LECTURE 4: The hebbian learning rule

Last week we looked at a number of different neuron models. We saw how a simple McCulloch Pitts neuron can "learn" to classify patterns using the perceptron learning rule.
In neural networks, learning is achieved mostly (but not exclusively) through changes in the strengths of the connections between neurons.
Mechanisms of learning include:

- changes in neural parameters (threshold, time constants)
- creation of new synapses
- elimination of synapses
- changes in the synaptic weights or connection strengths

Hebbian learning rule?
One common way to calculate changes in connection strengths in a neural network is the so called "hebbian learning rule", in which a change in the strength of a connection is a function of the pre – and postsynaptic neural activities. It is called the "hebbian learning rule" after D. Hebb ("When neuron A repeatedly participates in firing neuron B, the strength of the action of A onto B increases".; see Syllabus for the reference).

If xj is the output of the presynaptic neuron, xi the output of the postsynaptic neuron, and wij the strength of the connection between them, and γ learning rate, the one form of a learning rule would be:

$$\Delta W_{ij}(t) = \gamma * x_j * x_i$$

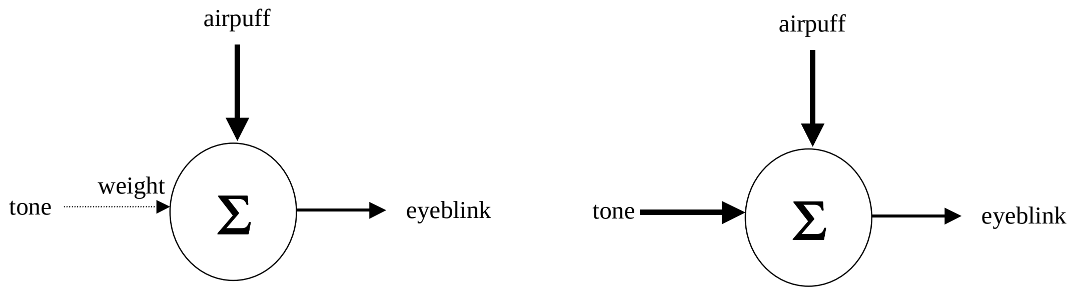
A more general form of a hebbian learning rule would be:

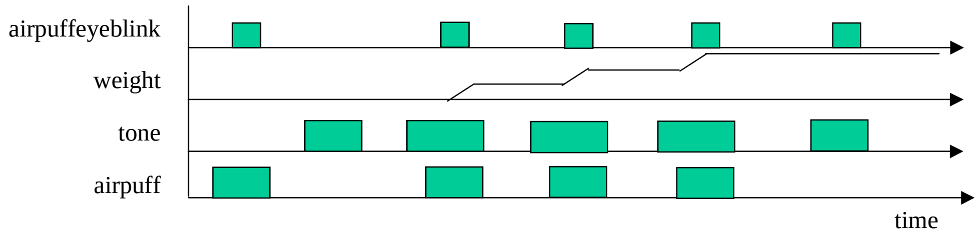
$$\Delta W_{ij}(t) = F(x_j, x_i, \gamma, t, \theta)$$

in which time and learning thresholds can be taken into account.

Time
We often say that the connection strength increases when the pre- and postsynaptic neurons are active "simultaneously". Simultaneous is very relative depending on the system under consideration! For example, when synaptic plasticity is induced in a brain slice preparation, simultaneous can be as short as several ms. However, when an animal learns an association between a food taste and sickness, simultaneous can be as long as several hours!

Example1
Classical conditioning can be modeled with a hebbian synapse. Consider an unconditioned stimulus (an air puff), an unconditioned response (eye blink), a conditioned stimulus (a tone) and a conditioned response (eye blink). Under normal conditions, an animal responds to an air puff with an eye blink. It does not respond to a tone with an eye blink. If the tone is paired with the air puff several times, then the animal acquires an association between the airpuff and the tone, and will now respond to the tone alone with an eye blink.
Consider a "black box" neural approach where one neuron receives input from the airpuff and from the tone. The neurons output represents the eye blink. The neuron is wired in such a way that at first, the airpuff but not the tone activates the neuron and produces an output. If we apply the airpuff input and the tone input together several times, then the neuron is active while the tone-input is active, and a hebbian learning rule will reinforce the strength of the connection between the tone and the neuron. This will lead to the fact that after a few trials, the tone alone will be able to activate the neuron. Neurons of this type can be recorded in the amygdala.





<u>Example 2</u>

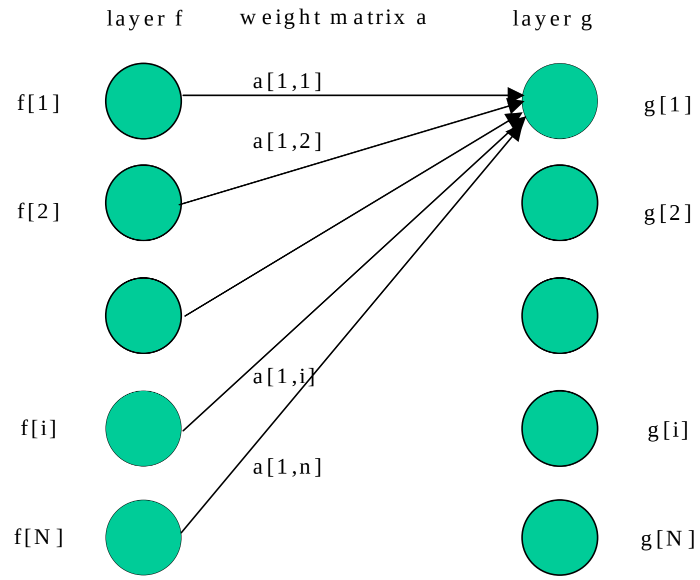
In brain slice preparations of hippocampus and cortex, changes in synaptic strength can be evoked by simultaneous depolarization of a postsynaptic neuron and activation of a presynaptic action potential. Long term potentation in brain slices can be evoked by a number of protocols including rapid stimulation (100 Hz) of the presynaptic fibers and simultaneous depolarization and firing of the presynaptic fibers. See readings in Anderson for details!

<u>Example 3</u>

Honeybees can be conditioned to respond to an odor stimulus by extending their proboscis (proboscis extension paradigm). In this paradigm the unconditioned stimulus is sucrose, the unconditioned response the proboscis extension and the conditioned stimulus the odor. A specific neuron in the honeybee brain, called VUMmx1 arborizes all over the bee brain . Under normal conditions, it responds to the presentation of sucrose to the proboscis by increased firing rates, but it does not respond to odors. However, after an odor has been presented simultaneously with the sucrose reward for several trials, the neuron responds to the odor alone.

Application of the hebbian learning rule: the linear associator

Take a network of neurons which are organized in two distinct layers (called f and g). Neurons in layer f project to neurons in layer g but not vice versa. Each neuron in layer g is a linear unit, its output is the sum of its inputs (hence the name linear associator). In vector notations, this is equivalent to writing: g = a*f (see readings in Anderson).
The strength of the connection form presynaptic neuron f[j] to postsynpatic neuron g[i] is given by a[i,j]. This notation may seem strange (w[i,j] denoting the connection going from neuron j to neuron i but is the most commonly used). The activation of each neuron in the output layer is given by is sum of weighted inputs. The strength of each connection is calculated from the product of the pre- and postsynaptic activities scaled by a "learning rate" γ (which determines how fast connection weights change).
$\Delta w_{ij} = \gamma * g[i] * f[j]$. in vector notations, this is equivalent to writing: $\Delta w = \gamma * g * f^T$. We will calculate an example in class!



g[i] = a[1,1]*f[1] + a[1,2]*f[2] ... etc

The linear associator stores associations between a pattern of neural activations in the input layer f and a pattern of activations in the output layer g. Once the associations have been stored in the connection weights between layer f and layer g, the pattern in layer g can be “recalled” by presentation of the input pattern in layer f.

An auto-associator stores association within a layer of neurons. Once the associations between the neural activities in a given pattern are stored in the connection weights, the auto-associator can recall the stored pattern from a noisy or incomplete input pattern.