Guiding Hidden Layer Representations for Improved Rule Extraction from Neural Networks

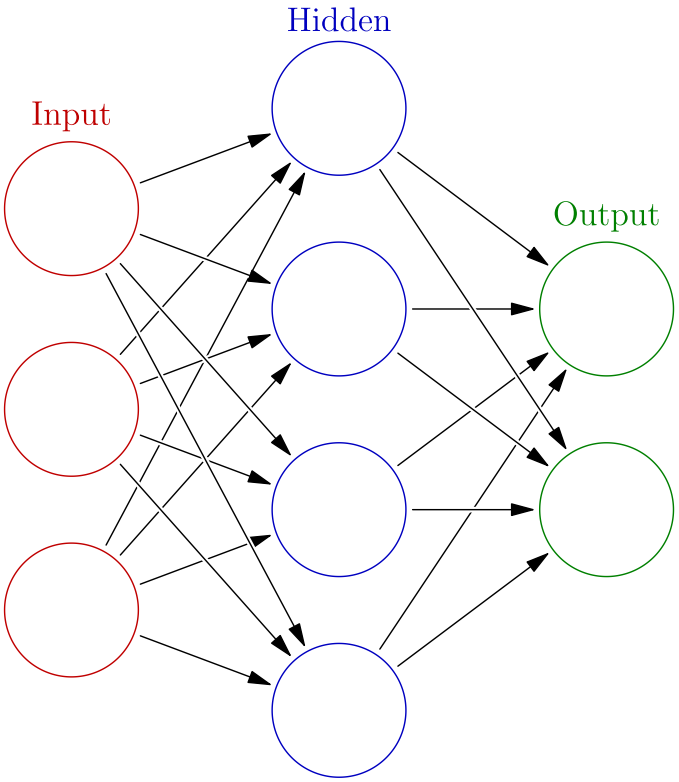
Thuan Q. Huynh and James A. Reggia

Artificial Neural Networks (ANNs) are flexible learning algorithms that are used to solve problems from speech synthesis to handwriting recognition or even self-driving cars. But how ANNs generate outputs can appear to be like a black box, which limits some areas of use of ANNs. Professor James Reggia at the University of Maryland has developed an approach to allow us to peer inside the black box, leading to more impactful applications of ANNs.

Background

ANNs are a family of algorithms inspired by biological neural networks. Unlike most algorithms, which compute results serially, step by step, neural networks are massively parallel systems of interconnected neurons (nodes) with synapses (connections) of varying connection strengths. Computation happens concurrently across the network, with each neuron only exchanging information with the neurons it is connected to. The global computation that a neural network performs is an emergent result from the small, local computations each neuron is computing and the varying strengths, often referred to as the "weights" of the synapses between neurons.

Learning algorithms for ANNs focus on efficient, robust and accurate ways to learn appropriate weights for the connections in the network. Typical training involves feeding the network an example input, computing the error of its output relative to an already known answer, and "backpropagating" the error through the network - making small adjustments to the weights throughout the network. Notably, most neural network learning algorithms only rely on local information to learn; calculating the appropriate changes to the weights for a particular neuron only requires information about the neurons immediately adjacent in the network topology. In this sense, ANNs closely mimick biological neural networks. Because so few assumptions were made about the problem, neural networks can solve a variety of problems.



(Source:
http://commons.wikimedia.org/wiki/File:Colored_neural_network.svg)

Predictable but not always understandable

However, a key weakness of artificial neural networks is the difficulty to understand what, exactly, a trained neural network has "learned". Most machine learning methods build, or can be easily adapted to build human-readable representations of their knowledge such as decision trees (similar to flowcharts) or decision rules, (often of the form "If *condition* then *outcome*"). The transparency and flexibility offered by a human-readable representation would allow many more potential applications. For example, a handwriting recognition app would have happier customers if it justified *why* a signature was rejected.

Unfortunately, the machine representation of a trained ANN is a large table of random-looking real numbers. The weight of each connection in the network contributes to the overall computation, but while learning, each neuron only receives information about the neurons it is connected to. The result is a meaningful *whole* network but with seemingly meaningless subcomponents. This presents enormous difficultly in extracting human-understandable rules.

Simplicity with little sacrifice

Huynh and Reggia's novel approach works by modifying the learning method used to the network. Instead of defining the error to be just the difference between the correct result and the predicted result, a new error term adds an additional penalty based on how close weight vectors are to each other. This

encourages the network to maximize the distances between each vector of weights, in a sense, "pushing" weight vectors away from each other.

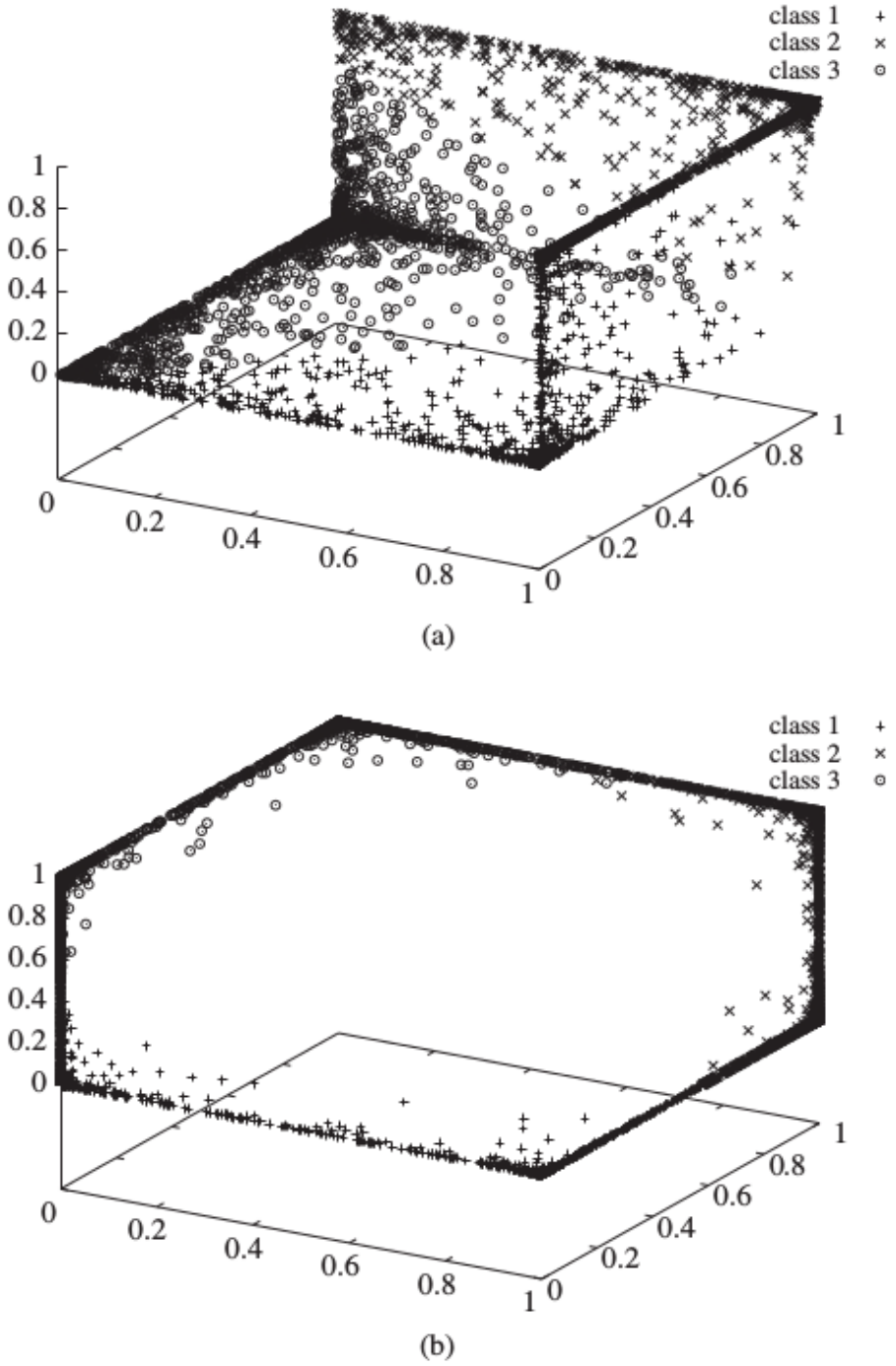


Fig. 2. Input patterns throughout hidden unit activation space for the *waveform* problem after training with (a) regular backpropagation ($E = E_1$) versus (b) same error function but augmented to include the new error term ($E = E_1 + E_2$).

(source: https://www.cs.umd.edu/sites/default/files/scholarly_papers/thuanhuynh_1.pdf

An interesting property of the new error term is that the entire calculation is still effectively local - each neuron still only requires information immediately

adjacent to it in the network. In addition, the new error term is agnostic as to the algorithm used in the rule-extraction phase. Experiments show that this approach can significantly reduce the number of rules generated during rule extraction without sacrificing classification accuracy in the neural network itself. So Professor Reggia's approach offers the best of both worlds: accurate predictions based upon relatively easy to understand rules.

References

[1] T. Huynh and J. Reggia "Guiding hidden layer representations for improved rule extraction from neural networks", IEEE Transactions on Neural Networks, vol. 22, no. 2, pp.264-275 2011

This report was prepared by Josh Brule, Neal Gupta, Assaf Magen, and Saeedreza Seddhighin. Click here to contact them.