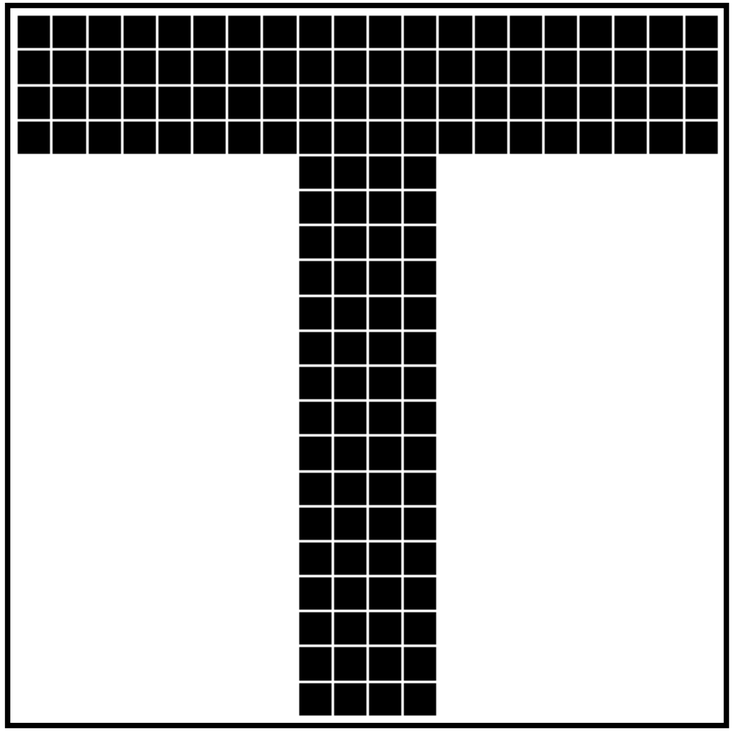
Large Associative Memory Problem
in Neurobiology & Machine Learning

"Tony" Runzhe Yang

Sept. 1, 2020

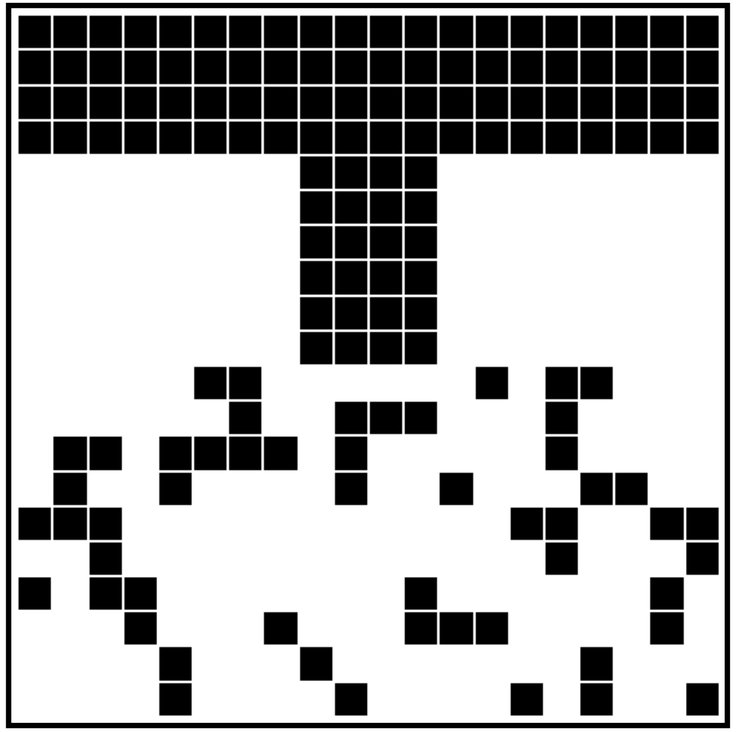
https://runzhe-yang.science

Associative memories in psychology: ability to remember sets of unrelated items



character “T”

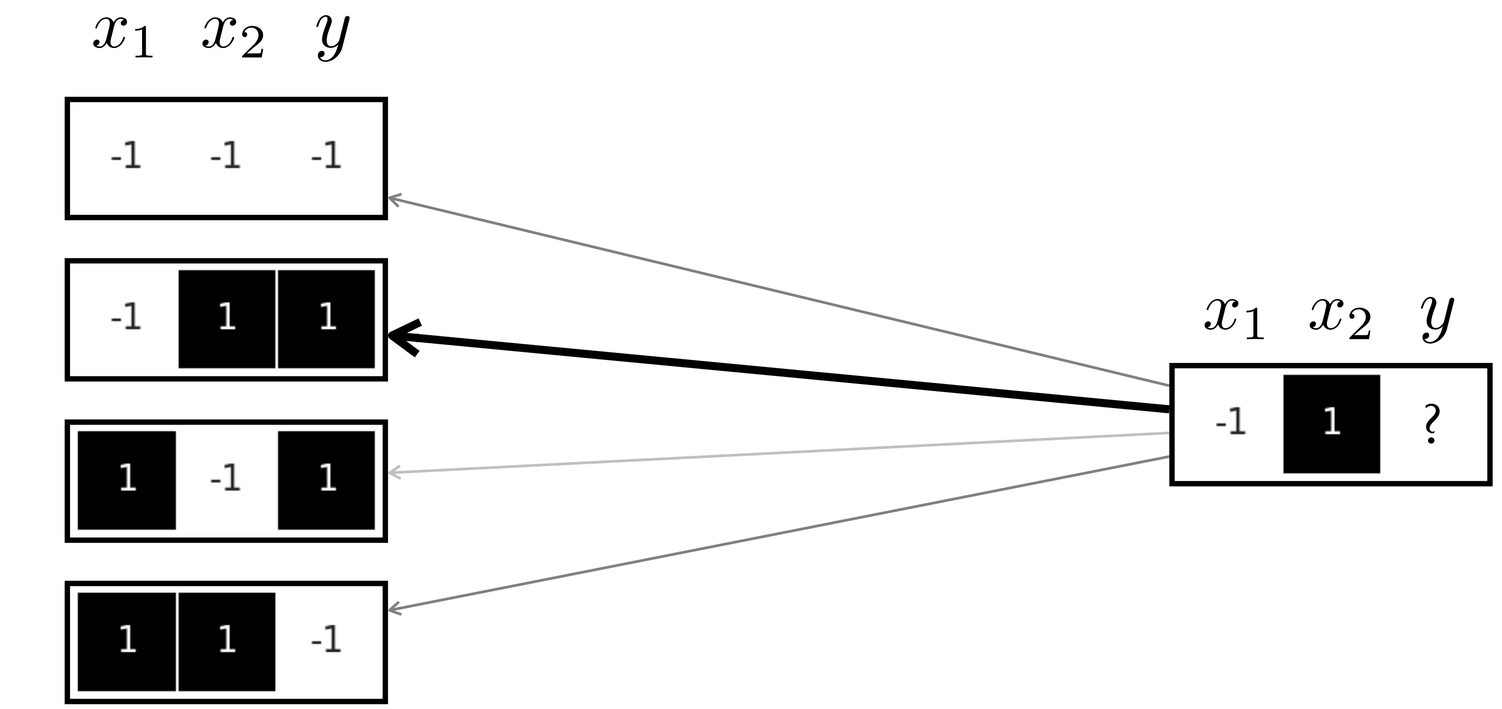
memory



corrupted “T”

stimulus

Pattern recognition as operation of associative memories



exclusive-or rules

memory

logical query

stimulus

Language models as operation of associative memories

large corpus

memory

The doctor ran to the emergency room to see [MASK] patient.

↓

BERT Predictions

44.3%	his/her
36.9%	the
8.1%	another
7.3%	a

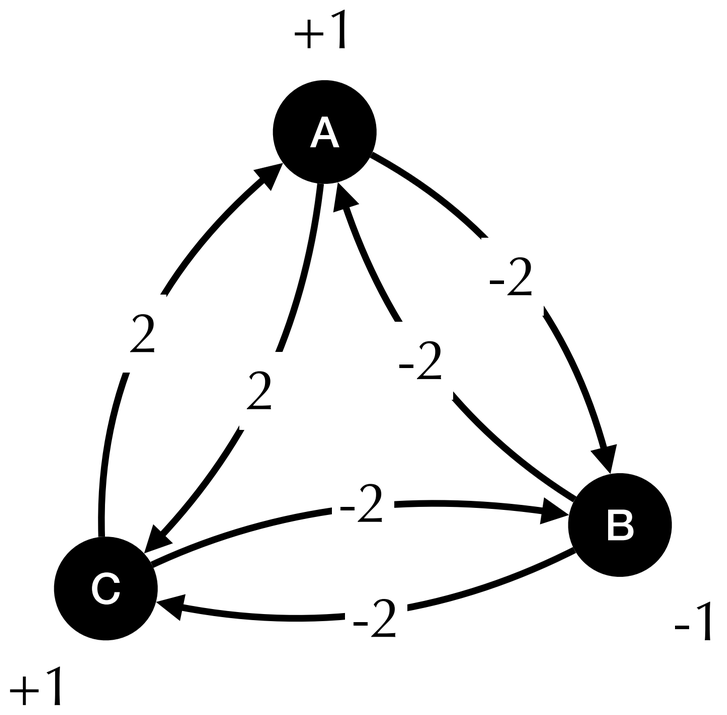
masked sentence

stimulus

Part I. Classic Hopfield Networks

How are memories stored and retrieved?

Hopfield Networks: a mathematical model abstracted from neurobiology



A Hopfield net with 3 neurons

- *N* binary neurons like McCullough-Pitts, and the **state** of neuron *i* is

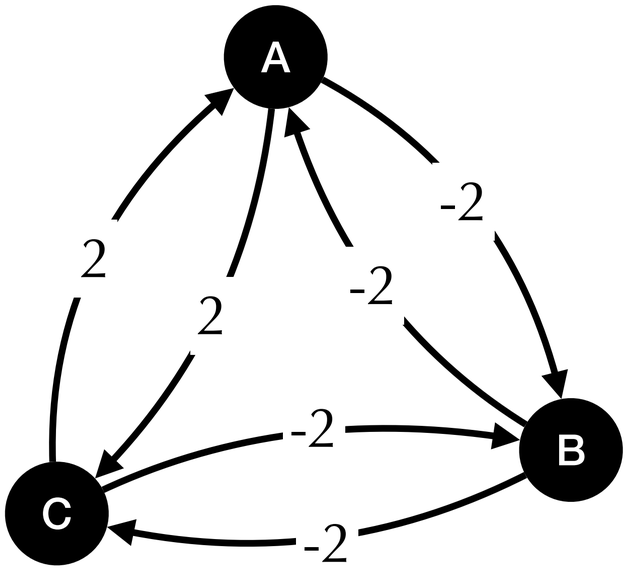
$$\begin{cases} \sigma_i = 1 & \text{(firing at maximum rate)} \\ \sigma_i = -1 & \text{(not firing)} \end{cases}$$

- The strength of **synaptic connection** from neuron *j* to neuron *i* is T_{ij}

$$T_{ii} = 0 \quad \text{and} \quad T_{ij} = T_{ji}$$

(no autapses) (symmetric reciprocal connections)

Training a Hopfield Network: how to store memories?



e.g., $\begin{cases} \boldsymbol{\xi}^1 = (1 \quad -1 \quad 1)^\intercal \\ \boldsymbol{\xi}^2 = (-1 \quad 1 \quad -1)^\intercal \end{cases}$

- Suppose we want to store K sets of neural states $\boldsymbol{\xi}^\mu \in \{-1, 1\}^N$ for $\mu = 1 \ldots K$. The **memories** are encoded by connectivities:

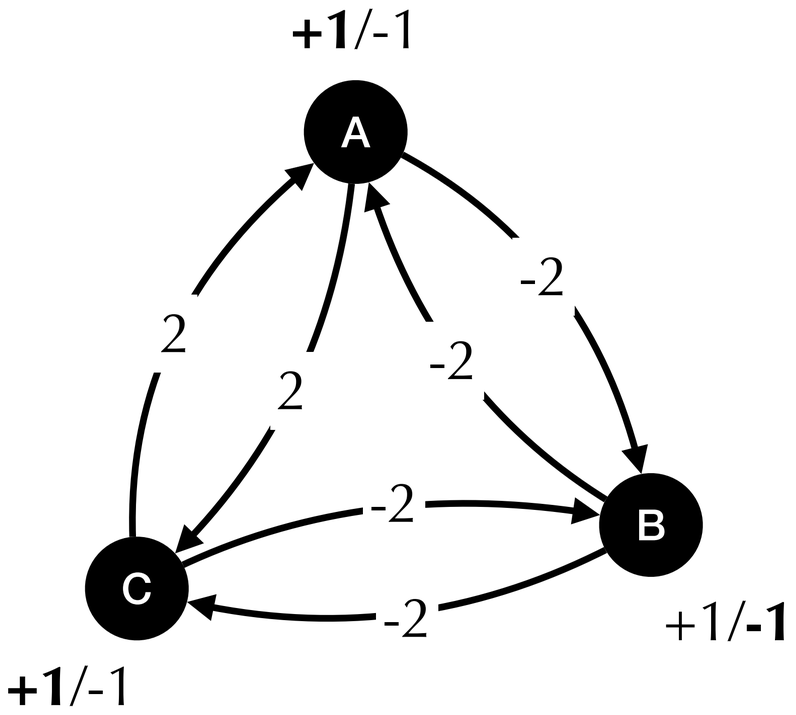
$$T_{ij} = \sum_{\mu=1}^{K} \xi_i^\mu \xi_j^\mu$$

- It is also related to Hebb's rule

$$\Delta T_{ij} \propto \langle \sigma_i(t) \sigma_j(t) \rangle_t$$

modification driven by **local correlation**

Updating a Hopfield Network: how to recall memories?



e.g., $\begin{cases} \boldsymbol{\xi}^1 = (1 \quad -1 \quad 1)^\intercal \\ \boldsymbol{\xi}^2 = (-1 \quad 1 \quad -1)^\intercal \end{cases}$

- All neurons are both inputs and outputs. **recurrent network & persistent activities.**
- Neural activity **dynamics**

$$\begin{matrix} \sigma_i \leftarrow 1 \\ \sigma_i \leftarrow -1 \end{matrix} \quad \text{if} \quad \sum_j T_{ij}\sigma_j \quad \begin{matrix} \geq 0 \\ < 0 \end{matrix}$$

  update neurons in a random order until converge.

- Which memories to recall? depends on 1) **initial state of neurons** and 2) **the order of update (random noise)**

Lyapunov (Energy) function of the neural dynamics

- Neural activity **dynamics**

$$\begin{array}{l}\sigma_i \leftarrow 1 \\ \sigma_i \leftarrow -1\end{array} \quad \text{if} \quad \sum_j T_{ij}\sigma_j \quad \begin{array}{l}\geq 0 \\ < 0\end{array}$$

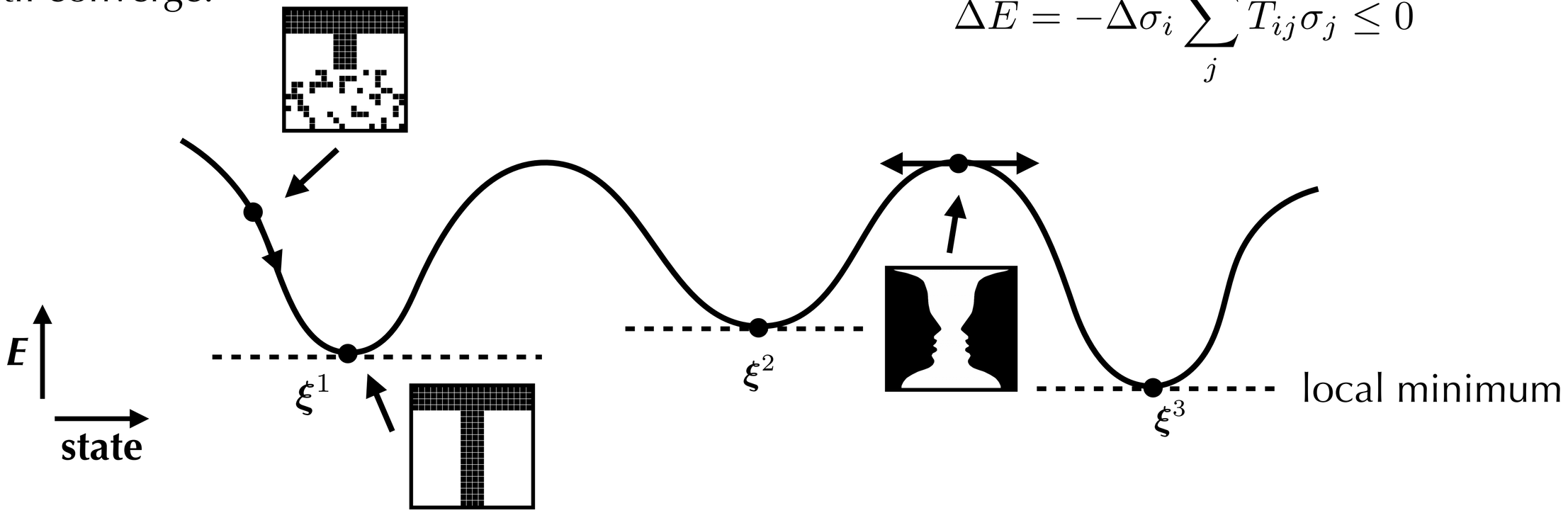
update neurons in a random order until converge.

- Quadratic Energy Function

$$E = -\sum_{ij} \sigma_i T_{ij} \sigma_j = -\sum_{\mu=1}^{K} \langle \boldsymbol{\xi}^\mu, \boldsymbol{\sigma} \rangle^2$$

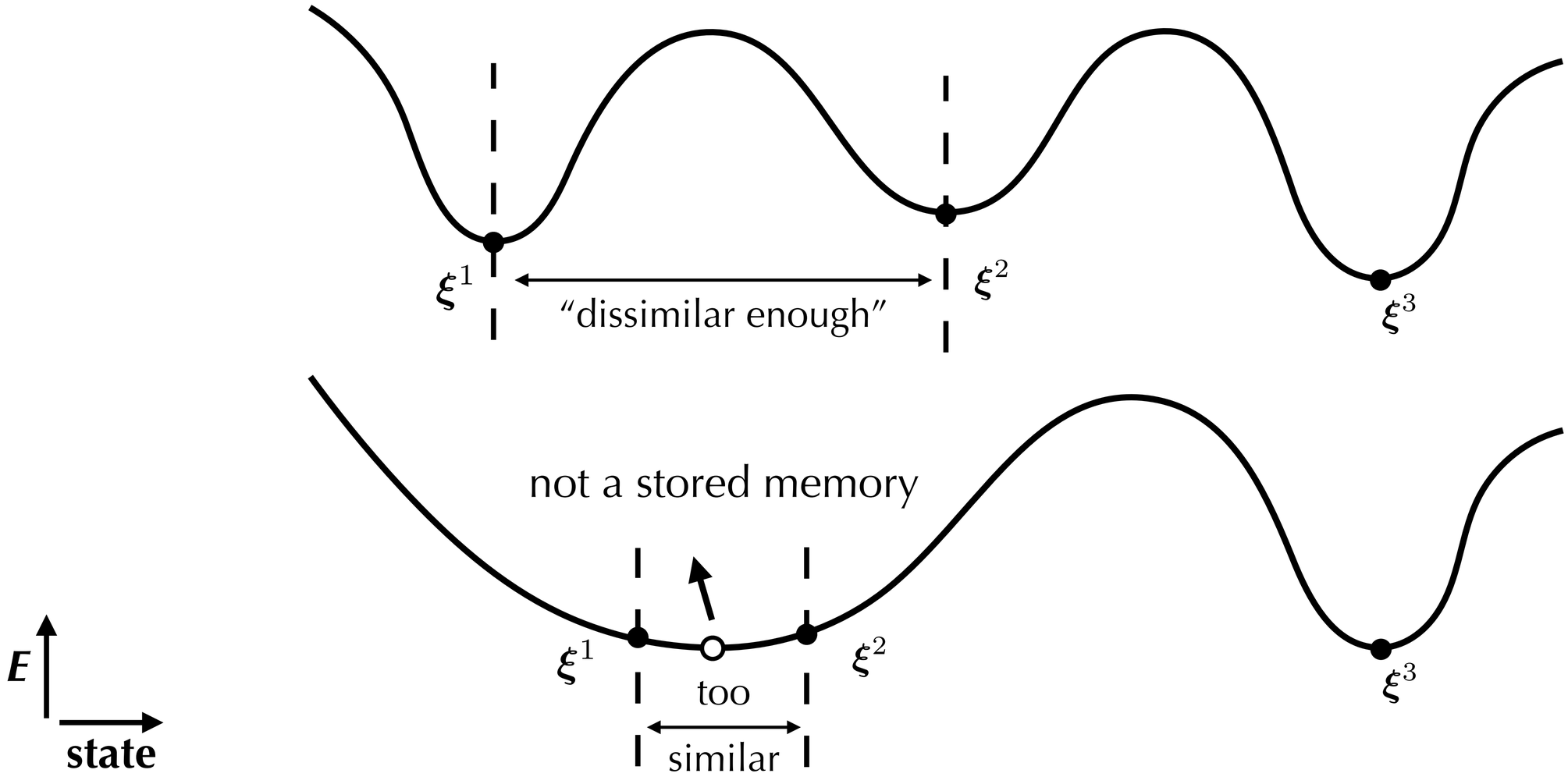
- Energy monotonically decreases

$$\Delta E = -\Delta\sigma_i \sum_j T_{ij}\sigma_j \leq 0$$

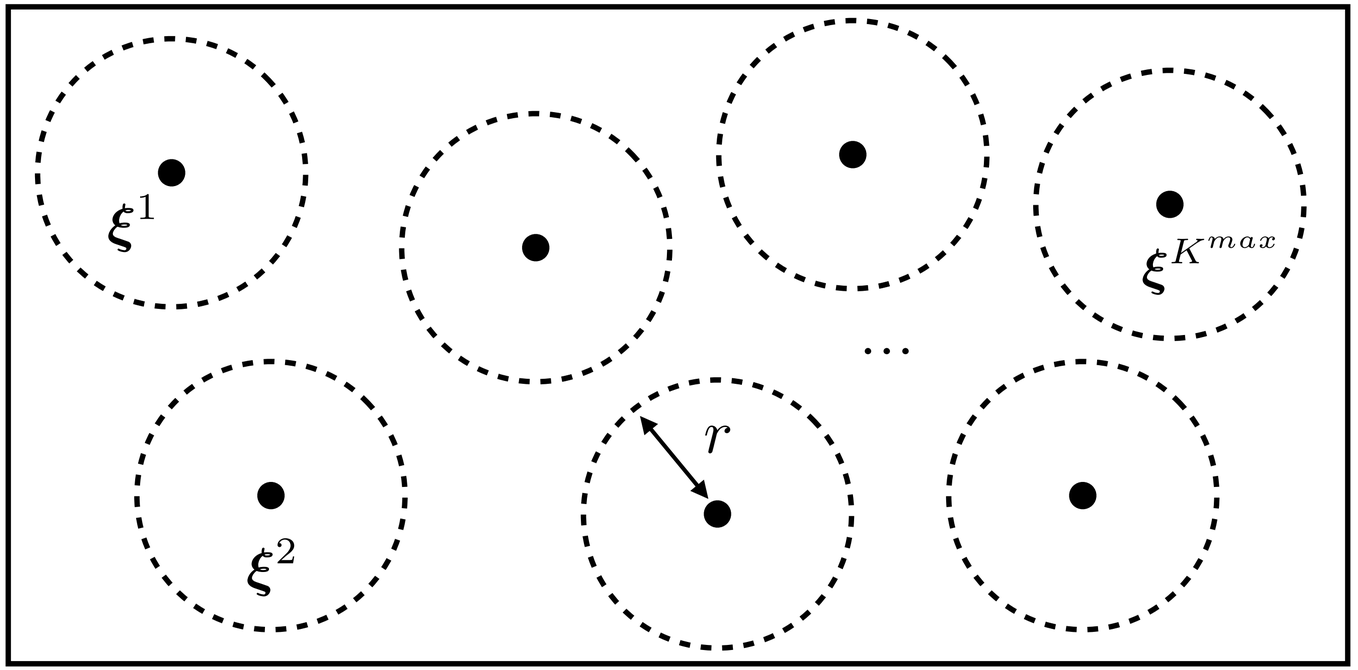


Network capacity: how many memories a Hopfield net can store?

$$E = -\sum_{ij} \sigma_i T_{ij} \sigma_j = -\sum_{\mu=1}^{K} \langle \boldsymbol{\xi}^\mu, \boldsymbol{\sigma} \rangle^2$$

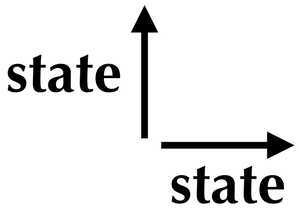


Network capacity: how many memories a Hopfield net can store?



“social distancing” for random reliable memories

$r = 1/2$ “safe distance”



$K^{max} \approx 0.138N$ (Hopfield, J.J., 1982)

$K^{max} = N$ (Kanter, I. and Sompolinsky, H., 1987)

Part II. Dense Associative Memory

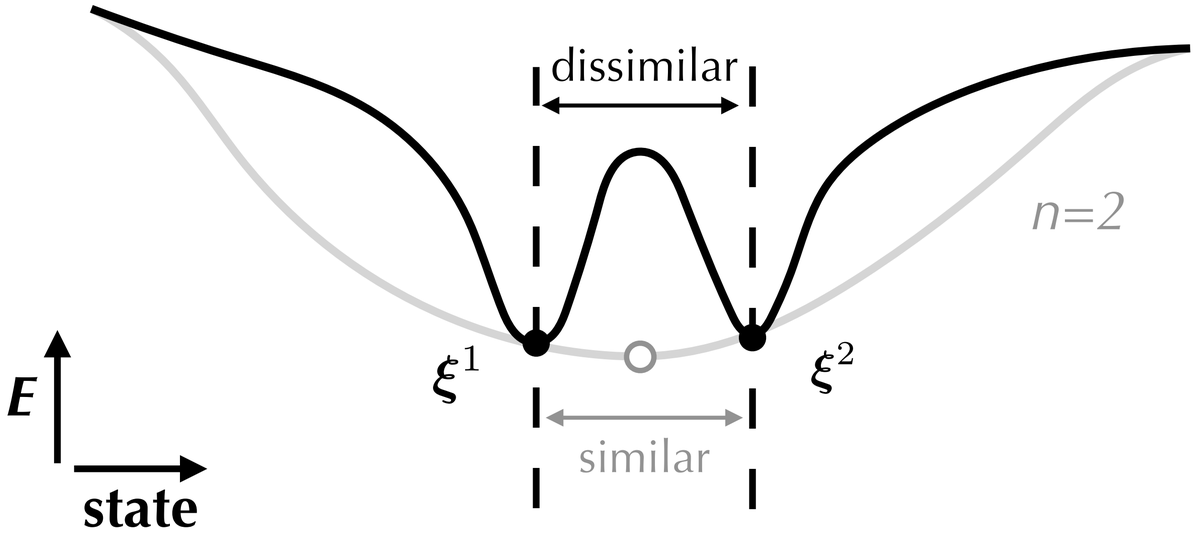
Can we increase the memory storage capacity?

Dense associative memory: dramatically increase the memory storage capacity

- Hopfield Energy Function

$$E = -\sum_{ij} \sigma_i T_{ij} \sigma_j = -\sum_{\mu=1}^{K} \langle \boldsymbol{\xi}^\mu, \boldsymbol{\sigma} \rangle^2$$

n>2: shaper energy landscape



- Polynomial Energy Function

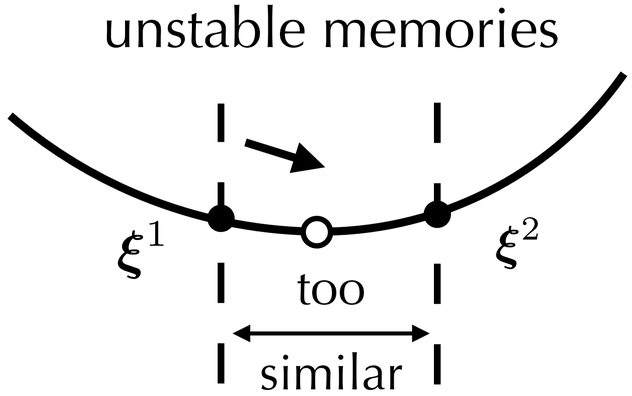
$$E = -\sum_{\mu=1}^{K} F_n(\langle \boldsymbol{\xi}^\mu, \boldsymbol{\sigma} \rangle)$$

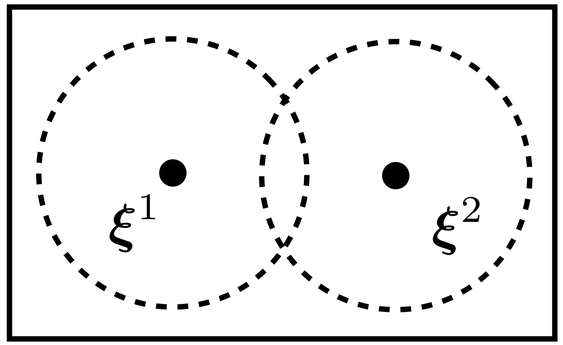
where *Fn* is the rectified polynomial

$$F_n(x) = \begin{cases} x^n, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

n=2 is the original Hopfield nets

Memory capacity of the polynomial energy function





random memories

$$K^{max} = \alpha_n N^{n-1}$$

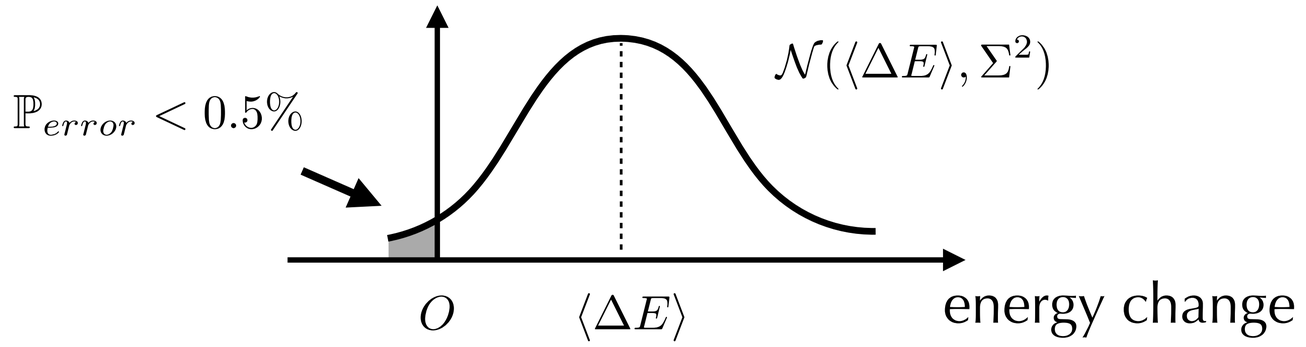
- Perturb one neuron *i*, from the initial state at memory $\boldsymbol{\xi}^\mu$

$$\Delta E = \sum_{\nu=1}^{K} (\langle \boldsymbol{\xi}^\mu, \boldsymbol{\xi}^\nu \rangle)^n - \sum_{\nu=1}^{K} (\langle \boldsymbol{\xi}^\mu_{-i}, \boldsymbol{\xi}^\nu \rangle)^n$$

- Mean fluctuation: $\langle \Delta E \rangle = N^n - (N-2)^n$

  Variance: $\Sigma^2 = 4n^2(2n-3)!!(K-1)N^n - 1$

- Memory is unstable when the fluctuation can be negative



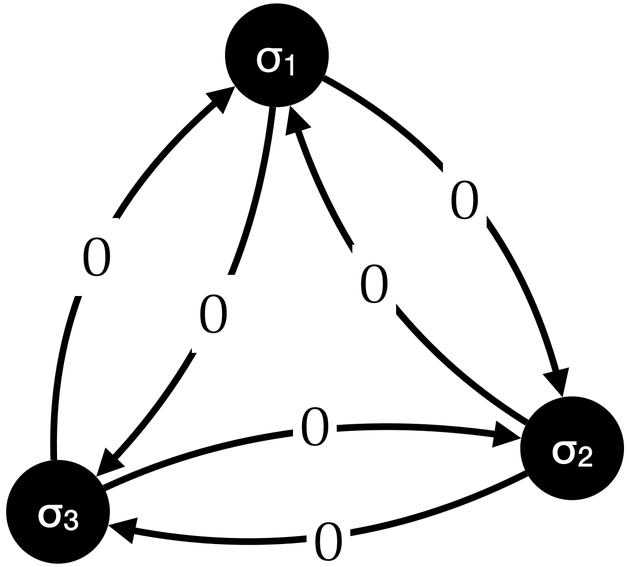
(Krotov, D. and Hopfield, J.J., 2016)

The case of XOR problem:

	σ_1	σ_2	σ_3
ξ^1	-1	-1	-1
ξ^2	-1	1	1
ξ^3	1	-1	1
ξ^4	1	1	-1

exclusive-or rules

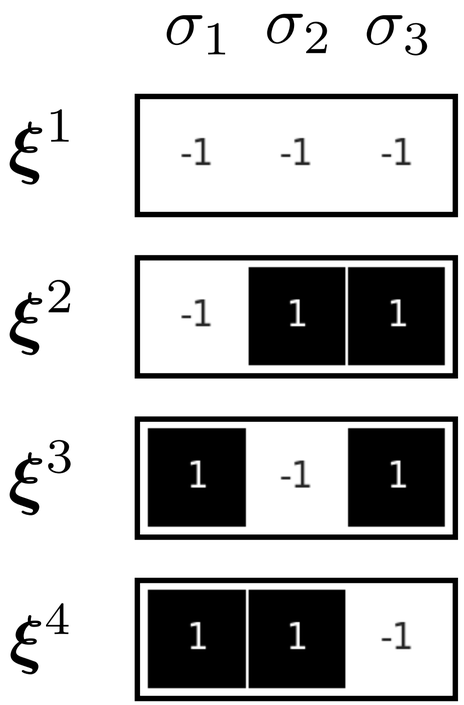
memory *K>N*



$$T_{ij} = \sum_{\mu=1}^{K} \xi_i^\mu \xi_j^\mu$$

3-neuron Hopfield net
cannot recall any XOR rules

The case of XOR problem:



exclusive-or rules

memory *K>N*

- Polynomial Energy Function

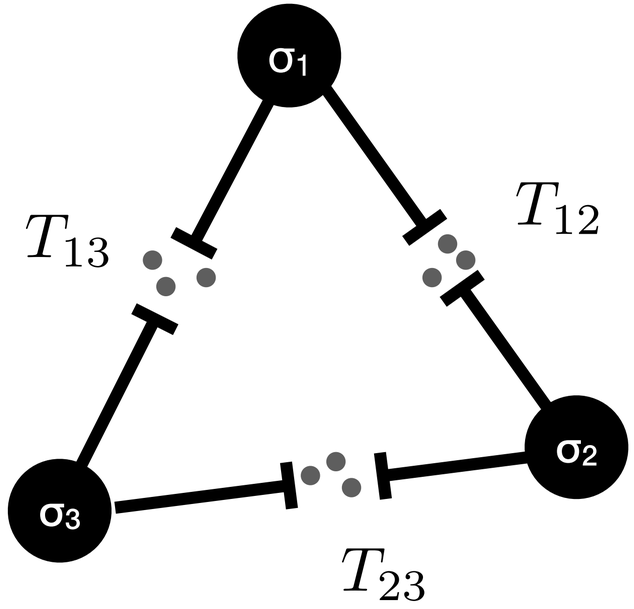
$$E_n = -\sum_{\mu=1}^{K} F_n(\langle \boldsymbol{\xi}^\mu, \boldsymbol{\sigma} \rangle)$$

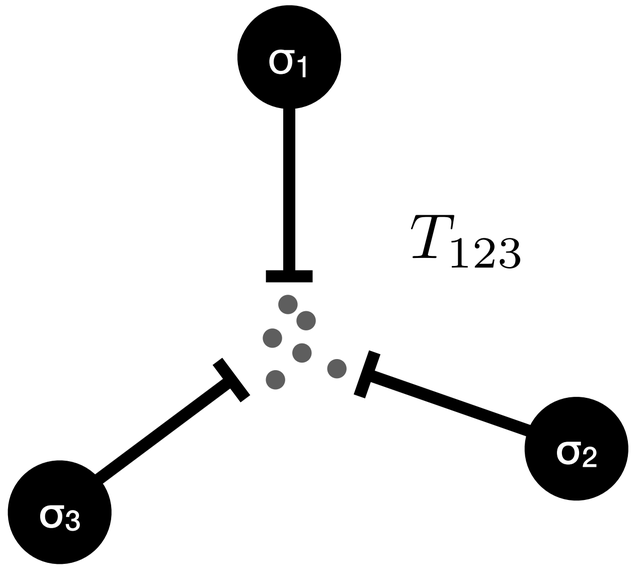
$\Rightarrow$

$$E_n(\boldsymbol{\sigma}) = \begin{cases} 0, & n = 1 \\ C_n, & n = 2, 4, 6, \dots \\ C_n \sigma_1 \sigma_2 \sigma_3, & n = 3, 5, 7, \dots \end{cases}$$

Dense associative memory network is capable of solving the XOR problem for higher odd values of *n*.

Dense associative memory nets allow higher-order interaction





(non-biological synapses)

$$E = -\sum_{i,j} \sigma_i T_{ij} \sigma_j = -\sum_{\mu=1}^{K} F_2(\langle \boldsymbol{\xi}^\mu, \boldsymbol{\sigma} \rangle)$$

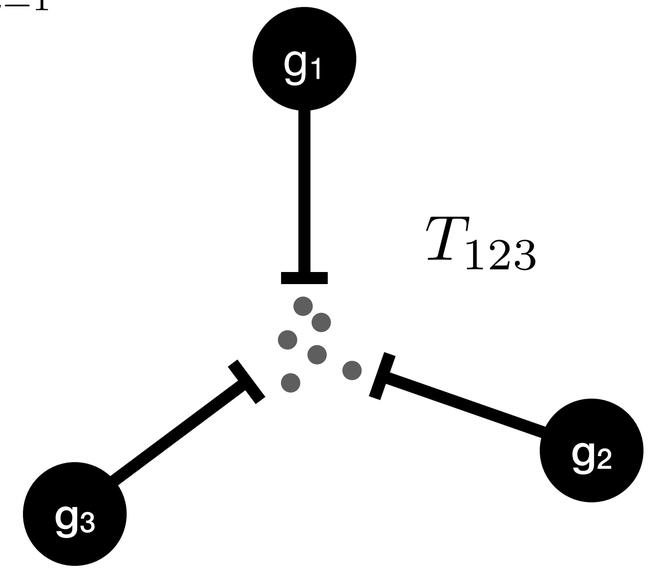
$$E = -\sum_{i,j,k} T_{ijk} \sigma_i \sigma_j \sigma_k = -\sum_{\mu=1}^{K} F_3(\langle \boldsymbol{\xi}^\mu, \boldsymbol{\sigma} \rangle)$$

Part III. Hidden Neuron Models

How to avoid many-cell synapses?

Effective network: *N* feature neurons + *K* hidden memory neurons

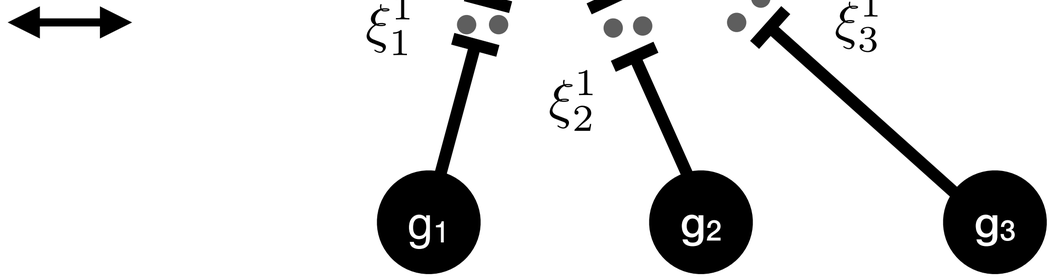
$$E = -\sum_{\mu=1}^{K} F(\langle \boldsymbol{\xi}^{\mu}, \boldsymbol{\sigma} \rangle)$$



(Krotov, D. and Hopfield, J.J., 2016)

(Demircigil et al, 2017: exponential *F*)

(Ramsauer et al, 2020:
Hopfield Networks is All You Need)



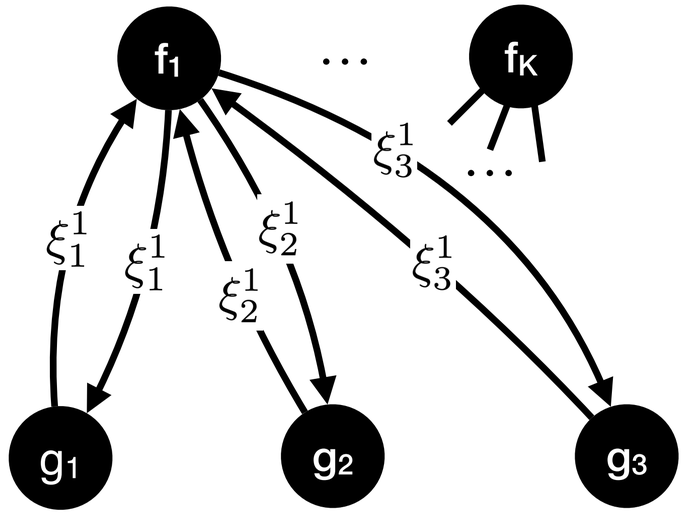
K hidden neurons

N feature neurons

(Krotov, D. and Hopfield, J.J., 2020)

RBM-like networks as general associative memory models

K hidden neurons



N feature neurons

(Krotov, D. and Hopfield, J.J., 2020)

- Extended to **continuous state** and **continuous time**
- Large memory capacity, and bio-plausible.
- Neural activity **dynamics**

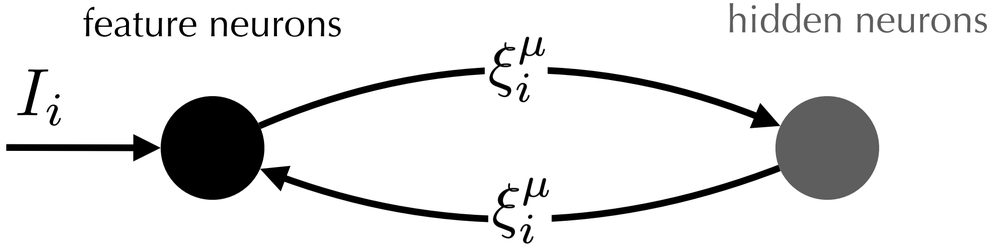
$$\begin{cases} \tau_v \dfrac{dv_i}{dt} = \displaystyle\sum_{\mu=1}^{K} \xi_i^\mu f_\mu - v_i + I_i \\ \tau_h \dfrac{dh_\mu}{dt} = \displaystyle\sum_{i=1}^{N} \xi_i^\mu g_i - h_\mu \end{cases}$$

where v_i, h_μ are the pre-activation currents of a feature neuron *i* and a hidden neuron μ, and g_i, f_μ are the outputs (e.g., firing rates) of them.

Activation function, contrastive normalization, and energy function

- Neural activity

$$\begin{cases} \tau_v \dfrac{dv_i}{dt} = \displaystyle\sum_{\mu=1}^{K} \xi_i^\mu f_\mu - v_i + I_i \\ \tau_h \dfrac{dh_\mu}{dt} = \displaystyle\sum_{i=1}^{N} \xi_i^\mu g_i - h_\mu \end{cases}$$



$$g_i = [\boldsymbol{g}(\boldsymbol{v})]_i \qquad f_\mu = [\boldsymbol{f}(\boldsymbol{h})]_\mu$$

- Energy Function

$$E = \left[\sum_{i=1}^{N} (v_i - I_i) g_i - L_v\right] + \left[\sum_{\mu=1}^{K} h_\mu f_\mu - L_h\right] - \sum_{\mu,i} f_\mu \xi_i^\mu g_i$$

only feature neurons | only hidden neurons | pairwise interaction

- Lagrangian functions

$$g_i = \frac{\partial L_v}{\partial v_i} \quad \text{and} \quad f_\mu = \frac{\partial L_h}{\partial h_\mu}$$

Case ① $= g(v_i)$ … activation function $\longleftrightarrow$ ① $= \sum_i G(v_i)$ … additive

Case ② $= \dfrac{\exp(v_i)}{\sum_j \exp(v_j)}$ … softmax $\longleftrightarrow$ ② $= \log(\sum_i e^{v_i})$ … non-additive

Conditions for convergence

- Neural activity dynamics

$$\begin{cases} \tau_v \dfrac{dv_i}{dt} = \displaystyle\sum_{\mu=1}^{K} \xi_i^\mu f_\mu - v_i + I_i \\ \tau_h \dfrac{dh_\mu}{dt} = \displaystyle\sum_{i=1}^{N} \xi_i^\mu g_i - h_\mu \end{cases} \longrightarrow$$

- Energy Function

$$E = \left[\sum_{i=1}^{N}(v_i - I_i)g_i - L_v\right] + \left[\sum_{\mu=1}^{K} h_\mu f_\mu - L_h\right] - \sum_{\mu,i} f_\mu \xi_i^\mu g_i$$

only feature neurons | only hidden neurons | pairwise interaction

$$\frac{dE}{dt} = -\tau_v \sum_{i,j=1}^{N} \frac{dv_i}{dt}\frac{\partial^2 L_v}{\partial v_i \partial v_j}\frac{dv_j}{dt} - \tau_h \sum_{\mu,\nu=1}^{K} \frac{dh_\mu}{dt}\frac{\partial^2 L_h}{\partial h_\mu \partial h_\nu}\frac{dh_\nu}{dt} \leq 0$$

The energy monotonically decreases on the dynamical trajectory, if Hessian matrices of the Lagrangian functions are positive semi-definite.

(boundedness needs to be check for specific choice of Lagrangian)

Dense Associative Memory Limit

- Neural activity dynamics

$$\begin{cases} \tau_v \dfrac{dv_i}{dt} = \displaystyle\sum_{\mu=1}^{K} \xi_i^\mu f_\mu - v_i + I_i \\ \tau_h \dfrac{dh_\mu}{dt} = \displaystyle\sum_{i=1}^{N} \xi_i^\mu g_i - h_\mu \end{cases} \xrightarrow[\text{no input currents}]{I_i = 0} \xrightarrow[\text{fast hidden neuron activity}]{\tau_h \to 0} \begin{cases} \tau_v \dfrac{dv_i}{dt} = \displaystyle\sum_{\mu=1}^{K} \xi_i^\mu F_n'(h_\mu) - v_i \\ h_\mu = \displaystyle\sum_{i=1}^{N} \xi_i^\mu \sigma_i \end{cases}$$

- Specific Lagrangian functions

$$\begin{cases} L_v = \sum_i |v_i| \\ L_h = \sum_\mu F_n(h_\mu) \\ \sigma_i = g_i = \frac{\partial L_v}{\partial v_i} = \text{sign}(v_i) \\ f_\mu = \frac{\partial L_h}{\partial h_\mu} = F_n'(h_\mu) \end{cases}$$

$$E = \underbrace{\left[\sum_{i=1}^{N} (v_i - I_i) g_i - L_v\right]}_{=0} + \left[\sum_{\mu=1}^{K} \cancel{h_\mu f_\mu} - L_h\right] - \cancel{\sum_{\mu,i} f_\mu \xi_i^\mu g_i}$$

$$\Rightarrow \quad E = -L_v = -\sum_{i=1}^{K} F_n(\langle \boldsymbol{\xi}^\mu, \boldsymbol{\sigma} \rangle)$$

(Krotov, D. and Hopfield, J.J., 2016)

Modern Hopfield Nets / Transformer Limit

- Neural activity dynamics

$$\begin{cases} \tau_v \dfrac{dv_i}{dt} = \displaystyle\sum_{\mu=1}^{K} \xi_i^\mu f_\mu - v_i + I_i \\ \tau_h \dfrac{dh_\mu}{dt} = \displaystyle\sum_{i=1}^{N} \xi_i^\mu g_i - h_\mu \end{cases} \xrightarrow[\tau_h \to 0]{I_i = 0} \begin{cases} \tau_v \dfrac{dv_i}{dt} = \displaystyle\sum_{\mu=1}^{K} \xi_i^\mu \mathrm{softmax}(h_\mu) - v_i \\ h_\mu = \displaystyle\sum_{i=1}^{N} \xi_i^\mu g_i \end{cases}$$

no input currents

fast hidden neuron activity

- Specific Lagrangian functions

$$\begin{cases} L_v = \frac{1}{2} \sum_i v_i^2 \\ L_h = \log\left(\sum_\mu e^{h_\mu} \right) \\ g_i = v_i \\ f_\mu = \frac{\partial L_h}{\partial h_\mu} = \mathrm{softmax}(h_\mu) \end{cases}$$

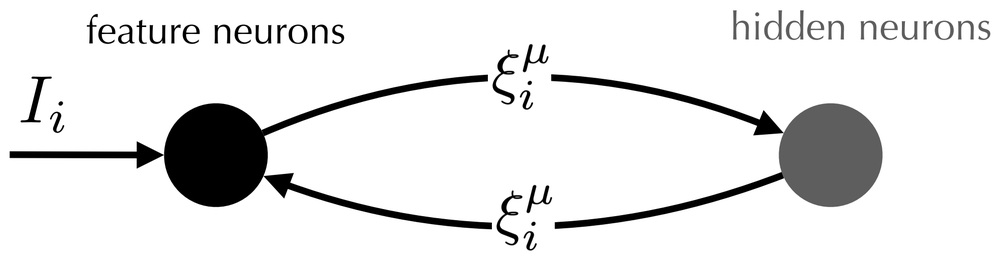
$$\Rightarrow \quad E = \frac{1}{2} \langle \boldsymbol{v}, \boldsymbol{v} \rangle - \log \left(\sum_{\mu=1}^{K} \exp(\langle \boldsymbol{\xi}^\mu, \boldsymbol{v} \rangle) \right)$$

$$v_i^{(t+1)} \leftarrow \sum_{\mu=1}^{K} \xi_i^\mu \mathrm{softmax}(\langle \boldsymbol{\xi}^\mu, \boldsymbol{v}^{(t)} \rangle)$$

(Ramsauer et al, 2020:
Hopfield Networks is All You Need)

Summary

Large Associative Memory Problem in Neurobiology & Machine Learning



$$g_i = [\boldsymbol{g}(\boldsymbol{v})]_i \qquad f_\mu = [\boldsymbol{f}(\boldsymbol{h})]_\mu$$

$$E = \left[\sum_{i=1}^{N}(v_i - I_i)g_i - L_v\right] + \left[\sum_{\mu=1}^{K} h_\mu f_\mu - L_h\right] - \sum_{\mu,i} f_\mu \xi_i^\mu g_i$$

- **Neurobiology:**
  - **biological plausibility** (two-cell synaptic connections)
  - **psychological plausibility** (large memory)
- **Machine Learning:**
  - unified various associative memory models in literature
  - inspired **new recurrent neural networks** architectures