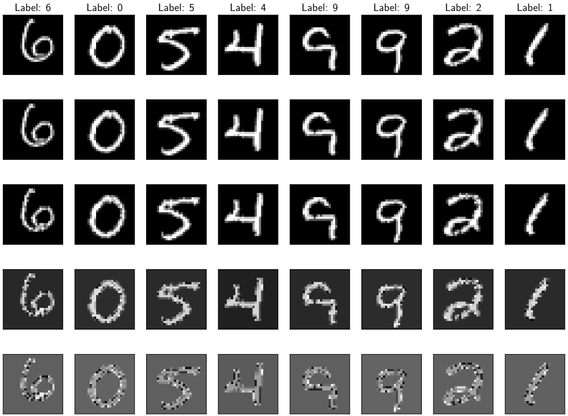
- MNIST dataset: 28×28 pixel handwritten digits between $0 - 9$, $60,000$ training images and $10,000$ test images.
- implicit neural network order: $n = 100$ and $\gamma = 0.95$
- loss function: cross entropy

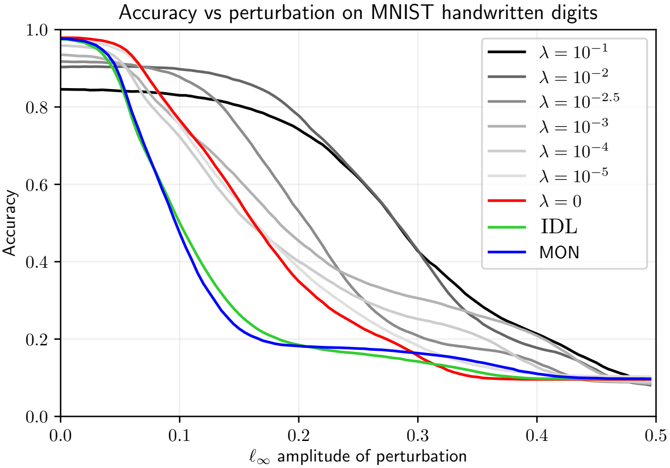


Improvements:

- ($\lambda = 0$): two orders of magnitude wrt. IDL and wrt. MON
- ($\lambda = 10^{-3}$): three orders of magnitude wrt. IDL and one order of magnitude wrt. MON
- ($\lambda = 10^{-2}$): four orders of magnitude wrt. IDL and two orders of magnitude wrt. MON

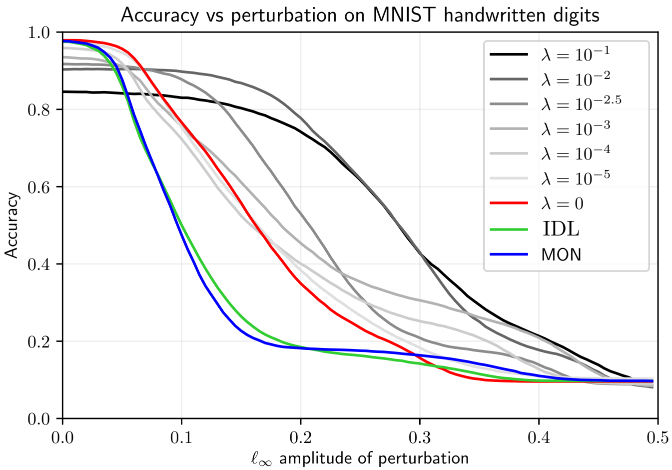
- perturbation: inversion attack $u_{\mathrm{adv}} = u + \epsilon\,\mathrm{sign}(\frac{1}{2}\mathbb{1}_{784} - u)$





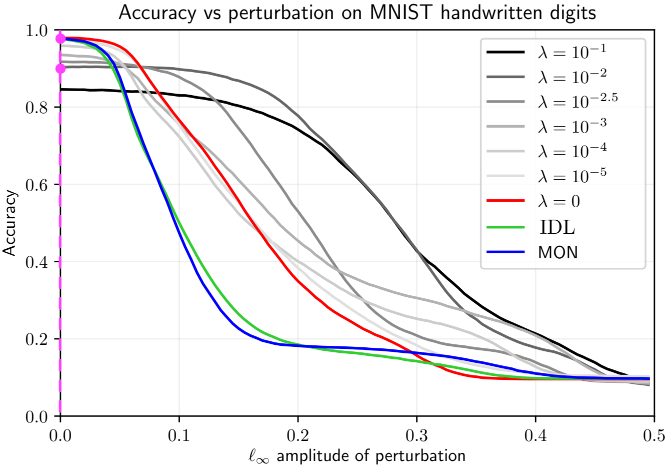
- ($\lambda = 0$): improved robustness than IDL and MON
- ($\lambda > 0$): improved robustness at sizable perturbations but losing some percentage accuracy in clean performance

Tradeoff between **clean performance** and **robustness**



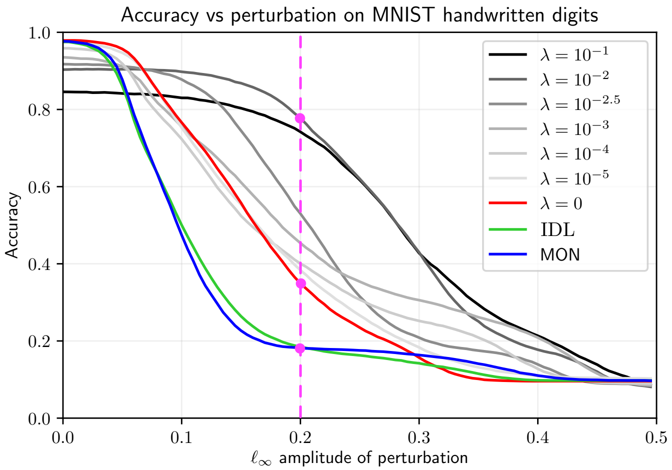
- $(\lambda = 0)$: improved robustness than IDL and MON
- $(\lambda > 0)$: improved robustness at sizable perturbations but losing some percentage accuracy in clean performance

Tradeoff between **clean performance** and **robustness**



- $(\lambda = 0)$: improved robustness than IDL and MON
- $(\lambda > 0)$: improved robustness at sizable perturbations but losing some percentage accuracy in clean performance

Tradeoff between **clean performance** and **robustness**



- $(\lambda = 0)$: improved robustness than IDL and MON
- $(\lambda > 0)$: improved robustness at sizable perturbations but losing some percentage accuracy in clean performance

Tradeoff between **clean performance** and **robustness**

Conclusions

- Extension of monotone operator theory to normed-spaces using Logarithmic norm
- Non-Euclidean contraction theory for well-posedness of INNs
- Lipschitz bounds of INNs using non-Euclidean monotone operator theory

Thank you for your attention!

Back up Slides

Differential condition

Logarithmic norm

Given a matrix $A \in \mathbb{R}^{n\times n}$ and a norm $\|\cdot\|$:

$$\mu_{\|\cdot\|}(A) := \lim_{h\to 0^+} \frac{\|I_n + hA\| - 1}{h}$$

- Directional derivative of norm $\|\cdot\|$ in direction of A,

$$\mu_2(A) = \tfrac{1}{2}\lambda_{\mathsf{max}}(A + A^\top)$$
$$\mu_1(A) = \max_j \left(a_{jj} + \textstyle\sum_{i\neq j} |a_{ij}|\right)$$
$$\mu_\infty(A) = \max_i \left(a_{ii} + \textstyle\sum_{j\neq i} |a_{ij}|\right)$$

[1] A. Davydov, **SJ**, F. Bullo, Non-Euclidean contraction theory for robust nonlinear stability, 2022

Differential condition

Logarithmic norm

Given a matrix $A \in \mathbb{R}^{n\times n}$ and a norm $\|\cdot\|$:

$$\mu_{\|\cdot\|}(A) := \lim_{h\to 0^+} \frac{\|I_n + hA\| - 1}{h}$$

- Directional derivative of norm $\|\cdot\|$ in direction of A,

$$\mu_2(A) = \tfrac{1}{2}\lambda_{\mathsf{max}}(A + A^\top)$$
$$\mu_1(A) = \max_j \left(a_{jj} + \sum\nolimits_{i\neq j} |a_{ij}|\right)$$
$$\mu_\infty(A) = \max_i \left(a_{ii} + \sum\nolimits_{j\neq i} |a_{ij}|\right)$$

Integral condition

Weak pairing[5]

Given a norm $\|\cdot\|$, the associated weak pairing is $[\![\cdot,\cdot]\!] : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$:

- Subadditive and weakly homogeneity
- Positive definite
- Cauchy-Schwarz inequality
- $[\![x, x]\!] = \|x\|^2$

$$[\![x, y]\!]_2 = y^\top x$$
$$[\![x, y]\!]_1 = \mathrm{sign}(y)^\top x$$
$$[\![x, y]\!]_\infty = \max_{i\in I_\infty(x)} x_i y_i$$

$$I_\infty(x) = \{i \mid |x_i| = \|x\|_\infty\}$$

[1] A. Davydov, **SJ**, F. Bullo, Non-Euclidean contraction theory for robust nonlinear stability, 2022

Theorem[6]

$\dot{x} = f(x,u)$ is contracting wrt $\|\cdot\|$ with rate c iff

Differential: $\mu_{\|\cdot\|}(D_x f(x,u)) \leq -c,$ for all x, u

Integral: $[\![f(x,u) - f(y,u), x-y]\!] \leq -c\|x-y\|^2,$ for all x, y, u

[2] A. Davydov, S. Jafarpour, F. Bullo, TAC 2022

Theorem

$\dot{x} = f(x,u)$ is contracting wrt $\|\cdot\|$ with rate c iff

Differential: $\mu_{\|\cdot\|}(D_x f(x,u)) \leq -c,$ for all x, u

Integral: $[\![f(x,u) - f(y,u), x-y]\!] \leq -c\|x-y\|^2,$ for all x, y, u

- Connection between **contraction theory** and **monotone operator theory**

f is a contracting vector field wrt to $\|\cdot\|_2$
iff
$-f$ is a strongly monotone operator wrt to the inner product $\langle\cdot,\cdot\rangle$.

Theorem

$\dot{x} = f(x,u)$ is contracting wrt $\|\cdot\|$ with rate c iff

Differential: $\mu_{\|\cdot\|}(D_x f(x,u)) \leq -c,$ for all x, u

Integral: $[\![f(x,u) - f(y,u), x - y]\!] \leq -c\|x-y\|^2,$ for all x, y, u

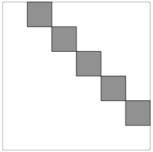
- Connection between **contraction theory** and **monotone operator theory**

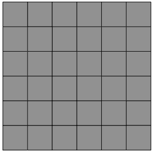
f is a contracting vector field wrt to $\|\cdot\|$
iff
$-f$ is a strongly monotone operator wrt to the weak pairing $[\![\cdot,\cdot]\!]$.

- Origins:

- Generalizing feedforward neural networks to fully-connected synaptic matrices

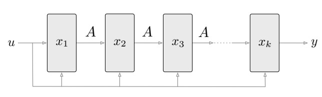
Intuition: $z^{i+1} = \phi_i(A_i z^i + b_i) \quad \Longleftrightarrow \quad z = \Phi(Ax + Bu + b)$, where A has upper diagonal structure.

$A_{\text{upper-diagonal}} =$ 

$A_{\text{complete}} =$ 

- comparable accuracy to traditional neural networks with significant memory reduction

Intuition: implicit neural network = weight-tied infinite-layer network



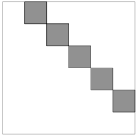
$z^{i+1} = \phi_i(Az^i + B_i x + b_i) \implies \lim_{i\to\infty} z^i = x^*$ solution to the implicit neural network

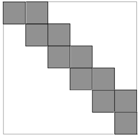
- suitable for learning constrained optimization problems

Intuition: casting KKT condition as an implicit layer

- vanishing and exploding gradient

Intuition: the notion of "autapse" (time-delayed self-feedback) from neuroscience

$A_{\text{upper-diagonal}} =$ 

$A_{\text{Autapse}} =$ 

- suitable for learning stiff problems or problems with discontinuity

Generalized Structure

Comparison with feedforward neural networks

- Feedforward neural networks:

$$z^{(\ell+1)} = \Phi(A_\ell z^{(\ell)} + b_\ell), \quad z^{(0)} = x$$
$$u = A_k z^{(k)} + b_k$$

- Implicit neural networks:

$$z = \Phi(Az + Bx + b)$$
$$u = Cz + c$$