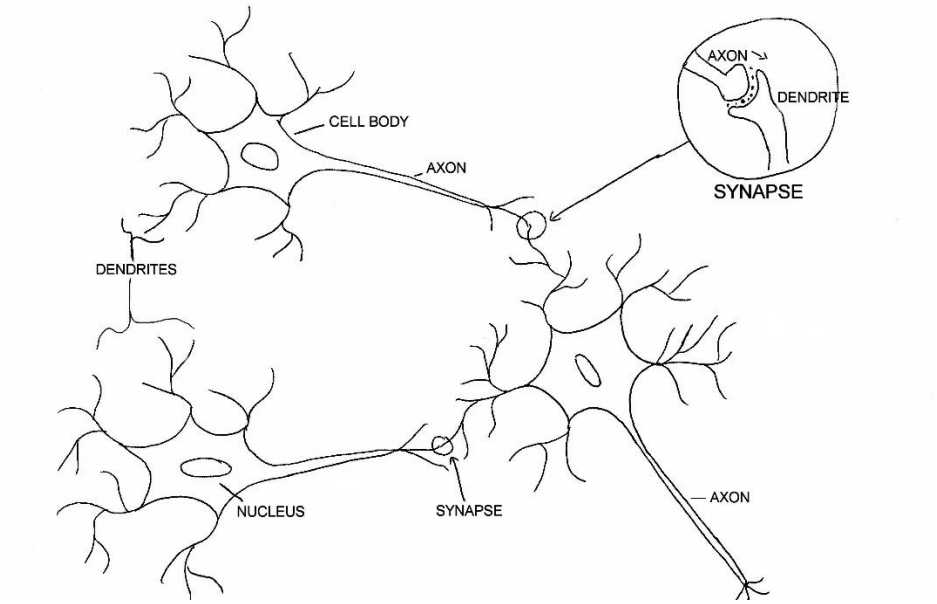
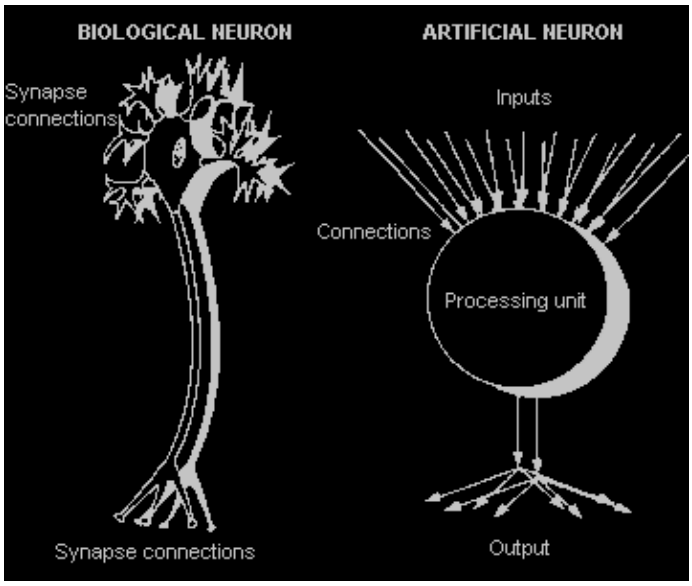
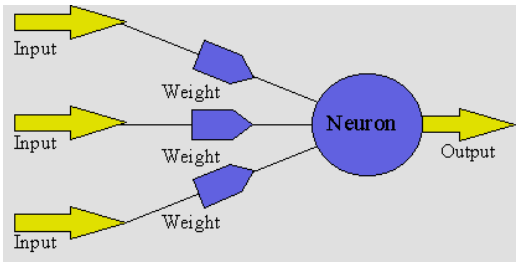
Introduction to Neural Networks

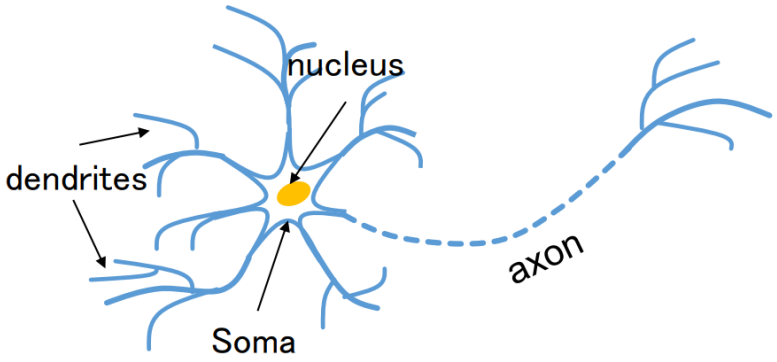






Human brain has been extensively studied by scientists. Its vast complexity prevents all but rudimentary understanding, even the behavior of an individual neuron is extremely complex. There are about ten billion neurons. Neuron's switching time $>10^{-3}$ secs. On average, each neuron has several thousand connections and hundreds of operations per second. A neuron only fires if its input signal exceeds a certain amount (the threshold) in a short time period. Synapses can vary in strength. Good connections allowing a large signal and slight connections allow only a weak signal. Synapses can be either excitatory or inhibitory. A neuron has a cell body, a branching input structure (the dendrite) and a

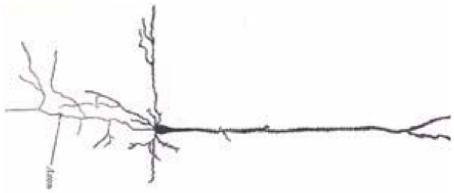
branching output structure (the axon). Axons connect to dendrites via synapses. Electro-chemical signals are propagated from the dendritic input, through the cell body, and down the axon to other neurons.

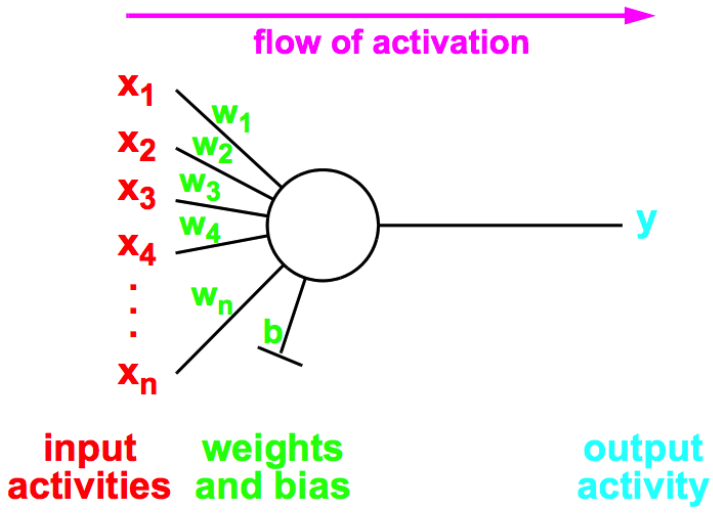


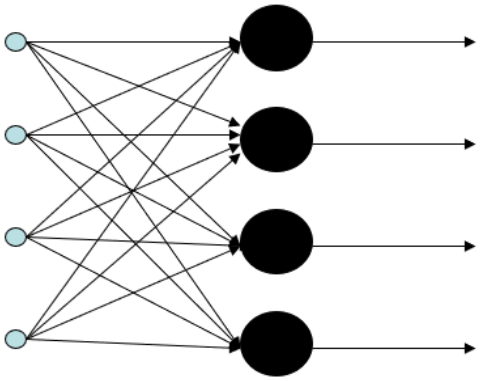
Artificial neural networks (ANN) are modeled taking inspiration from human brains.

Properties of ANN:

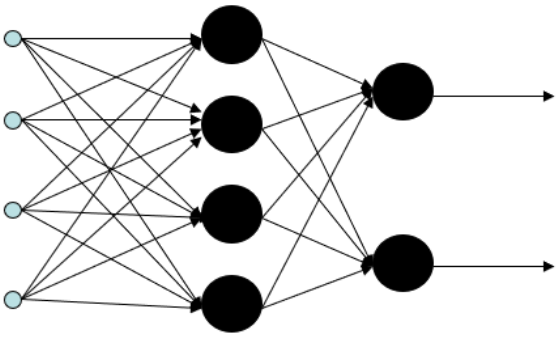
- ✓ Many simple neuron-like threshold switching units
- ✓ Many weighted interconnections among units
- ✓ Highly parallel, distributed processing
- ✓ Learning by tuning the connection weights



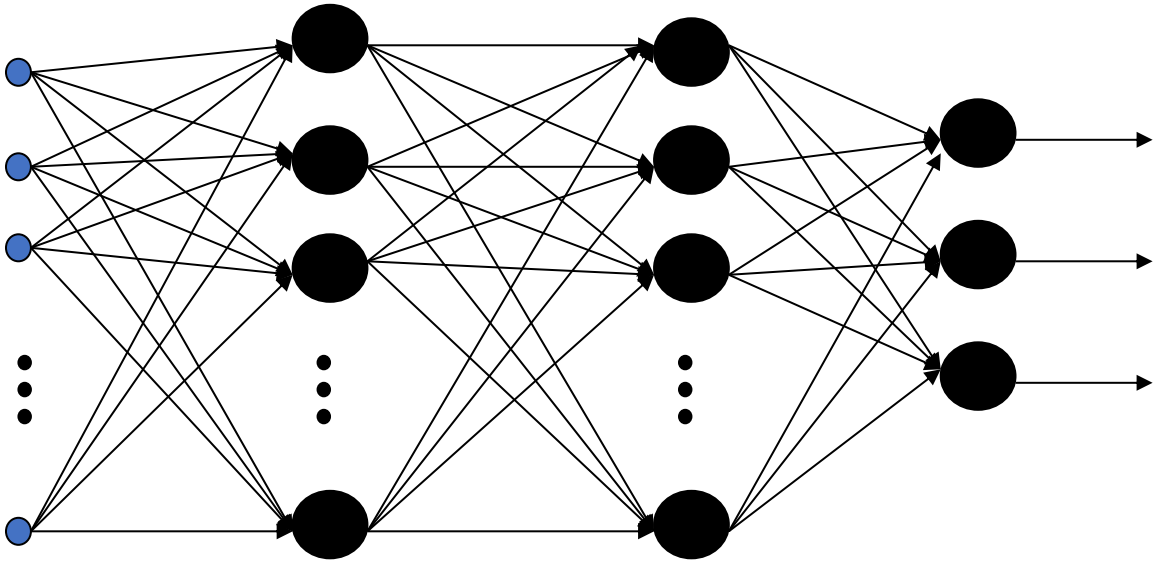




Multiple Inputs & Single Layer



Multiple Inputs and layers

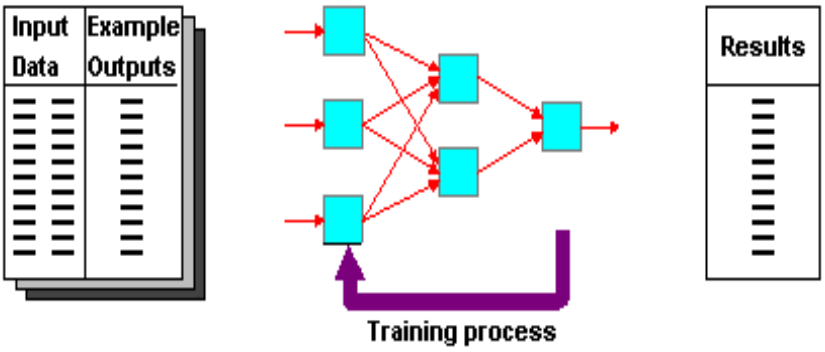


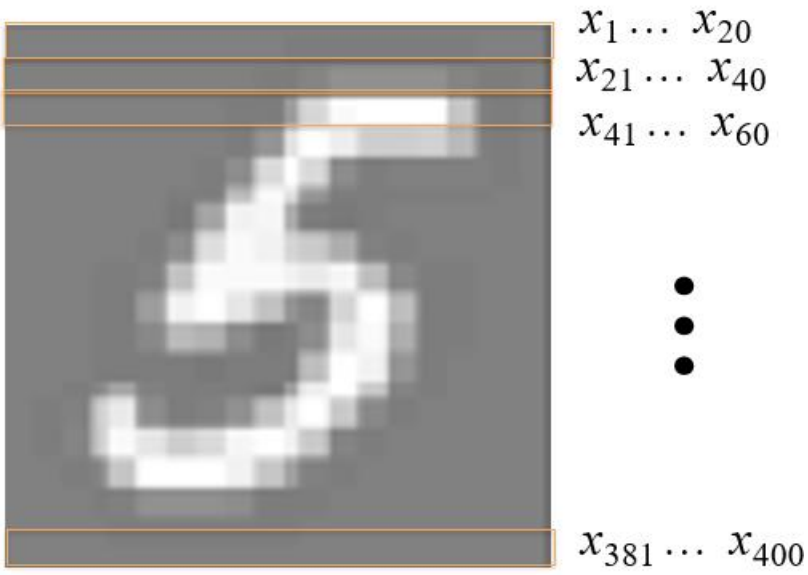
Inputs

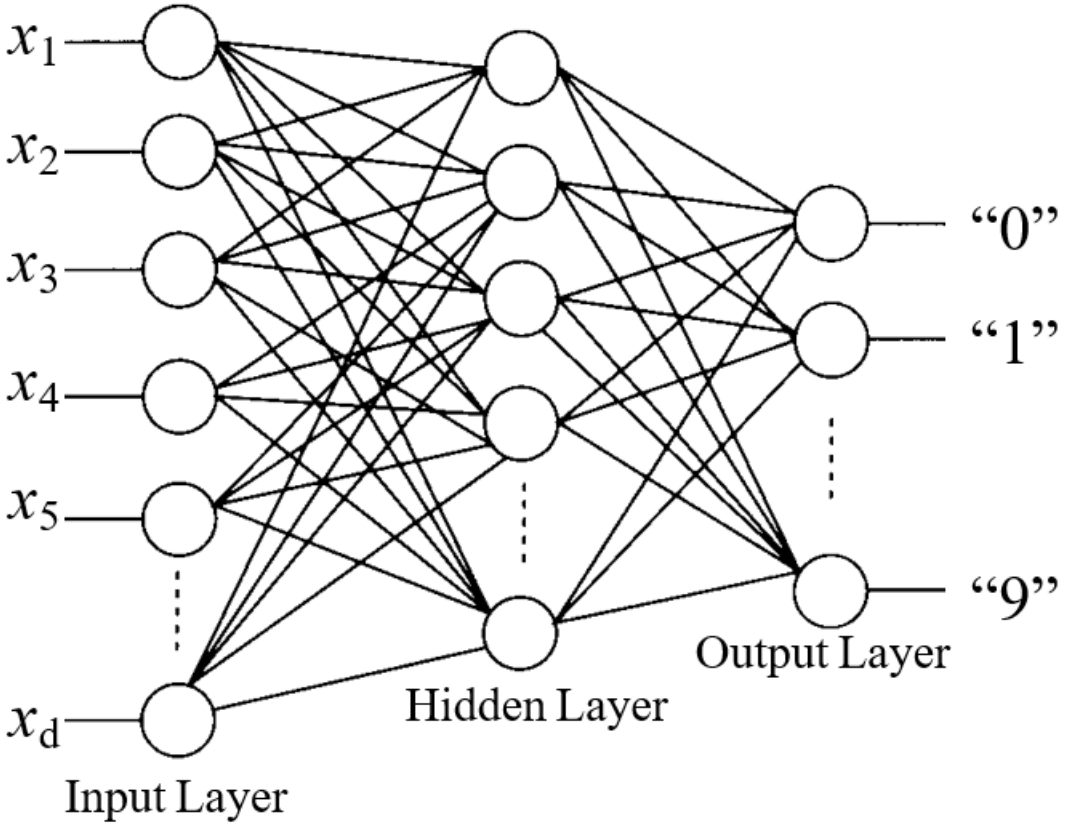
First Hidden

Second Hidden

Output




$x_1 \dots x_{20}$
$x_{21} \dots x_{40}$
$x_{41} \dots x_{60}$
$x_{381} \dots x_{400}$


x_1
x_2
x_3
x_4
x_5
x_d
"0"
"1"
"9"
Input Layer
Hidden Layer
Output Layer

When a neural network is used, the inputs are processed by the neurons using certain weights to yield the output. This is like a signal propagating through the network. When training the network, an error signal is generated when the inputs are `propagated' through to the outputs (usually the difference between outputs and the expected known values). Now the errors are used to change the weights, that is, the errors are processed to generate a change in the weights, also called an update. It's like the errors are propagating backwards through the network to yield a better set of weights that would match the inputs to the outputs, at least on the training data. That's a crude way to understand `back propagation'. The back propagation step yields new weights for the neurons, through a process of optimization. Back-propagation is nothing but similar to how humans learn from their mistakes.