

Neural Networks Pt. 1

Terms & Definitions

Lecture 5 I400/I590

Artificial Life as an approach to Artificial Intelligence

Larry Yaeger

Professor of Informatics, Indiana University

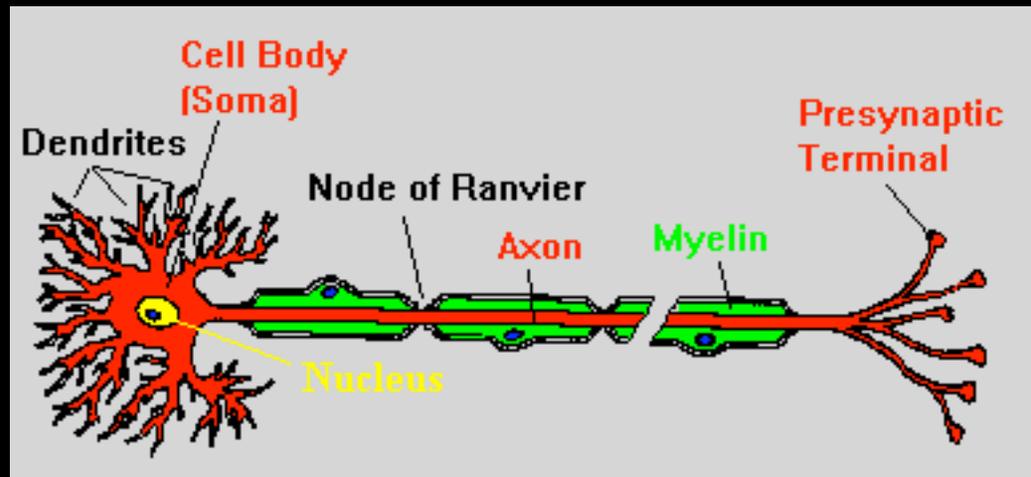
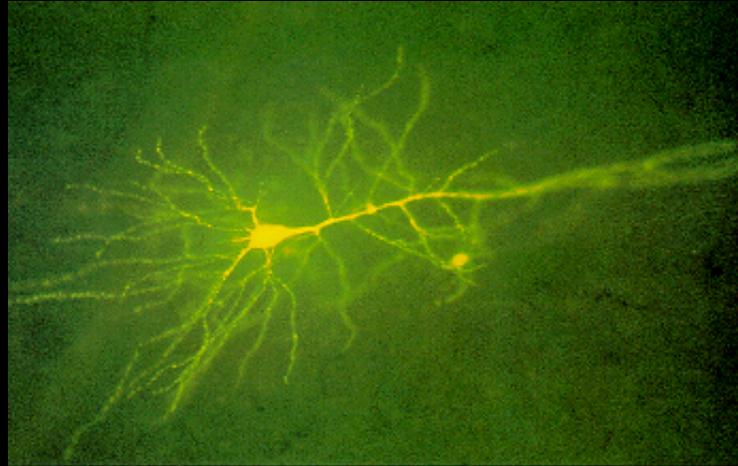
What are neural networks?

- "... systems that are deliberately constructed to make use of some of the organizational principles that are felt to be used in the human brain." — Anderson
- "... a neural network is a system composed of many simple processing elements operating in parallel whose function is determined by network structure, connection strengths, and the processing performed at computing elements or nodes." — DARPA Neural Network Study (1988)
- "A Neural Network is an interconnected assembly of simple processing elements, units or nodes, whose functionality is loosely based on the animal neuron. The processing ability of the network is stored in the inter-unit connection strengths, or weights, obtained by a process of adaptation to, or learning from, a set of training patterns." — Gurney

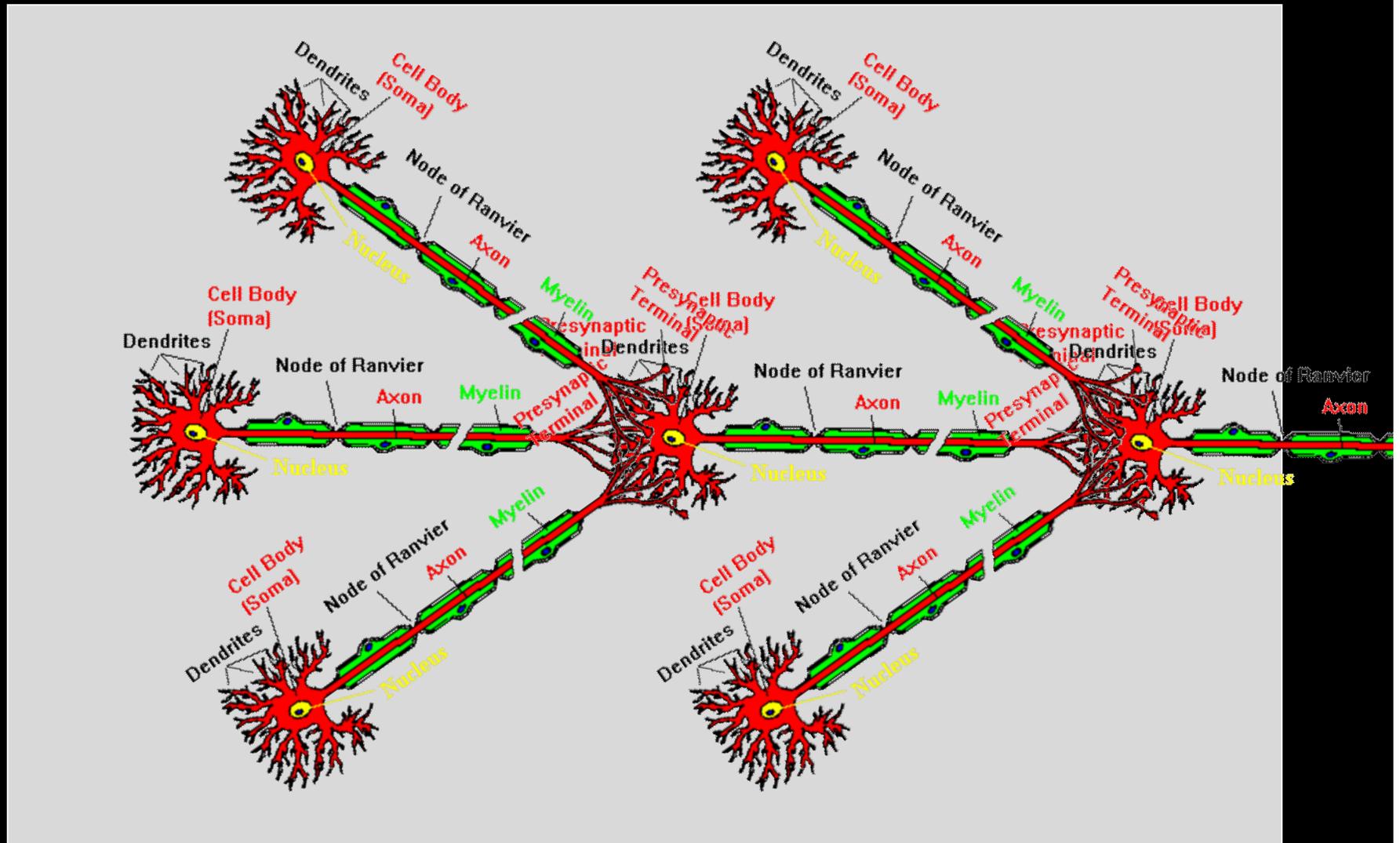
Motivations

- Humans have always wanted to build intelligent machines — stories of golems, automata, mechanical men, and robots have been around for millenia
- Computer technology is rapidly approaching the processing power of human brains
 - But organization and function are still huge unknowns
- Biological brains use neurons that are 10^5 or 10^6 times slower than silicon logic gates, yet perform computations better and faster than our best computational algorithms
- (Artificial) Neural Networks are an attempt to harness the massively parallel, distributed computation of biological brains for a variety of purposes

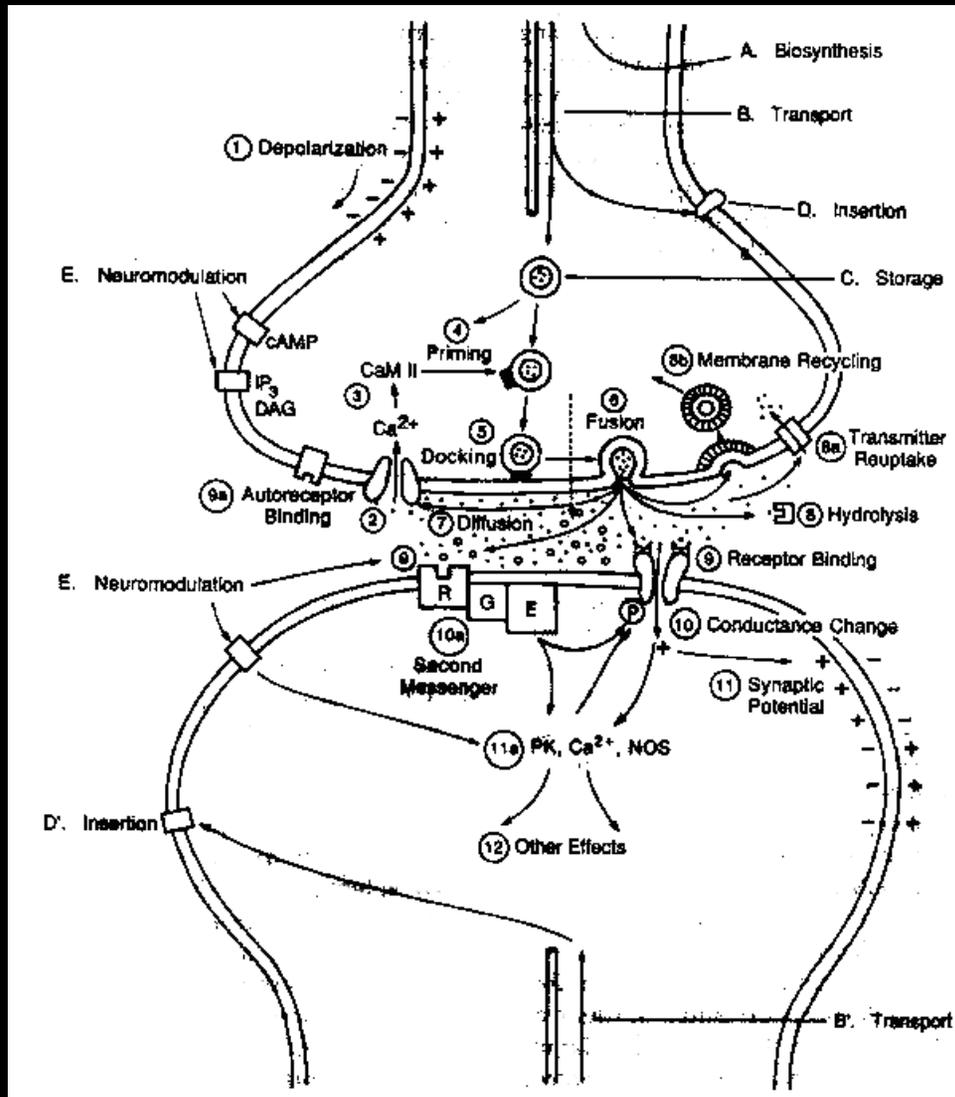
What is a neuron?



Network of neurons



What is a synapse?

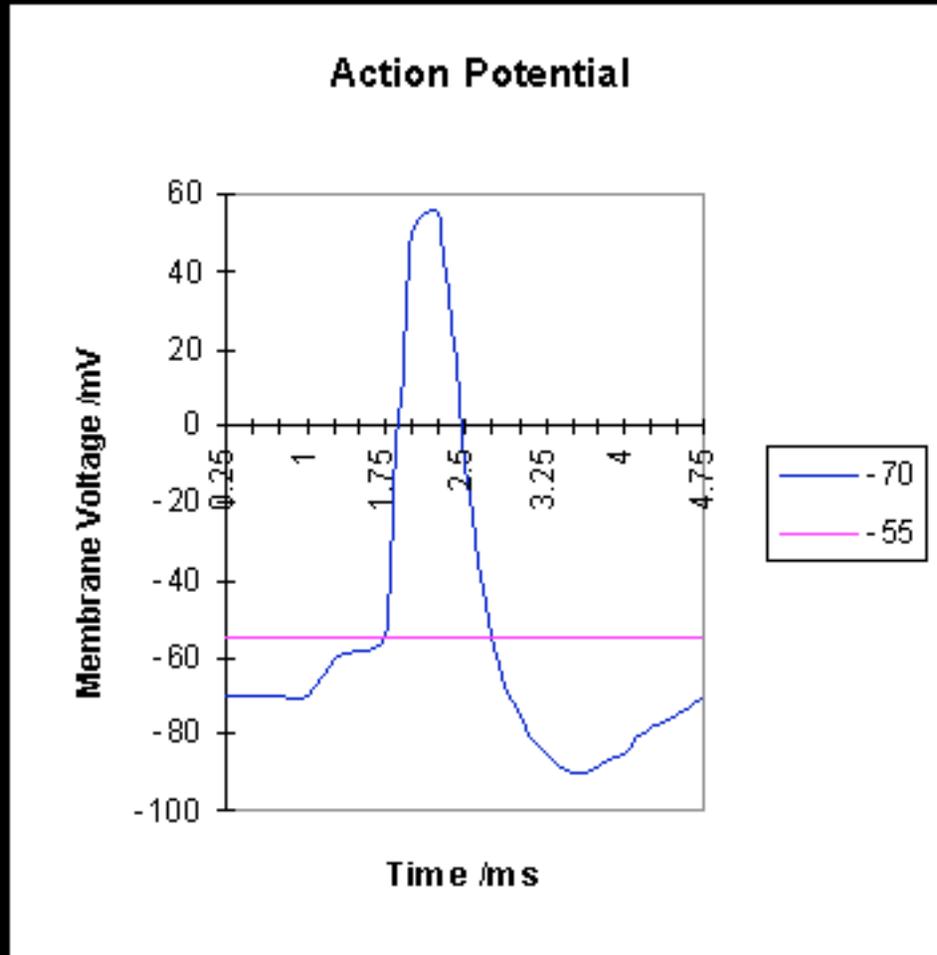


Neurotransmitters emitted from the presynaptic terminal drive ion uptake into the postsynaptic cell, affecting its *membrane potential* (the difference in electrical charge between the cell and its surround)

Kinds of synapses

- Excitatory
 - Increases membrane potential
 - Makes it more likely that (postsynaptic) neuron will fire its action potential
- Inhibitory
 - Decreases membrane potential
 - Makes it less likely that (postsynaptic) neuron will fire its action potential

How do neurons propagate signals?



Neurons normally have a negative *resting potential* of about -65 – 80 mV, with a negative cytoplasm and positive ions lining the cell membrane (K^+ inside, Na^+ out). Neurotransmitters emitted from the presynaptic terminal drive positive ion uptake (extracellular K^+) into the postsynaptic cell, making its *membrane potential* less negative, until the postsynaptic neuron reaches a threshold and fires its *action potential*—a faster influx of positive ions (Na^+) that sweeps along the cell membrane, temporarily driving the cell to have a positive membrane potential. This action potential causes the cell to emit its own neurotransmitters, thus propagating the electrical signal. At the height of the membrane potential reversal, a rapid efflux of K^+ occurs, returning the cell to its negative polarization. And finally sodium-potassium pumps work continuously to restore the balance of K^+ and Na^+ along the cell membrane and restore the nominal resting potential.

How big is a human brain?

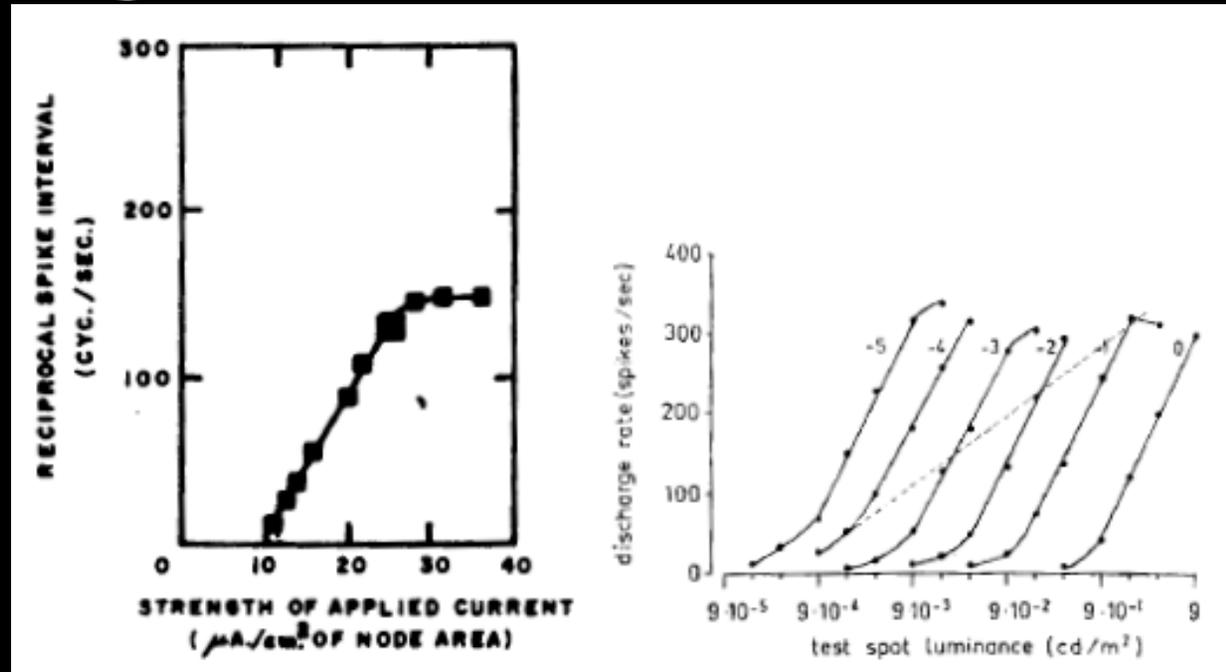
- 10^{10} to 10^{11} neurons (10 to 100 billion)
- 10^2 to 10^4 synapses per neuron (100 to 10 thousand)
- 10^{12} to 10^{15} synapses total
- 10^{11} neurons and 10^{14} synapses are probably reasonable ballpark values
- Note: 5×10^8 transistors on a chip routinely, 10^9 or greater with modern 45 nm process chips, and Intel predicts 10^{10} by 2010

Neural activity

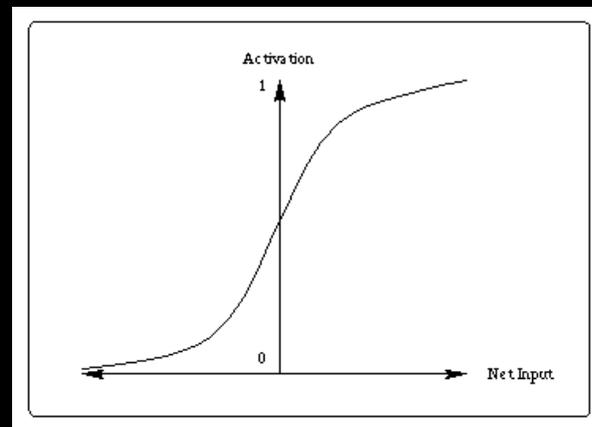
- Some scientists believe that temporal activity—the neuron integrate and fire behavior, aka the *spiking* behavior—is critical to understanding and modeling brain function
 - There does seem to be evidence that the brain uses the spike timing and other temporal characteristics of neural dynamics
- Some modeling work has fit data and made successful predictions about brain function paying attention purely to the *firing rate* (spike frequency) of neurons
 - Spikes/second (action potentials/second)
- Until we learn more about complex network dynamics, both models seem to have their uses

Neural firing rates

Real neural firing rates (from different brain areas, in response to different stimuli)



Artificial neuron firing rate determined from *sigmoidal squashing function (logistic function)* and degree of stimulation



$$\frac{1}{(1 + e^{-x})}$$

Learning in neural networks

- Knowledge is represented in neural networks by the strength of the synaptic connections between neurons (hence "*connectionism*")
- Learning in neural networks is accomplished by adjusting the synaptic strengths (aka *synaptic weights*, *synaptic efficacy*)
- There are three primary categories of neural network learning algorithms (in increasing order of biological plausibility):
 - Supervised — *exemplar* pairs of inputs and (known, labeled) target outputs are used for training
 - Reinforcement — single good/bad training signal used for training
 - Unsupervised — no training signal; self-organization and clustering produce (and are produced by) the "training"

History of neural networks

- Alexander Bain, Scottish inventor of the first electrical clock and the "chemical telegraph" (fax machine), in 1873 published *Mind and Body: The Theories of Their Relation*, proposing networks of artificial neurons that used a Hebb-like adaptive rule to produce associations for inputs that are "made together, or flow in close succession"
 - In 1876 he founded the first psychological journal entitled *Mind*
- William James, American psychologist, in 1890 published *Principles of Psychology*, restating Bain's thesis: "When two elementary brain-processes have been active together or in immediate succession, one of them, on reoccurring, tends to propagate its excitement into the other."

History of neural networks

- Nicolas Rashevsky, in addition to early work on a statistical theory of neural fields, proposed the first neural logic circuit in 1938 with an EXCLUSIVE-OR network based on binary logic, and proposed that the brain could be organized around binary logic operations, since the action potential could be seen as a binary operator
- Warren McCulloch (a psychiatrist and neuroanatomist) and Walter Pitts (a mathematician), started the modern era of neural networks when, in 1943, they proposed a neuron model that summed its inputs and fired whenever that sum exceeded a threshold
 - Using a network of such threshold neurons they showed it was possible to construct any logical function

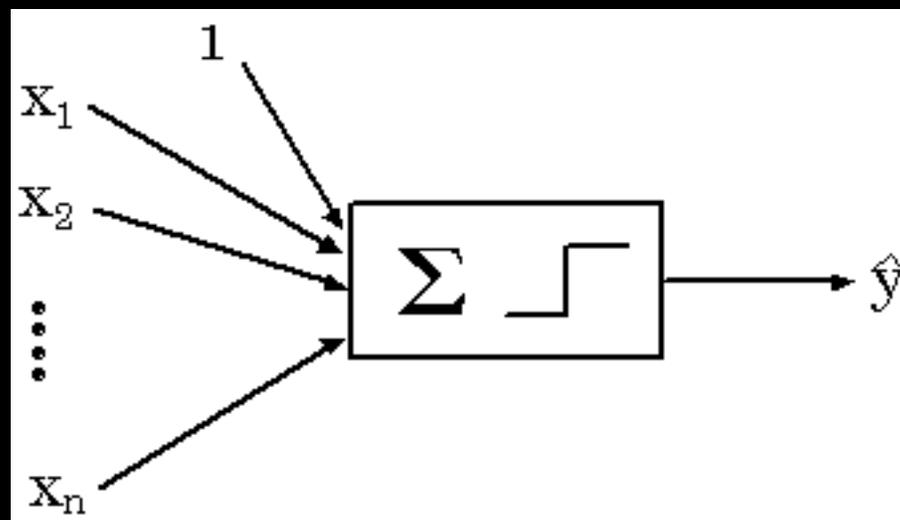
History of neural networks

- Donald Hebb, in 1949, published *The Organization of Behavior*, in which he proposed that the effectiveness of a synapse between two neurons might be increased by correlated activity of those two neurons
 - Used widely in artificial neural models today
 - There is also evidence of a biological mechanism in which neural activity modulates calcium ion flow, which, in turn, affects the physical structure of synapses
- Farley and Clark, in 1954 on an early digital computer at MIT, modeled a network of randomly connected neurons employing a modified Hebb learning rule that was able to learn to discriminate between two input patterns

History of neural networks

- Frank Rosenblatt, in 1958, introduced the single-layer *perceptron* with adjustable synaptic weights and a threshold output neuron

- Rosenblatt conceptualized perceptrons as consisting of three layers, for *sensory*, *association*, and *response*, but the weights only varied on the synapses leading to the single threshold neuron



- He also introduced an error-correction rule (the *perceptron learning rule*) to adapt the weights and proved that if two input classes were linearly separable the algorithm would converge to a solution (the *perceptron convergence algorithm*)

History of neural networks

- Bernard Widrow and Ted Hoff, in 1960, introduced the *Least-Mean-Squares algorithm* (*delta-rule* or *Widrow-Hoff rule*) and used it to train ADALINE (ADaptive LINear Elements or ADaptive LINear NEurons)
 - ADALINE was similar to a perceptron, except for using a linear activation function instead of a threshold
 - MADALINE (Multiple ADALINE) used multiple ADALINE units, still in a single conceptual layer, and combined their outputs in a fixed way (AND, OR, majority vote takers)
 - MADALINE and the delta-rule were used to eliminate echo in phone lines in the first practical application of ANNs, and they are still used today in telecommunications

History of neural networks

- Marvin Minsky and Seymour Papert, in 1969, published *Perceptrons*, in which they mathematically proved that single-layer perceptrons were only able to distinguish linearly separable classes of patterns
 - While true, they also (mistakenly) speculated that an extension to multiple layers would lose the "virtue" of the perceptron's simplicity and be otherwise "sterile"
- In the 1950s and 1960s, symbolic AI and sub-symbolic Connectionism competed for prestige and funding
 - AI investigated higher-order cognitive problems—logic, rational thought, problem solving
 - Connectionism investigated neural models like the perceptron and struggled to find a learning algorithm for multi-layer perceptrons
 - As a result of Minsky & Papert's *Perceptrons*, research in neural networks was effectively abandoned in the 1970s and early 1980s

History of neural networks

- Shun-ichi Amari, in 1967, and Christoph von der Malsburgh, in 1973, published ANN models of self-organizing maps, but the work was largely ignored
- Paul Werbos, in his 1974 PhD thesis, first demonstrated a method for training multi-layer perceptrons, essentially identical to *Backprop* but the work was largely ignored
- Stephen Grossberg and Gail Carpenter, in 1980, established a new principle of self-organization called Adaptive Resonance Theory (ART), largely ignored at the time
- John Hopfield, in 1982, described a class of recurrent networks as an associative memory using statistical mechanics; now known as Hopfield networks, this work and Backprop are considered most responsible for the rebirth of neural networks

History of neural networks

- Teuvo Kohonen, in 1982, introduced *SOM* algorithms for *Self-Organized Maps*, that continue to be explored and extended today
- David Parker, in 1982, published an algorithm similar to Backprop, which was ignored
- Kirkpatrick, Gelatt and Vecchi, in 1983, introduced *Simulated Annealing* for solving combinatorial optimization problems
- Barto, Sutton, and Anderson in 1983 popularized *reinforcement learning* (it had been addressed briefly by Minsky in his 1954 PhD dissertation)
- Valentino Braitenberg, in 1984, published *Vehicles*, in which he advocates for a bottom-up, synthesis approach to understanding complex systems, and bases his mental models on neurophysiological data
- Yann LeCun, in 1985, published an algorithm similar to Backprop, which was again ignored

History of neural networks

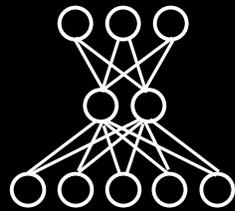
- David Ackley, Geoffrey Hinton, and Terrence Sejnowski, in 1985, introduced the Boltzmann machine (the first successful realization of a multilayered neural network)
- David Rumelhart, Geoffrey Hinton, and Christopher Williams, in 1986, introduced *Backprop*, the first widely recognized learning algorithm for multilayer perceptrons or neural networks
- David Rumelhart, James (Jay) McClelland, and the PDP Research Group, in 1986, published *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volumes 1 and 2*, and officially resuscitated neural network research
- Terrence Sejnowski and Charles Rosenberg, in 1986, introduced NETtalk, a text-to-speech neural network system trained using Backprop, and one of the earliest practical applications of multilayer perceptrons
- In 1987 NIPS, INNS, IJCNN neural network conferences began

Derivation of Backprop

Output layer

Hidden layer

Input layer



Define:

a_i = activation of neuron i

w_{ij} = synaptic weight from neuron j to neuron i

x_i = excitation of neuron i (sum of weighted activations coming into neuron i , before squashing)

t_i = target vector

o_i = activation of output neuron i (same as a_i)

By definition:

$$x_i = \sum_j w_{ij} a_j$$

$$a_i = 1 / (1 + e^{-x_i})$$

Summed, squared error at output layer: $E = 1/2 \sum_i (t_i - o_i)^2$

Derivation of Backprop

By chain rule:

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial o_i} \frac{\partial o_i}{\partial x_i} \frac{\partial x_i}{\partial w_{ij}}$$

$$\frac{\partial E}{\partial o_i} = (1/2) \cdot 2 \cdot (t_i - o_i) \cdot (-1) = (o_i - t_i) \quad E = 1/2 \sum_i (t_i - o_i)^2$$

$$\begin{aligned} \frac{\partial o_i}{\partial x_i} &= \frac{\partial}{\partial x_i} (1 + e^{-x_i})^{-1} = - (1 + e^{-x_i})^{-2} \cdot (-e^{-x_i}) = e^{-x_i} / (1 + e^{-x_i})^2 \\ &= \frac{(1 + e^{-x_i}) - 1}{(1 + e^{-x_i})} \cdot \frac{1}{(1 + e^{-x_i})} = [1 - 1 / (1 + e^{-x_i})] \cdot [1 / (1 + e^{-x_i})] \\ &= (1 - o_i) o_i \end{aligned}$$

$$\frac{\partial x_i}{\partial w_{ij}} = a_j \quad x_i = \sum_j w_{ij} a_j$$

Derivation of Backprop

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial o_i} \frac{\partial o_i}{\partial x_i} \frac{\partial x_i}{\partial w_{ij}}$$

$$= \underbrace{(o_i - t_i)}_{\text{raw error term}} \underbrace{(1 - o_i)o_i}_{\text{saturation due to sigmoid}} \underbrace{a_j}_{\text{modulation due to incoming (pre-synaptic) activation}}$$

$$\Delta w_{ij} = - \eta \frac{\partial E}{\partial w_{ij}} \quad (\text{where } \eta \text{ is an arbitrary learning rate})$$

$$w_{ij}^{t+1} = w_{ij}^t + \eta \underbrace{(t_i - o_i)}_{d_i} \underbrace{(1 - o_i) o_i a_j}_{e_i}$$

Derivation of Backprop

Now need to compute weight changes in the hidden layer, so, as before, we write out the equation for the error function slope w.r.t. a particular weight leading into the hidden layer:

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial a_i} \frac{\partial a_i}{\partial x_i} \frac{\partial x_i}{\partial w_{ij}}$$

(where i now corresponds to a unit in the hidden layer and j now corresponds to a unit in the input or earlier hidden layer)

From previous derivation, last two terms can simply be written down:

$$\frac{\partial a_i}{\partial x_i} = (1 - a_i) a_i$$

$$\frac{\partial x_i}{\partial w_{ij}} = a_j$$

Derivation of Backprop

However, the first term is more difficult to understand for this hidden layer. It is what Minsky called the *credit assignment problem*, and is what stumped connectionists for two decades. The trick is to realize that the hidden nodes do not themselves make errors, rather they contribute to the errors of the output nodes. So, the derivative of the total output error w.r.t. a hidden neuron's activation is the sum of that hidden neuron's contributions to the errors in all of the output neurons:

$$\frac{\partial E}{\partial a_i} = \sum_k \frac{\partial E}{\partial o_k} \frac{\partial o_k}{\partial x_k} \frac{\partial x_k}{\partial a_i} \quad (\text{where } k \text{ indexes over all output units})$$

contribution of each output neuron

contribution of all inputs to the output neuron (from the hidden layer)

contribution of the particular neuron in the hidden layer

Derivation of Backprop

From our previous derivations, the first two terms are easy:

$$\frac{\partial E}{\partial o_k} = (o_k - t_k)$$

$$\frac{\partial o_k}{\partial x_k} = (1 - o_k) o_k$$

For the third term, remember:

$$x_k = \sum_i w_{ki} a_i$$

And since only one member of the sum involves a_i :

$$\frac{\partial x_k}{\partial a_i} = w_{ki}$$

Derivation of Backprop

Combining these terms then yields:

$$\frac{\partial E}{\partial a_i} = - \sum_k \underbrace{(t_k - o_k) (1 - o_k) o_k}_{d_k} w_{ki}$$

Weights between hidden and output layers

And combining with previous results yields:

$$\frac{\partial E}{\partial w_{ij}} = - \left(\sum_k d_k w_{ki} \right) (1 - a_i) a_i a_j$$
$$w_{ij}^{t+1} = w_{ij}^t + \eta \underbrace{\left(\sum_k d_k w_{ki} \right)}_{d_i} (1 - a_i) a_i a_j$$

Derivation of Backprop

Propagation to multiple hidden layers follows the same pattern, using d and w from the layer above and a from the layers on either side of the weights being updated.

Voila, you have derived Backprop!

Credits

- Real neuron photo on slide 4 from <http://coe.sdsu.edu/eet/Articles/anns/start.htm>
- Schematic neuron drawing on slides 4 & 5 from <http://faculty.washington.edu/chudler/cells.html>
- Schematic synapse drawing on slide 6 from http://www.neurocomputing.org/html/real_neurons.html
- Action potential graph on slide 7 from <http://www.emc.maricopa.edu/faculty/farabee/BIOBK/BioBookNERV.html>
- Real neural firing rate graphs on slide 11 from *Neurocomputing*, Ed. Anderson
- Sigmoid graph on slide 11 from <http://www2.psy.uq.edu.au/~brainwav/Manual/BackProp.html>
- Perceptron image from <http://documents.wolfram.com/applications/neuralnetworks/NeuralNetworkTheory/2.4.0.html>

References

- General introduction and history
 - http://www.dacs.dtic.mil/techs/neural/neural_ToC.html
 - <ftp://ftp.sas.com/pub/neural/FAQ.html>
 - <http://www.neurocomputing.org/>
 - <http://www.statsoft.com/textbook/stneunet.html>
 - <http://www.cs.stir.ac.uk/~lss/NNIntro/InvSlides.html>
 - <http://rfhs8012.fh-regensburg.de/~saj39122/jfroehl/diplom/e-index.html>
 - <http://www.shef.ac.uk/psychology/gurney/notes/l1/l1.html>
 - <http://www-cse.stanford.edu/classes/sophomore-college/projects-00/neural-networks/index.html>
- BackProp
 - http://courses.cs.tamu.edu/rgutier/cs790_w02/l11.pdf (+history)
 - <http://www2.psy.uq.edu.au/~brainwav/Manual/BackProp.html>
 - http://www.ccs.fau.edu/~bressler/EDU/CogNeuro/Perceptrons_hbttn.htm