Synaptic Learning Rules

Computational Models of
Neural Systems
Lecture 4.1

David S. Touretzky
November, 2023

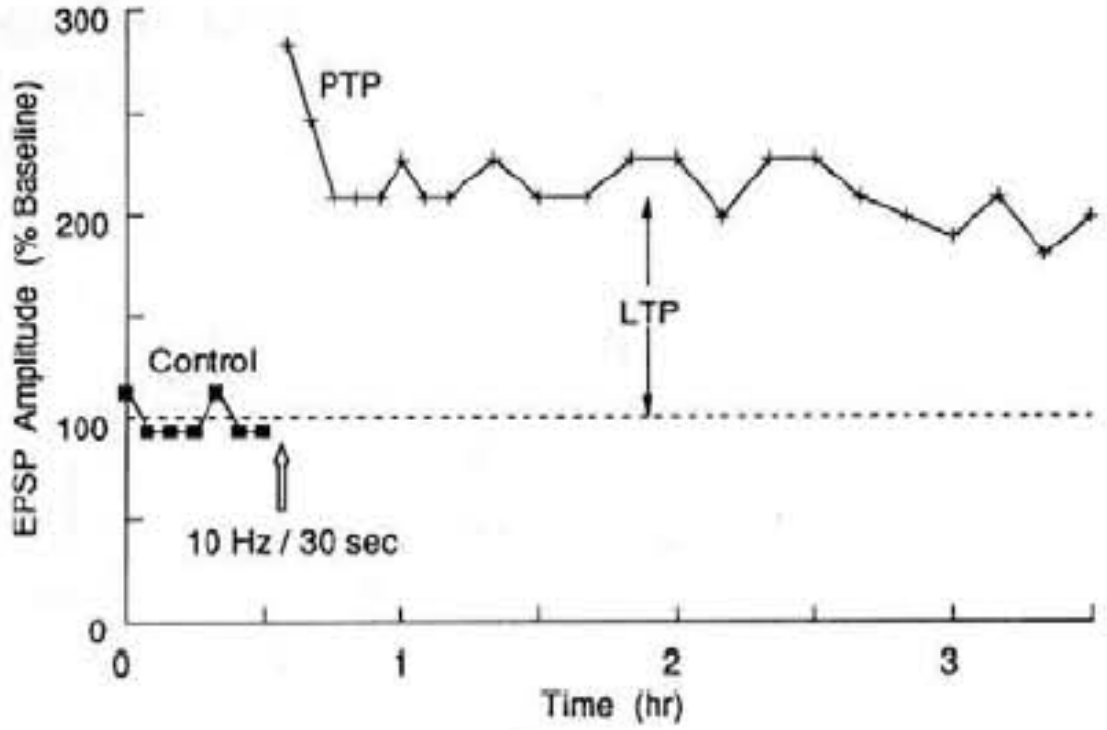
Why Study Synaptic Plasticity?

- Synaptic learning rules determine the information processing capabilities of neurons.
- Synaptic learning rules can implement mechanisms like gain control.
- Simple learning rules can even extract information from a noisy dataset, via a technique called *Principal Components Analysis.*

Terms

- LTP: Long Term Potentiation
  - A synapse increases in strength, above its baseline value.
- LTD: Long Term Depression
  - A synapse decreases in strength, below its baseline value.
- PTP: Post-Tetanic Potentiation
- STP: Short-Term Potentiation

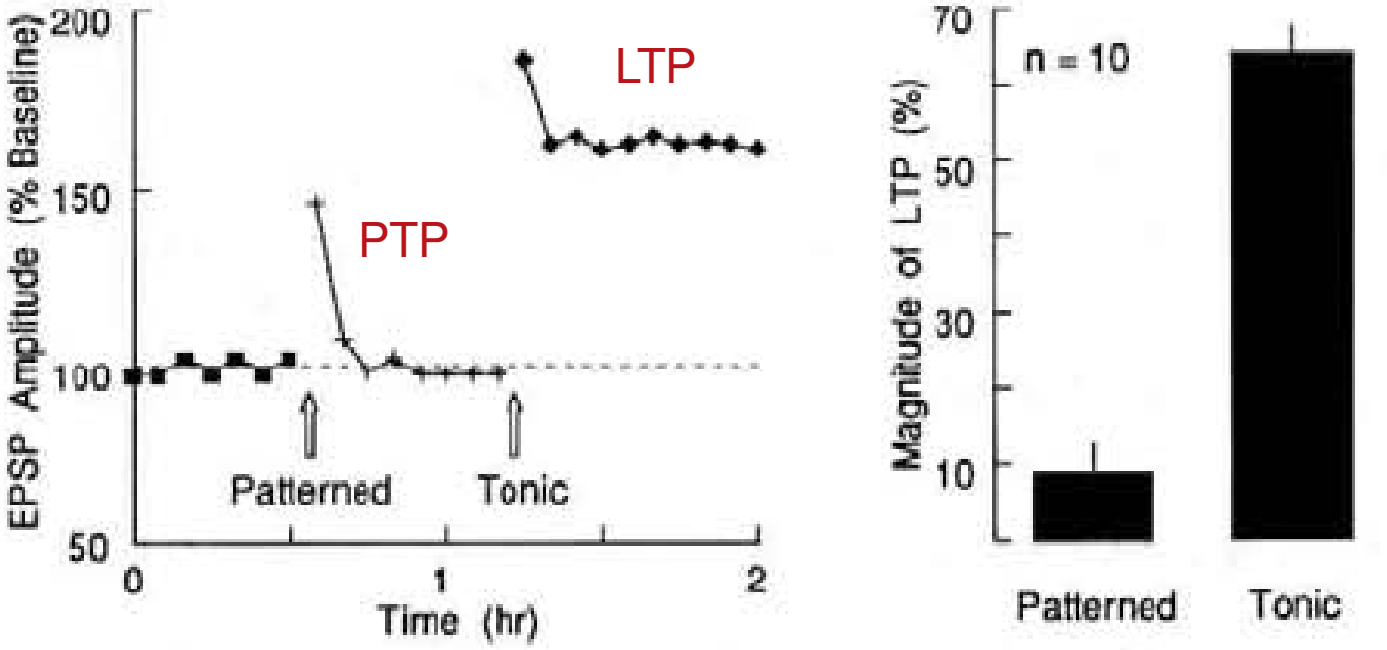
PTP vs. LTP



Baxter & Byrne (1993)

Optimal Stimulus Pattern for LTP

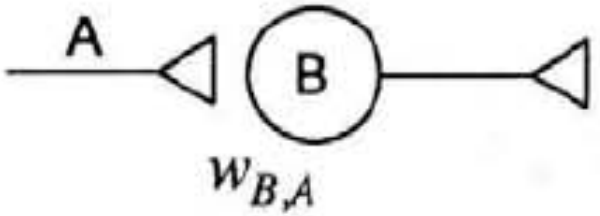
- Tonic stimulus: 30 secs @ 10 Hz = 300 spikes.
- Patterned stimulus: 30 secs of evenly spaced 2-5 spike 100 Hz bursts, for a total of 300 spikes.



Types of Synaptic Modification Rules

- Non-associative vs. Associative
  - Non-associative: based on activity of a single cell: either presynaptic or postsynaptic
  - Associative: based on correlated activity between cells
- Homosynaptic (action at the same synapse) vs. Heterosynaptic (activity at one synapse affects another)
- Potentiation vs. Depression

Non-Associative Homosynaptic Rules



presynaptic change

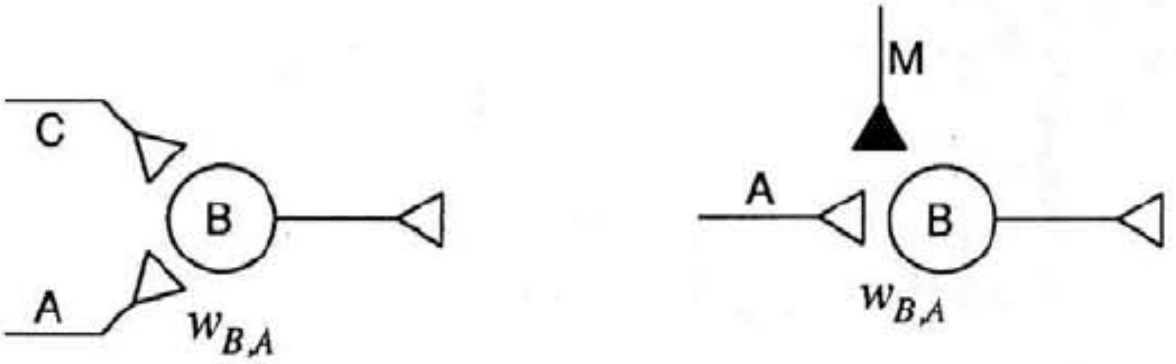
postsynaptic change

activation

A	B	Rule 1a	Rule 1b	Rule 2a	Rule 2b
+		↑	↓		
	+			↑	↓

What biophysical mechanisms could cause these changes in strength?

Non-Associative **Hetero**synaptic Rules

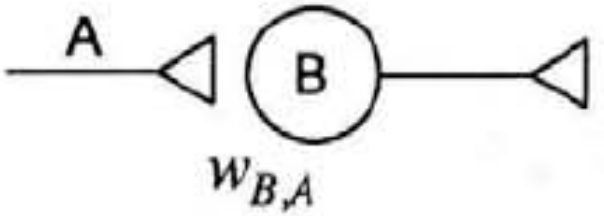


C	M	Rule 3a	Rule 3b	Rule 4a	Rule 4b
+		↑	↓		
	+			↑	↓

Modification of the A→B synapse depends on activity in presynaptic neuron C or modulatory neuron M.

Homosynaptic Presynaptic Potentiation

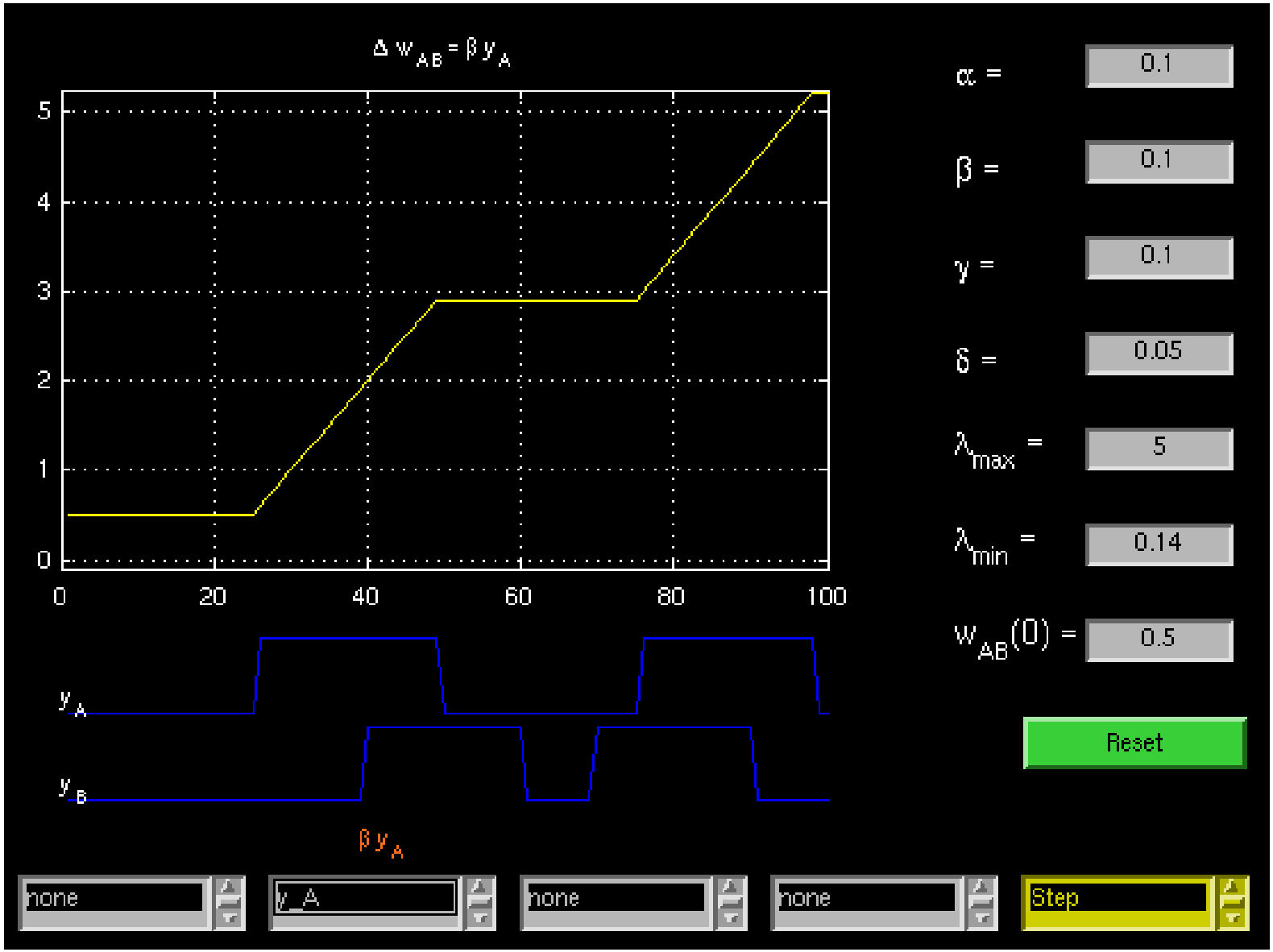
$$\Delta w_{B,A}(t) = \epsilon \cdot y_A(t)$$



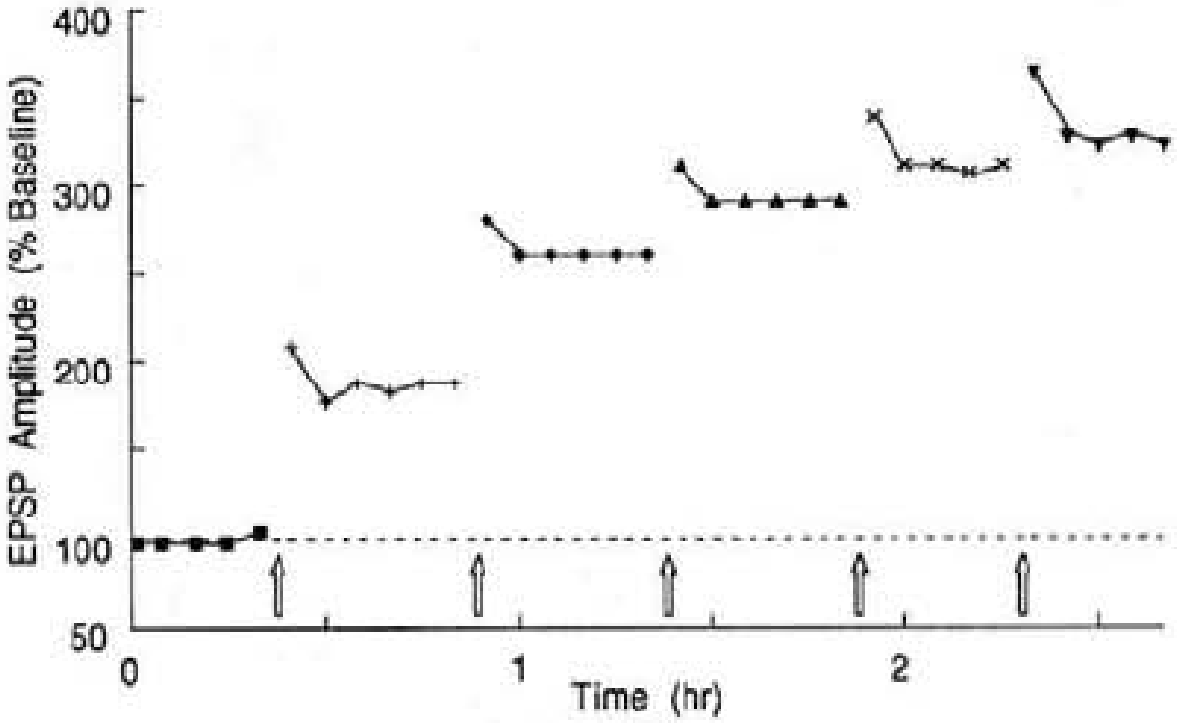
- $y_A(t)$ is the firing frequency of the presynaptic cell, i.e., spike activity averaged over a few seconds.
- This rule may apply to mossy fiber synapses in hippocampus.
- But this rule causes $w_{B,A}$ to grow without bound.
  - In real cells, the weight approaches an upper limit.

Matlab Learning Rule Simulator

- Find it in the matlab/ltp directory.



Saturation of LTP



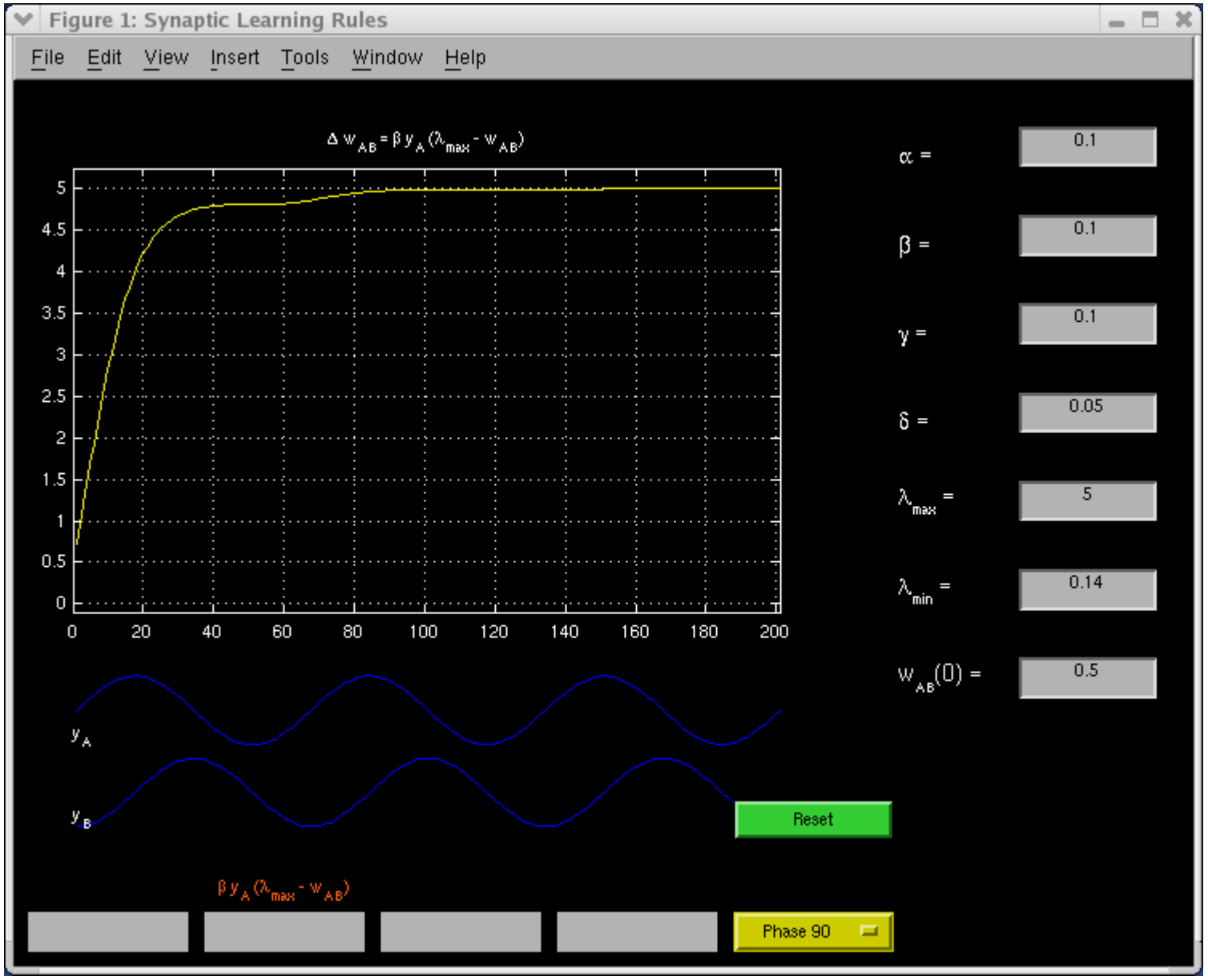
Baxter & Byrne (1993)

Homosynaptic Presynaptic Potentiation with Asymptote

$$\Delta w_{B,A}(t) = \epsilon \cdot y_A(t) \cdot \left(\lambda_{max} - w_{B,A}(t)\right)$$

- λ_{max} is the asymptotic strength.
- The weights are now bounded from above by λ_{max}
- But the weights can never decrease, so they will saturate.
- Still a very abstract model.
- λ_{max} < 6 to 10 times w_0.

Presynaptic Potentiation with Asymptote



Homosynaptic **Pre**synaptic Depression

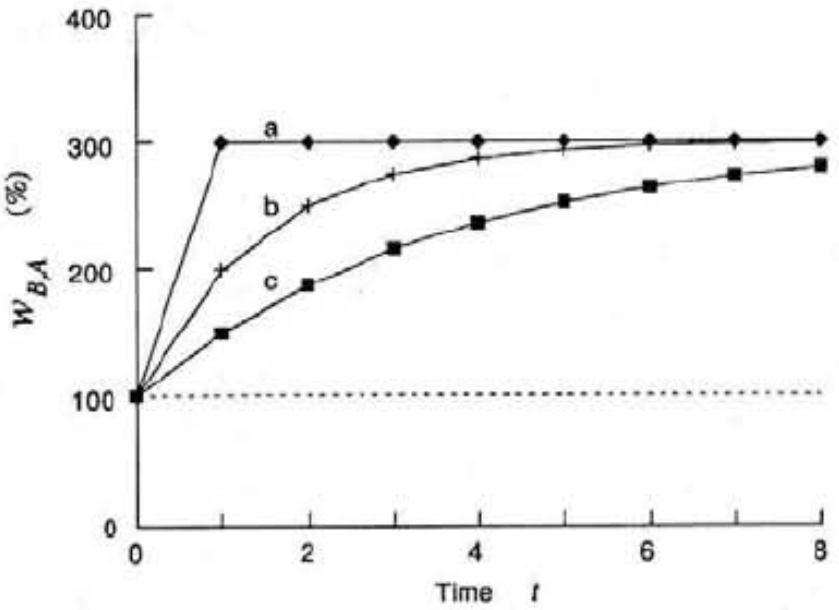
- By analogy with potentiation, but use the inverse of activity, so that low frequency stimulation (0.1 Hz) produces more depression than high frequency (> 1 Hz).

$$\Delta w_{B,A}(t) = \left(\epsilon \cdot y_A(t)\right)^{-1} \cdot \left(\lambda_{min} - w_{B,A}(t)\right)$$

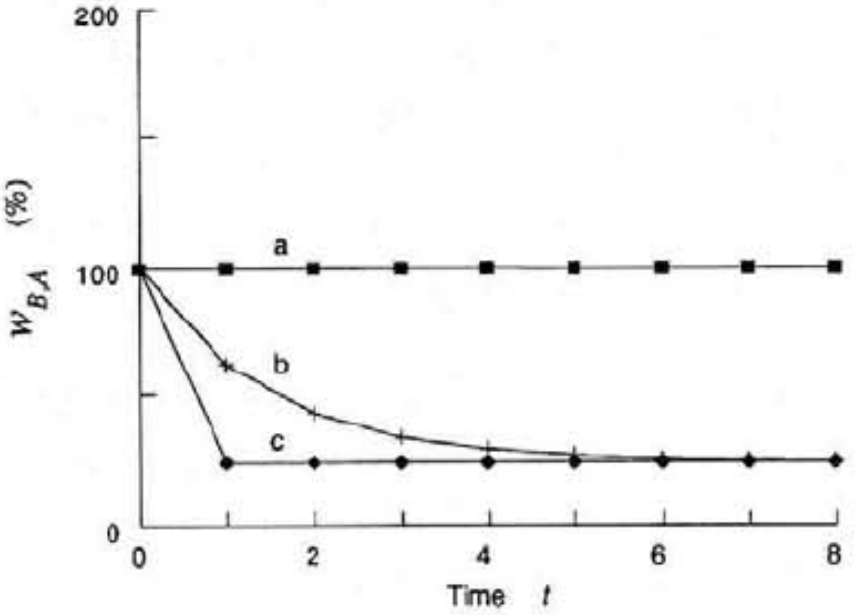
- Larger y_A means less weight change.
- ε is positive; asymptote term is negative.

Effects of Stimulus Strength

a = 100, b = 50, c = 25



A stronger stimulus potentiates more quickly.



A weaker stimulus depresses more quickly.

Homosynaptic **Post**synaptic Modification

- Depends on activity of the postsynaptic cell, $y_B(t)$

$$\Delta w_{B,A}(t) = \epsilon \cdot y_B(t) \cdot \left(\lambda_{max} - w_{B,A}(t)\right)$$

$$\Delta w_{B,A}(t) = \left(\epsilon \cdot y_B(t)\right)^{-1} \cdot \left(\lambda_{min} - w_{B,A}(t)\right)$$

- λ_{max} is around 3 times the initial weight w_0.
- For depression, λ_{min} is around 0.14 times w_0.

Non-Associative Heterosynaptic Rules

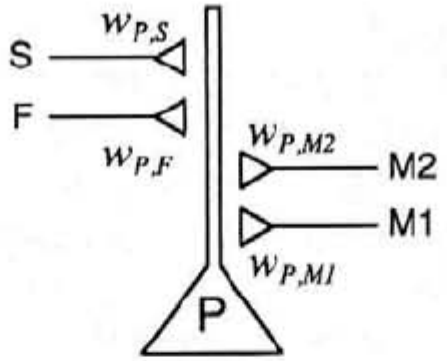
- Weight change occurs when a third neuron C fires.

$$\Delta w_{B,A}(t) = F\left(y_C(t)\right)$$

- Exact formula by analogy again.

- There are also modulatory neurons that can affect synapses by secreting neurotransmitter onto them.

Several Types of Non-Associative Learning Are Observed in Hippocampus CA3 or CA1



M1	S	F	$w_{P,M1}$	$w_{P,M2}$	$w_{P,S}$	$w_{P,F}$
+			Homosynaptic Potentiation	Heterosynaptic Depression	Heterosynaptic Potentiation	Heterosynaptic Potentiation
	+		Heterosynaptic Depression		Homosynaptic Potentiation	
		+	Heterosynaptic Depression		Heterosynaptic Depression	Homosynaptic Potentiation

M1, M2	mossy fiber pathways	F	fimbrial pathway
S	Schaffer collateral / commissual pathways	w	synaptic weight
P	CA3 pyramidal neuron	+	tetanized pathway

Associative Learning Rules

- Basic Hebb rule
- Anti-Hebbian rule
- Bilinear Hebb rule
- Asymptotic Hebb rule
- Temporal specificity
- Covariance rule
- BCM (Bienenstock, Cooper, and Munro) rule

Hebbian Learning

"When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, in increased."

-- D. O. Hebb, 1949

$$\Delta w_{B,A}(t) = F\left(y_A(t), y_B(t)\right)$$

- Purely local learning rule (good).
- Weights can grow without bound (bad).
- No decrease mechanism is mentioned (bad).

Basic Hebbian and Anti-Hebbian Rules

- Basic Hebbian rule produces monotonically increasing weights with no upper limit:

$$\Delta w_{B,A}(t) = \epsilon \cdot y_A(t) \cdot y_B(t)$$

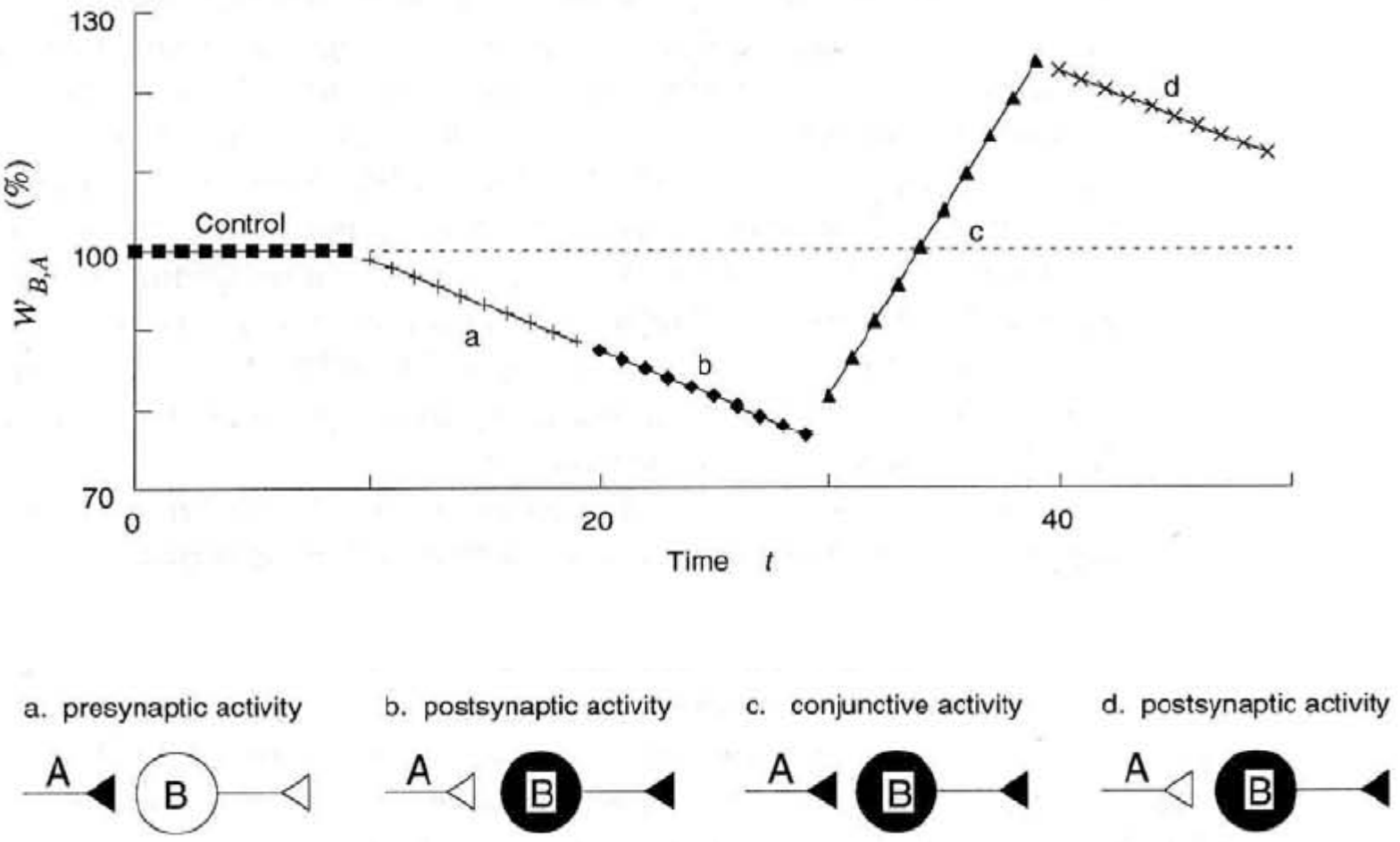
- Anti-Hebbian rule uses $\varepsilon < 0$. Also called “inverse Hebbian” or “reverse Hebbian”.
  - If the presynaptic and postynaptic neurons fire together, decrease the weight.

Bilinear Hebb Rule

$$\Delta w_{B,A}(t) = \epsilon \cdot y_A(t) \cdot y_B(t) - \beta \cdot y_A(t) - \gamma \cdot y_B(t) - \delta$$

- Increase based on product of activity.
- Linear decrease if either neuron fires.
- General decay term δ should probably be $\delta \cdot w_{B,A}$ for asymptotic decay.
- ε must be large enough to outweigh β and γ for this to work.

Simulation of Bilinear Rule

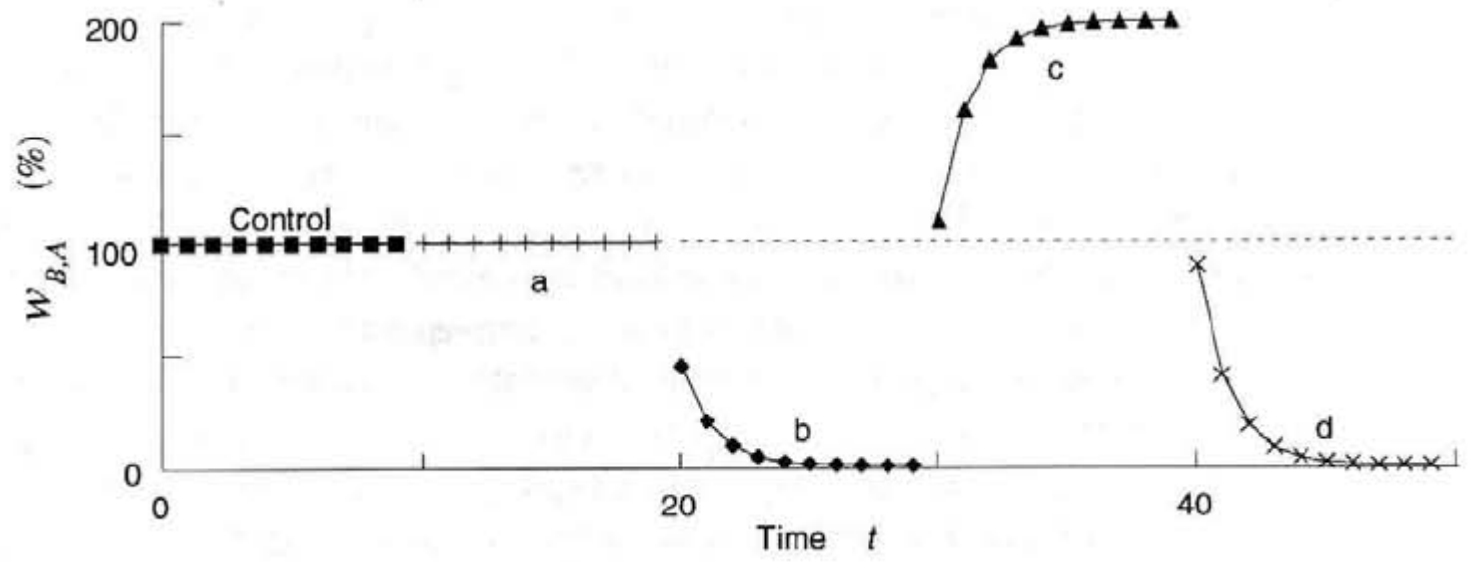


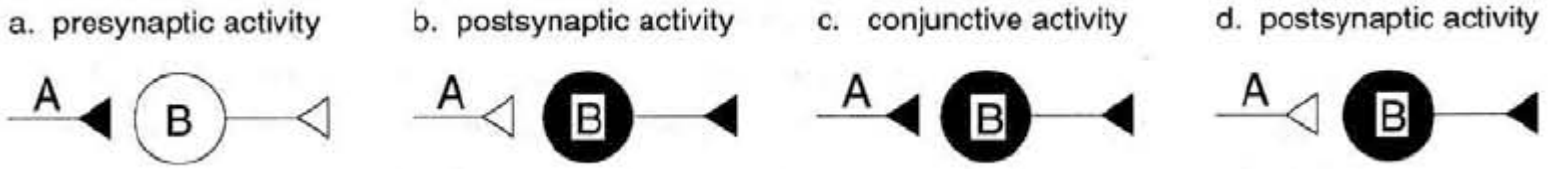
Asymptotic Hebb Rule

$$\Delta w_{B,A}(t) = \epsilon \cdot G(y_B(t)) \cdot \left(c \cdot y_A(t) - w_{B,A}(t) \right)$$

- Allows weight increases and decreases, like bilinear rule.
- Incorporates an asymptotic limit.
- If y_B is 0 there is no weight change.
- If neuron B fires, then neuron A's state determines the weight change.

Hebbian Rule with Asymptotic Limits On Both Potentiation and Depression



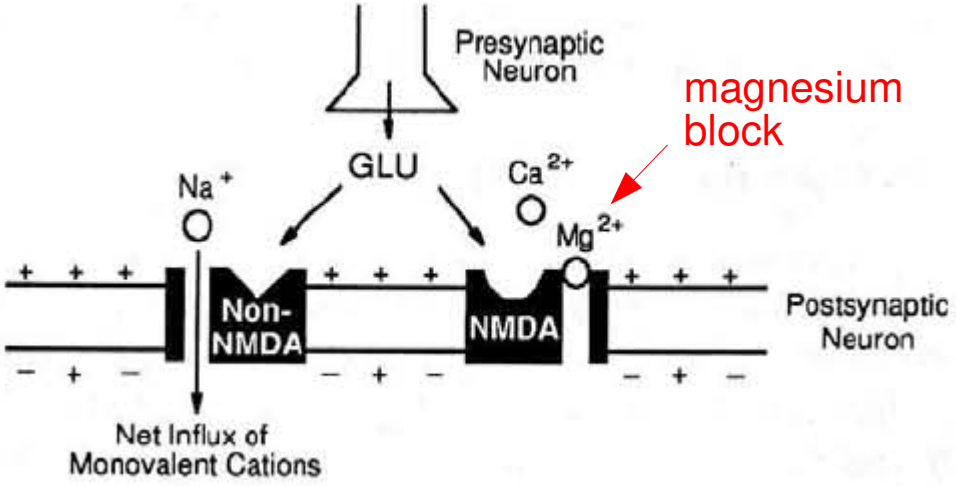


Temporal Specificity

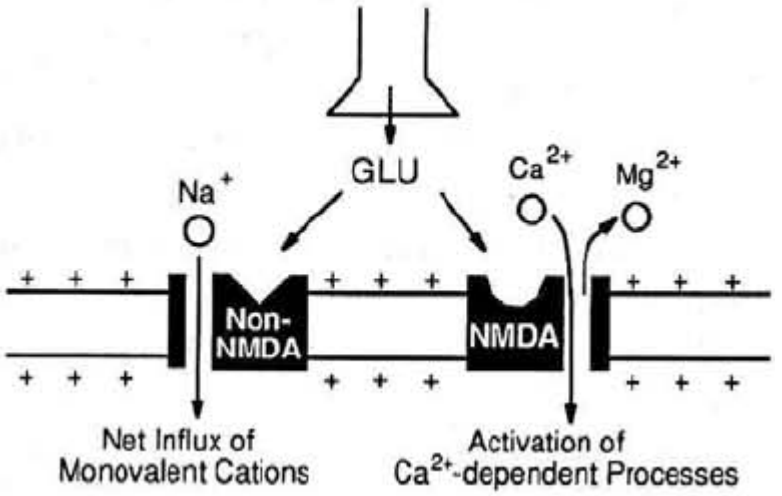
- Hebb's formulation refers to neuron A causing neuron B to fire. Can't measure causality directly.
- Instead, look for correlated activity.
- Traces of a presynaptic spike will linger for a short while after the spike has passed.
- Can use this to detect correlation:
  - *k* is how far back to look
  - F(*t*-τ,*x*) is a weighting function based on age of the spike (*t*-τ)

$$\Delta w_{B,A}(t) = \epsilon \underbrace{\sum_{\tau=0}^{k} F\left(\tau, y_A(t-\tau)\right)}_{\text{Memory trace}} \cdot G\left(y_B(t)\right)$$

The NMDA Receptor Detects Correlated Activity



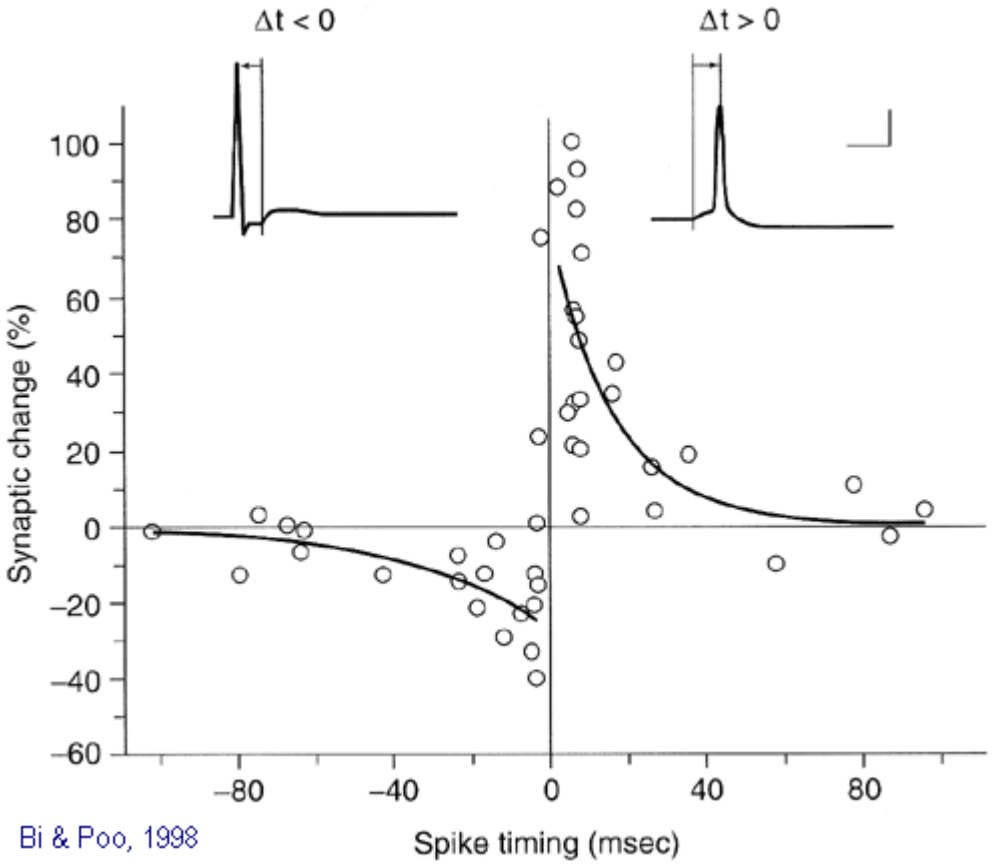
Small postsynaptic depolarization: no Ca^{2+} influx due to Mg^{2+} block



Large postsynaptic depolarization brings Ca^{2+} influx

Spike-Timing Dependent Plasticity

- Weight increase vs. decrease depends on relative timing of pre- and post-synaptic activity.



Hebbian Covariance Learning Rule

- Subtract mean rate from current firing rate of each cell.
- Then use a Hebbian rule to update the weight.
- Weight will increase if pre- and post-synaptic firing are positively correlated (both above or both below their means).
- Will decrease if they are negatively correlated.
- No change if firing is uncorrelated.
- Summary: weight change is proportional to the covariance of the firing rates.

Covariance Learning Rule

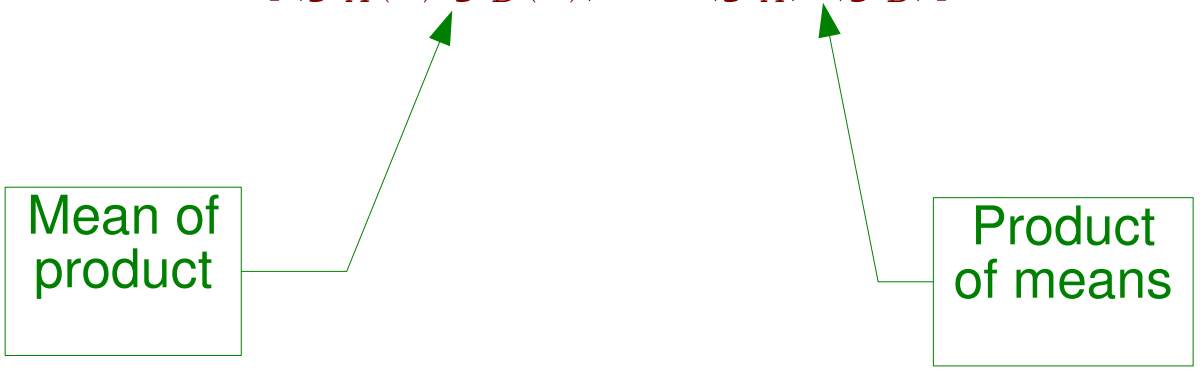
$$\Delta w_{B,A}(t) = \epsilon \cdot \left[y_A(t) - \langle y_A \rangle\right] \cdot \left[y_B(t) - \langle y_B \rangle\right]$$

$$= \epsilon \cdot \left[\, y_A(t) \cdot y_B(t) - \langle y_A \rangle \cdot y_B(t) - y_A(t) \cdot \langle y_B \rangle + \langle y_A \rangle \cdot \langle y_B \rangle \,\right]$$

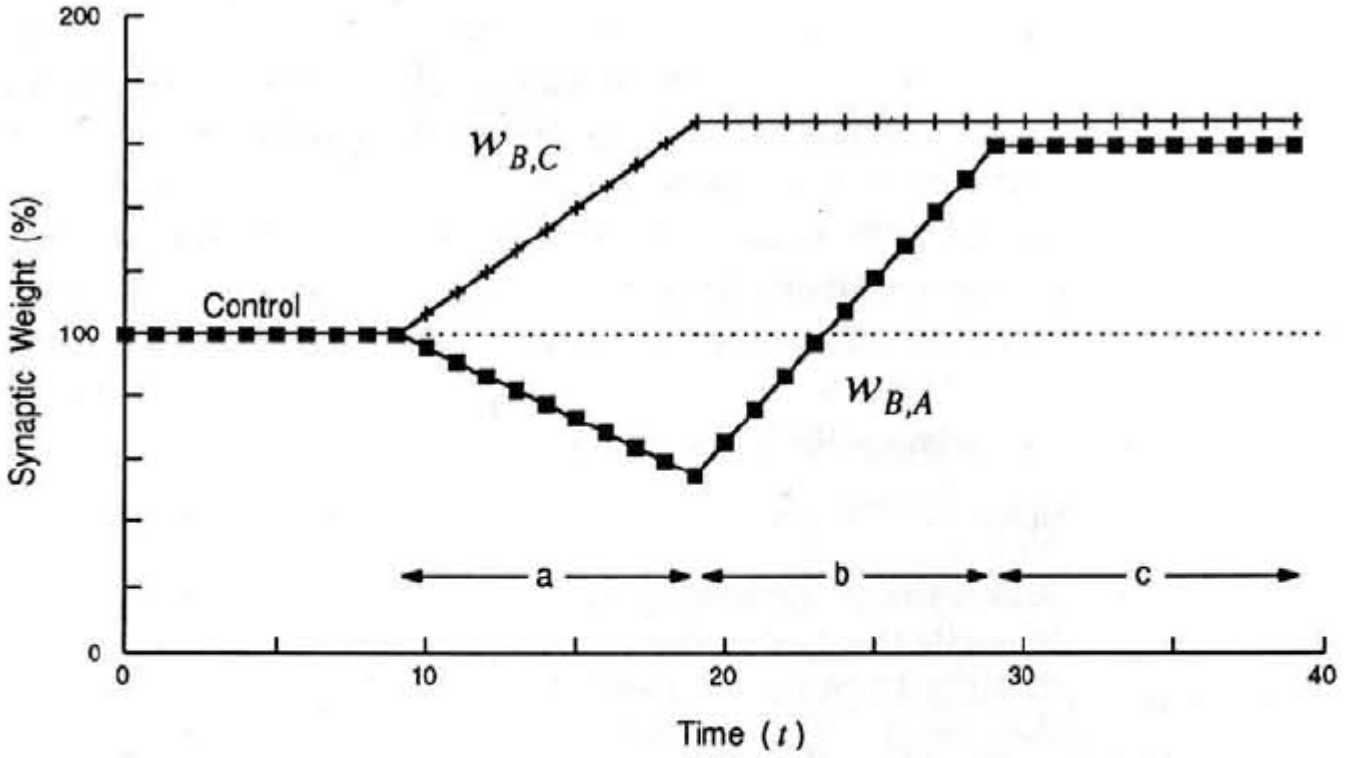
$$\langle \Delta w_{B,A}(t) \rangle = \epsilon \cdot [\langle y_A(t) \cdot y_B(t) \rangle - \langle \langle y_A \rangle \cdot y_B(t) \rangle - \langle y_A(t) \cdot \langle y_B \rangle \rangle + \langle \langle y_A \rangle \cdot \langle y_B \rangle \rangle]$$

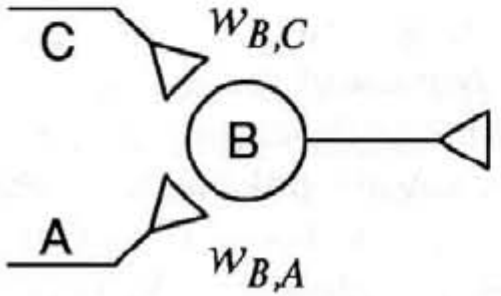
$$= \epsilon \cdot [\langle y_A(t) \cdot y_B(t) \rangle - \langle y_A \rangle \cdot \langle y_B \rangle - \langle y_A \rangle \cdot \langle y_B \rangle + \langle y_A \rangle \cdot \langle y_B \rangle]$$

$$= \epsilon \cdot [\langle y_A(t) \cdot y_B(t) \rangle - \langle y_A \rangle \cdot \langle y_B \rangle]$$



Simulation of Covariance Rule





a. Cells C and B positively correlated
Cells A and B negatively correlated

b. Cells C and B uncorrelated
Cells A and B positively correlated

c. Cells A, C and B Uncorrelated

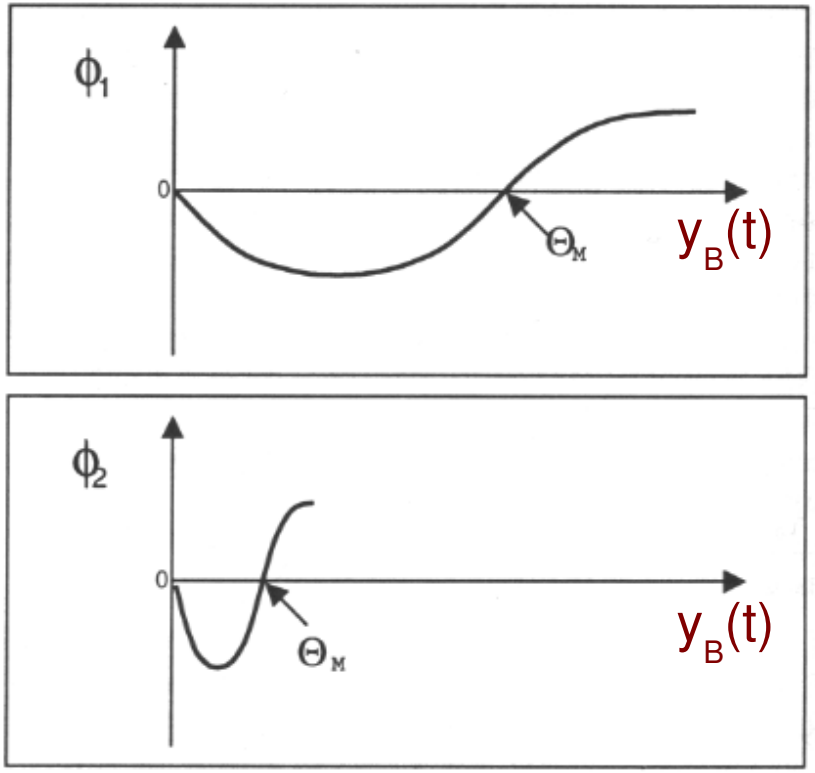
BCM Rule

- Bienenstock, Cooper, and Munro learning rule

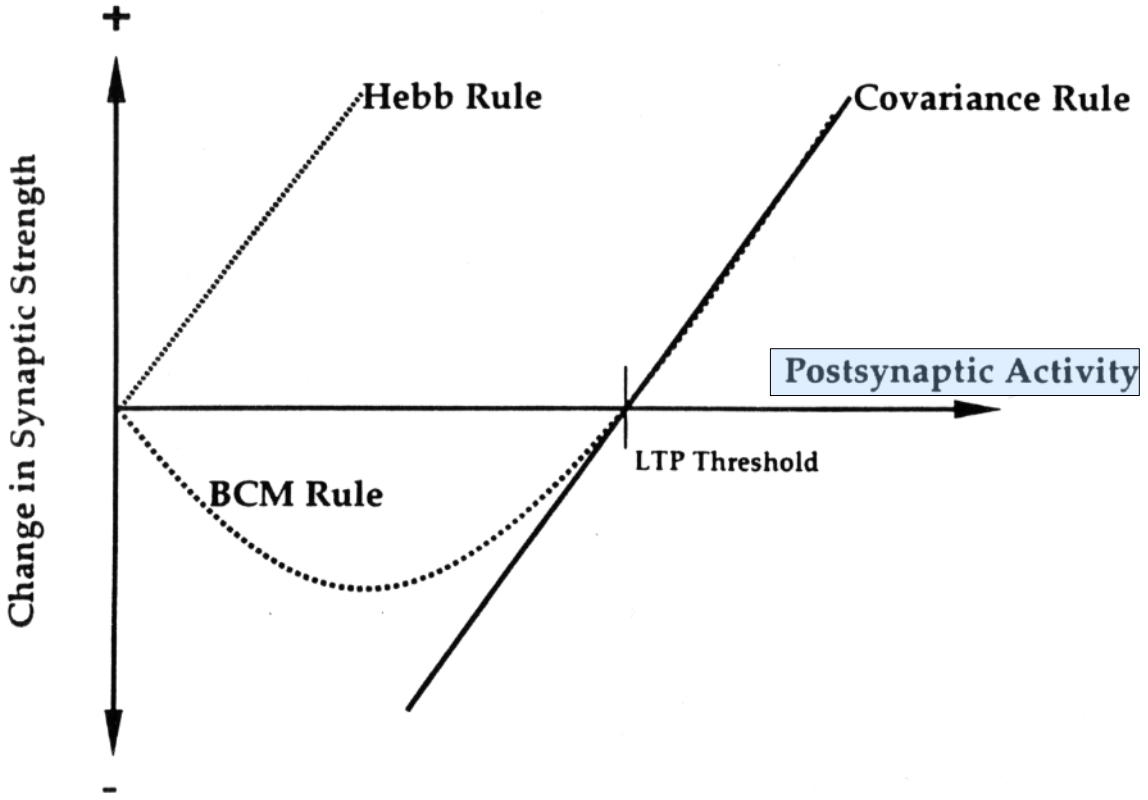
$$\Delta w_{B,A} = \phi\left(y_B(t), \Theta(t)\right) \cdot y_A(t)$$

$$\Theta(t) = \langle y_B^2 \rangle$$

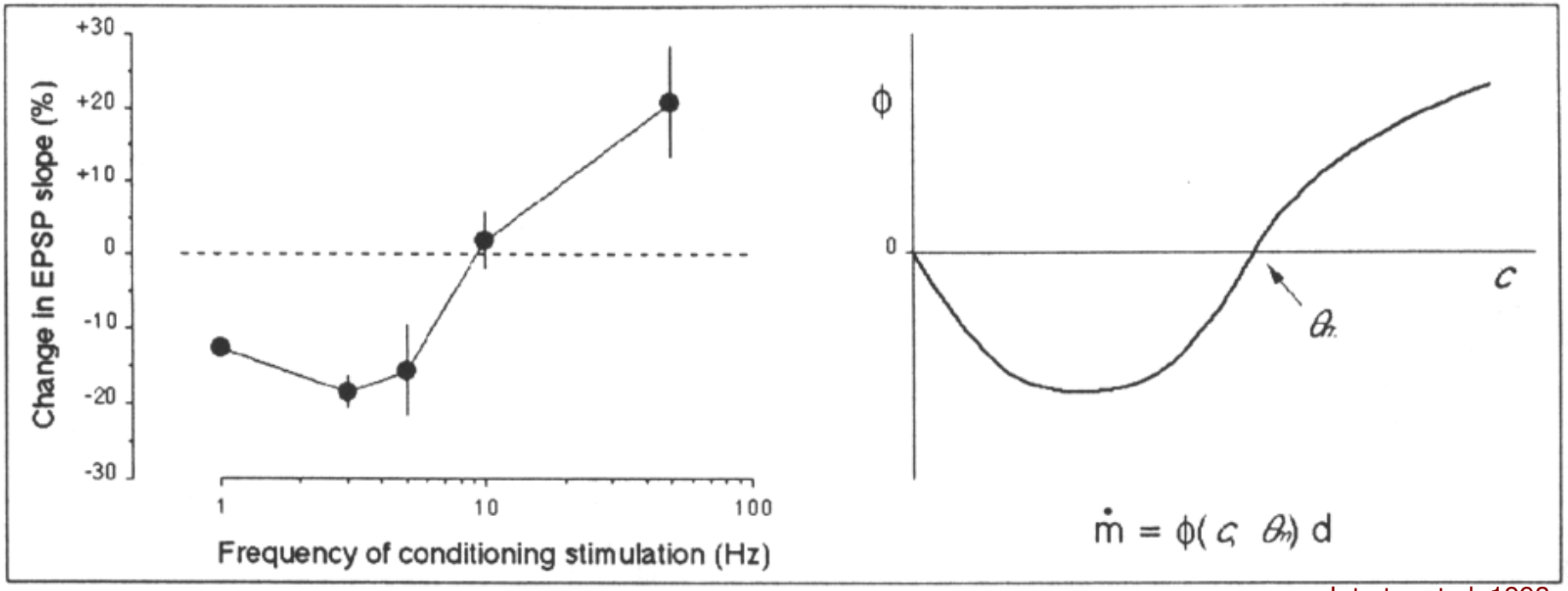
- θ is a variable threshold.
- Similar to covariance rule
- No weight change unless presynaptic cell A fires.



Comparison of BCM and Related Rules, Assuming Fixed Presynaptic Activity



Evidence for BCM Learning in Visual Cortex

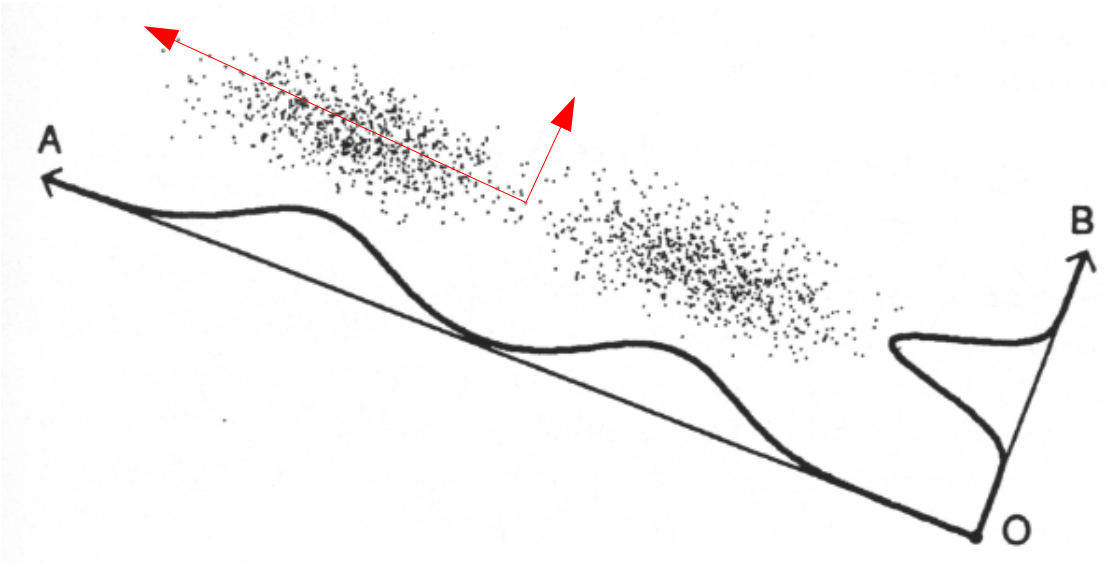


Intrator et al. 1993

- Weight increase/decrease matches BCM rule.
- But does the threshold θ adapt?
  - If so, what is the physiological basis?
  - Might be calcium concentration $[Ca^{2+}]_i$.

Principal Components Analysis

- N-dimensional data has up to N principal components.
- Principal components are mutually orthogonal.
- The first principal component is the direction along which the (zero-meaned) data has the greatest variance.
- The first few components capture the essence of the data, i.e., they provide an efficient encoding.

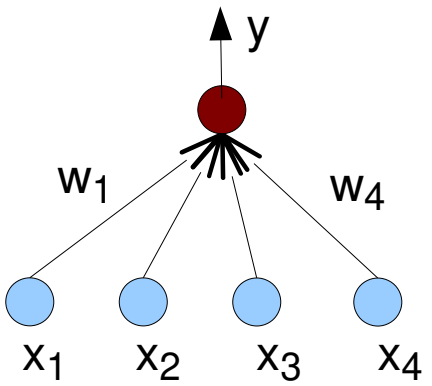


PCA with a Linear Unit

- Assume inputs x_i normalized to have zero mean, so that Hebbian learning is equivalent to a covariance learning rule.
  - Then the variance of x_i is equal to $\langle x_i^2 \rangle$.

$$y = \sum_i w_i x_i$$

$$\Delta w_i = \eta x_i y = \eta \cdot \left(\ldots + w_i x_i^2 + \ldots \right)$$



- Weight grows without bound, but in the direction of the first principal component, i.e., the component with greatest variance $\langle x_i^2 \rangle$.

Oja's Rule

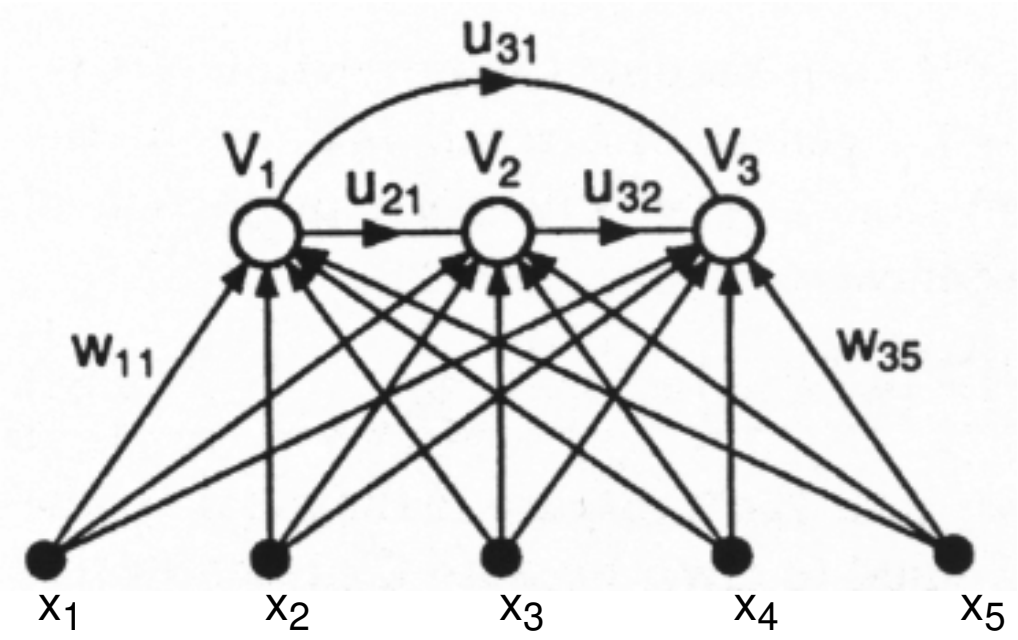
$$\Delta w_{B,A} = \epsilon \cdot y_B(t) \cdot \left(y_A(t) - y_B(t) \cdot w_{B,A}(t)\right)$$

$$= \epsilon \cdot y_b(t) \cdot y_a(t) - \epsilon \cdot y_b^2(t) \cdot w_{B,A}(t)$$

- Weight vector **w** is bounded.
- **w** approaches a unit length vector in the direction of the eigenvector with largest eigenvalue, i.e., the first principal component.

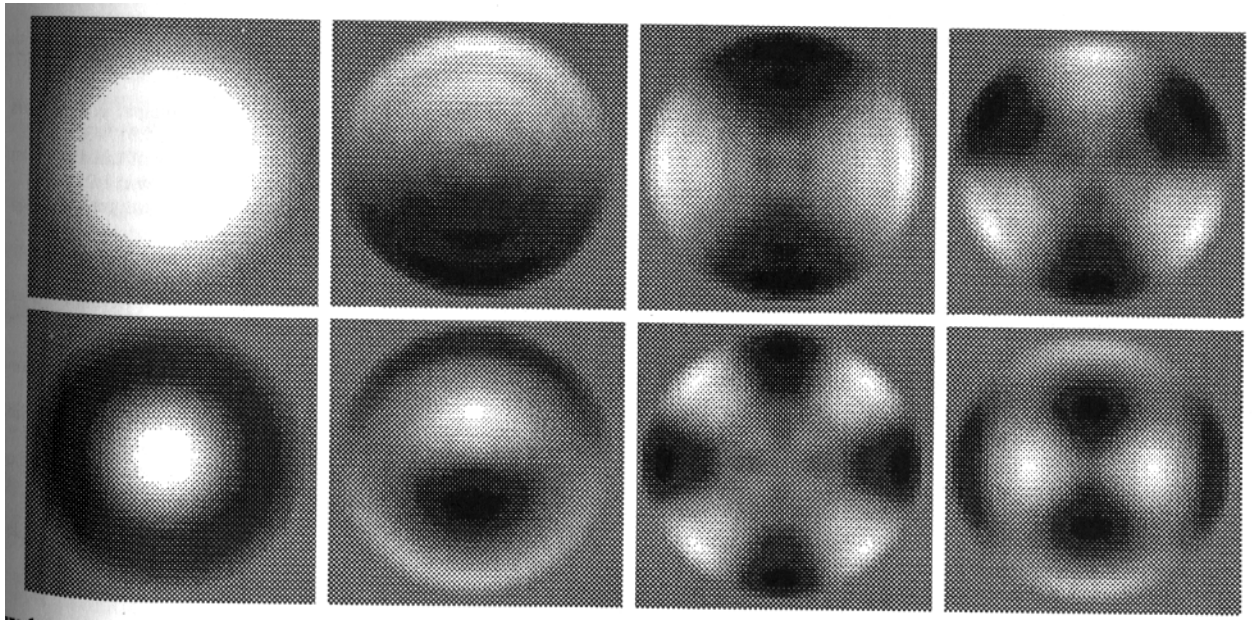
Extracting Multiple Components

- A network of k neurons can be used to extract the first k principal components.
- Use Hebbian learning for the w_i connections.
- Use anti-Hebbian for the u_i connections.



Does the Brain Really Do PCA?

- PCA can train feature detectors that efficiently encode high-dimensional data, such as images.
- But the receptive fields learned by Hebbian covariance neurons don't look like the receptive fields of real neurons.

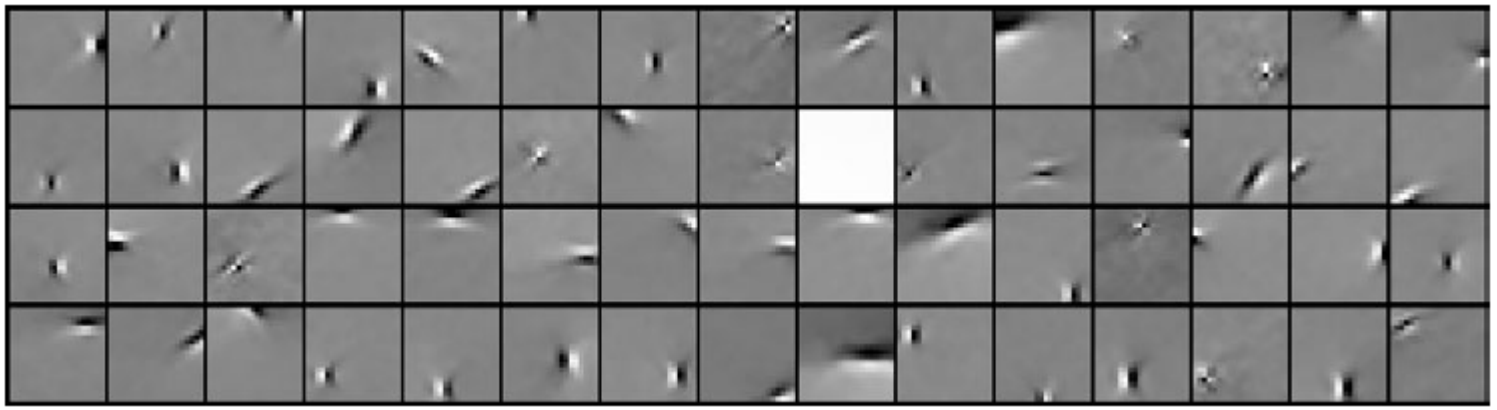


The first 8 principal components extracted from visual data using symmetric connections.

Independent Components Analysis

- A more sophisticated learning algorithm, called Independent Components Analysis, does produce realistic looking receptive fields.

Karklin & Lewicki (2003)



Tries to maximize the variance of each component while minimizing their correlation; they needn't be orthogonal.

- Does the brain do ICA? Possibly.