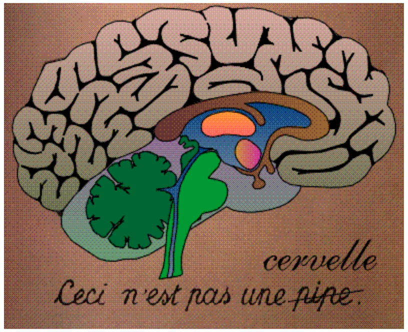
Lecture 14 – Population dynamics and associative memory; stable learning

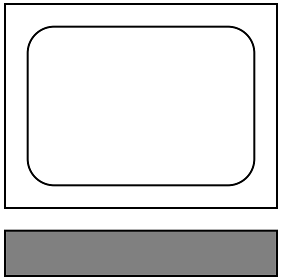
-Introduction
-Associative Memory
-Dense networks (mean-field)
-Population dynamics and Associative Memory
-Discussion

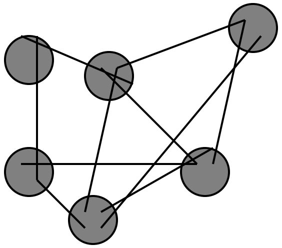
Systems for computing and information processing



Brain

Computer

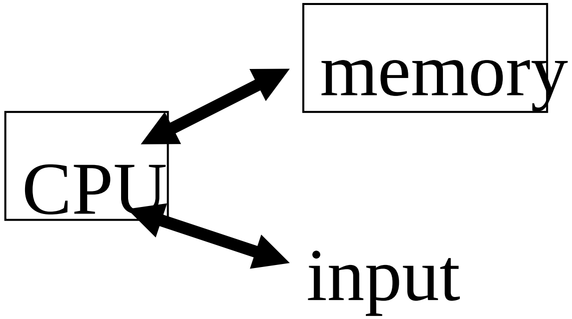




Distributed architecture

(10^{10} proc. Elements/neurons)

No separation of
processing and memory



memory

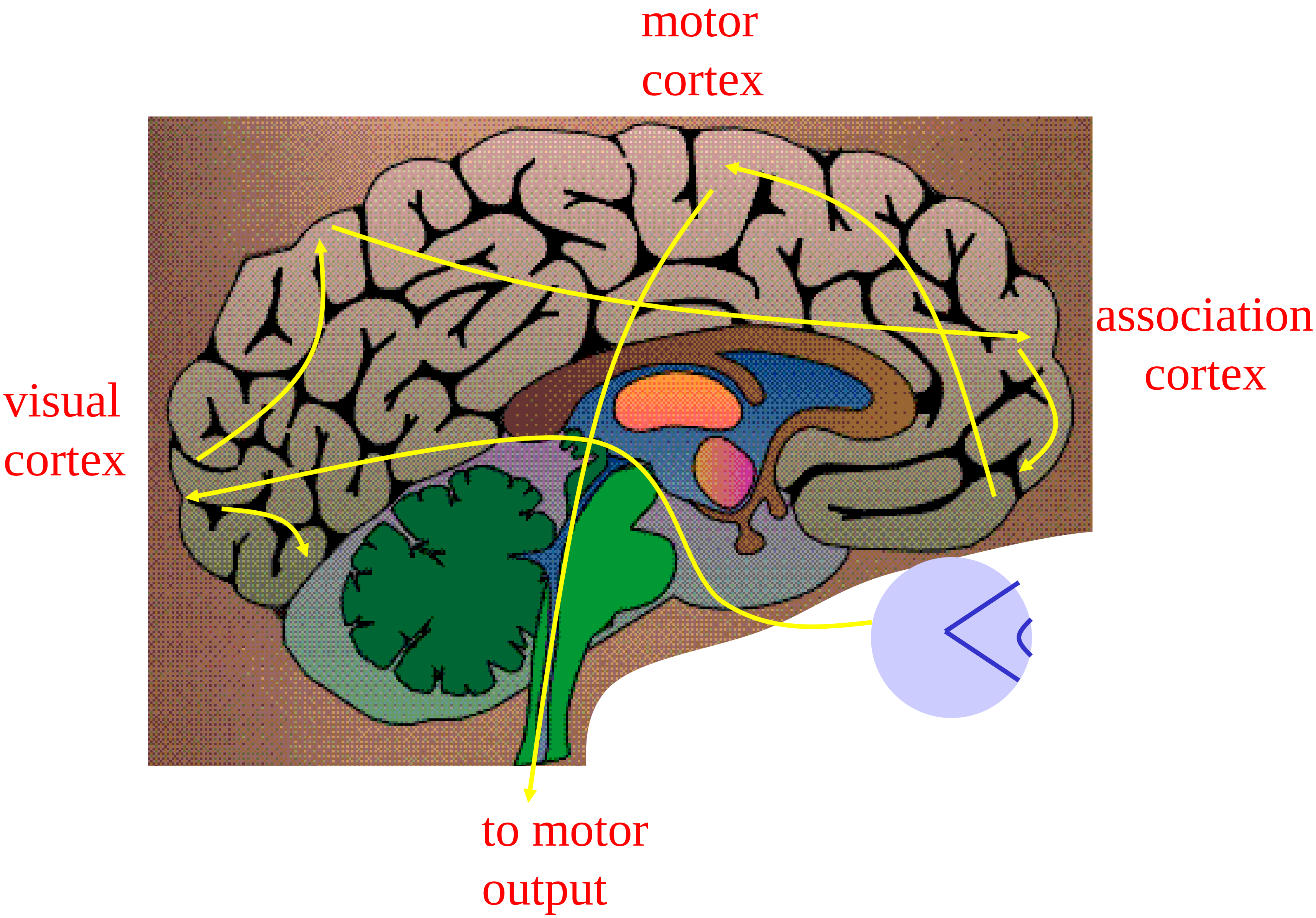
CPU

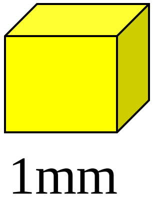
input

Von Neumann architecture

1 CPU

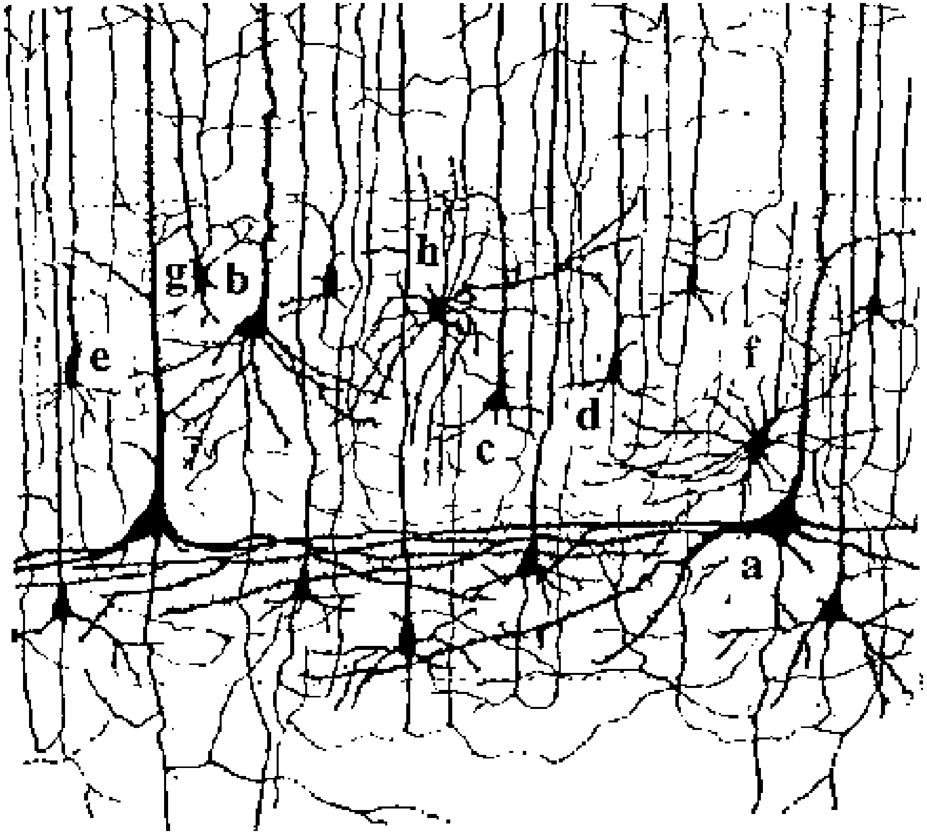
(10^{10} transistors)

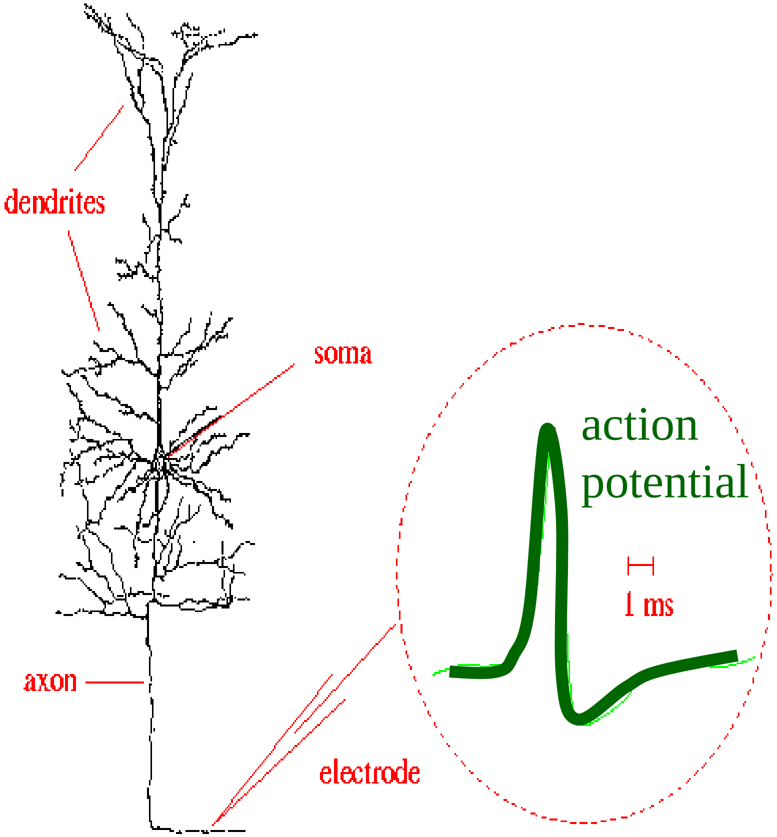

motor cortex
association cortex
visual cortex
to motor output



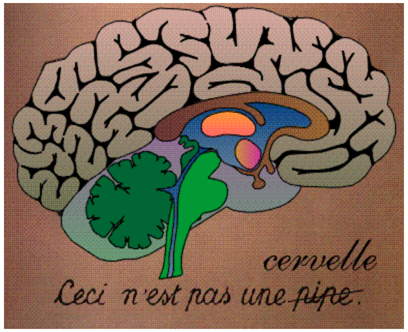
10 000 neurons
3 km wires

Signal:
action potential (spike)

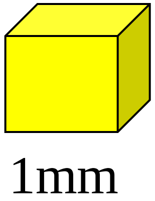




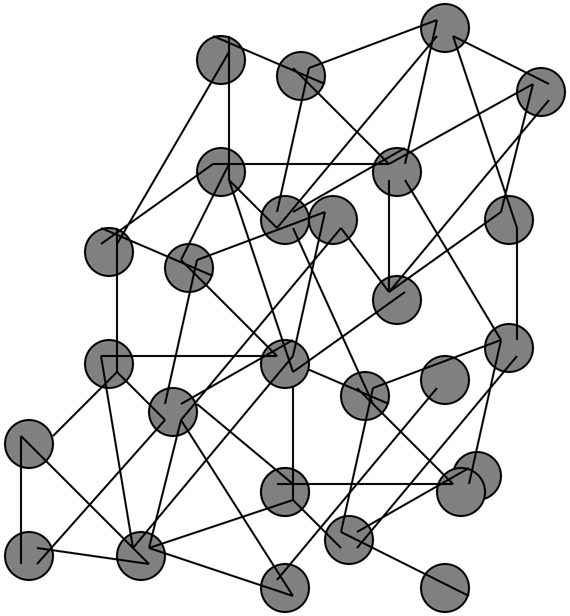
Systems for computing and information processing



Brain



10 000 neurons
3 km wires



Distributed architecture

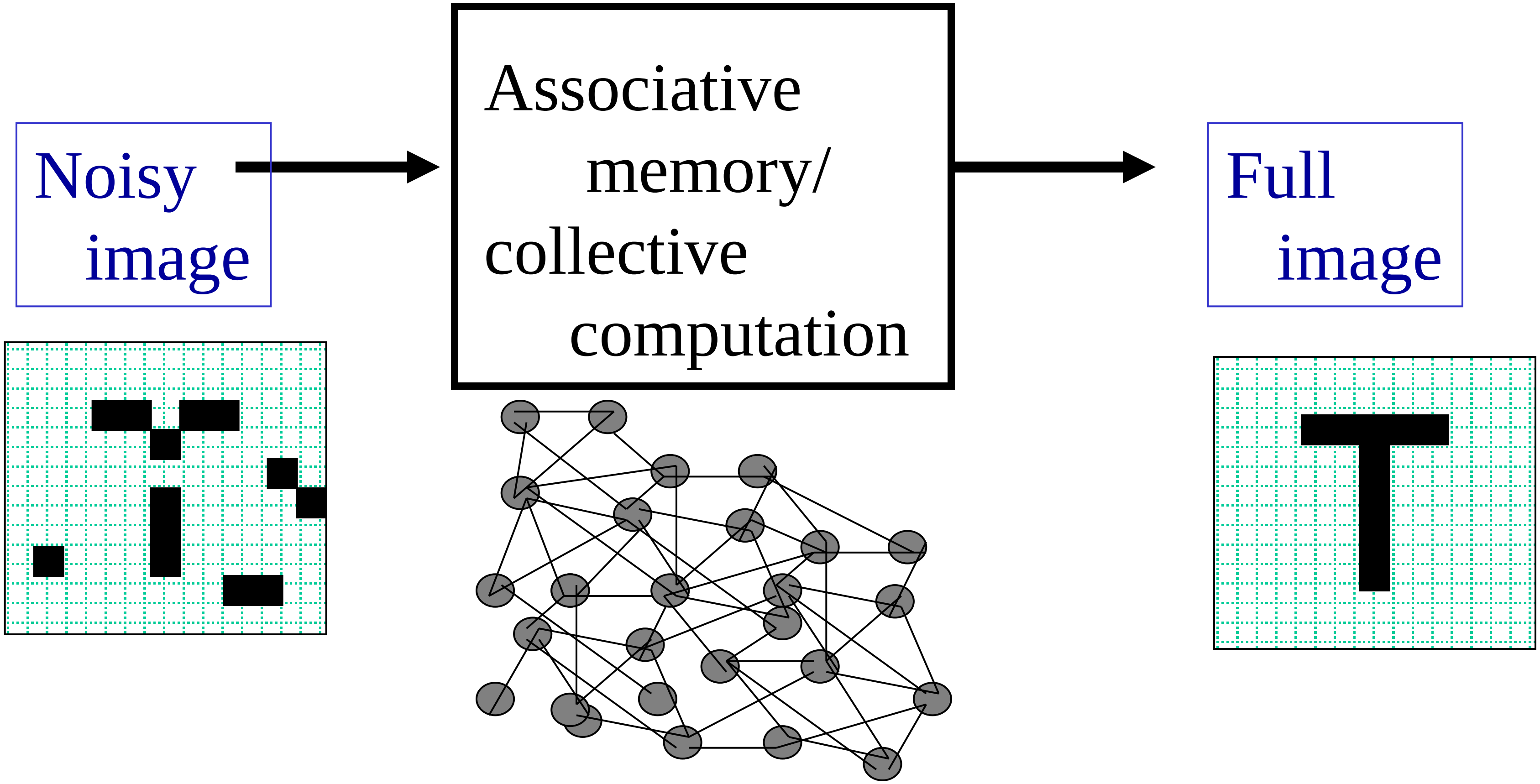
10^{10} neurons

10^{4} connections/neurons

No separation of
processing and memory

- recognize/understand images:

pattern recognition



Noisy image

Associative memory/ collective computation

Full image

Brain-style computation

Lecture 14 – Population dynamics and associative memory; stable learning

✓ -Introduction

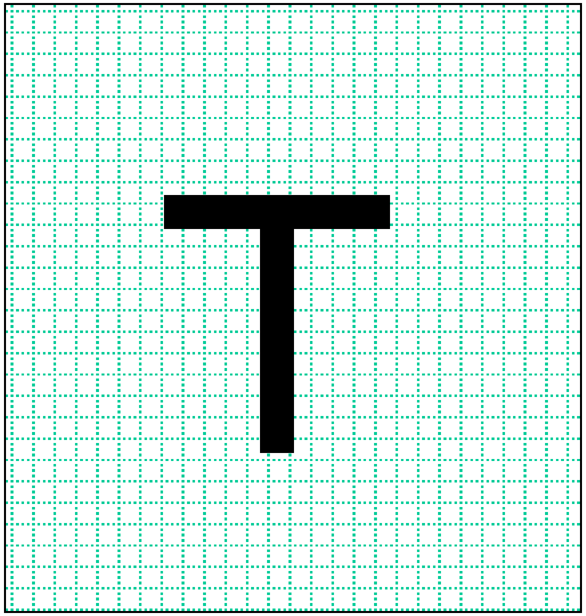
→ -Associative Memory

-Dense networks (mean-field)

-Population dynamics and Associative Memory

-Final discussion

Associative memory



Elementary pixel

■ $S_i = +1$

□ $S_i = -1$

■↔■ $w_{ij} = +1$

□↔□ $w_{ij} = +1$

□↔■ $w_{ij} = -1$

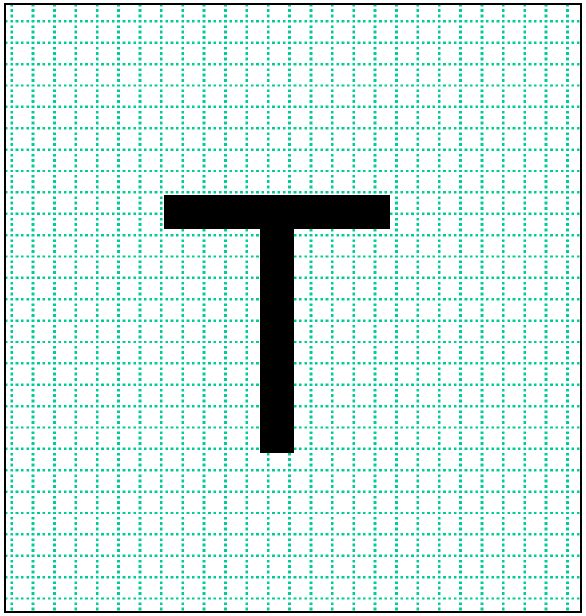
dynamics

$$S_i(t+1) = \mathrm{sgn}\Big[\sum_j w_{ij} S_j \big(\Big.$$

Sum over all interactions with i

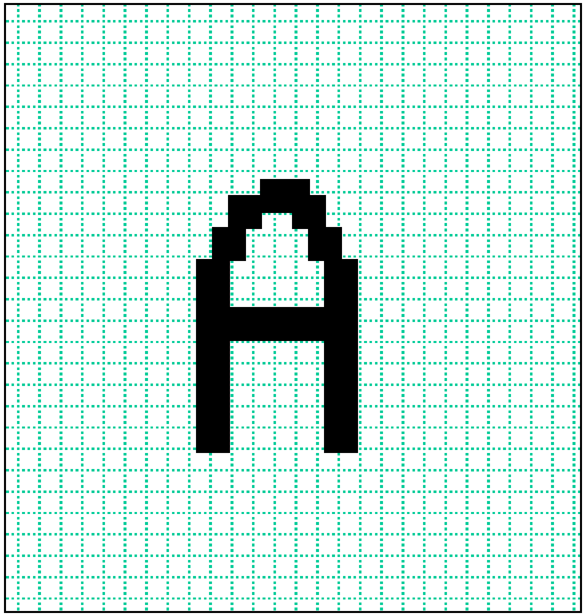
Hopfield model

Associative memory



Prototype

$\vec{p}^1$



Prototype

$\vec{p}^2$

interactions

$$w_{ij} = \sum_{\mu} p_i^{\mu} p_j^{\mu}$$

Sum over all prototypes

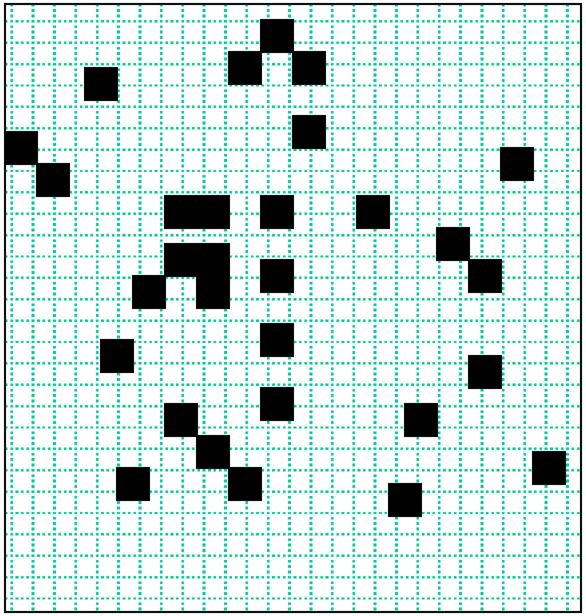
dynamics

$$S_i(t+1) = \mathrm{sgn}\left(\sum_j w_{ij} S_j\right.$$

Sum over all interactions with i

Hopfield model

Associative memory



Prototype

$\vec{p}^{1}$

interactions

$$w_{ij} = \sum_{\mu} p_i^{\mu} p_j^{\mu}$$

Sum over all prototypes

This rule is very good for **random** patterns

It does not work well for correlated patters

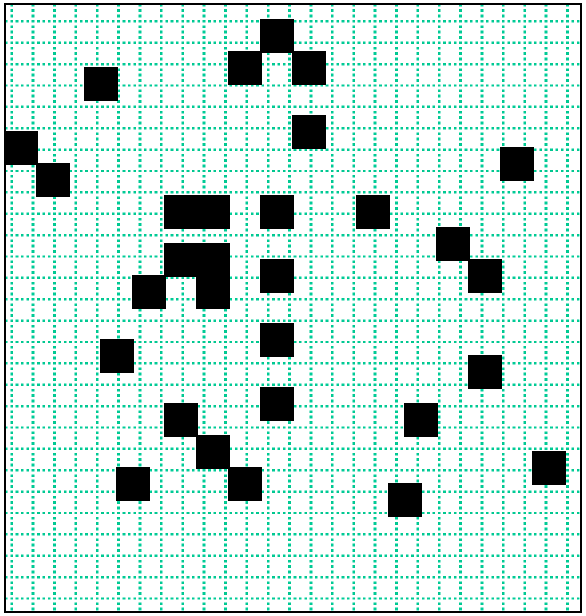
dynamics

$$S_i(t+1) = \mathrm{sgn}\left(\sum_j w_{ij} S_j\right)$$

Sum over all interactions with i

Hopfield model

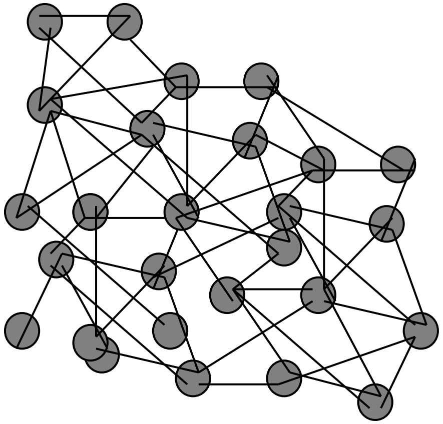
Associative memory



Prototype

$\vec{p}^1$

Interacting neurons



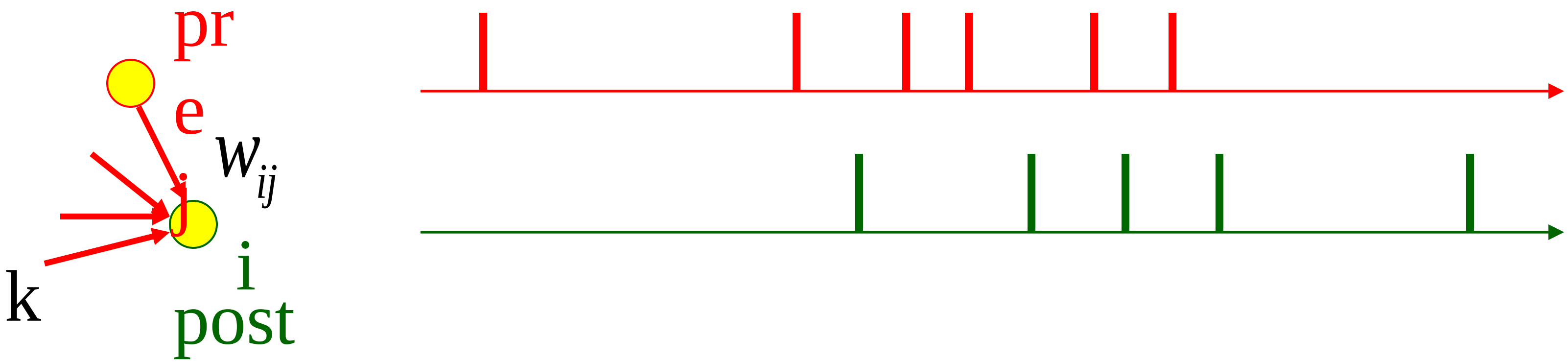
Finds the closest prototype
i.e. maximal overlap
(similarity) m^μ

Computation

- without CPU,
- without explicit memory unit

Hopfield model

Hebbian Learning

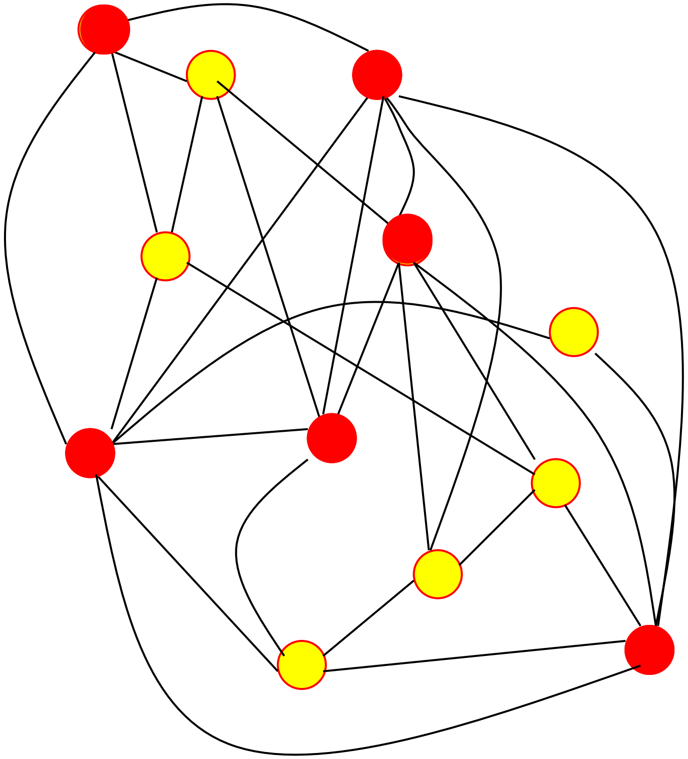


When an axon of cell **j** repeatedly or persistently takes part in firing cell **i**, then j's efficiency as one of the cells firing i is increased

Hebb, 1949

- local rule
- simultaneously active (correlations)

Hebbian Learning



Elementary pixel

🔴 $S_i = +1$

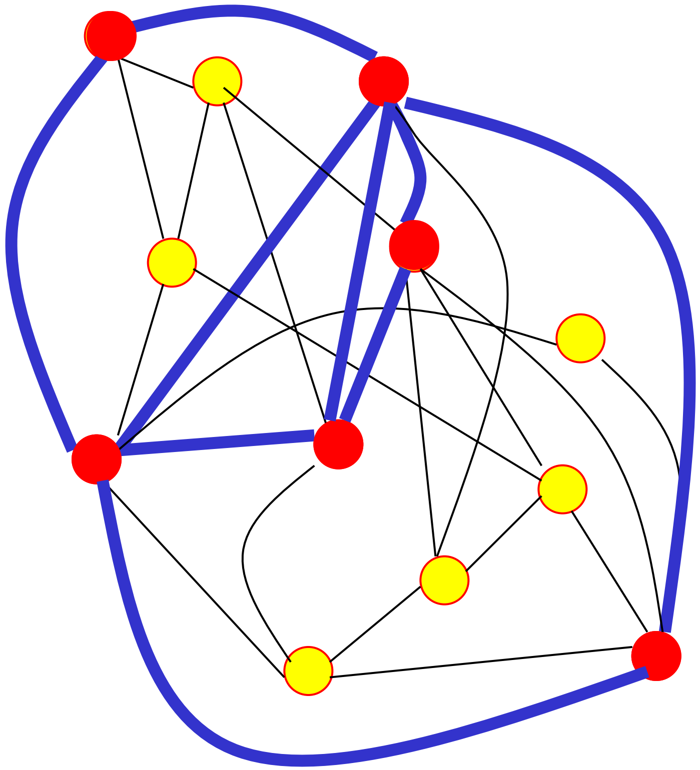
🟡 $S_i = -1$

🔴↔🔴 $w_{ij} = +1$

🟡↔🟡 $w_{ij} = +1$

🟡↔🔴 $w_{ij} = -1$

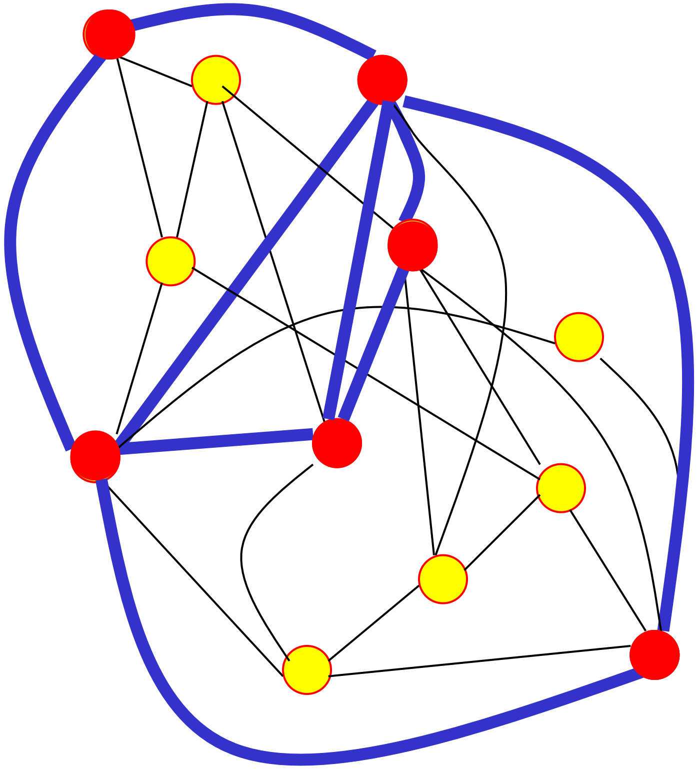
Hebbian Learning



item memorized

Hebbian Learning

Recall:
Partial info



item recalled

Lecture 14 – Population dynamics and associative memory; stable learning

-Introduction

✓ -Associative Memory

So far: neuron=spin=pixel=binary=on/off
BUT:
what about more realistic spiking neurons?

Lecture 14 – Population dynamics and associative memory; stable learning

-Introduction

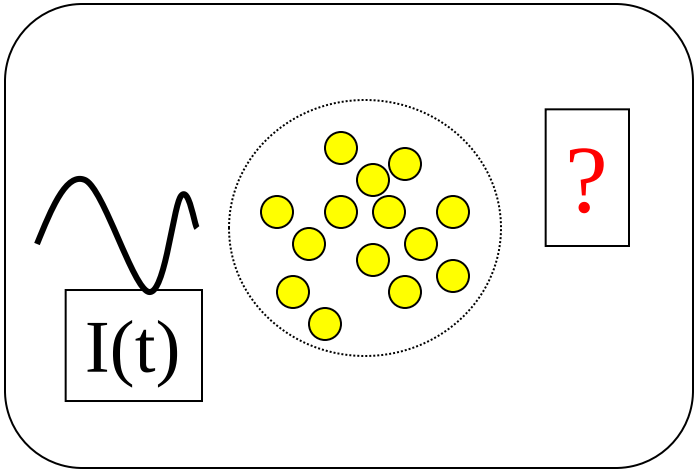
✓ -Associative Memory

→ -Dense networks (mean-field)

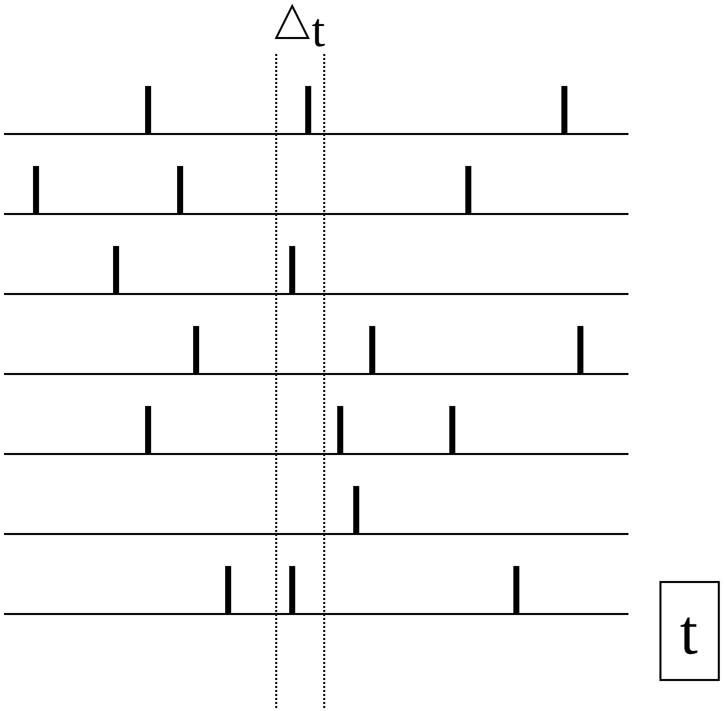
-Population dynamics and Associative Memory

-Stable learning

Populations of spiking neurons



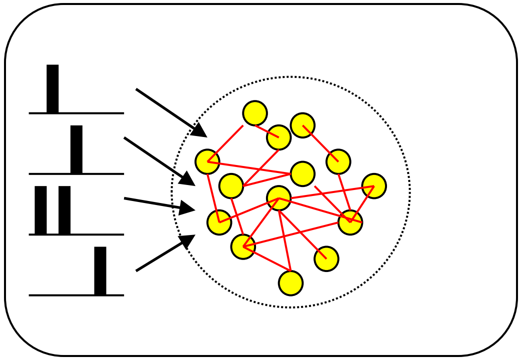
population dynamics?



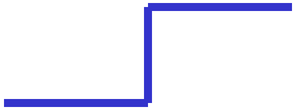
population activity

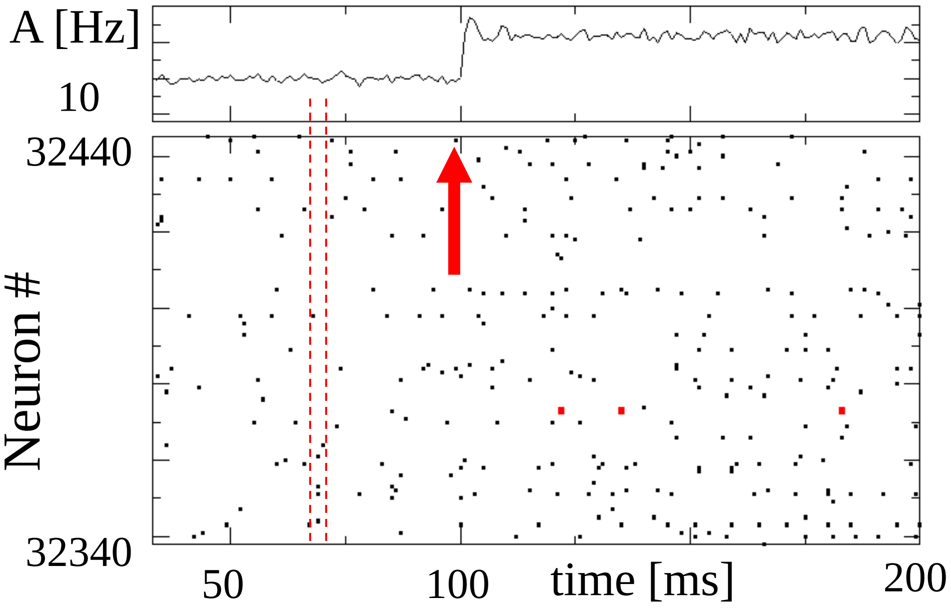
$$A(t) = \frac{n(t;t+\Delta t)}{N\Delta t}$$

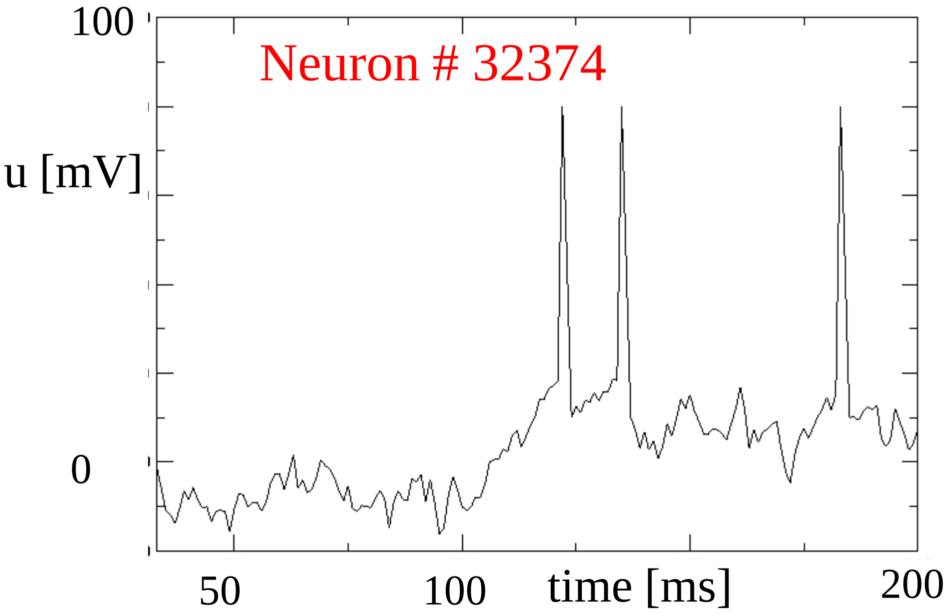
Activity in a populations of neurons



input {-low rate
-high rate



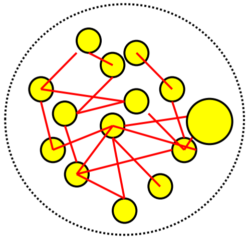




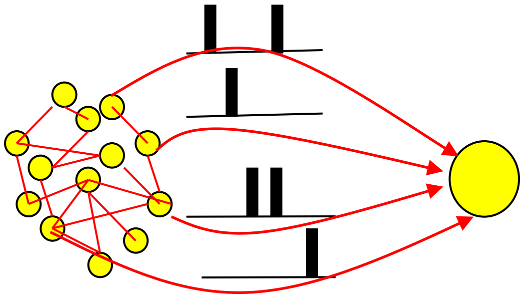
Population

- 50 000 neurons
- 20 percent inhibitory
- randomly connected

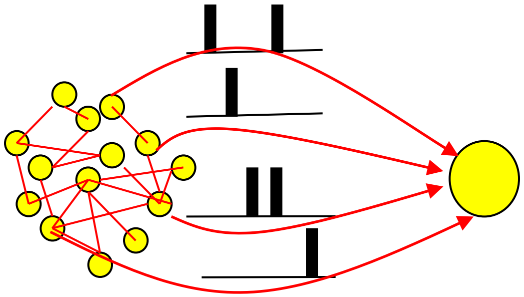
Homogeneous network (I&F)



Consider 1 neuron



Homogeneous network (I&F)



Assumption of Stochastic spike arrival:
network of exc. neurons,
total spike arrival rate *A(t)*

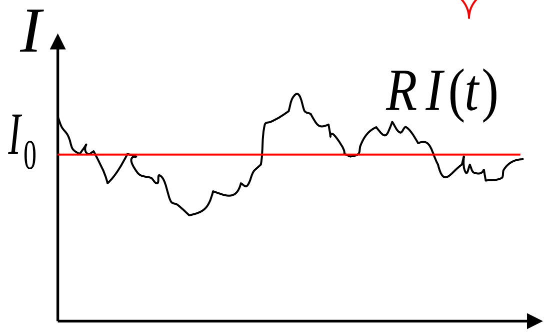
$$C\frac{d}{dt}u = -\sum_k I_k^{ion} \quad + I(t)$$

← *Membrane equation*

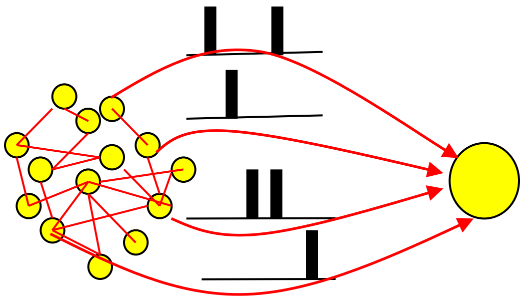
Synaptic current pulses of shape α

$$I(t) = \sum_k \frac{J_0}{N} \sum_f \underbrace{\alpha(t - t_k^f)}_{\text{EPSC}}$$

All synapses have identical weight



Homogeneous network (I&F)

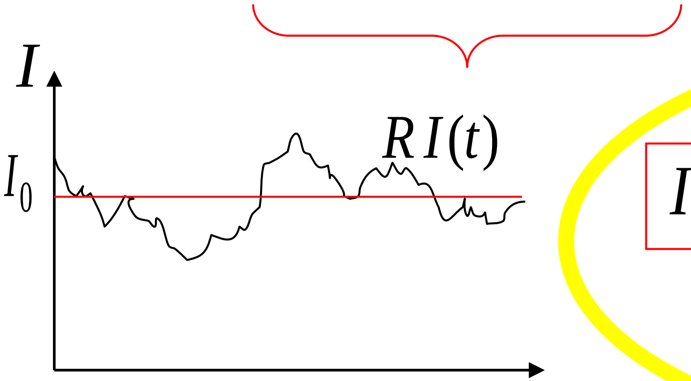


Assumption of Stochastic spike arrival:
network of exc. neurons,
total spike arrival rate *A(t)*

$$C\frac{d}{dt}u = -\sum_k I_k^{ion} \qquad + I(t)$$

← *membrane equation*

$$I(t) = I_o + I_{noise}$$

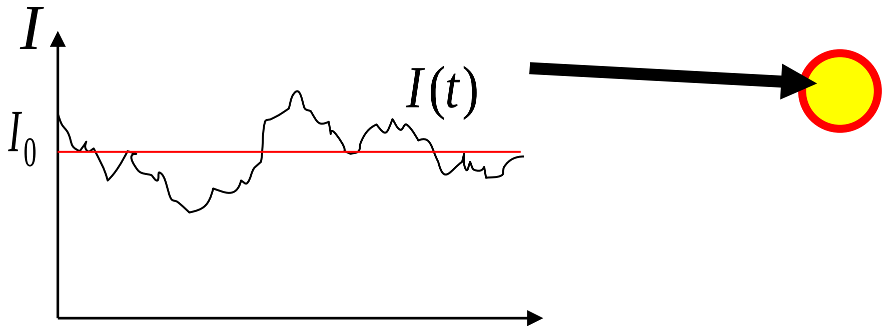


$$\boxed{I_0 = \gamma A_0}$$

Population activity

Analysis of Homogeneous Population

Step 1: Single neuron property: Inject noise current



Hodgkin-Huxley

$$C\frac{d}{dt}u = -\sum_k I_k^{ion} \quad + I(t)$$

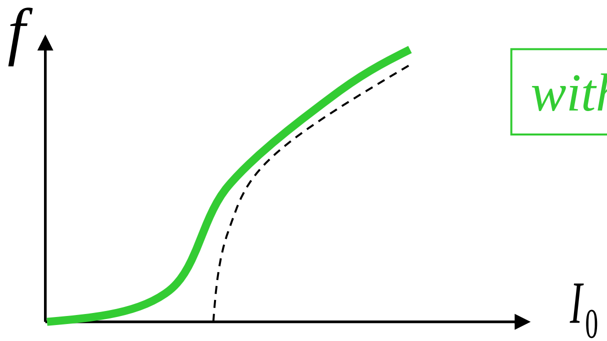
$$I(t) = I_o + I_{noise}$$

Leaky I&F

$$\tau \frac{d}{dt}u = -(u - u_{rest}) \quad + R\,I(t)$$

$$I(t) = I_o + I_{noise}$$

Measure frequency

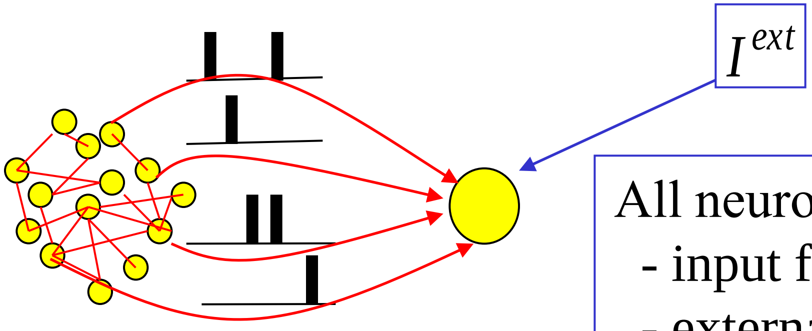


with noise

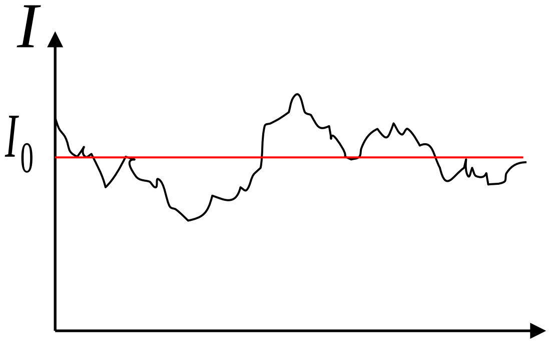
$$\boxed{f = g(I_0)}$$

Analysis of Homogeneous Population

Step 2 : consider 1 neuron in the network



All neurons receive the same input ('mean field'):
- input from network
- external input



$$I_0 = \gamma A_0 + I^{ext}$$

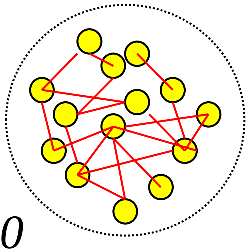
Population activity

Mean input from network prop. to population activity

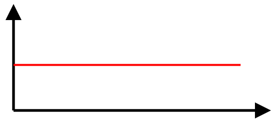
Step 3: assume Stationary State/Asynchronous State

$$\boxed{A_0 = f}$$

All neurons are the same



A(t)=const

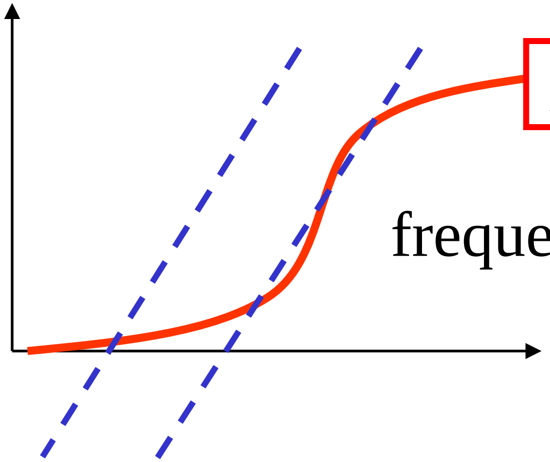


Step 4: close equation – calculate *A₀*

$$\boxed{A_0 = f = g(I_0) = g(\gamma A_0 + I^{ext})}$$

$$I_0 = \gamma A_0 + I^{ext}$$

$$\boxed{A_0 = \frac{1}{\gamma}\left[I_0 - I^{ext} \right]}$$



A_0

typical mean field
(Curie Weiss)

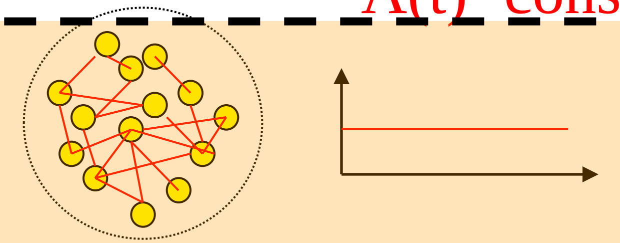
$$\boxed{f = g(I_0)}$$

frequency (single neuron)

Blackboard

Exercise (some time ago): find stationary state

A(t)=const

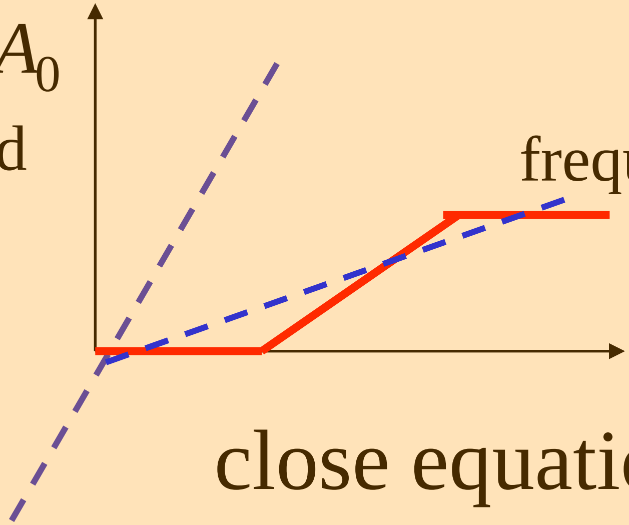


$$I_0 = \gamma A_0 + I^{ext}$$

$$\boxed{A_0 = \frac{1}{\gamma}\left[I_0 - I^{ext} \right]}$$

fully connected
coupling J₀/N

typical mean field



frequency (single neuron)

$$\boxed{f = g(I_0)}$$

close equations – calculate A_0

Lecture 14 – Population dynamics and associative memory; stable learning

-Introduction

-Associative Memory

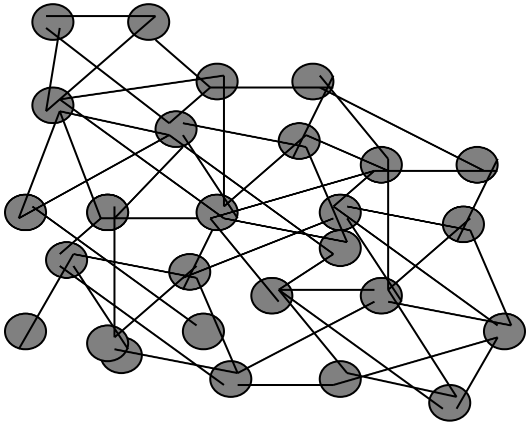
✓ -Dense networks (mean-field)

→ -Population dynamics and Associative Memory

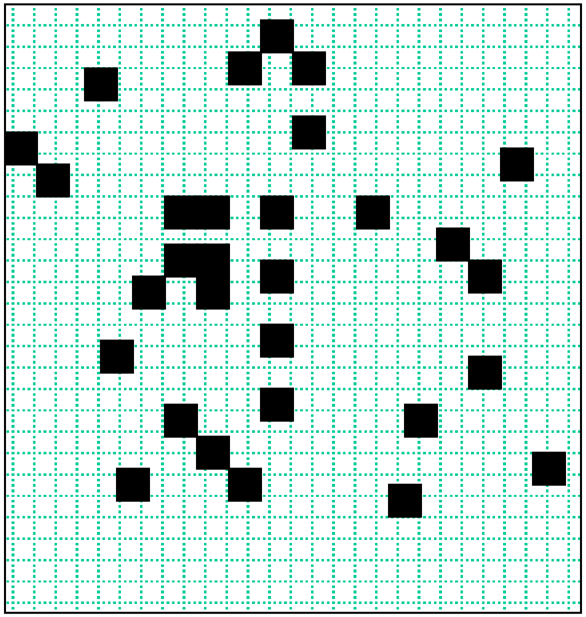
review

Back to Associative memory

Interacting neurons



-Possible with spiking neurons
-Calculation: mean-field
-Prototypes = random patterns



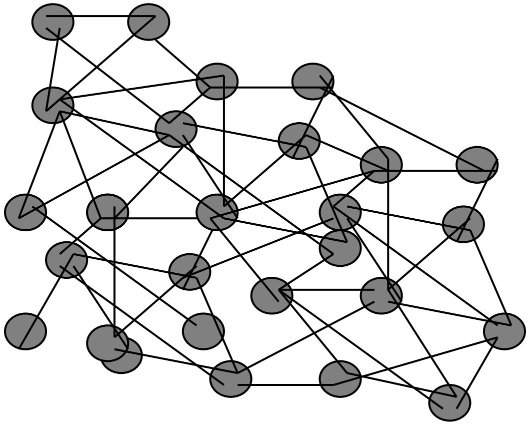
Computation

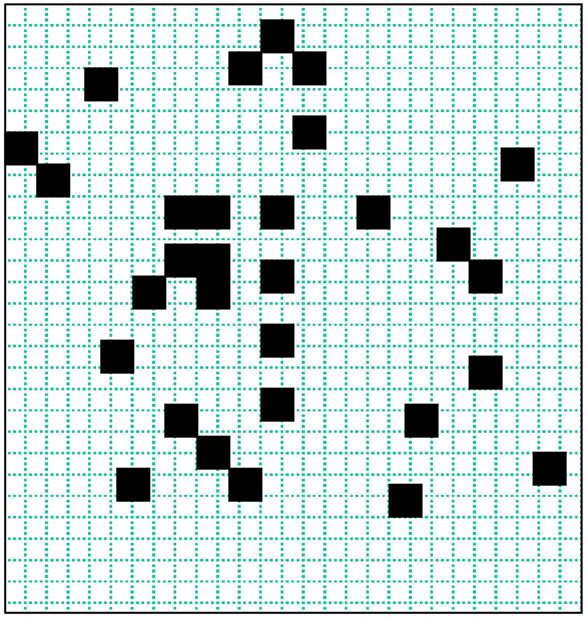
- without CPU,
- without explicit memory unit

Associative memory - simple model

Interacting neurons

-Rate model/population activity
-Calculation: mean-field
-Prototypes = random patterns





rate model

Single-neuron firing rate

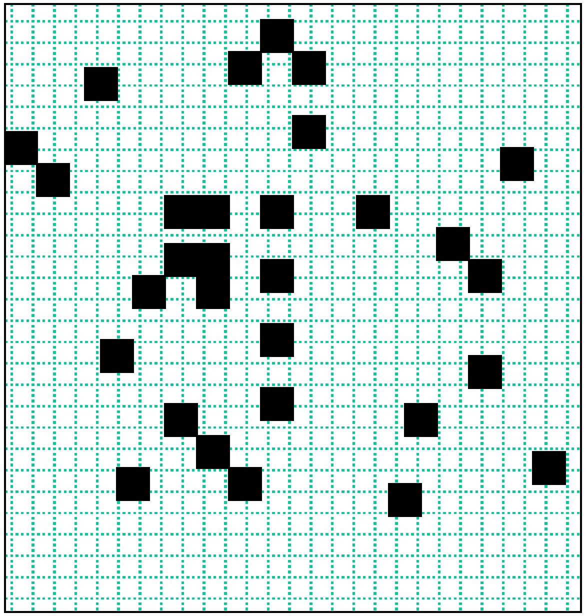
$$\boxed{f = g(I_0)}$$

$$\boxed{A_i = g(\sum_j w_{ij} A_j)}$$

population rate

$$w_{ij} = \frac{2}{N f^{\max}} \sum_\mu p_i^\mu p_j^\mu$$

Associative memory



For comparison with Spin Sj=+/-1

$$m^{\mu} = \tfrac{1}{N} \sum_j p_j^{\mu} S_j$$

Prototype $\vec{p}^1$

Task: Find the prototype with maximal overlap

$$\boxed{m^{\mu} = \frac{1}{N} \sum_j \underset{\uparrow}{p_j^{\mu}} \left(2 \frac{f_j}{f^{\max}} - 1\right)}$$

+/-1

Blackboard

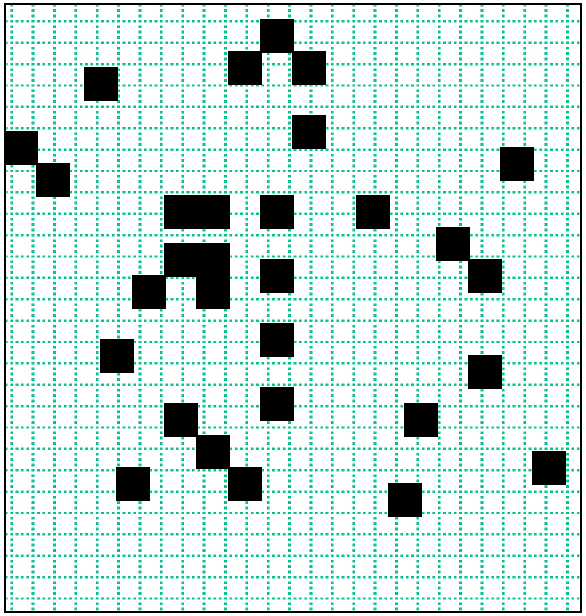
Associative memory

$\boxed{f = g(I_0)}$

Input current

$\boxed{f = g(h_0)}$

Input potential



Prototype

$\vec{p}^{1}$

$$\boxed{m^\mu = \frac{1}{N}\sum_j p_j^\mu \left(2\frac{f_j}{f^{\max}} - 1\right)}$$

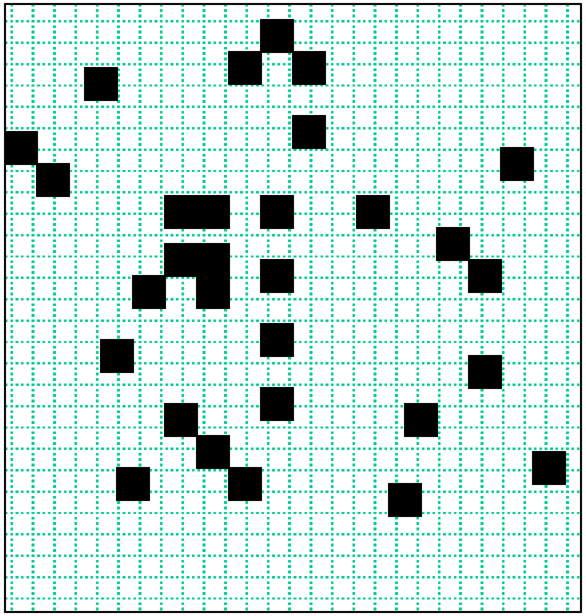
overlap

$$\boxed{m^\mu = \frac{2}{N f^{\max}}\sum_j p_j^\mu f_j - \cancel{\frac{1}{N}\sum_j p_j^\mu}}$$

+/-1

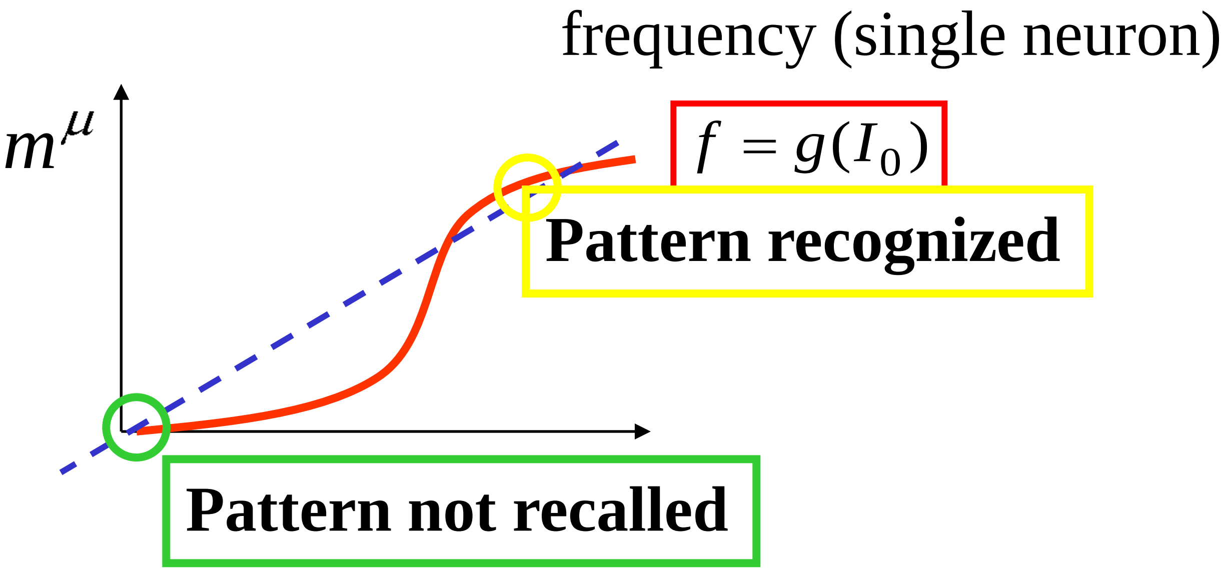
with prob. 0.5 +/-1

Associative memory – main idea



Prototype

$\vec{p}^1$



Pattern recognized

Pattern not recalled

Task: Find the prototype with maximal overlap

$$m^\mu = \frac{1}{N}\sum_j p_j^\mu \left(2\frac{f_j}{f^{\max}} - 1\right)$$

+/-1

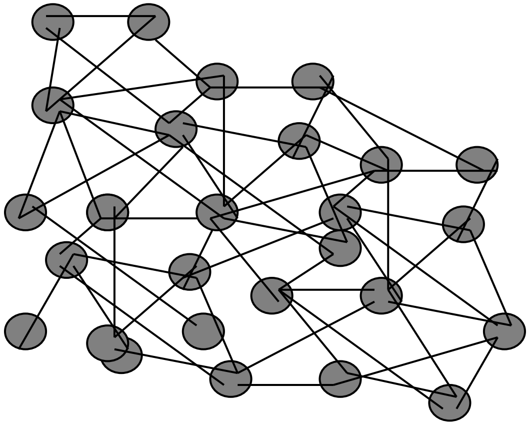
Conclusion - Associative memory

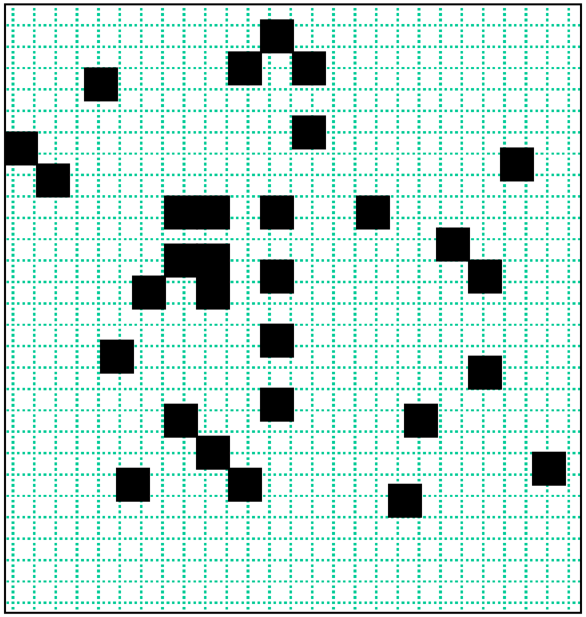
Interacting neurons

-Possible with spiking neurons

-Calculation: mean-field

-Prototypes = random patterns

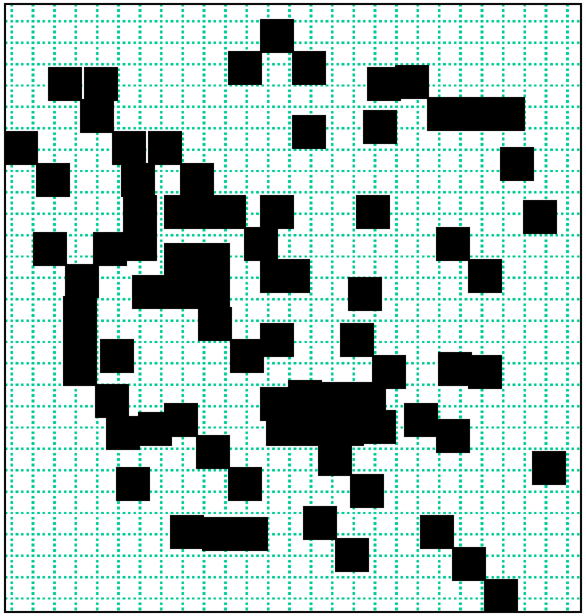


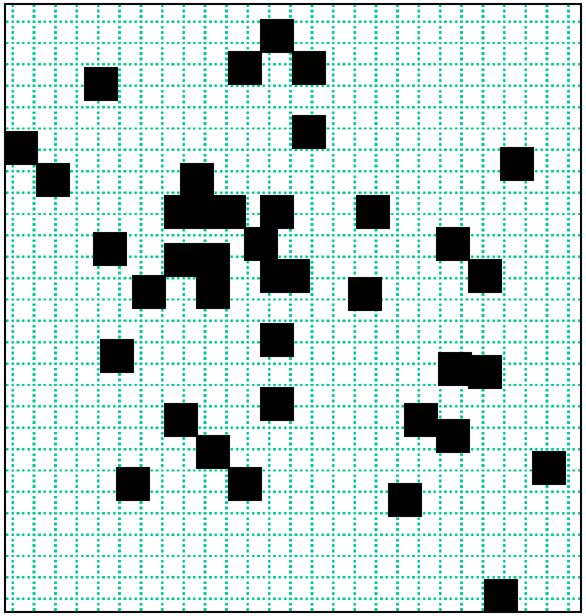


Computation

- without CPU,
- without explicit memory unit

Associative memory





interactions

$$w_{ij} = \sum_{\mu} p_i^{\mu} p_j^{\mu}$$

Sum over all prototypes

This rule is very good for **random** Patterns (a=0.5)

It does not work well for correlated patters

for low activity **random** Patterns (e.g., a=0.1)

$$w_{ij} = \sum_{\mu} (p_i^{\mu} - c)(p_j^{\mu} - a)$$

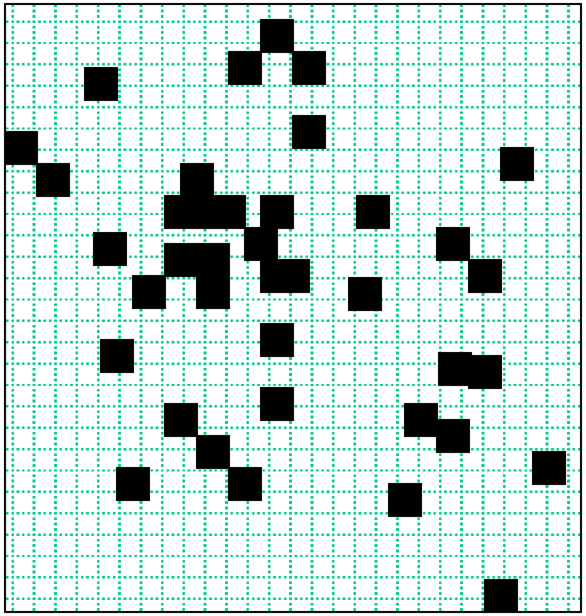
e.g., in each pattern exactly 10 percent of neurons are active

Associative memory

$$w_{ij} = \frac{1}{N} K(\underset{\uparrow}{\vec{p}_i}, \underset{\uparrow}{\vec{p}_j})$$

Locally stored information

Blackboard



for low activity **random** Patterns (e.g., a=0.1)

$$w_{ij} = \sum_{\mu} (p_i^{\mu} - c)(p_j^{\mu} - a)$$

e.g., in each pattern exactly 10 percent of neurons are active

Associative memory

Blackboar

$$w_{ij}=\frac{1}{N}K(\underset{\uparrow}{\vec{p}_i}\,,\underset{\uparrow}{\vec{p}_j})$$

Locally stored information

$$w_{ij}=\sum_{\mu}(p_i^{\mu}-c)(p_j^{\mu}-a)$$

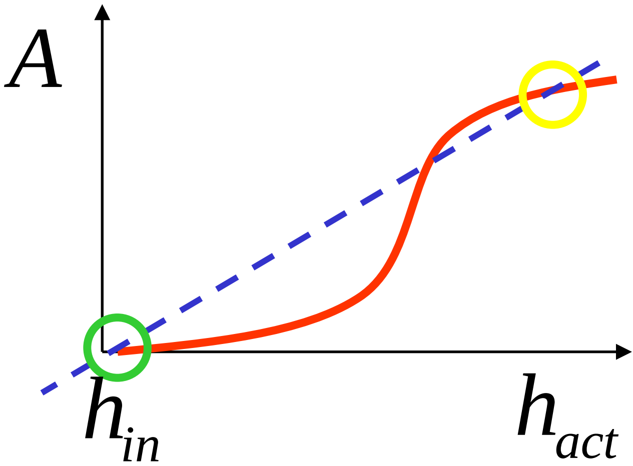
e.g., a=0.1 means in each pattern
exactly 10 percent of neurons are active

$$h(\vec{z})=\sum_{\vec{z}\,'}\frac{|L(\vec{z})|}{N}K(\vec{z},\vec{z}\,')\int\varepsilon(s)A(\vec{z}\,',t-s)ds$$

$$\boxed{A(\vec{z}\,')=g(h(\vec{z}\,'))}$$

Associative memory –
For a given pattern only 2 populations!

frequency (single neuron)



$\boxed{f = g(h_0)}$

Blackboard

$$h(\vec{z}) = \sum_{\vec{z}'} \frac{|L(\vec{z})|}{N} K(\vec{z}, \vec{z}') \int \varepsilon(s) A(\vec{z}', t-s) ds$$

$$w_{ij} = \frac{1}{N} K(\vec{p}_i, \vec{p}_j) = \frac{1}{N} \sum_{\mu} (p_i^{\mu} - c)(p_j^{\mu} - a)$$

$$\boxed{A(\vec{z}') = g(h(\vec{z}'))}$$

Lecture 14 – Population dynamics and associative memory; stable learning

-Introduction

-Associative Memory

-Dense networks (mean-field)

✓ -Population dynamics and Associative Memory

→ -Final discussion

Nearly the end:
what can I improve for the students next year?

→ Integrated exercises?

→ Miniproject?

→ Overall workload ?(4 credit course = 6hrs per week)

→ Background/Prerequisites?

-Physics students
-SV students
-Math students

Exam:

- -written exam, 23.6 from 8:15-11:15
- - miniprojects counts 1/3 towards final grade

For written exam:

-bring 1 page A5 of own **handwritten** notes

The end

The end

Exercise now: Associative memory

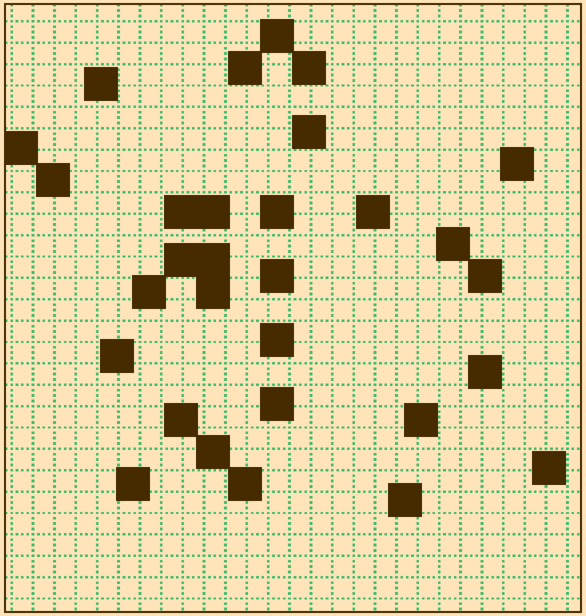
$$\boxed{f = g(I_0)}$$

$$f_i = g\left(\sum_j w_{ij} f_j\right)$$

$$w_{ij} = \frac{2}{N f^{\max}} \sum_\mu p_i^\mu p_j^\mu$$

$$\boxed{f_i = g\left(\sum_\mu p_i^\mu m^\mu\right)}$$

$$m^\mu = \frac{1}{N} \sum_j p_j^\mu \left(2 \frac{f_j}{f^{\max}} - 1\right)$$



Prototype

$\vec{p}^1$

Assume 4 patterns. At time t=0, overlap with Pattern 3, no overlap with other patterns.

discuss temporal evolution

(assume that patterns are orthogonal)

The end