Title: Learning and Memory in the Brain
(Les Gastins 2025)

(A) Machine learning

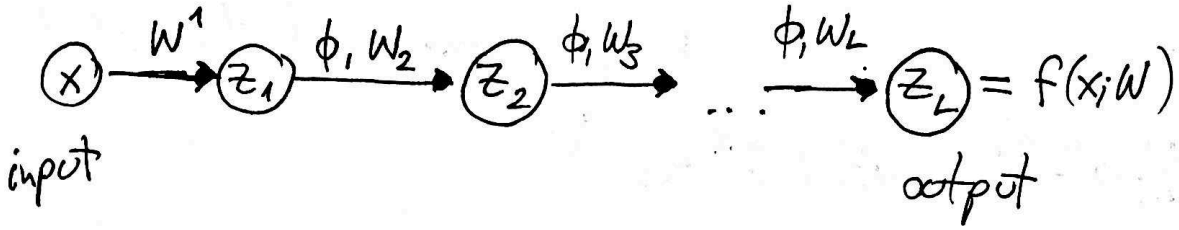
• Feedforward network with $L-1$ hidden layers $z_1, \dots, z_{L-1}$

Output $f(x;W) = z^L(x)$, $z^\ell = W^\ell \phi(z^{\ell-1}) \in \mathbb{R}^{N_\ell}$

$z^1 = W^1 x$ ← nonlinearity

input $x \in \mathbb{R}^{N_0}$ ↑ input

Weights $\{W^1, \dots, W^L\}$ ↑ $W^\ell \in \mathbb{R}^{N_\ell \times N_{\ell-1}}$



• training

desired output ↘ ↙ network output

Objective function $C(y, f(x;W))$: How much does the network output $f(x;W)$ for given input x differ from desired output y.

E.g. $C(y, f(x;W)) = \|y - f(x;W)\|^2$ (MSE)

Gradient descent

Given training data $\{x^\mu, y^\mu\}_\mu$, loss $\mathcal{L}(W) := \sum_\mu C(y^\mu, f(x^\mu;W))$

Gradient descent

$$W^\ell \leftarrow W^\ell - \lambda \left[\frac{\partial \mathcal{L}(W)}{\partial W^\ell_{ij}}\right]_{ij}$$

↑ learning rate

• / Backpropagation of the error : efficient recursive algo. to compute $\frac{\partial \mathcal{L}}{\partial w^l}$ for all $l = 1, \dots, L$

Needed : $\frac{\partial C}{\partial w^l_{ij}} = \frac{\partial C}{\partial f_{i_L}} \frac{\partial z^L_{i_L}}{\partial w^l_{ij}}$ (sum over repeated indices $i_L, i_{L-1}, \dots$)

$$= \underbrace{\frac{\partial C}{\partial f_{i_L}} W^L_{i_L i_{L-1}} \phi'(z^{L-1}_{i_{L-1}}) \dots W^{l+1}_{i_{l+1} i} \phi'(z^l_i)}_{\equiv \delta^l_i \text{ (error signal at layer } l)} \phi(z^{l-1}_j)$$

recursion :

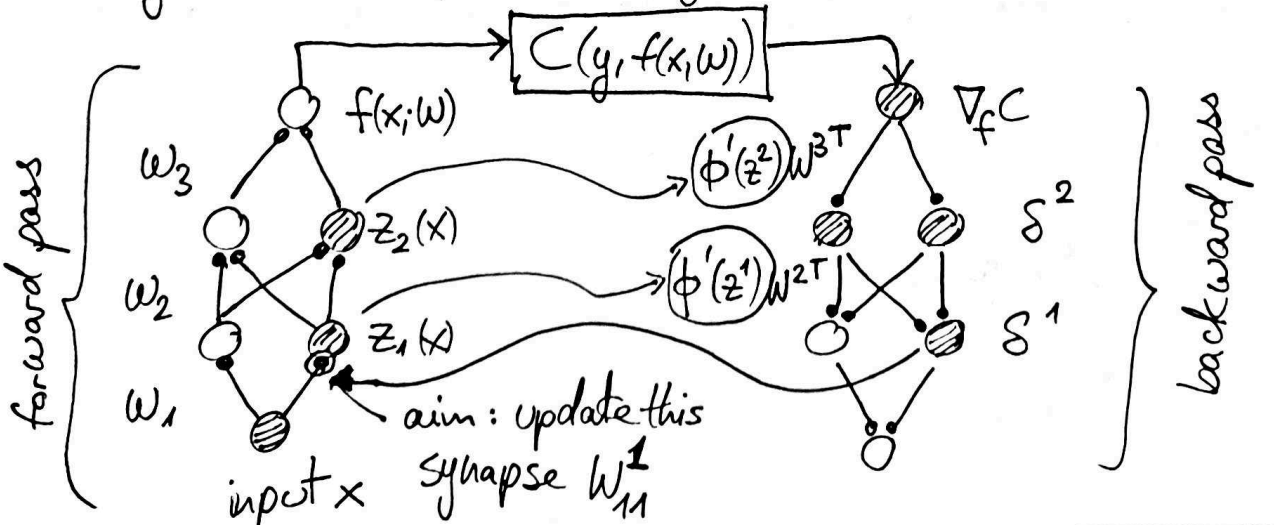
$$\delta^{l-1}_i = \sum_k \delta^l_k W^l_{ki} \phi'(z^{l-1}_i) \quad \text{with } \delta^l \in \mathbb{R}^{N_l}$$

or $$\boxed{\delta^{l-1} = \text{diag}\left[\phi'(z^{l-1}_i)\right] (W^l)^T \delta^l \text{ with } \delta^L = \nabla_f C}$$

$$\boxed{\frac{\partial C}{\partial w^l} = \delta^l \phi(z^{l-1})^T \left(\text{and } \frac{\partial C}{\partial w^1} = \delta^1 x^T\right)}$$

• / In the Brain : How could a synapse W^l_{ij} receive the error signal δ^l_i which determines how it should be updated to optimize the objective function $C(y, f(x,w))$?

E.g. x : visual stimulus, y : desired motor output (eg. move your arm)



Several issues:

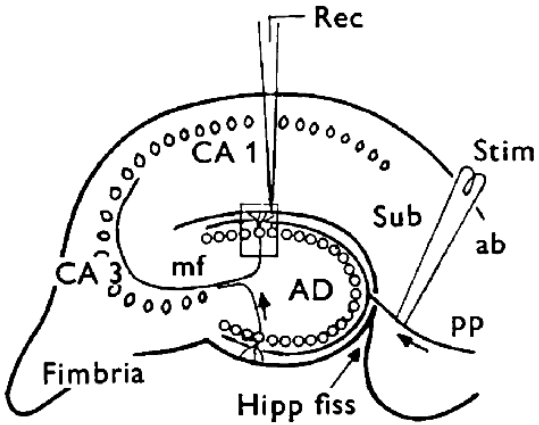
- no nonlinearity in the backward pass:

$$z^l = w^l \phi(z^{l-1}) \quad \text{(forward)}$$

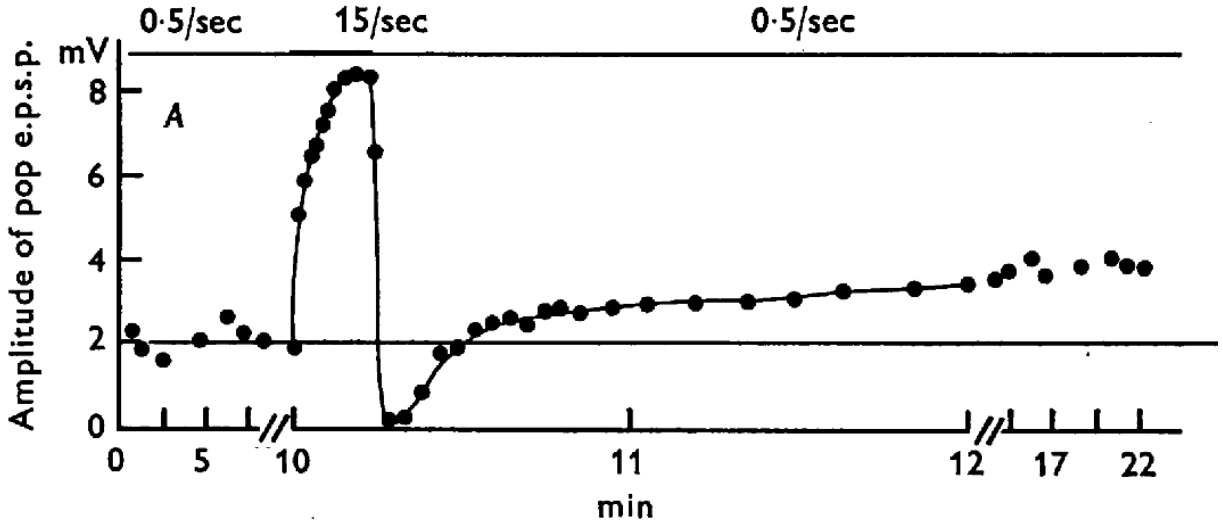
but $\delta^{l-1} = \phi'(z^{l-1}) w^{l^T} \delta^l$ (backward)

- Either forward and backward pass happen in the same network, then synapses need to be symmetric $w^l_{ij} = w^l_{ji}$, but:
  → that's usually not the case and
  → the network would need to precisely time and alternate forward and backward passes
- Or the backward pass happens in a separate network with exactly the same (but reversed) synapses. But then it would also need knowledge of $\phi'(z^{l-1})$ and still time the alternating forward/backward passes
- (Not so clear what exactly the output in the brain would be ≠)

Long Term Potentiation (LTP)



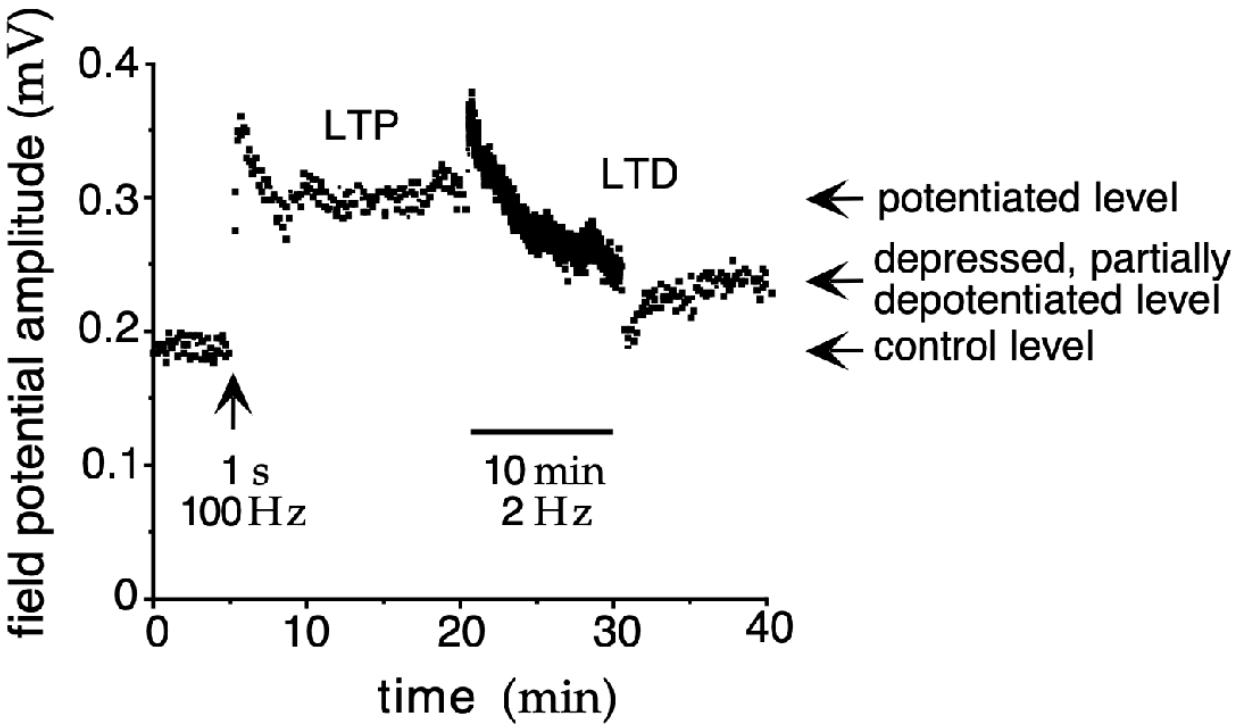
Perforant path fibres (pp) in hippocampus of rabbits anaesthetized with urethane.



Increasing the stimulation frequency from 5/sec to 15/sec during a short period (~2min) increases the neuron's response for a long period

Bliss and Lomo (1973)

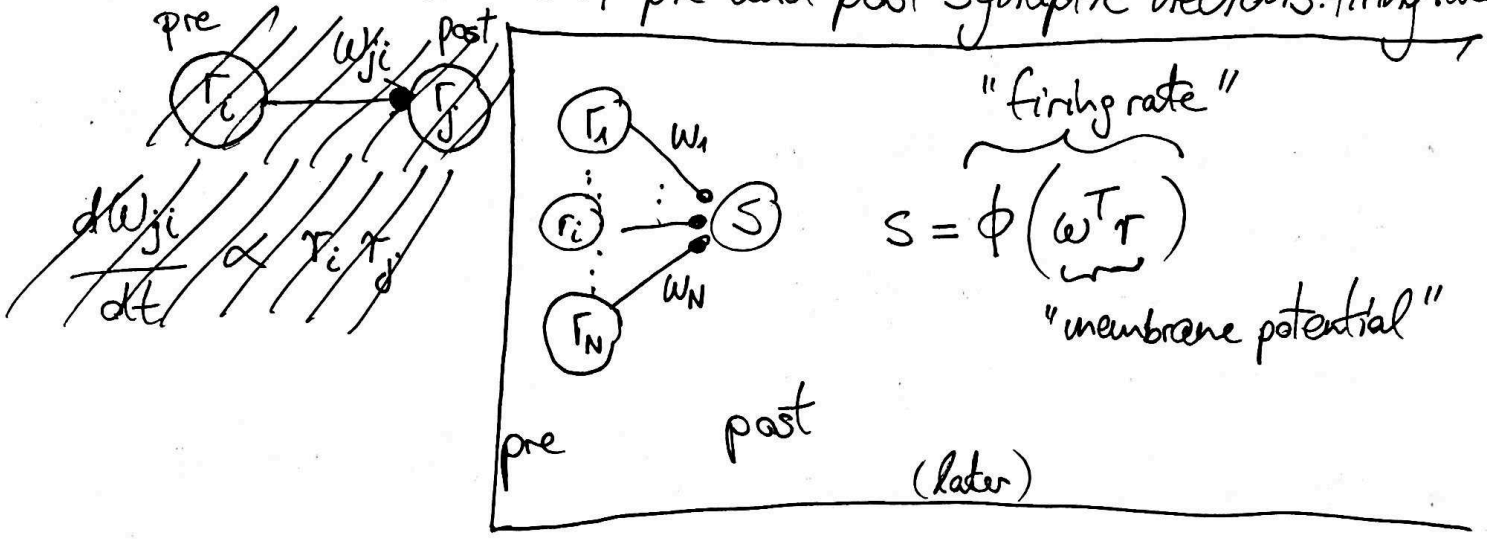
Long Term Depression



(B) Neuroscience

- Hebb (1949) conjectures that if input from neuron A often contributes to the firing of neuron B, then the synapse A → B is strengthened (A)—•(B)
- Experimental evidence: Bliss & Lomo (1973) and many studies afterwards

/ theoretical models: Hebbian learning / plasticity based on correlations of pre and post synaptic neurons firing rates



$$\frac{dW_i}{dt} \propto r_i S \quad \text{(basic hebbian learning)}$$

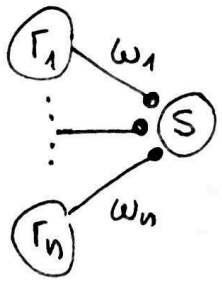
/ Problems

- Weights grow unconstraint ~~$\frac{d\vec{w}}{dt} = (\vec{r}^T S + \vec{w})$ (normalization)~~
- Symmetric weights per default $\frac{dw_{ij}}{dt} \propto r_i r_j \rightarrow f(r_i) g(r_j)$

~~What can this do?~~

- Unsupervised: How to optimize an objective function without access to any error signal?
- Can this do interesting things? (Yes → Hopfield)

- Oja's modification of Hebbs rule : Normalize weights + small updates γ



$$\omega(t+1) = \frac{\omega(t) + \gamma\, r(t)\, s(t)}{\| \omega(t) + \gamma\, r(t)\, s(t) \|} \approx \omega(t) + \gamma \underbrace{(sr}_{\text{"learning"}} - \underbrace{\omega s^2)}_{\text{"forgetting"}}$$

here: lin. approx

where $s(t) = \phi(r^T\omega) \approx r^T\omega$

↑ transfer function: Membrane pot → Firing rate

Take away: Weights ω converge to the eigenvector of

$C := \langle r r^T \rangle_t$ (~~correlation~~ covariance matrix of the input to neuron s)

with largest eigvalue, i.e. ω becomes the 1st PC. (principle component).

Proof:

Note that $\|\omega(t)\| = 1 \;\forall t$ (Def. of the update rule)

$$\omega(t+1) \approx \omega(t) + \gamma \left.\frac{\partial \omega(t+1)}{\partial \gamma}\right|_0$$

$$\left.\frac{\partial \omega(t+1)}{\partial \gamma}\right|_0 = sr - \omega \frac{\partial}{\partial \gamma} \|\omega + \gamma r s\|$$

$$= sr - \omega \left.\frac{(\omega + \gamma r s)^T r s}{\|\omega + \gamma r s\|}\right|_{\gamma=0} = sr - \omega \underbrace{\omega^T r s}_{s}$$

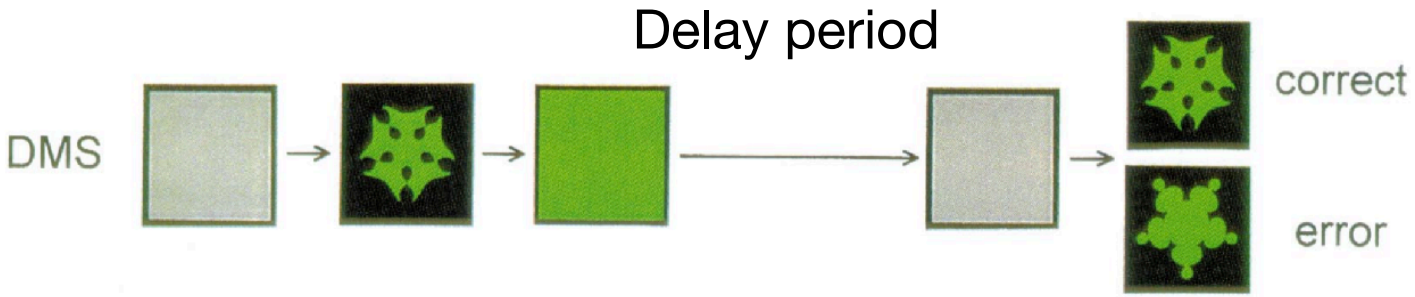
$$\Delta\omega(t) = \gamma(sr - \omega s^2) \leftarrow \text{substitute } s = r^T\omega$$

$$\langle \Delta\omega(t) \rangle_t = 0 = \gamma \left(\langle r r^T \rangle_t \omega - \left(\omega^T \langle r r^T \rangle_t \omega\right) \omega \right)$$

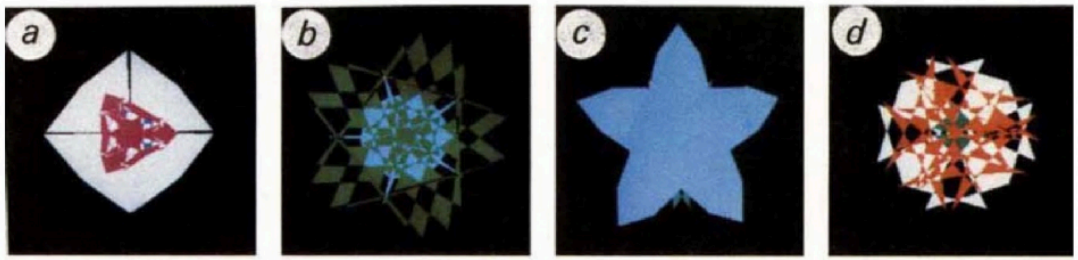
$$\Leftrightarrow C\omega = (\omega^T C \omega)\,\omega$$

ω is evec. of C with eval $\omega^T C \omega$. One can show that the dynamic converges to ω st. $\omega^T C \omega$ is maximal ∎

Delay Match to Sample Tasks

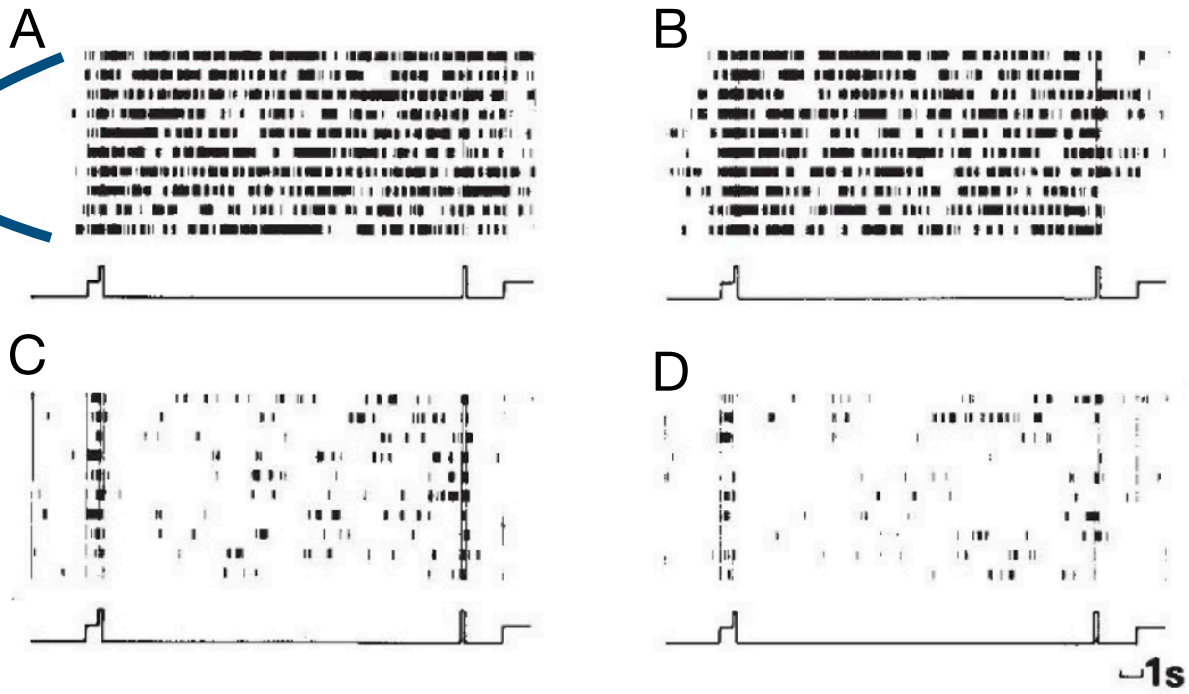


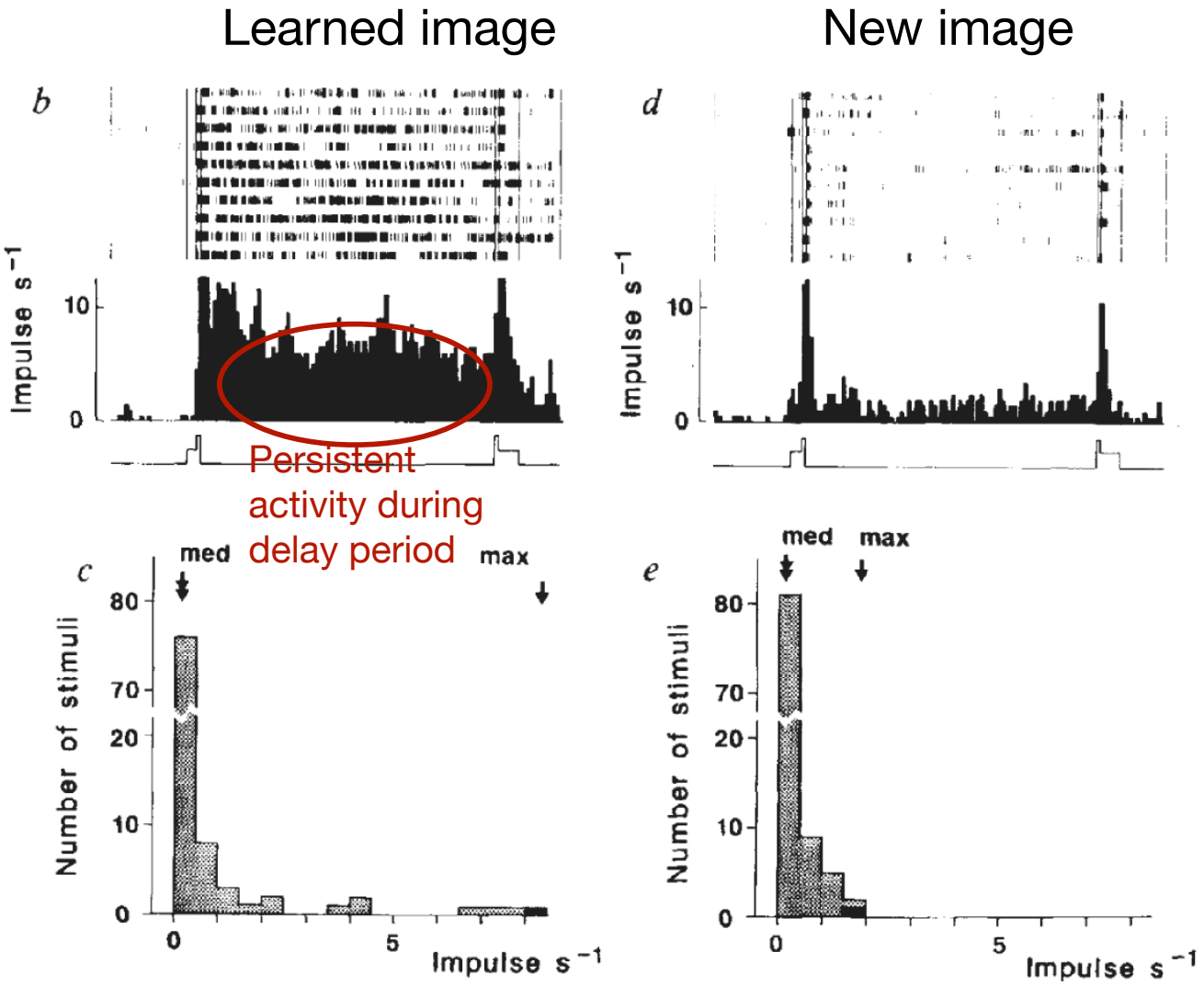
Adapted from Y Naya, K Sakai, Y Miyashita (1996)



Y. Miyashita, H.S. Chang,
“Neuronal correlate of pictorial short-term memory in the primate temporal cortex”
1988

Same neuron, different trials





Y Miyashita, “Neuronal correlate of visual associative long-term memory in the primate temporal cortex” (1988)

(b,d) Trial averaged activity of the neuron that is most active for the given image

(c,d) Distribution of firing rates (Impulse/s) during delay period for a given neuron and 100 images

• Hopfield network (1982)

- Binary neurons $S_i(t) = \begin{cases} 1, & \text{active} \\ -1, & \text{inactive} \end{cases}$ $\quad i = 1, \dots, N$

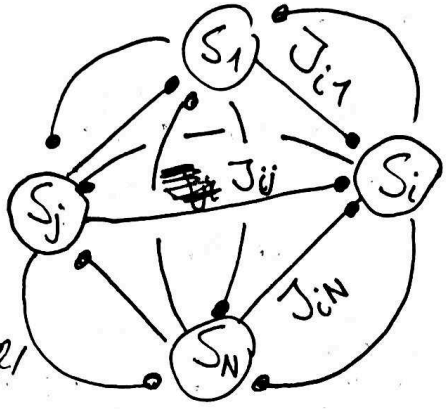
  Connected with synaptic weights J_{ij}.

  Evolving (synchronously or assync.)

  $$S_i(t+1) = \operatorname{sgn}\left(\sum_j J_{ij} \cdot S_j(t) \right)$$

- Biological interpretation

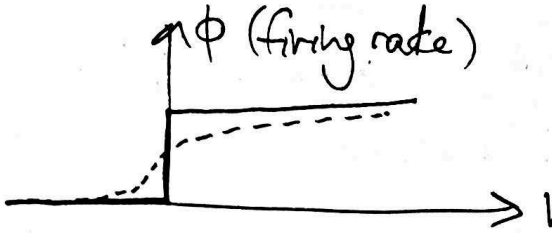
  S_i : firing rate of neuron i



fully connected recurrent network

$h_i = \sum_j J_{ij} S_j$: membrane voltage / potential of neuron i

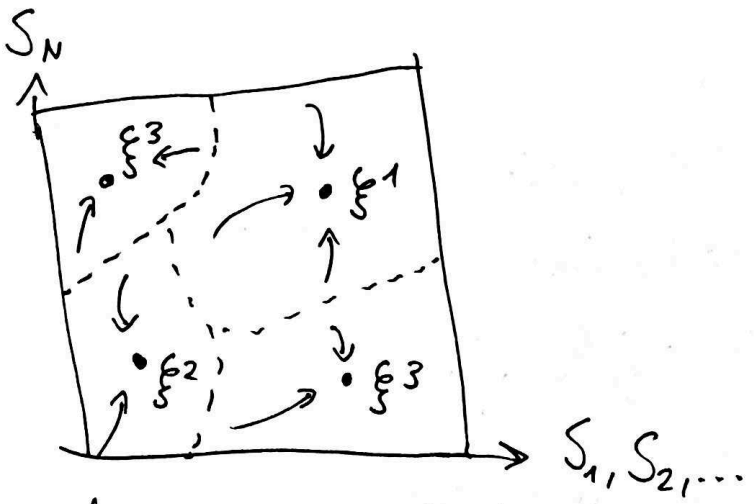
$\phi = \operatorname{sgn}$: transferfunction (potential $\rightarrow$ firing rate)



- How to store memories (here patterns of activity $\xi_i^\mu = \pm 1$; μ: pattern index, i: neuron) in the synapses J_{ij} of this network ~~that~~ ?

  ~~can be recalled as the form of persistent activity~~

  Idea: Memories correspond to fixed points of the dynamics and can be recalled by the dynamics as soon as the network state is close enough / in the bassin of attraction of the fixed point.



- Correct J corresponds to learning the patterns with Hebb's rule one by one: $J^{(0)} = 0$

$$J_{ij}^{(1)} = \frac{1}{N} \xi_i^{(1)} \xi_j^{(1)}$$

($\xi_i^{(1)}$: postsyn.; $\xi_j^{(1)}$: presyn. firing rate)

(the pattern to be learned is presented to the neurons of the network, i.e. they are forced to the corresp. values and the connections are updated)

$$J_{ij}^{\text{final}} = \frac{1}{N} \sum_\mu \xi_i^\mu \xi_j^\mu \quad \text{and} \quad J_{ii} = 0$$

- Dynamics (show that it works)

Can be understood as minimizing an energy

$$E = -\frac{1}{2} \sum_{i \neq j} J_{ij} S_i S_j \overset{J_{ii}=0}{=} -\left(\sum_j J_{kj} S_j\right) S_k - \frac{1}{2} \sum_{i \neq j \neq k} J_{ij} S_i S_j$$

This always decreases: If neuron k is updated

$$S_k(t+1) = \operatorname{sgn}\left(\sum_j J_{kj} S_j(t)\right) = \operatorname{sgn}(h_k(t))$$

$$E(t) - E(t+1) = -\left(\sum_j J_{kj} S_j\right)\left(S_k(t) - S_k(t+1)\right)$$

If S_k does not flip ($S_k(t+1) = S_k(t)$): $\Delta E = 0$

Otherwise: $\Delta E = E(t+1) - E(t) = -2 S_k(t+1) h_k(t) < 0$

($S_k(t+1)$: same sgn as h_k)

- Patterns (if not too many) are stable:

If $S_i(t) = \xi_i^1$ then $S_i(t+1) = \operatorname{sgn}\left(\frac{1}{N} \sum_\mu \sum_j \xi_i^\mu \underbrace{\xi_j^\mu \xi_j^1}_{\text{EW}}\right) \simeq \operatorname{sgn}\left(\xi_i^1\right) = \xi_i^1$

- max number of patterns P that can be stored and perfectly retrieved with N neurons

$$\alpha := \frac{P}{N} = \frac{1}{2 \ln N}$$

Why? Study SNR

Assume $S_i(t) = \xi_i^1$. Then $S_i(t+1) = \mathrm{sgn}(h_i(t))$

$$h_i(t) = \sum_{j \neq i} J_{ij} \xi_j^1 = \underbrace{\xi_i^1}_{\text{Signal}} + \underbrace{\frac{1}{N} \sum_{j \neq i} \sum_{\mu=2}^{P} \xi_i^\mu \xi_j^\mu \xi_j^1}_{\equiv \delta_i \text{ noise}}$$

$\langle \cdot \rangle$ ← avg. over $\xi_i^\mu = \pm 1$ wp $1/2$

$$\langle \delta_i \rangle = 0$$

$$\langle \delta_i^2 \rangle \simeq \frac{P}{N} \equiv \alpha \quad \rightarrow \quad \mathrm{SNR} = \frac{1}{\sqrt{\alpha}}$$

$$\rho \equiv \mathrm{Prob}(\text{Noise } \delta_i < \text{signal} = 1) = \int_{-\sqrt{1/\alpha}}^{\infty} \frac{dx}{\sqrt{2\pi}} e^{-x^2/2}$$

(δ_i is approx Gaussian with std $\sqrt{\alpha}$)

(write $\delta_i = \sqrt{\alpha} X$ with $X \sim N(0,1)$, then $\Pr(\delta < 1) = \Pr(X < \frac{1}{\sqrt{\alpha}})$ $= \Pr(X > -\frac{1}{\sqrt{\alpha}})$)

$$\rho \approx 1 - \exp\left(-\frac{1}{2\alpha}\right)\left(\sqrt{\alpha} + \mathcal{O}(\alpha^{3/2})\right)$$

$$\mathrm{Prob}(\delta_i < 1 \ \forall i) = \rho^N \sim \exp\left(-N\sqrt{\alpha} \exp\left(-\frac{1}{2\alpha}\right)\right)$$

If $\alpha < \frac{1}{2 \ln N}$: $\rho^N \to 1$

If $\alpha > \frac{1}{2 \ln N}$: $\rho^N \to 0$

- if not perfect retrieval is required: $\alpha \sim \mathcal{O}(1)$

- Critiques of Hopfield's model

- No separation between exitatory and inhibitory neurons (Dale's law: All synapses of a given neuron are either inhibitory or exhibitory)
- Electrophys. recordings show rather a sparse coding of memories (most $\xi_i^\mu = 0$)
- Symmetric connections are not biologically plausible (Sompolinsky showed that $J_{ij} + \eta_{ij}$ is robust where $\eta_{ij} \sim N(0, \Delta)$ with $\Delta \sim O(1)$)
- Black-out catastrophe: All patterns are lost if one goes over the capacity. Numerous modifications have been proposed
  - Either: stop learning by some supervised signal when capacity is reached
  - Or: modify st. continual learning is possible where older patterns get forgotten as newer ones are learned.