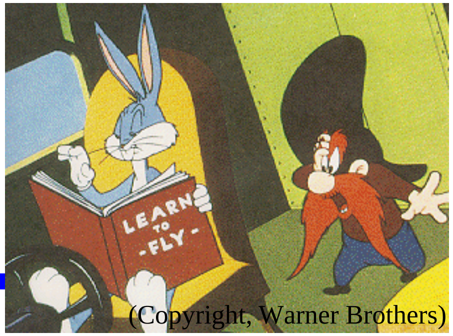
CSE/NB 528

Lecture 11: Plasticity and Learning

(Chapter 8)

Gameplan for Today



(Copyright, Warner Brothers)

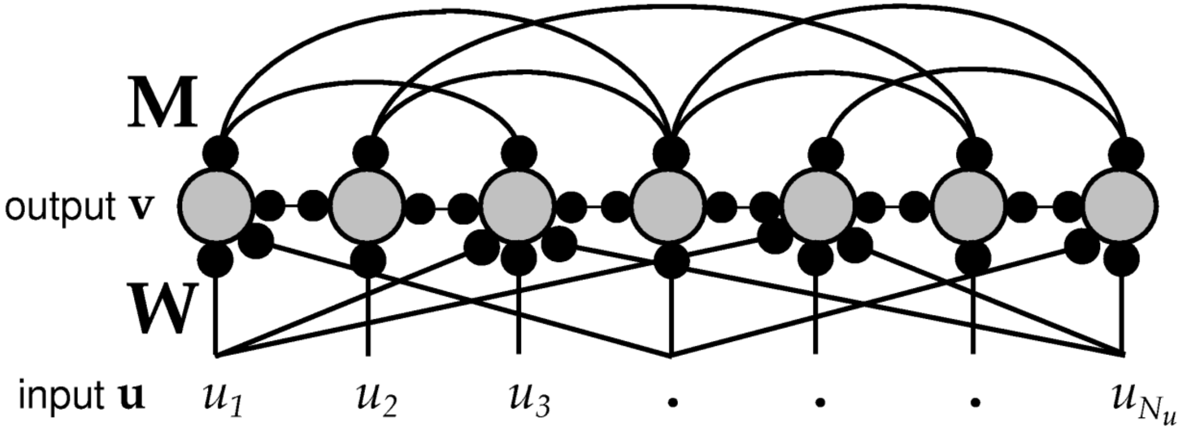
- Plasticity and Learning
  - Types: Unsupervised, Supervised, and Reinforcement learning
- Unsupervised Learning
  - Hebb rule and its variants (Covariance, Oja rule)
  - Mathematical formulation
  - Stability analysis of learning rules

So far, we have been analyzing networks with *fixed* sets of synaptic weights W and M

(based on eigenvalues of M etc.)

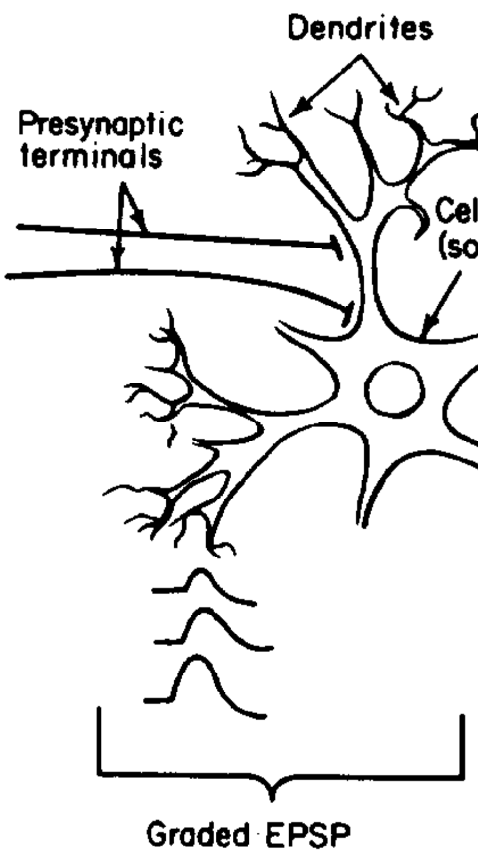
Can synaptic weights be adapted in response to inputs?

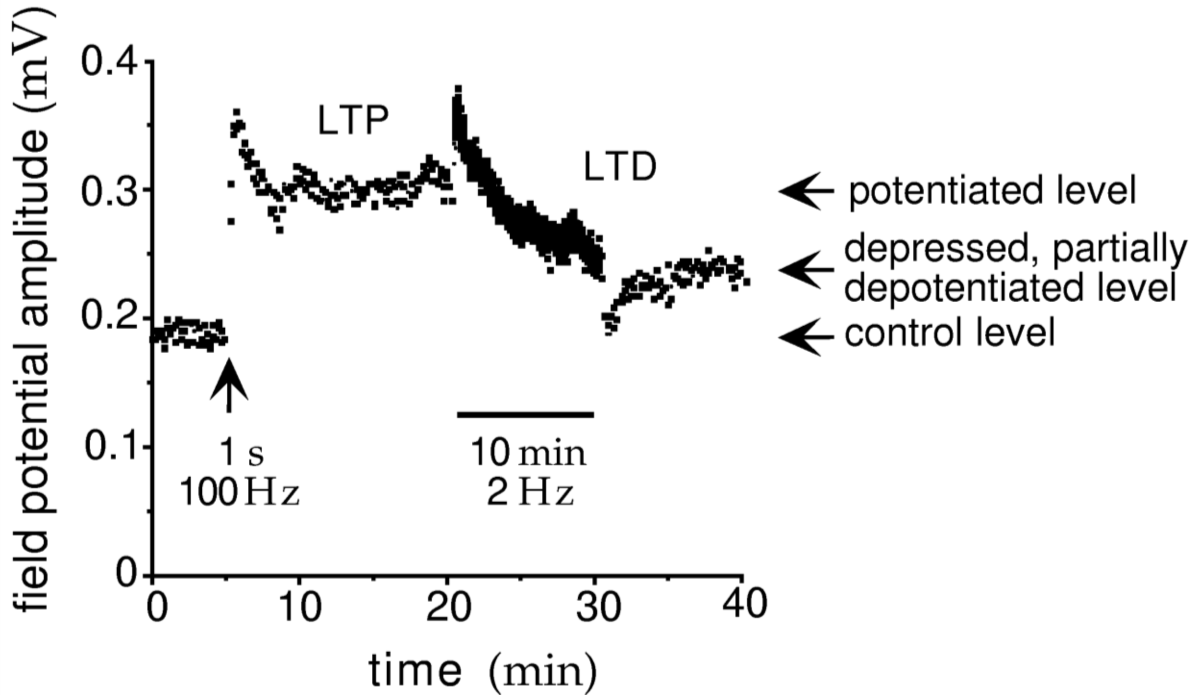
Plasticity and Learning: Adapting the Connections



- Question 1: How do we adapt the synaptic weights W and M to solve useful tasks?
- Question 2: How does the brain do it?

Synaptic Plasticity in the Brain

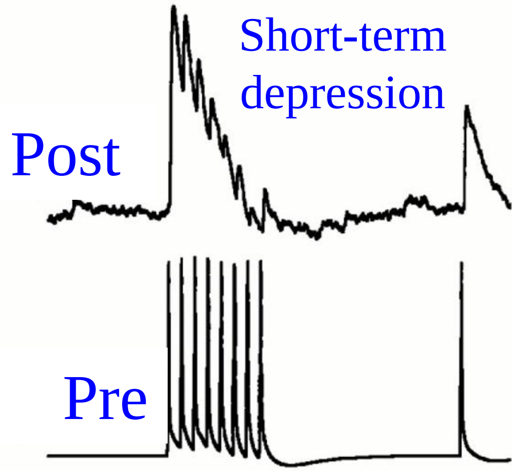


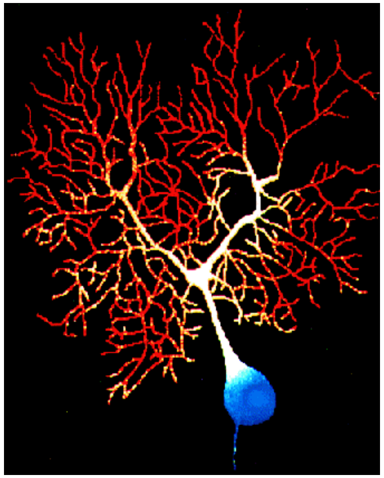


LTP = Long Term Potentiation
LTD = Long Term Depression

Other Forms of Plasticity in the Brain

- Short-Term Synaptic Plasticity
  - Short-term depression/facilitation
  - Dynamics may change on a long-term basis via LTP/LTD
- Changes to intrinsic excitability of cell
  - Density and distribution of various channels (ionic conductances)
  - Currently active research area
- Growth and morphological changes in dendrites
  - Currently active research area
- Addition of new neurons?
  - Hot topic of research in recent years…





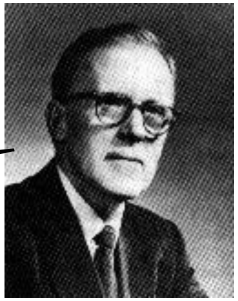
The Theory: Classification of Learning Algorithms

- **Unsupervised Learning**
  - Synapses adapted based solely on inputs
  - Network self-organizes in response to *statistical patterns* in input
  - Similar to Probability Density Estimation in statistics
- **Supervised Learning**
  - Synapses adapted based on inputs and desired outputs
  - External "teacher" provides desired output for each input
  - Goal: Function approximation
- **Reinforcement Learning**
  - Synapses adapted based on inputs and (delayed) reward/punishment
  - Goal: Pick outputs that *maximize total expected future reward*
  - Similar to optimization based on Markov decision processes

Let’s start with Unsupervised Learning

Consider a single neuron receiving feedforward inputs from other neurons (e.g. from the retina)

The Grand-Daddy of Unsupervised Learning



- Rule hypothesized by Donald Hebb in 1949
- Hebb's learning rule:

"If neuron A frequently contributes to the firing of neuron B, then the synapse from A to B should be strengthened"

- Related Mantra: *Neurons that fire together wire together*
- Hebb's goal: Produce clusters of neurons ("*cell assemblies*") that fire together in response to a stimulus

Mathematical Formulation of Hebb's Rule

On-Board Derivation

Formalizing Hebb's Rule

- Consider a linear neuron (steady state): $v = \mathbf{w}^T \mathbf{u} = \mathbf{u}^T \mathbf{w}$
- Basic Hebb Rule: $\tau_w \dfrac{d\mathbf{w}}{dt} = \mathbf{u}v$ (or $\mathbf{w} \leftarrow \mathbf{w} + \varepsilon \cdot \mathbf{u}v$)
- What is the average effect of this rule?

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

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

Variants of Hebb's Rule

- Pure Hebb only increases synaptic weights (LTP)
  - What about LTD?

- Covariance rule:

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

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

  - where θ_v can be set to the average value of v.
  - Why is this called the covariance rule?

Are these learning rules stable?

On Board Analysis, leading up to Oja’s rule

Next Class: Unsupervised Learning

- Things to do:
  - Finish Chapter 8 and Start Chapter 10
  - Homework 3 due on Friday May 20
  - Start mini-project