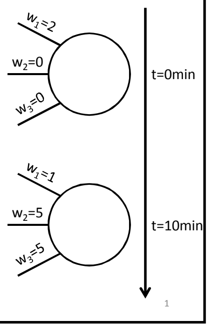
CISC 3250
Systems Neuroscience

Neuroplasticity: Learning in Neurons

Professor Daniel Leeds
dleeds@fordham.edu
JMH 328A



1

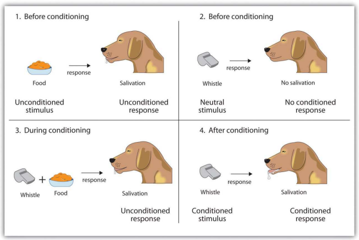
Two forms of plasticity

- **Structural plasticity**: generation of new connections between neurons
- $\boxed{\textbf{Functional plasticity}\text{: changing strength of connections between neurons}}$

Hebbian plasticity:
"cells that fire together, wire together"

2

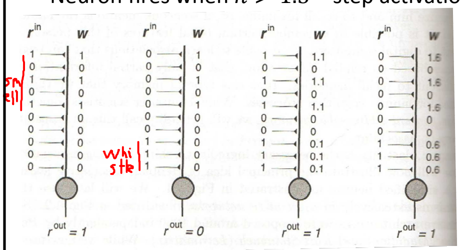
Cognitive level: Conditioning



Associating both smell and whistle with food
- **Unconditioned stimulus**: smell – already associated with food
- **Conditioned stimulus**: whistle – indicates food coming

Computing level: Associator network

Define input $h = \sum_i w_i r_i^{in}$
Neuron fires when $h > 1.5$ – step activation function



At each learning step, add 0.1 to weights of pre-synaptic inputs co-occuring with post-synaptic firing

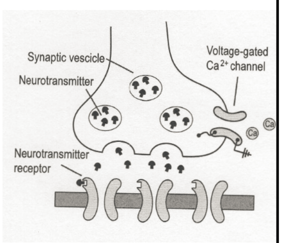
Smell input | Whistle input | Whistle+smell input | Whistle input, **after** 6 learning steps

4

Chemical level: NT receptors

Increase weight by improving NT detection
Post-synaptic:
- Insert more receptors into dendrite membrane
- Improve performance of receptors

Pre-synaptic:
- Increase amount of NT released



5

Marr's levels of analysis

- **Computational theory:** Learn associations among sensations
- **Representation and algorithm:** Associate each sense with set of neural outputs, adjust weights on these outputs into another neuron
- **Hardware implementation:** Insert/remove NT receptors from dendrites

6

Features of associators

- Pattern completion/ generalization

- Fault tolerance
  - Selected dendrites miss input, post-synaptic neuron still fires
- Learning prototypes

  - Neuron firing for common combinations

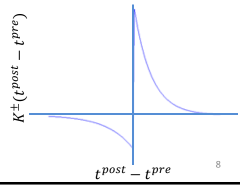
7

Math of Hebbian spike learning

- Pre-synaptic spike followed by post-synaptic spike -> increase weight
- Post-synaptic spike followed by pre-synaptic spike -> decrease weigh

$$\Delta w_{ij}^{\pm} = \underline{\epsilon^{\pm}(w)}K^{\pm}(t^{post} - t^{pre})$$

Prevent weights from increasing to ∞

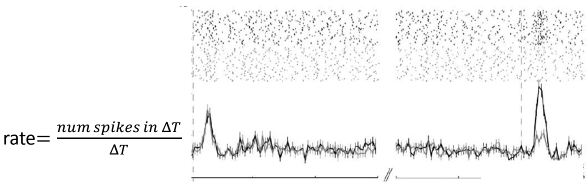


8

Math of Hebbian rate learning

"Cells that fire together, wire together"

$$\Delta w_{ij} = \epsilon r_i r_j$$ – always ↑Δw



9

Math of Hebbian rate learning

"Cells that fire together, wire together"

$$\Delta w_{ij} = \underline{\epsilon} r_i r_j$$ – always ↑Δw

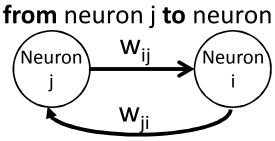
ε(w_ij)

Prevent weights from increasing to ∞

Notation correction:

w_{ij} is weight of synapse **from** neuron j **to** neuron i

$$w_{ij} = w_{j \to i}$$

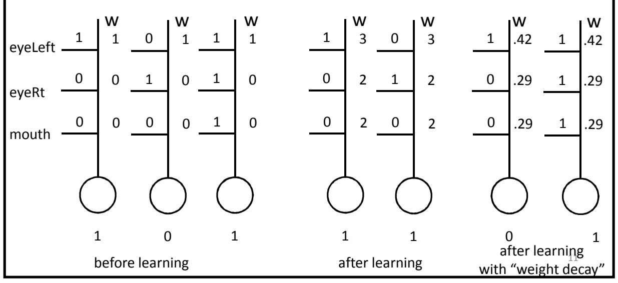


10

Learning a face detector

$$h = \sum_j w_{ij} r_j^{in}$$

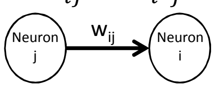
Neuron fire when h>0.5



11

Weight decay

$$\Delta w_{ij} = \epsilon r_i r_j$$



At each learning step, add $\epsilon r_i r_j$ to weight of synapse from neuron j firing at rate r_j onto neuron i firing at rate r_i

Assume default weight w_{ij}=0, weights tend to return to 0

- Require multiple pre-synaptic neural inputs for post-synaptic excitation – prototypes
- Allow weights on unused synapses to shrink – unlearn associations

12

Weight decay:

- Move weight towards 0

push to 0

in proportion to post-syn firing

$$\Delta w_{ij} = r_i r_j - r_i w_{ij} - Willshaw$$

Weight normalization:

- Force all incoming weights to add up to 1 after each weight update:

$$w_{ij} \leftarrow \frac{w_{ij}}{\sum_j w_{ij}}$$

$\sum_j w_{ij} = 1$

13