A path-norm toolkit for modern networks: consequences, promises and challenges

Antoine Gonon, Nicolas Brisebarre, Elisa Riccietti, Rémi Gribonval

Univ Lyon, ENS de Lyon, UCBL, CNRS, Inria, LIP, F-69342 Lyon

SUMMARY

Modern Challenges

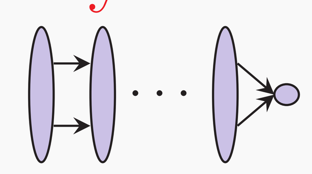
1. *Generalization* [1, 2]
2. *Robustness* [1]
3. Implicit bias [7]
4. Identifiability [5, 6]

Existing: "path-norm", a promising tool?

Theory: ☺ partial answers to 1-4

Practice: ☺ correlates with *generalization* [3, 4]
☺ easy to compute [4]

☹ ...only developed for toy networks



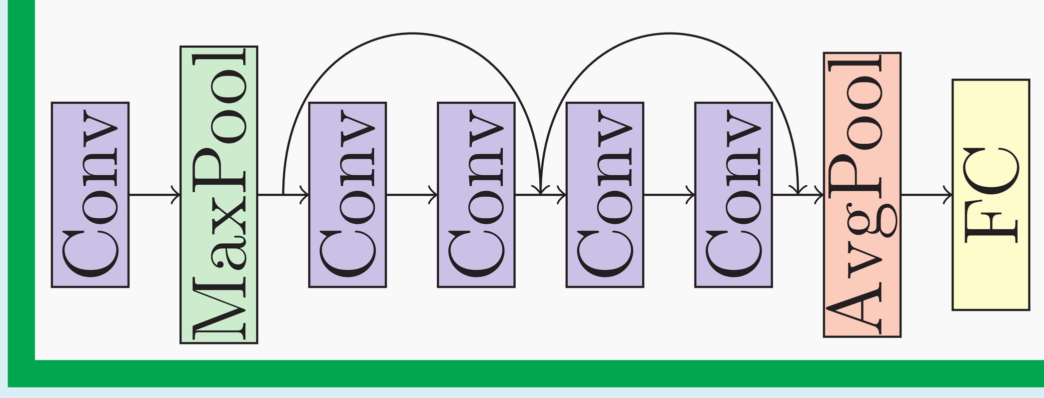
This work: tools for modern ReLU networks

ResNets, VGGs, U-nets, ReLU MobileNets, Inception nets, AlexNet...

Theory: ☺☺ sharper *generalization* bound

Practice: ☺☺ PyTorch implementation
☺☺ first numerical assessment on ResNets/ImageNet

☺ **now for modern networks**



"Path-norm" philosophy

$\boldsymbol{\theta} =$ network parameters (weight, biases)

Measure $\begin{cases} \textit{Generalization} \\ \textit{Robustness} \end{cases} \leq f(\|\boldsymbol{\theta}\|)$

$\rightarrow$ replace with

$\Phi(\boldsymbol{\theta}) =$ path-lifting

$\leq g(\|\Phi(\boldsymbol{\theta})\|)$ "path-norm" based bound

☺ invariant under neuron-wise rescaling
☺ sharper than $\|\boldsymbol{\theta}\|$

Contribution 1: definition of path-lifting and path-activations for modern NNs

$\begin{cases} \Phi(\boldsymbol{\theta}) \text{ path-lifting (see paper)} \\ A(\boldsymbol{\theta}, x) \text{ path-activations (binary matrix)} \end{cases}$ s.t. $\boxed{R_{\boldsymbol{\theta}}(x) := \text{output}(\boldsymbol{\theta}, x) = \langle \Phi(\boldsymbol{\theta}), A(\boldsymbol{\theta}, x)x\rangle}$

Corollary: Measure(*Robustness*)=Lipschitz constant$= \sup_{x\neq x'} \frac{\|R_{\boldsymbol{\theta}}(x)-R_{\boldsymbol{\theta}}(x')\|_1}{\|x-x'\|_\infty} \leq \|\Phi(\boldsymbol{\theta})\|_1 =:$ path-norm (easy to compute)

Contribution 2: first path-norm based generalization bound valid for modern networks

Our bound: $C \times \|\Phi(\boldsymbol{\theta})\|_1$

$$C = \frac{\sup_x \|x\|_\infty}{\sqrt{n}} L \sqrt{D\ln(K) + \ln(d_{\text{in}} d_{\text{out}})}$$

☺ Valid for modern networks
☺ Sharper than $\prod_{k=1}^{D} \|M_k\|_{\infty\to\infty}$ for layered networks $x \mapsto M_D \operatorname{ReLU}(\dots\operatorname{ReLU}(M_1 x))$
☺ Improves on previous generalization bounds

D = depth, K = k-max-pooling kernel size, L= loss Lipschitz constant, n = number of samples, $d_{\text{in}}/d_{\text{out}}$=input/output dimension

Contribution 3: First assessment of the promises of path-norm on modern networks

☺ Assessing the promises of path-norm for the first time on ResNets/ImageNet $C \simeq 0.1$

PyTorch pretrained ResNet18

$\|\Phi(\boldsymbol{\theta})\|_1$	1.3×10^{30}
$\|\Phi(\boldsymbol{\theta})\|_2$	2.5×10^{2}
$\|\Phi(\boldsymbol{\theta})\|_4$	7.2×10^{-6}

$\leftarrow$

What's next?

worst-case measure $\|\Phi(\boldsymbol{\theta})\|_1$ $\rightarrow$ average-case complexity$(\Phi(\boldsymbol{\theta}))$

References

[1] Neyshabur et al., Norm-Based Capacity Control in Neural Networks, COLT (2015)
[2] Barron et al., Complexity, Statistical Risk, and Metric Entropy of Deep Nets Using Total Path Variation, preprint (2019)
[3] Jiang et al., Fantastic Generalization Measures and Where to Find Them, ICLR (2020)
[4] Dziugaite et al., In search of robust measures of generalization, NIPS (2020)
[5] Bona-Pellissier et al., Local Identifiability of Deep ReLU Neural Networks: the theory, NIPS (2022)
[6] Stock et al., An embedding of ReLU networks and an analysis of their identifiability, Constr. Approx. (2023)
[7] Marcotte et al., Abide by the Law and Follow the Flow: Conservation Laws for Gradient Flows, NIPS (2023)