Implementing Convolutional Neural Networks in Cyber Security

University of New Hampshire
College of Engineering and Physical Sciences

[1]Landon Buell
Adviser: [2]Prof. Qiaoyan Yu
[1]Dept. of Physics and Astronomy
[2]Dept. of Electric and Computer Engineering
University of New Hampshire, Durham New Hampshire, USA

Introduction

- Neural Networks are implemented all over the modern world [3,5]
  - US Postal Service in 1980's for hand-written digits [2]
  - Goal of network is to train by minimizing a *Loss Function* [4]
  - Frequent use makes them targets for Cyber Attacks
  - Possibly prone to security threats
- What Happens when someone *Attacks* a Neural Network?
  - How does the performance change?
  - What signs show an attack has taken place?
  - Can we take protective measures?
- We demonstrate initial concepts that Network designers can consider
  - Prepare defense mechanisms against attacks

Mathematical Models

$$\vec{x}^{(l+1)} = f\left[\hat{W}^{(l)}\vec{x}^{(l)} + \vec{b}^{(l)}\right] \qquad (1)$$

$$\vec{x}^{(l+1)} = f\left[A\left(\hat{W}^{(l)}\vec{x}^{(l)}\right) + \vec{b}^{(l)}\right] \qquad (2)$$

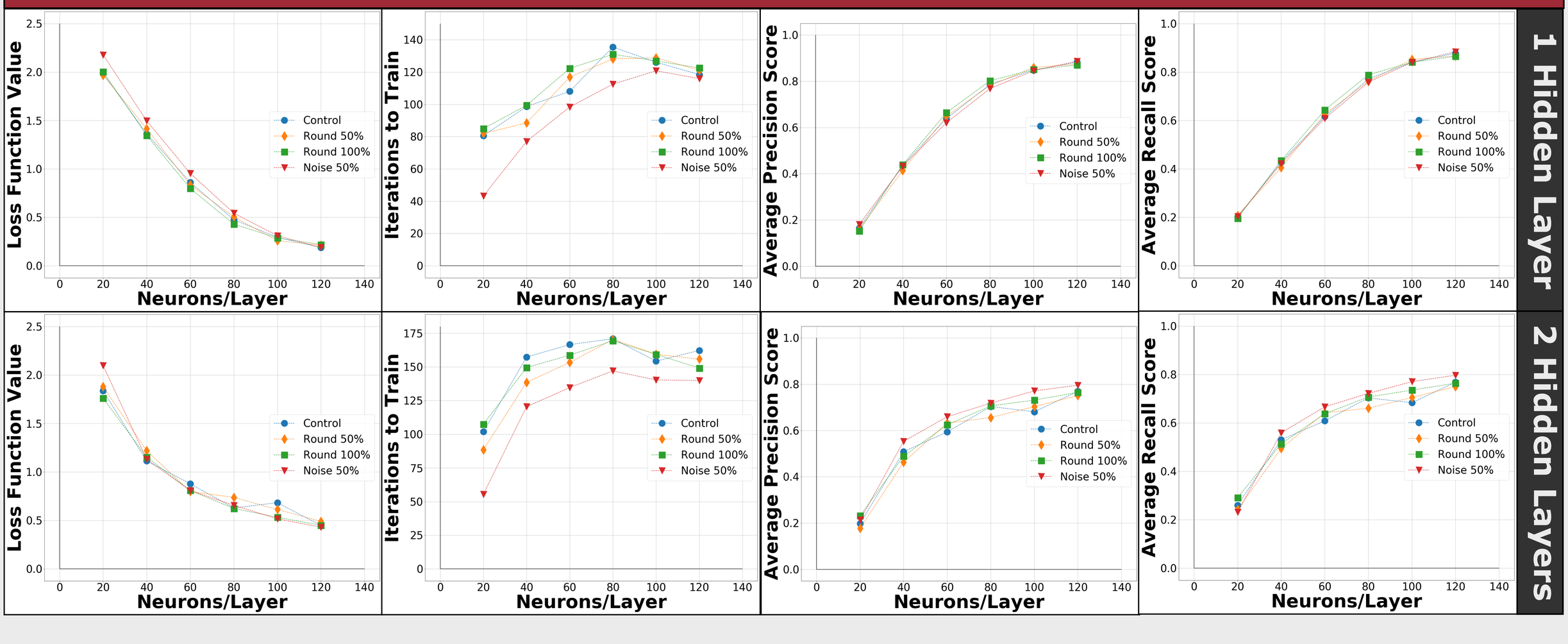
Data Set Examples

- Use MNIST data set—images of digits
  - 28 x 28 Pixels, 8-bit color scale [2]
  - Train w/ 12,000 samples, Test w/ 4,000

Experimental Methodology

- Multilayer Perceptron is a type of Linear Neural Network Architecture [2,3]
  - Layers of neurons, modeled by vectors [5]
  - Scikit-Learn "*MLPClassifier*" in Python 3.8 [1,7]
  - Information is passed through network with Eqn. (1)
  - We introduce *Attack Function* in Eqn. (2)
- Attack Function Variants
  - Reduce Numerical Accuracy (round FP numbers)
  - Introduce *Noise* Function
  - Combine with different *Trigger Conditions*
  - Compare with "Control" baseline
- Changing Model Architecture
  - Test each variant for single & double hidden layers
  - Neuron densities (20,40,60,80,100,120)
  - Train 100 models and average results for each

Experiment Results



Experiment Conclusions

- Attack Functions Affect Lower Level Metrics
  - Changes prevalent in *Loss Function Value* and *Training Iterations*
  - Models converge on higher loss function values, with few iterations
  - No substantial changes in classification metrics
  - *Stopping parameter* [3] met before parameters converge near minimum
- Other Considerations and Questions
  - Attack only acts on decision process, not correction process
  - Only forward pass is modified, Back Propagation still operates
  - Results about neuron density or layer depth is still inclusive
  - Results are averages, can a *single* attack be detected?

References

[1] Buitinck et al. *API design for machine learning software: experiences from the scikit-learn project*, 2013
[2] Géron Aurélien. *Hands-on Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly, 2017.
[3] Goodfellow, Ian, et al. *Deep Learning*. MIT Press, 2017.
[4] James, Gareth, et al. *An Introduction to Statistical Learning with Applications in R*. Springer, 2017.
[5] Loy, James. *Neural Network Projects with Python: The Ultimate Guide to Using Python to Explore the True Power of Neural Networks through Six Projects*. Packt Publishing, 2019.
[6] McCulloch, Warren S., and Walter Pitts. *"A Logical Calculus of the Ideas Immanent in Nervous Activity."* The Bulletin of Mathematical Biophysics, vol. 5, no. 4, 1943, pp. 115–133.
[7] Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.

We would like to acknowledge support from the National Science Foundation, award number CNS-1652474.