Brendan Ritter
4/26/13
Artificial Intelligence

Topic Mastery-Neural Networks

What are neural networks?

Neural networks are an interconnected group of nodes that preform calculations by either firing or not firing given a number of inputs to the node. Together these nodes form a system that can adapt to different problems.

History of neural networks

The history of neural networks started longer ago than one might expect. Mathematicians and biologists alike have been trying to understand the secrets of the brain since the discovery of the axon, the tendril like cells that compose our brain. This work was started in the 1930's.

However, McCulloch and Pitts in 1943, has become the definitive starting point for the computational neural network. In this paper they introduced, in mathematical and computational terminology (rather than psychological) the description of a neuron that, when a threshold of potential was reached, would fire. This firing, would be received and acted upon by other neurons and so forth.

Further work was done by Frank Rosenblatt in the late 1950's. His model, called the perceptron, attracted much interest and fostered further research in the field.

Perceptrons, a book authored by Marvin Minsky and Seymour Papert highlighted some of the capabilities and downfalls of neural networks. In particular, their example that a simple neural network could never learn the XOR operation lead many to believe that, since simple neural networks were flawed in this aspect, that further development of more complicated networks was pointless. Despite this flawed assumption (it was known by Minsky and Papert that more complicated networks could be trained to make the XOR) and the eventual discovery with other flaws in the book's theory, it still lead to the decline of the use of neural networks for decades.

However, in recent year, more developments have been made with neural networks. In addition, a limiting factor of the early neural networks, processing power, has increased significantly, allowing further research into their abilities.

Some simple examples

Several functions for this threshold idea have been proposed. One example is the binary neuron, which is purely a step function. This means that the neuron either fires or it doesn't, giving it binary nature. It this situation the number -1 or 0 commonly maps to False and 1 maps to True.

Although there are multiple more models, the only other one that seems to still be commonly used is the sigmoid neuron in which the threshold function follows the equation:

$$\sigma(t) = \frac{1}{1 + e^{-\beta t}}$$

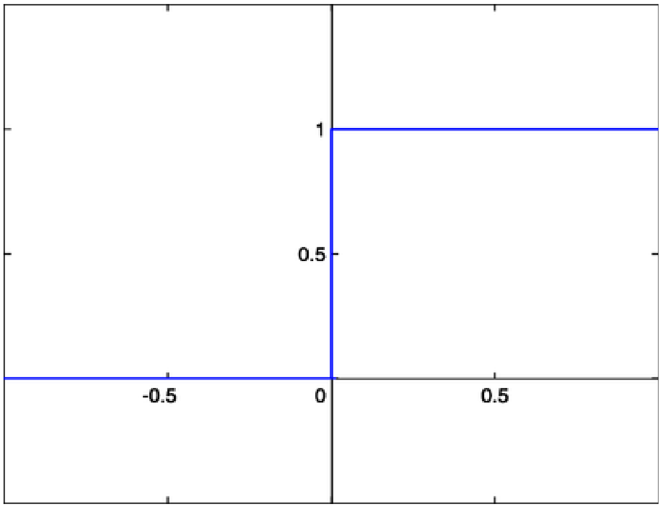

Illustration 1: Binary Neuron

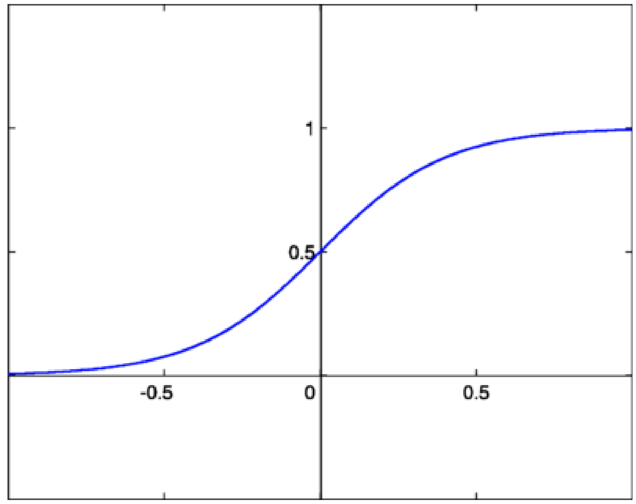

Illustration 2: Sigmoid Neuron

Regardless of the function used, the structure of the computation remains the same. An additional factor in neural networks is the weight that each neuron puts on the input from another. Furthermore, one can also assign the output of the neuron a probability rather than a fixed value. In this case the neural network is no longer deterministic, although this fact is sometimes desired for more complicated behavior. This is usually the case for a subset of Neural networks, Bayesian Networks.

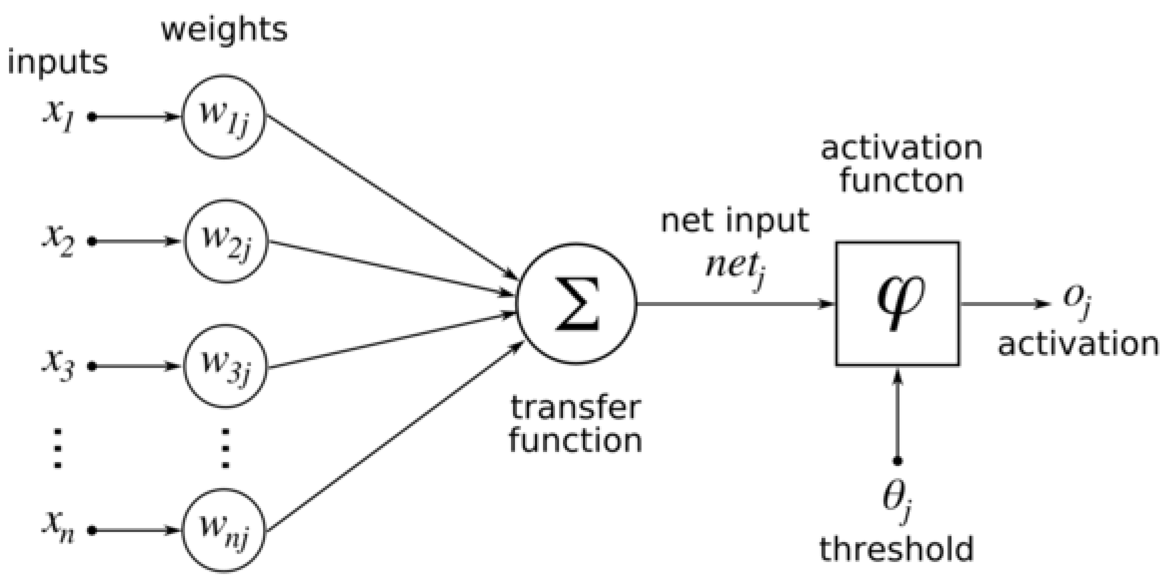

Illustration 3: A Example of a General Neural Network Structure

In the example above, the neuron receives inputs x_1 through x_c and assigns each a weight w_{1j} through w_{nj}. Commonly the weights and inputs are multiplied. This product is then summed along with the rest of the input-weight products. The sum of these input-weights is the (net) input to the activation function of the neuron. This information is then compared to the threshold of the neuron and the result is fired. This firing is either the input to another neuron, or the output to the network itself.

When there exist nodes which are neither input nor output, the network has what is called a hidden layer. The size of the input, output and hidden layers can change, but generally most problems that are neural net friendly can be solved with one hidden layer.

Neural networks can also be more complex through the addition of loops or cycles in the network. These networks are called recurrent neural networks. If a network has none of these cycles, it is referred to as a feedfoward neural network.

Recurrent neural networks are important since their cyclic nature allows them to have state and thus 'memory' of what they have seen.

Training Neural Networks

Although the structure of neural networks may seem simple, their functionality and ability to correctly solve a problem derives from repeated input and then evaluation of the network's performance. The repeated input allows refinement what the weights are on each of the nodes. This procedure is called 'training' a neural network.

Training a network with a large amount of data in crucial for correct behavior, especially for edge cases. For instance, a neural network taught to drive and only presented with courses that had left turns might not behave correctly when it encountered a right turn and certainly would not stop at a stop light.

What neural networks are good/bad at

In general, neural networks are good at tasks such as image recognition and more generally, pattern recognition, classification of data and predicting the next value in a series of data. 'Expert' systems, like medical diagnosis can be preformed by neural networks. Neural networks are also good at data filtering and data mining. Another feature are networks ability to preform computation through purely analog means. Unlike traditional programs, the inputs and outputs of neural networks can be analog, allowing theoretically almost infinite precision.

Both a strength and weakness of neural networks is their ability to recognize patterns. Although neural networks are a good tool for pattern recognition, some users have found that neural networks will at times be over zealous in that recognition. Simply put, they find patterns in data that don't actually exist. Some of this is due to error in the data itself, which, because neural networks are trained, leads to the phrase "garbage in, garbage out" meaning that regardless of the construction of the network, if it is trained on inferior data, it will result in inferior answers.

Neural networks are the wrong approach for systems where there are a series of well defined steps like mathematical formula. In addition since the system must be trained, a large quantity of data must be obtained. If this data is not available, neural networks can't be used, for instance the analysis of very rare occurrences like super nova would not be a problem for neural networks.

Furthermore, since neural networks simply adapt to solve the problems they are given, there is no validation of a neural network and no further understanding gained of the process, the neural network simply outputs the answer. A close study of a neural network would only yield the things that make it compute, the weights of the inputs and the structure of the net. Neither of these things allows for advancement of theory.

Future of neural networks

Unlike many other topics in AI , neural networks are still used today, especially for their ability for pattern recognition. Companies like Google and Toyota for example are using this ability in autonomous vehicles for reacting to possible collisions and other detrimental events.
Banks use the optical character recognition abilities of neural networks to process handwritten checks.

Farther reaching theories have extended neural networks to quantum computing, extending neural network's pattern matching abilities significantly, allowing for things like real time face tracking. Furthermore, due to the superposition of memory in a quantum computer, the teaching of a quantum neural network would not require repetition in order to teach it!

Other recent advances include the creation of specialized hardware necessary for analog neural networks. Because these systems operate differently from traditional digital circuits, time is necessary before these new hardwares become more wide spread. However, they offer huge increases in speed and preciseness due to their analog nature.

Recent work at the University of Maryland in applying neural networks to climate models has also resulted in some success.

Another application of neural networks is their use in computational models of non-linear systems. Current computers have some trouble even with numerical methods in tackling some differential equations, and some even remain unsolved, even with extensive computing time. Given enough training data, and helped by their analog operation, analog neural networks have been proposed as a solution for certain non-linear equations.

Finally, because the topic of neural networks has not been exhausted and because the inception of neural networks was derived from biological systems, further research into neural networks can tell us more about the nature of our brain, how it calculates, how it functions. This line of research may yet lead to a theory of emergent consciousness. The end result being a strong Ai. One that can think and preform any task a human can.

Works Cited

Weigang, Li. *Entangled Neural Networks*. Diss. University of Brasilia, Print.

Vemuri, Rao. "Main Problems and Issues in Neural Networks Application." *Artificial Neural Networks and Expert Systems*. (1993): n. page. Print.

Beelen, Wesley. "Ground Plane Classification using edge based segmentation and artificial neural networks." (2007): n. page. Print.

Heaton, Jeff. "Solving Problems with Neural Networks ." *Heaton Research*. N.p., 23 12 2007. Web. 28 Apr 2013. <http://www.heatonresearch.com/articles/1/page3.html>.

"Neural Networks." *IBM Infocenter*. IBM. Web. 28 Apr 2013. <http://publib.boulder.ibm.com/infocenter/iseries/v5r3/index.jsp?topic=/rzahx/reference/com/ibm/able/beans/doc-files

"Analog Neural Networks: Computational Power." *BINDS Lab*. Umass Amhearst, n.d. Web. 28 Apr 2013. <http://binds.cs.umass.edu/anna_cp.html>.

Weigang, Li. "Quantum Neural Computing Study." . N.p., n.d. Web. 28 Apr 2013. <http://www.cic.unb.br/~weigang/qc/aci.html>.