A Dynamical System Perspective for Lipschitz Neural Networks

Laurent Meunier[*,1,2], Blaise Delattre[*,1,3], Alexandre Araujo[*,4], Alexandre Allauzen[1,5]

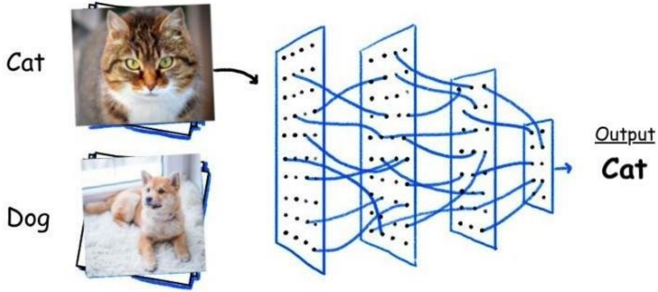
[*] Equal contribution

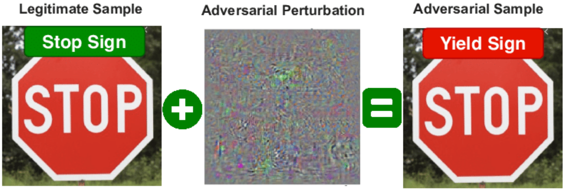
[1] Université Paris-Dauphine, PSL University, Paris, France
[2] Meta AI Research, Paris, France
[3] Foxstream, Lyon, France
[4] INRIA, Ecole Normale Supérieure, CNRS, PSL University, Paris, France
[5] ESPCI, Paris, France

Introduction

Classification in Machine Learning

- Input space $\mathcal{X} \subset \mathbb{R}^d$ to a label space $\mathcal{Y} := \{1, \dots, K\}$.
- Classifier function $\mathbf{f} := (f_1, \dots, f_K) : \mathcal{X} \to \mathbb{R}^K$ such that the predicted label for an input x is $\operatorname{argmax}_k f_k(x)$.
- Input-label (x, y) is correctly classified if $\operatorname{argmax}_k f_k(x) = y$.



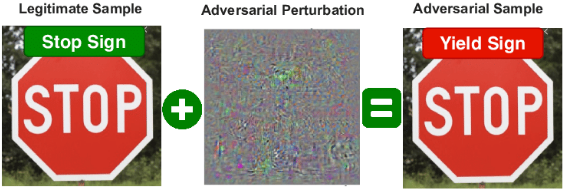


Definition (Adversarial Attacks)

Let $x \in \mathcal{X}$, $y \in \mathcal{Y}$ the label of x and let $\mathbf{f}$ be a classifier. An adversarial attack at level ε is a perturbation τ such that $\|\tau\| \leq \varepsilon$ such that:

$$\underset{k}{\operatorname{argmax}}\, f_k(x+\tau) \neq y$$

Adversarial Attacks



Definition (Adversarial Attacks)

Let $x \in \mathcal{X}$, $y \in \mathcal{Y}$ the label of x and let $\mathbf{f}$ be a classifier. An adversarial attack at level ε is a perturbation τ such that $\|\tau\| \leq \varepsilon$ such that:

$$\operatorname*{argmax}_k f_k(x+\tau) \neq y$$

A classifier $\mathbf{f}$ is said to be certifiably robust at radius $\varepsilon \geq 0$ at point x with label y if for all τ such that $\|\tau\| \leq \varepsilon$:

$$\operatorname*{argmax}_k f_k(x+\tau) = y$$

Background

Lipschitz constant and Certification

Proposition (Tsuzuku et al. (2018))

Let $\mathbf{f}$ *be an* L*-Lipschitz continuous classifier for the* ℓ_2 *norm. Let* $\varepsilon > 0$, $x \in \mathcal{X}$ *and* $y \in \mathcal{Y}$ *the label of* x*. If at point* x*, the margin* $\mathcal{M}_{\mathbf{f}}(x)$ *satisfies:*

$$\mathcal{M}_{\mathbf{f}}(x) := \max(0, f_y(x) - \max_{y' \neq y} f_{y'}(x)) > \sqrt{2} L \varepsilon$$

then we have for every τ *such that* $\|\tau\|_2 \leq \varepsilon$*:*

$$\operatorname*{argmax}_k f_k(x + \tau) = y$$

Trade-off between a large margin and a small Lipschitz constant.

Related Work on 1-Lipschitz Neural Networks

Previous approaches on 1-Lipschitz Neural Networks

- Spectral norm of weights matrices:
  $\rightarrow$ Yoshida et al. (2017); Farnia et al. (2019); Anil et al. (2019)

- Orthogonal weights:
  $\rightarrow$ Li et al. (2019); Trockman et al. (2021); Singla et al. (2021)

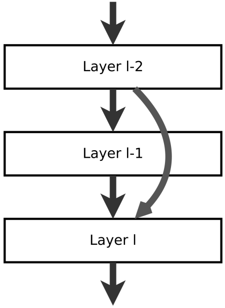
Residual Networks flows

A Residual Network is defined as:

$$\begin{cases} x_0 & = x \in \mathcal{X} \\ x_{t+1} & = x_t + F_t(x_t) \end{cases}$$

where $F_t(x_t)$ is typically a two layer neural networks:

$$F_t(x_t) = W_{2,t}\sigma(W_{1,t}x_t)$$



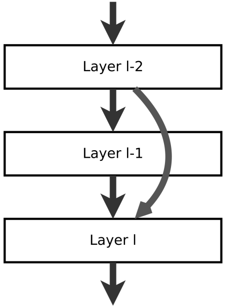
Residual Networks flows

A Residual Network is defined as:

$$\begin{cases} x_0 & = x \in \mathcal{X} \\ x_{t+1} & = x_t + F_t(x_t) \end{cases}$$

where $F_t(x_t)$ is typically a two layer neural networks:

$$F_t(x_t) = W_{2,t}\sigma(W_{1,t}x_t)$$



Definition (Continuous Residual Networks Haber et al. (2017))

Let $(F_t)_{t\in[0,T]}$ be a family of functions on $\mathbb{R}^d$, we define the continuous time Residual Networks flow associated with F_t as:

$$\begin{cases} x_0 & = x \in \mathcal{X} \\ \frac{dx_t}{dt} & = F_t(x_t) \text{ for } \ t \in [0,T] \end{cases}$$

Lipschitzness of Continuous Residual Networks

Proposition

Let $(F_t)_{t\in[0,T]}$ be a family of functions on $\mathbb{R}^d$. Let us assume that $\mu_t I \preceq \mathit{Sym}(\nabla_x F_t(x)) \preceq \lambda_t I$ for all $x \in \mathbb{R}^d$, and $t \in [0,T]$. Then the flow associated with F_t satisfies for all initial conditions x_0 and z_0:

$$\|x_0 - z_0\| e^{\int_0^t \mu_s ds} \leq \|x_t - z_t\| \leq \|x_0 - z_0\| e^{\int_0^t \lambda_s ds}$$

Proposition

Let $(F_t)_{t\in[0,T]}$ be a family of functions on $\mathbb{R}^d$. Let us assume that $\mu_t I \preceq Sym(\nabla_x F_t(x)) \preceq \lambda_t I$ for all $x \in \mathbb{R}^d$, and $t \in [0,T]$. Then the flow associated with F_t satisfies for all initial conditions x_0 and z_0:

$$\|x_0 - z_0\| e^{\int_0^t \mu_s ds} \leq \|x_t - z_t\| \leq \|x_0 - z_0\| e^{\int_0^t \lambda_s ds}$$

Corollary

Let $(f_t)_{t\in[0,T]}$ be a family of convex differentiable functions on $\mathbb{R}^d$ and $(A_t)_{t\in[0,T]}$ a family of skew symmetric matrices. Let us define

$$F_t(x) = -\nabla_x f_t(x) + A_t x,$$

then the flow associated with F_t satisfies for all initial conditions x_0 and z_0:

$$\|x_t - z_t\| \leq \|x_0 - z_0\|$$

Hybrid schemes

Discretization Problem: Forward Euler Discretization:

$$x_{t+1} = x_t + F_t(x_t)$$

does not satisfy the previous Lipschitz property and Backward Euler is hardly tractable. Solution: **Hybrid schemes!**

$$\begin{cases} x_{t+\frac{1}{2}} & = \text{STEP1}(x_t, \nabla_x f_t) \\ x_{t+1} & = \text{STEP2}(x_{t+\frac{1}{2}}, A_t) \end{cases}$$

Proposition

Let $t \in \{1, \cdots, T\}$ Let us assume that f_t is L_t-smooth. We define the following discretized ResNet gradient flow using h_t as a step size $x_{t+\frac{1}{2}} = x_t - h_t \nabla_x f_t(x_t)$. Consider now two trajectories x_t and z_t with initial points $x_0 = x$ and $z_0 = z$ respectively, if $0 \leq h_t \leq \frac{2}{L_t}$, then $\|x_{t+\frac{1}{2}} - z_{t+\frac{1}{2}}\|_2 \leq \|x_t - z_t\|_2$

Discretization schemes for A_t

Midpoint Euler method. We thus propose to use Midpoint Euler method, defined as follows:

$$x_{t+1} = x_{t+\frac{1}{2}} + A_t \frac{x_{t+1} + x_{t+\frac{1}{2}}}{2}$$
$$\iff x_{t+1} = \left(I - \frac{A_t}{2}\right)^{-1} \left(I + \frac{A_t}{2}\right) x_{t+\frac{1}{2}}.$$

$\rightarrow$ **Cayley Transform** studied by Trockman et al. (2021) of $\frac{A_t}{2}$ that induces an orthogonal mapping.

Exact Flow.

$$\frac{du_t}{ds} = A_t u_s, \quad u_0 = x_{t+\frac{1}{2}},$$

By taking the value at $s = \frac{1}{2}$, we obtained the following:

$$x_{t+1} := u_{\frac{1}{2}} = e^{\frac{A}{2}} x_{t+\frac{1}{2}}.$$

$\rightarrow$ **Skew Orthogonal Convolution** (SOC) studied by Singla et al. (2021).

Parametrization for Convex Potential Layers

Gradient of ICNN (Amos et al., 2017):
Let ϕ a convex real function. $F_{w,b} : x \in \mathbb{R}^d \mapsto \sum_{i=1}^{k} \phi(w_i^\top x + b_i)$ defines a convex function in x as the composition of a linear and a convex function. Its gradient with respect to its input x is

$$x \mapsto \sum_{i=1}^{k} w_i \phi'(w_i^\top x + b_i) = \mathbf{W}^\top \sigma(\mathbf{W}x + \mathbf{b})$$

with $\sigma := \phi'$. Assuming σ is L-Lipschitz, we have that $F_{w,b}$ is $L\|\mathbf{W}\|_2^2$-smooth. $\|\mathbf{W}\|_2$ is the spectral norm of $\mathbf{W}$: $\|\mathbf{W}\|_2 := \max_{x\neq 0} \frac{\|\mathbf{W}x\|_2}{\|x\|_2}$

New 1-Lipschitz Layer: Convex Potential Layer

$$z = x - \frac{2}{\|\mathbf{W}\|_2^2} \mathbf{W}^\top \sigma(\mathbf{W}x + b)$$

Use of Power Iteration algorithm for computing Spectral Norms.

- Quasi-free at training: single iteration for each layer at each step.
- Free at inference: we make 100 iterations for each layer but only once!

Algorithm 1 Computation of a Convex Potential Layer

Require: **Input:** x, **vector:** u, **weights:** $\mathbf{W}$, b
Ensure: Compute the layer z and return u

$$\left.\begin{array}{l} v \leftarrow \mathbf{W}u / \|\mathbf{W}u\|_2 \\ u \leftarrow \mathbf{W}^\top v / \|\mathbf{W}^\top v\|_2 \\ h \leftarrow 2 / \left(\sum_i (\mathbf{W}u \cdot v)_i\right)^2 \end{array}\right\} \begin{array}{l} \text{1 iter. for training} \\ \text{100 iter. for inference} \end{array}$$

return $x - h\left[\mathbf{W}^\top \sigma(\mathbf{W}x + b)\right], u$

Experimental Results (1)

4 versions of CPL networks (S, M, L, XL) with various depths and widths.

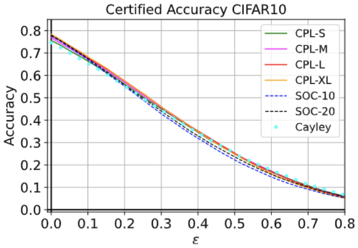


Figure 1: Certified Accuracy in function of the perturbation ε for our CPL networks and its concurrent approaches on CIFAR10 and CIFAR100 datasets.

Experimental Results (2)

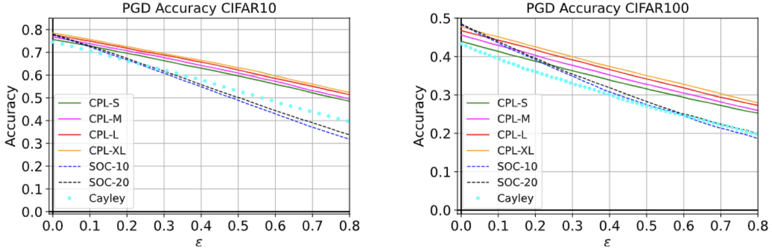


Figure 2: Accuracy against PGD attack with 10 iterations in function of the perturbation ε for our CPL networks and its concurrent approaches on CIFAR10 and CIFAR100 datasets.

Scalability of the Approach

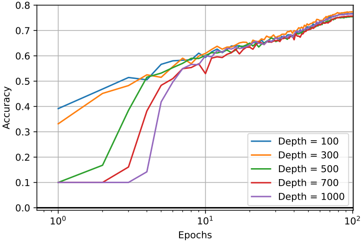


Figure 3: Standard test accuracy in function of the number of epochs (log-scale) for various depths for our neural networks.

Conclusion

Conclusion

A Dynamical System Perspective for Lipschitz Neural Networks

- Perspective from Dynamical System explain previous approaches
- SOTA results in classification and robustness in comparison which other existing 1-Lipschitz approaches
- Our layers provides scalable approaches without further regularizations to train very deep architectures

Thank You!

References

Brandon Amos et al. Input convex neural networks. In International Conference on Machine Learning, 2017.

Cem Anil et al. Sorting out lipschitz function approximation. In International Conference on Machine Learning, 2019.

Farzan Farnia et al. Generalizable adversarial training via spectral normalization. In International Conference on Learning Representations, 2019.

Eldad Haber et al. Stable architectures for deep neural networks. Inverse problems, 2017.

Qiyang Li et al. Preventing gradient attenuation in lipschitz constrained convolutional networks. In Advances in Neural Information Processing Systems. 2019.

Sahil Singla et al. Skew orthogonal convolutions. In Proceedings of the 38th International Conference on Machine Learning, 2021.

Asher Trockman et al. Orthogonalizing convolutional layers with the cayley transform. In International Conference on Learning Representations, 2021.

References

Yusuke Tsuzuku et al. Lipschitz-margin training: Scalable certification of perturbation invariance for deep neural networks. In Advances in Neural Information Processing Systems, 2018.

Yuichi Yoshida et al. Spectral norm regularization for improving the generalizability of deep learning. arXiv preprint arXiv:1705.10941, 2017.