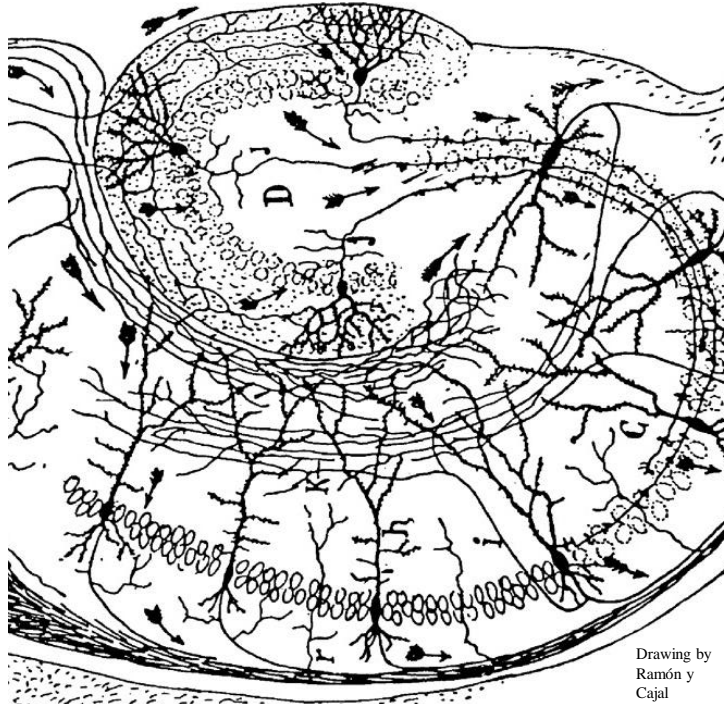
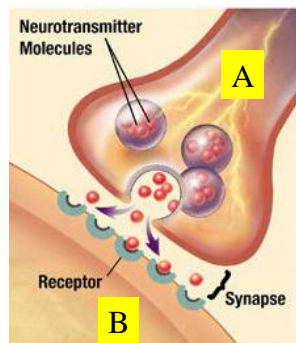
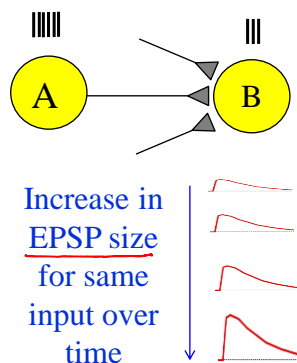


Synaptic Plasticity and Learning



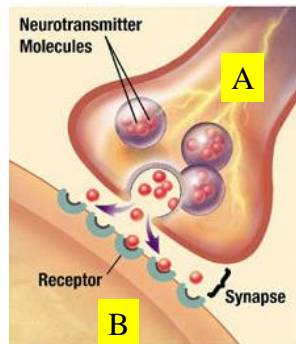
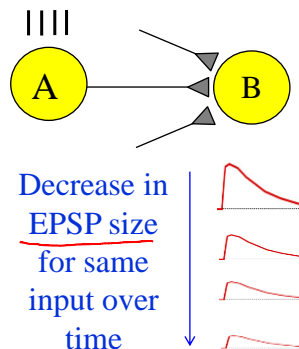
Long Term Potentiation (LTP)

LTP = Experimentally observed increase in synaptic strength that lasts for hours or days



Long Term Depression (LTD)

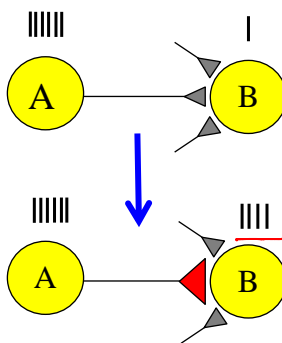
LTD = Experimentally observed decrease in synaptic strength that lasts for hours or days



3

Image Source: Wikimedia Commons

Hebb's Learning Rule



If neuron A repeatedly takes part in firing neuron B, then the synapse from A to B is strengthened



"Neurons that fire together wire together!"

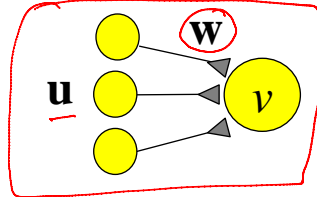
4

Image Source: Wikimedia Commons

Formalizing Hebb's Rule

- Consider a single linear neuron with steady state output:

$$v = \mathbf{w} \cdot \mathbf{u} = \mathbf{w}^T \mathbf{u} = \mathbf{u}^T \mathbf{w}$$



- Basic Hebb Rule: $\tau_w \frac{d\mathbf{w}}{dt} = \underline{\mathbf{u}v}$

Discrete Implementation:

$$\tau_w \frac{\mathbf{w}(t + \Delta t) - \mathbf{w}(t)}{\Delta t} = \mathbf{u}v \quad (\text{or } \mathbf{w}(t + \Delta t) = \mathbf{w}(t) + \frac{\Delta t}{\tau_w} \mathbf{u}v)$$

$$\mathbf{w}_{i+1} = \mathbf{w}_i + \varepsilon \cdot \mathbf{u}v \quad (\text{or } \Delta \mathbf{w} = \varepsilon \cdot \mathbf{u}v)$$

5

What is the average effect of the Hebb rule?

- Hebb Rule: $\tau_w \frac{d\mathbf{w}}{dt} = \mathbf{u}v$

- Average effect of the rule:

$$\tau_w \frac{d\mathbf{w}}{dt} = \langle \underline{\mathbf{u}v} \rangle_{\mathbf{u}} = \langle \mathbf{u} \mathbf{u}^T \mathbf{w} \rangle_{\mathbf{u}} = \langle \mathbf{u} \mathbf{u}^T \rangle_{\mathbf{u}} \mathbf{w} = \underline{Q} \mathbf{w}$$

- Q is the input correlation matrix: $\underline{Q} = \langle \mathbf{u} \mathbf{u}^T \rangle_{\mathbf{u}}$

6

Covariance Rule

- Hebb rule only increases synaptic weights (LTP)

⇒ What about LTD?

- Covariance rule:

$$v > \langle v \rangle \Rightarrow \text{LTP}$$

$$v < \langle v \rangle \Rightarrow \text{LTD}$$

$$\tau_w \frac{d\mathbf{w}}{dt} = \mathbf{u}(v - \langle v \rangle)$$

(Note: LTD for low or no output given some input)

- Average effect of the rule:

$$\begin{aligned} \tau_w \frac{d\mathbf{w}}{dt} &= \langle \mathbf{u}(v - \langle v \rangle) \rangle_{\mathbf{u}} = \langle \mathbf{u}(\mathbf{u}^T - \langle \mathbf{u} \rangle^T) \mathbf{w} \rangle_{\mathbf{u}} = (\langle \mathbf{u} \mathbf{u}^T \rangle - \langle \mathbf{u} \rangle \langle \mathbf{u} \rangle^T) \mathbf{w} \\ &= \mathbf{C} \mathbf{w} \quad (\mathbf{C} \text{ is the input covariance matrix } \langle \mathbf{u} \mathbf{u}^T \rangle - \langle \mathbf{u} \rangle \langle \mathbf{u} \rangle^T) \end{aligned}$$

Are these learning rules stable?

- Does \mathbf{w} converge to a stable value or explode?

⇒ Look at what happens to the length of \mathbf{w} over time

- Hebb rule: $\tau_w \frac{d\mathbf{w}}{dt} = \mathbf{u}v$

$$\frac{d\|\mathbf{w}\|^2}{dt} = 2\mathbf{w}^T \frac{d\mathbf{w}}{dt} = 2\mathbf{w}^T (\mathbf{u}v / \tau_w) = \frac{2}{\tau_w} v^2 > 0 \quad \mathbf{w} \text{ grows without bound!}$$

- Covariance rule: $\tau_w \frac{d\mathbf{w}}{dt} = \mathbf{u}(v - \langle v \rangle)$

$$\frac{d\|\mathbf{w}\|^2}{dt} = 2\mathbf{w}^T \frac{d\mathbf{w}}{dt} = 2\mathbf{w}^T (\mathbf{u}(v - \langle v \rangle) / \tau_w) = \frac{2}{\tau_w} (v^2 - v\langle v \rangle)$$

CONSTRAINT

$$\|\mathbf{w}\| = 1$$

$$\frac{d\|\mathbf{w}\|}{dt}$$

$$\text{Averaging RHS, } \frac{d\|\mathbf{w}\|^2}{dt} = \frac{2}{\tau_w} (\langle v^2 \rangle - \langle v \rangle^2) = \frac{2}{\tau_w} \sigma_v^2 > 0$$

\mathbf{w} grows without bound!

Oja's Rule for Hebbian Learning

♦ Oja's rule: $\tau_w \frac{d\mathbf{w}}{dt} = \underline{\mathbf{u}v} - \underline{\alpha v^2 \mathbf{w}}$ ($\alpha > 0$)

♦ Stable?

$$\frac{d\|\mathbf{w}\|^2}{dt} = 2\mathbf{w}^T \frac{d\mathbf{w}}{dt} = \frac{2}{\tau_w} \mathbf{w}^T (\mathbf{u}v - \alpha v^2 \mathbf{w}) = \frac{2}{\tau_w} (v^2 - \alpha v^2 \mathbf{w}^T \mathbf{w})$$

i.e., $\tau_w \frac{d\|\mathbf{w}\|^2}{dt} = 2v^2(1 - \alpha\|\mathbf{w}\|^2)$

At steady state $\|\mathbf{w}\|^2 = \frac{1}{\alpha}$ ($\|\mathbf{w}\| = \frac{1}{\sqrt{\alpha}}$)

\mathbf{w} does not grow without bound, i.e.,
Oja's rule is stable!

9

Summary: Hebbian Learning

♦ Hebb rule:

$$\tau_w \frac{d\mathbf{w}}{dt} = \mathbf{u}v$$

Unstable

(unless constraint on $\|\mathbf{w}\|$ is imposed)

♦ Covariance rule:

$$\tau_w \frac{d\mathbf{w}}{dt} = \mathbf{u}(v - \langle v \rangle)$$

Unstable

(unless constraint on $\|\mathbf{w}\|$ is imposed)

♦ Oja's rule:

$$\tau_w \frac{d\mathbf{w}}{dt} = \mathbf{u}v - \alpha v^2 \mathbf{w}$$

Stable

$$\|\mathbf{w}\| \rightarrow \frac{1}{\sqrt{\alpha}}$$

10

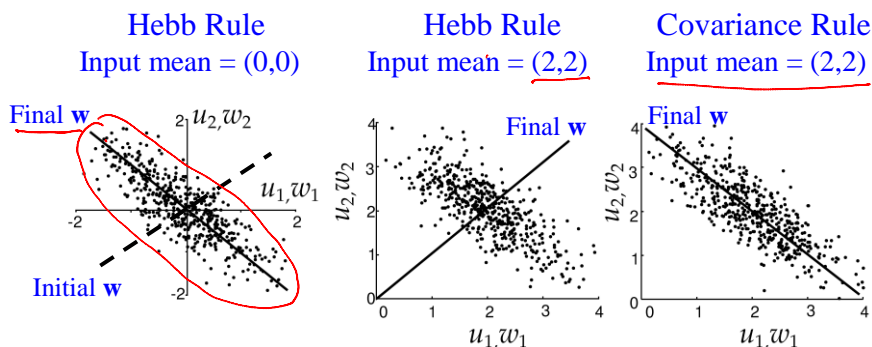
What does Hebbian Learning do anyway?

- Start with the averaged Hebb rule: $\tau_w \frac{d\mathbf{w}}{dt} = Q\mathbf{w}$
- How do we solve this equation to find $\mathbf{w}(t)$?
 ⇒ **Eigenvectors** to the rescue (again)!
- Write $\mathbf{w}(t)$ in terms of eigenvectors of Q : $\mathbf{w}(t) = \sum_i c_i(t) \mathbf{e}_i$
- Substitute in Hebb rule diff. eq. and simplify as before:
 $\tau_w \frac{dc_i}{dt} = \lambda_i c_i$ i.e., $c_i(t) = c_i(0) \exp(\lambda_i t / \tau_w)$
 $\mathbf{w}(t) = \sum_i c_i(t) \mathbf{e}_i = \sum_i c_i(0) \exp(\lambda_i t / \tau_w) \mathbf{e}_i$ ←
 For large t , largest eigenvalue term dominates: $\mathbf{w}(t) \propto \mathbf{e}_1$
 (For Oja's rule: $\mathbf{w}(t) = \frac{\mathbf{e}_1}{\sqrt{\alpha}}$)

11

The Brain can do Statistics!*

Hebbian Learning implements Principal Component Analysis (PCA)

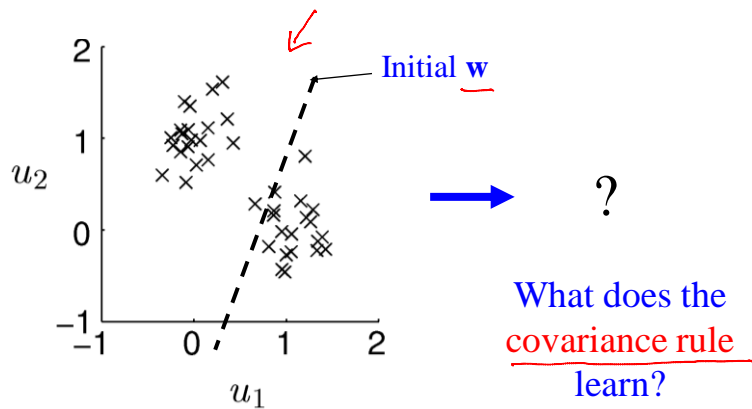


Hebbian learning learns a weight vector aligned with the principal eigenvector of input correlation/covariance matrix (i.e., direction of maximum variance)

12

*See last week's lecture for "The Brain can do Calculus!"

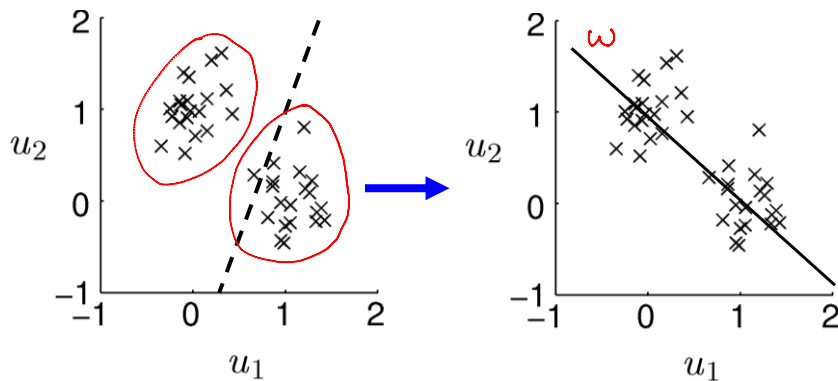
What about this input data?



13

Image Source: Dayan & Abbott textbook

PCA does not correctly describe the data



What should a network of neurons learn from such data?

Next Lecture: Competitive Learning, Generative Models, and
Unsupervised Learning

14

Image Source: Dayan & Abbott textbook