Neural Network Inspired Data Feature Extraction (DO-14)

Wojciech Czaja, *Radu Balan*, Pierre-Emmanuel Jabin

Department of Mathematics, CSCAMM and NWC
University of Maryland, College Park, MD

LTSPoC: Steve Boerner

January 7, 2020
LTS Annual Review Meeting

Collaborators:

Naveed Haghani (UMD)
Maneesh Singh (Versik)
Dongmian Zou (UMD, IMA)

Problem

Outcome of this project: D.Zou, R. Balan, M. Singh, *On Lipschitz Bounds of General Convolutional Neural Networks*, IEEE Trans. Info. Theory, to appear in 2020.

Problem: Estimate the Lipschitz constant of a Deep Convolutive Network.

Types of Lipschitz Constants (LC) for a M-layer Deep Network Φ, $y = \Phi(x)$:

1. Theoretical LC:

$$L^2 = \sup_{x_1 \neq x_2} \frac{\|y_2 - y_1\|_2^2}{\|x_2 - x_1\|_2^2}$$

2. Analytical estimate using a Linear Program:

$$L_c^2 \approx \prod_{k=1}^{M} \sigma_{max}^2(A_k)$$

3. Local LC:

$$Lip(x)^2 = \lim_{r \to 0} \sup_{x_2 : \|x_2 - x\|_2 < r} \frac{\|y_2 - y\|_2^2}{\|x_2 - x\|_2^2}$$

4. Empirical LC:

$$L_{emp} = \sup_{x_1 \neq x_2;\ x_1, x_2 \in DataBase} \frac{\|y_2 - y_1\|_2^2}{\|x_2 - x_1\|_2^2}$$

Results
Findings - 1

1. Computation of Local Lipschitz bound and relationship with the global bound, for DNN with ReLU activation map:

$$Lip(x) = \sigma_{max}\left(\prod_{k=1}^{M} D_k(x) T_k\right) \quad L = \max_{\|x\| \leq R} Lip(x)$$

2. Numerical values for AlexNet on ImageNet database:

Method	Lip const
Analytical estimate L_c: compute Bessel bounds and solve a linear program	2.51×10^3
Empirical bound L_{emp}: take quotient from pairs of samples	7.32×10^{-3}
Numerical approximation L: compute local Lipschitz constants and take the maximum	1.44

Results

Findings - 2

3. Discrepancy between L and L_{emp} We introduce and compute an *effective Jacobian* J_{eff} that accounts for mid-range interactions:

$$J(x) = P_M D_M T_M \cdots P_2 D_2 T_2 P_1 D_1 T_1$$

$$J_{eff} = \mathbb{E}[P_M]\mathbb{E}[D_M] T_M \cdots \mathbb{E}[P_1]\mathbb{E}[D_1] T_1 = \frac{p_1 \cdots p_M}{\tau^m} A_M \cdots A_1$$

Then: $L_{eff} = \sigma_{max}(J_{eff})$. For AlexNet, the number of layers $M = 5$, the Pool Tile Size $\tau = 9$ and the number of Pooling Layers $m = 3$. Experimentally over 10,000 pairs we obtained:

$$p_1 = 0.4115\ ,\ p_2 = 0.3184\ ,\ p_3 = 0.3587\ ,\ p_4 = 0.2733\ ,\ p_5 = 0.1943$$

The estimated effective Lipschitz constant:

$$L_{eff} = 1.78 \cdot 10^{-2}$$

which is about twice the emprirical constant $7.32 \cdot 10^{-3}$.

Next Steps

Open questions and future steps:

1. How to speed up the computation of the effective Jacobian?
2. Adaptive/On-line algorithm for tunable deep networks
3. How to use this estimate as a constraint or a penalty in learning deep networks?
4. Connections with the Fisher Information Matrix and the mean-field theory.