Faculty of Mathematics, Physics and Informatics
Comenius University Bratislava

Neural Networks

Lecture 8

Hopfield's auto-associative memory

Igor Farkaš

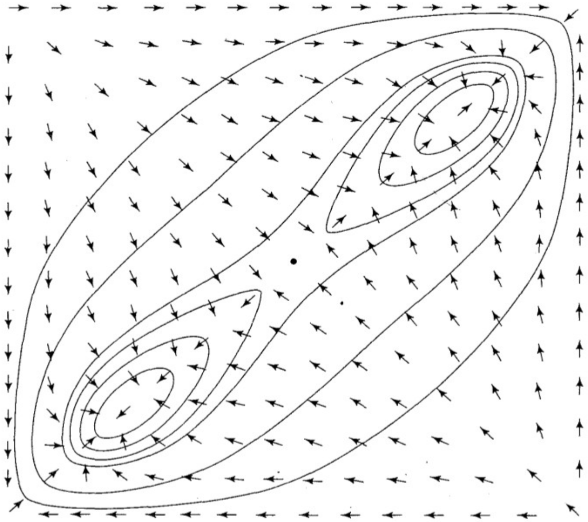
2024

Introduction

- Two functional uses of recurrent neural networks:
  - input-output mapping networks
  - associative memories
- Here we focus on the associative memories
- Key concept – stability (depends on feedback in the model):
- The presence of stability implies some coordination among elements of the dynamic system. Two views:
  - engineering: bounded-input-bounded-output (BIBO) criterion
  - nonlinear dynamic systems: in Lyapunov's (1892) sense
- Neurodynamics: deterministic or stochastic

Dynamic systems

- State-space model – uses state variables that unfold in time
- the state $\boldsymbol{x}(t) = [x_1(t),\ x_2(t),\ \ldots,\ x_n(t)]^T$, n = order of the system
- In continuous time: $d\boldsymbol{x}(t)/dt = \mathbf{F}(\boldsymbol{x}(t))$
- In discrete time: $\boldsymbol{x}(t+1) = \mathbf{F}(\boldsymbol{x}(t))$
- $\mathbf{F}$ is a vector function; its each component is a nonlinear function (whose arguments are any elements of $\boldsymbol{x}$)
- System unfolding ~ trajectory in state space
- State portrait ~ all trajectories superimposed
- Stability analysis – identification of equilibria



example in 2D space

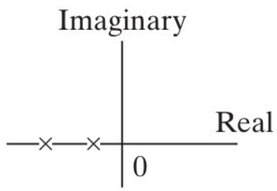
(Hopfield, 1984)

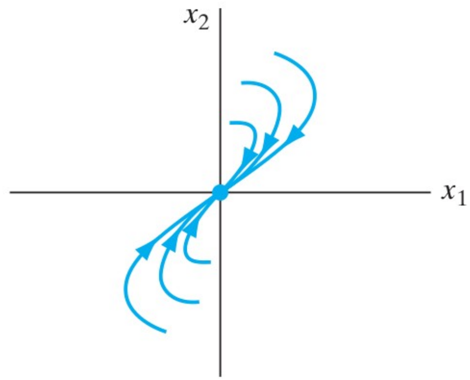
Analyzing equilibrium states

- Analysis of equilibria – attractor points – is important for understanding a nonlinear dynamic system
- Equillibrium state: $\mathbf{F}(\boldsymbol{x}^*) = \mathbf{0}$
- use 1st order approximation of $\boldsymbol{x}(t+1) = \mathbf{F}(\boldsymbol{x}(t))$ of each $\boldsymbol{x}^*$
- $\mathbf{F}(\boldsymbol{x}(t))$ is assumed to be smooth enough in the neighborhood, to allow linearization:
- $\mathbf{F}(\boldsymbol{x}(t)) \approx \boldsymbol{x}^* + \mathbf{A}(\boldsymbol{x}(t) - \boldsymbol{x}^*)$, where $\mathbf{A} = \partial\mathbf{F}/\partial\mathbf{x}\,|\boldsymbol{x} = \boldsymbol{x}^*$
- properties of (Jacobian) matrix $\mathbf{A}$ important $\rightarrow$ eigenvalues
- 2D example:

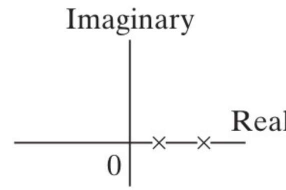
Attractor types in 2D space

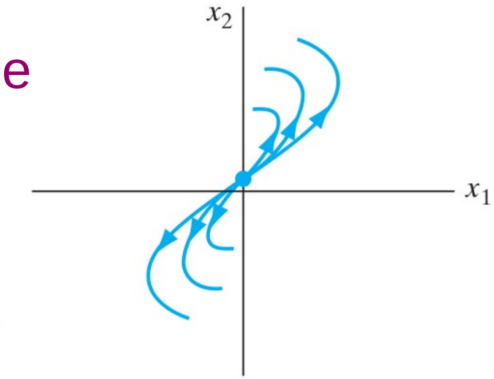
stable node



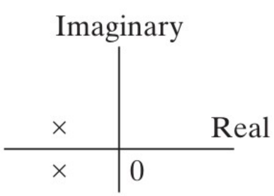


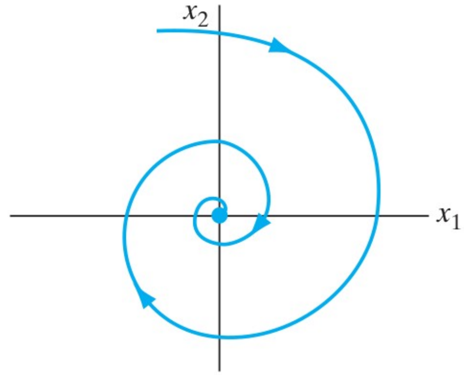
unstable node



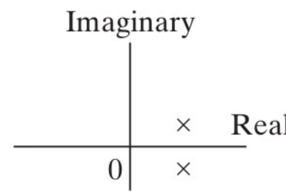


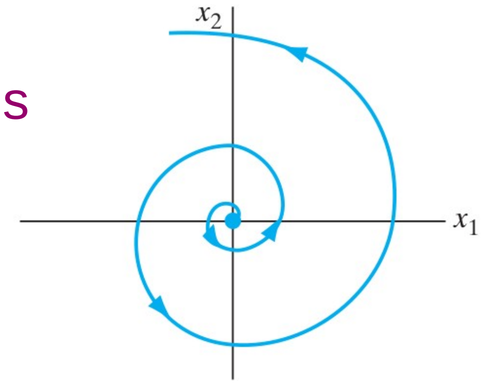
stable focus



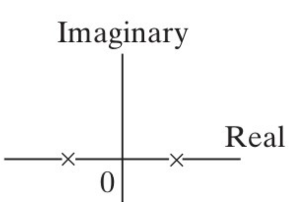


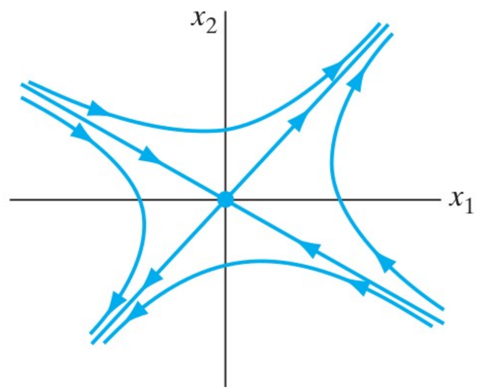
unstable focus



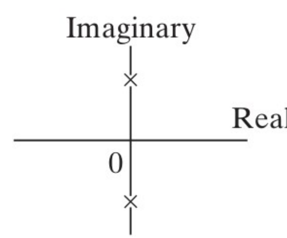


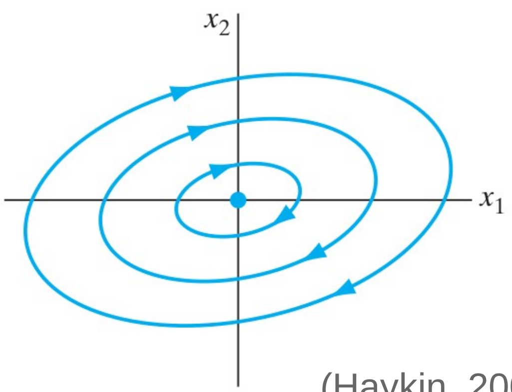
saddle point





center





(Haykin, 2009)

Definitions of stability

- Linearization is useful, but we need more precise definitions for autonomous nonlinear dynamic systems (Khalil, 1992):
- *Def.1:* Equilibrium state $\boldsymbol{x}^*$ is said to be uniformly stable if $\forall \varepsilon > 0$, $\exists \delta > 0$, that if $\|\boldsymbol{x}(0) - \boldsymbol{x}^*\| < \delta$, then $\|\boldsymbol{x}(t) - \boldsymbol{x}^*\| < \varepsilon$, $\forall t > 0$.
- *Def.2:* Equilibrium state $\boldsymbol{x}^*$ is convergent if $\exists \delta > 0$, such that if $\|\boldsymbol{x}(0) - \boldsymbol{x}^*\| < \delta$, then $\boldsymbol{x}(t) \to \boldsymbol{x}^*$, for $t \to \infty$.
- *Def.3:* Equilibrium state $\boldsymbol{x}^*$ is said to be asymptotically stable, if it is both stable and convergent.
- *Def.4:* Equilibrium state $\boldsymbol{x}^*$ is said to be globally asymptotically stable, if it is stable and all trajectories of the system converge to $\boldsymbol{x}^*$ as $t \to \infty$.

Determining stability

- Lyapunov function – scalar function of the system state:
- Equilibrium state $\boldsymbol{x}^*$ is stable if there exists a positive-definite function $V(\boldsymbol{x})$ such that $dV/d\boldsymbol{x} \leq 0$ for $\boldsymbol{x} \in nbh(\boldsymbol{x}^*)$
- Equilibrium state $\boldsymbol{x}^*$ is asymptotically stable if there exists a positive-definite function $V(\boldsymbol{x})$ such that $dV/d\boldsymbol{x} < 0$ for $\boldsymbol{x} \in nbh(\boldsymbol{x}^*)$.
- Function $V(\boldsymbol{x})$ is positive-definite if: (1) there exist continuous $\partial V(\boldsymbol{x})/\partial x_i$ for $i = 1,2,\ldots,n$, (2) $V(\boldsymbol{x}^*) = 0$, (3) $V(\boldsymbol{x}) > 0$ for $\boldsymbol{x} \in nbh(\boldsymbol{x}^*)$.
- no indication of how to find a Lyapunov function in general
- Existence of a Lyapunov function is a sufficient, but not a necessary, condition for stability.

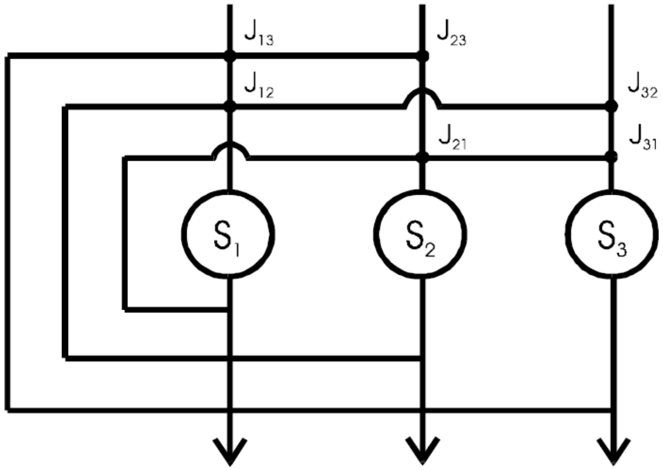
Neurodynamic models

- The systems of interest (i.e. with continuous state variables, and dynamics described by differential/difference equations) have these characteristics (Peretto and Niez, 1986; Pineda, 1988a):
- Large number of degrees of freedom – both the computational power and the fault-tolerant capability of such a system are the result of the collective dynamics of the system.
- Nonlinearity – essential for creating a universal computing machine.
- Dissipation – characterized by the convergence of the state-space volume onto a manifold of lower dimensionality as time goes on.
- Noise – an intrinsic characteristic; in real neurons, membrane noise is generated at synaptic junctions (Katz, 1966).

Towards a deterministic Hopfield model

- physical inspiration (ordering states in magnetic materials)
- model of spin glasses (Kirkpatrick a Sherrington, 1978)
- Hopfield (1982) network: (influential)
  - content-addressable memory (“given a cue, retrieve a pattern”)
  - an example of cellular automaton
- Attractive features of AAM:
  - model of a cognitive processing (attractors)
  - emergent behavior
- emphasis is on pattern retrieval dynamics, rather than learning

Hopfield model: basic concepts



$\mathbf{J} \equiv \mathbf{W}$

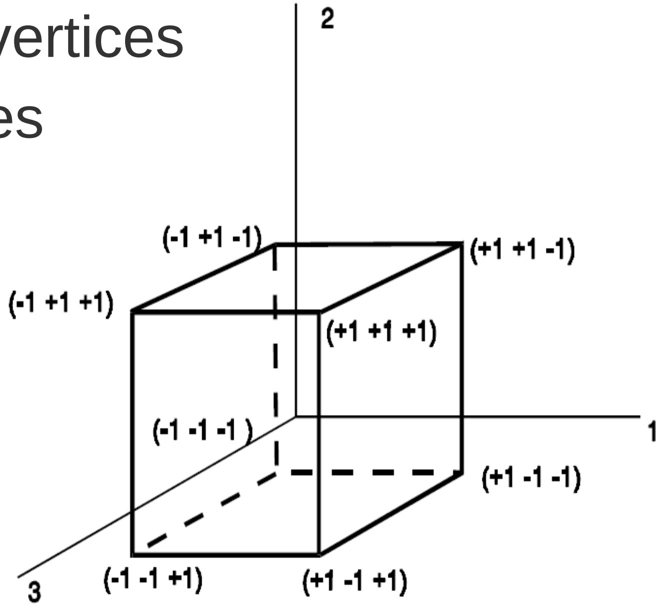
- One (fully connected) layer with n neurons
- Neuron: two states $S_i \in \{-1, +1\}$, $i = 1 \ldots n$
- Configuration: $\mathbf{S} = [S_1, S_2, \ldots, S_n]$
- Weight: $w_{ij} \sim j \rightarrow i$, if $w_{ij} > 0$, then excitatory,
  $w_{ii} = 0$ if $w_{ij} < 0$, inhibitory,
- Postsynaptic potential: $h_i^{\mathrm{int}} = \sum_j w_{ij} S_j$ (~ internal magnetic field)
- Neuron excitation threshold: h_i^{ext} (~ external field)
- Effective postsynaptic potential: $h_i = h_i^{\mathrm{int}} - h_i^{\mathrm{ext}}$ (~ postsynaptic potential)
- Neuron state update (deterministic version): $S_i \rightarrow S_i' = \mathrm{sgn}(h_i) \in \{-1, +1\}$
  if $h_i \geq 0$, $\mathrm{sgn}(h_i) = 1$, else -1.

Model dynamics

- Synchronous (parallel): $S_i(t) = \mathrm{sgn}(\sum_{j\neq i} w_{ij} S_j(t-1) - h_i^{\mathrm{ext}}) \qquad \forall i$
  - one relaxation cycle = update of all neurons: $\mathbf{S}(t-1) \rightarrow \mathbf{S}(t)$
- Asynchronous (sequential): $S_i(t) = \mathrm{sgn}(\sum_{j\neq i} w_{ij} S_j(t-1) - h_i^{\mathrm{ext}}) \qquad i \sim \mathrm{rnd}$
  - randomly chosen neurons
- Evolution of configuration: $\mathbf{S}(0) \rightarrow \mathbf{S}(1) \rightarrow \mathbf{S}(2) \rightarrow \ldots$ (relaxation process)
  - Sync dynamics: trajectory over hypercube vertices
  - aSync dynamics: along the hypercube edges
- Energy of configuration: (as Lyapunov f.)

$$E(\mathbf{S}) = -\tfrac{1}{2} \sum_i \sum_j w_{ij} S_i S_j - \sum_i S_i h_i^{\mathrm{ext}}$$

- non-increasing for sym. $\mathbf{W}$ and $h_i^{\mathrm{ext}} = 0, \forall i$



Energy function does not increase

(in case of symmetric weights and no external field)

… until the network reaches stable state. Why?

$E(\mathbf{S}) = -\frac{1}{2} \sum_i \sum_j w_{ij} S_i S_j$, let us change $S_m \rightarrow S'_m$, i.e. $\Delta E(\mathbf{S}) = E(\mathbf{S}') - E(\mathbf{S})$

$E(\mathbf{S}) = -\frac{1}{2} \sum_{i \neq m} \sum_{j \neq m} w_{ij} S_i S_j - \frac{1}{2} \sum_i w_{im} S_i S_m - \frac{1}{2} \sum_j w_{mj} S_m S_j$

$E(\mathbf{S}) = -\frac{1}{2} \sum_{i \neq m} \sum_{j \neq m} w_{ij} S_i S_j - \sum_i w_{im} S_i S_m$ (since $w_{im} = w_{mi}$)

$\Delta E(\mathbf{S}) = -\sum_i w_{im} S_i S'_m - \sum_i w_{im} S_i S_m = -(S'_m - S_m) \sum_i w_{im} S_i = (S_m - S'_m)\, h_m$

$+1 \rightarrow +1$: $\Delta E(\mathbf{S}) = 0$

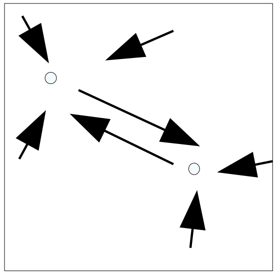
$-1 \rightarrow -1$: $\Delta E(\mathbf{S}) = 0$

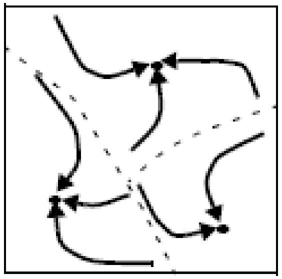
$+1 \rightarrow -1$: $\Delta E(\mathbf{S}) = +2\, h_m < 0$, since $h_m < 0$ (such that S_m change occurs)

$-1 \rightarrow +1$: $\Delta E(\mathbf{S}) = -2\, h_m < 0$, since $h_m > 0$ (such that S_m change occurs)

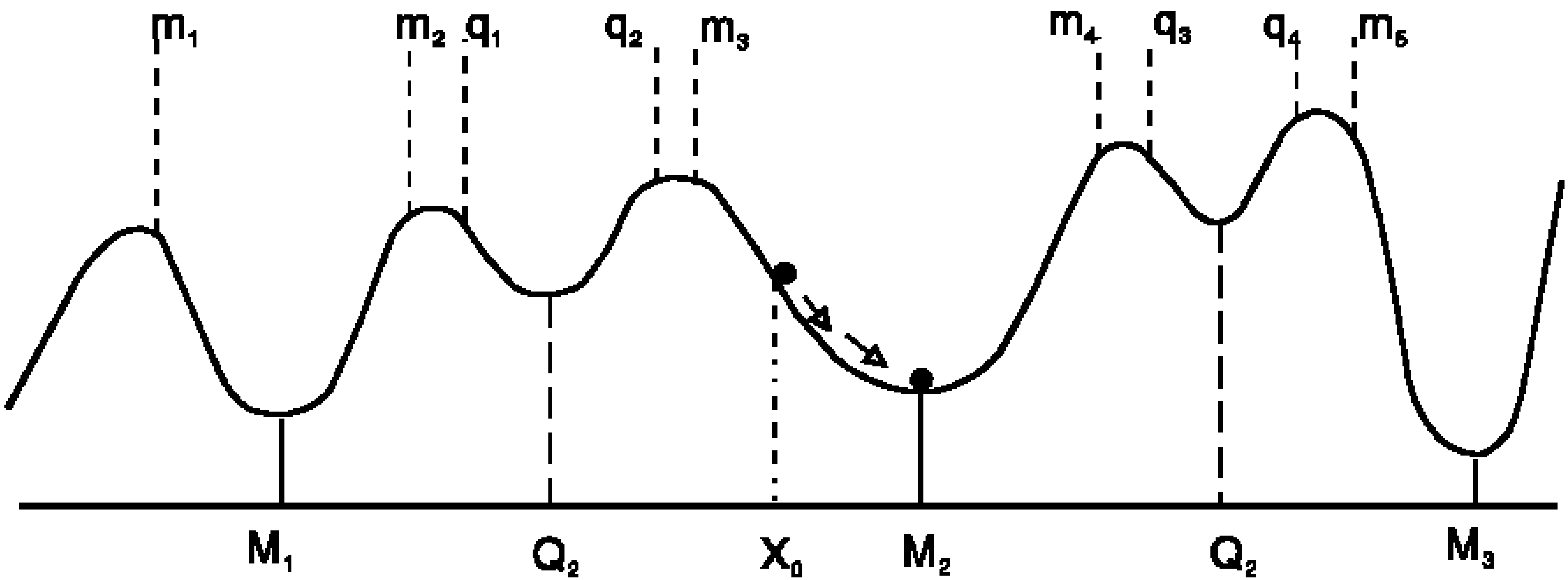
Asymptotic behavior

- Randomly set initial config. $(P_{\text{bit}=1} = 0.5)$, $h_i^{\text{ext}} \in [-1,+1]$, $w_{ij} \in [-1,+1]$
- Chaotic behavior: E rises and descends
  - typical param.: $\mathbf{W} \neq \mathbf{W}^T$, SyncDyn, arbitrary $h_i^{\text{ext}} \in [-1,+1]$
- Limit cycles: (with period 2, 4, ...)
  - typical param.: syncDyn (rarely for aSyncDyn)
- (Fixed) points: local minima of E
  - typical param.: $\mathbf{W} = \mathbf{W}^T$, asyncDyn, $h_i^{\text{ext}} = 0$
  - E descends only





Energy landscape



(Kvasnička et al., 1997)

- basins of attraction, depend on **W**
- attractors: true (M_k), spurious (Q_l)
- energy decreases monotonously (in fixed point dynamics)
- spurious attractors – undesirable (linear combinations of odd number of patterns)

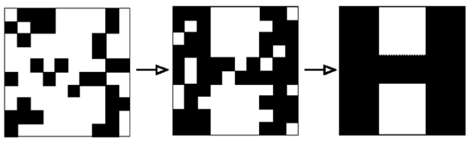
Autoassociative memory

- Point attractors = stationary states ⇔ memorized patterns
- Content-addressable memory
- Assume (binary) patterns: $\boldsymbol{x}^{(p)} = [x_1^{(p)}, x_2^{(p)}, \ldots, x_n^{(p)}]$, $p = 1\ldots N$ (patterns)
- Set symmetric weights: $w_{ij} = 1/n \sum_p x_i^{(p)} x_j^{(p)}$ for $i \neq j$, $w_{ii} = 0$
- $w_{ij} \in \{-N/n, \ldots, 0, \ldots, N/n\}$
- Recall (retrieval) of pattern $\boldsymbol{x}^{(r)}$ occurs ⇔ $\mathbf{S}(0) \rightarrow \ldots \rightarrow \boldsymbol{x}^{(r)}$
- Stability requirement for $\boldsymbol{x}^{(r)}$: $x_i^{(r)}.\, h_i^{(r)} > 0$ for $i = 1\ldots n$ (units)
- $x_i^{(r)}.\, h_i^{(r)} = x_i^{(r)} \sum_j w_{ij}\, x_j^{(r)} = \ldots = 1 + C_i^{(r)} > 0.$

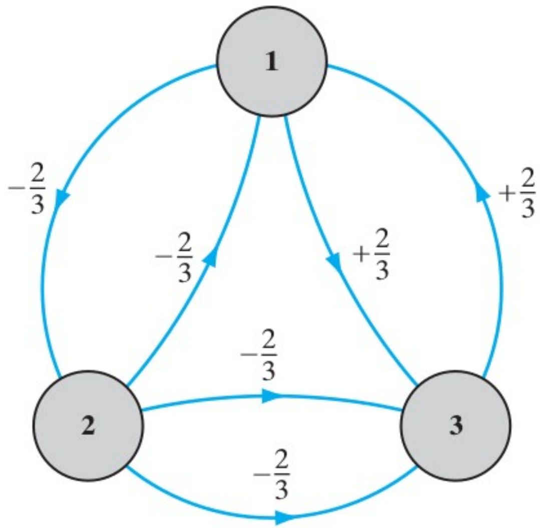
$$C_i^{(r)} = x_i^{(r)} \sum_{p \neq r} x_i^{(p)} \left(\frac{1}{N} \sum_j x_j^{(p)} x_j^{(r)} \right)$$

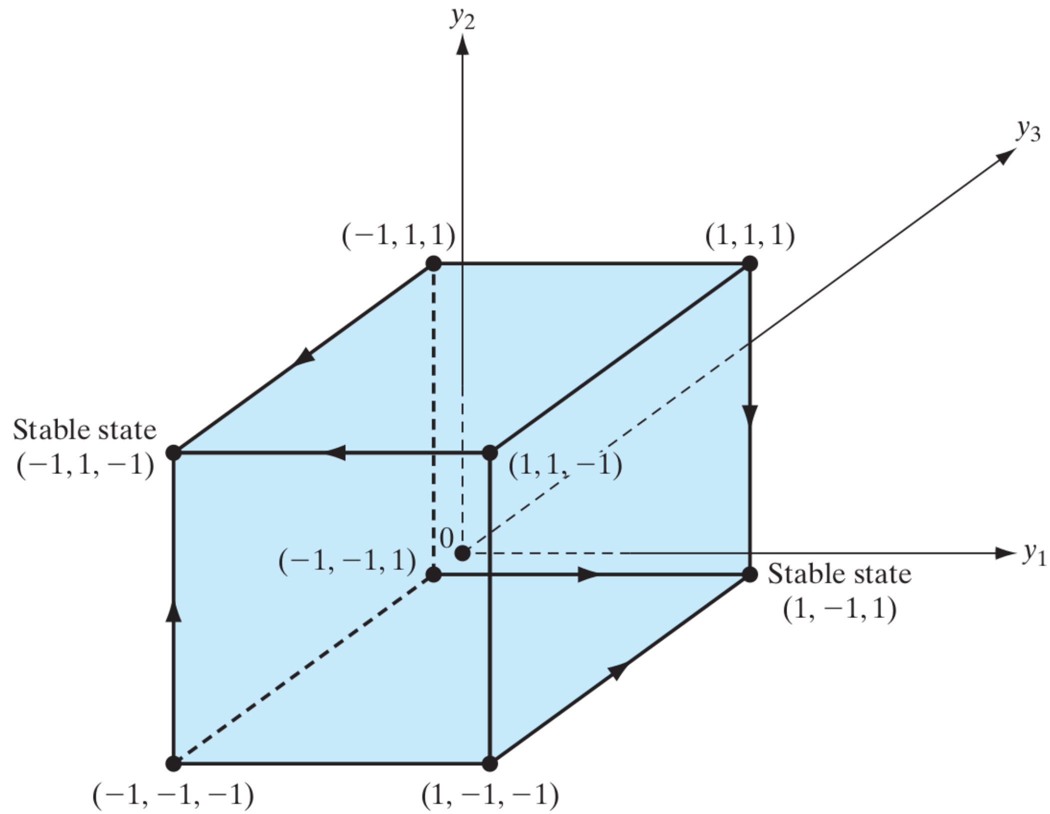
in limit for large *n*

crosstalk



Small example with 3 neurons





$$\mathbf{W} = \frac{1}{3}\begin{bmatrix} 0 & -2 & +2 \\ -2 & 0 & -2 \\ +2 & -2 & 0 \end{bmatrix}$$

$$\mathbf{Wy} = \frac{1}{3}\begin{bmatrix} 0 & -2 & +2 \\ -2 & 0 & -2 \\ +2 & -2 & 0 \end{bmatrix}\begin{bmatrix} +1 \\ -1 \\ +1 \end{bmatrix} = \frac{1}{3}\begin{bmatrix} +4 \\ -4 \\ +4 \end{bmatrix}$$

$$\mathbf{Wy} = \frac{1}{3}\begin{bmatrix} 0 & -2 & +2 \\ -2 & 0 & -2 \\ +2 & -2 & 0 \end{bmatrix}\begin{bmatrix} -1 \\ +1 \\ -1 \end{bmatrix} = \frac{1}{3}\begin{bmatrix} -4 \\ +4 \\ -4 \end{bmatrix}$$

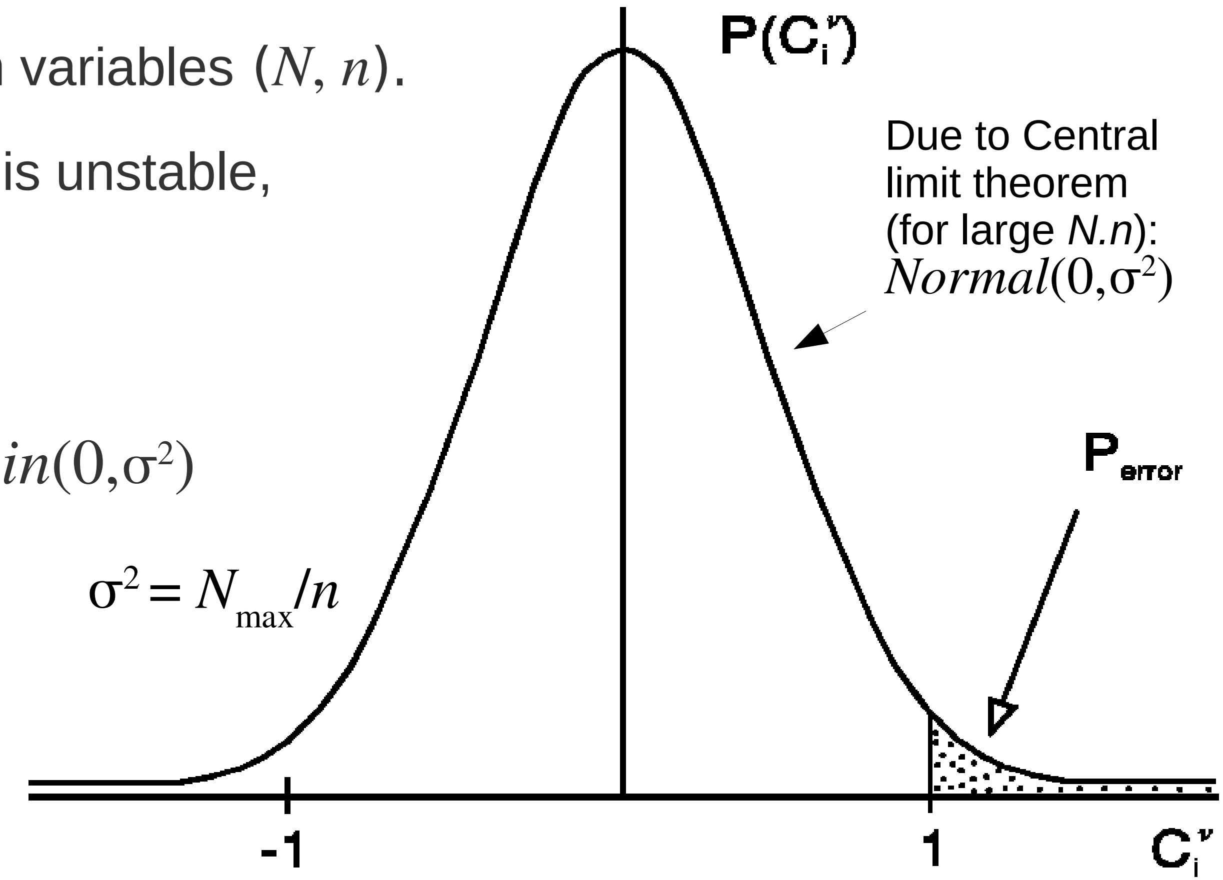
- two states are stable →
- other 6 states are unstable

(Haykin, 2009)

Memory capacity

- for orthogonal patterns $C_i^{(r)} = 0 \;\; => \;\; N_{\max} = n$
- for pseudoorthogonal patterns: i.e. if $\langle \boldsymbol{x}^{(p)\mathrm{T}} . \boldsymbol{x}^{(r)} \rangle \approx 0 \;\wedge\; |C_i^{(r)}| < 1$ (stability)
  - What's the capacity in this case?
- Treat pattern bits as random variables (N, n).
- What's the prob. that *i*-th bit is unstable,
  i.e. $P_{\mathrm{error}} = P(C_i^{(r)} > 1)$?
- for large N and n,

  $C_i^{(r)} \sim$ binomial distribution $Bin(0,\sigma^2)$
- $Bin \approx Normal(0,\sigma^2)$



(Kvasnička et al, 1997)

Memory capacity (ctd)

- Relationship between P_{error} and capacity:

P_{error}	$N_{\max}/n$
0.001	0.105
0.0036	0.138
0.01	0.185
0.05	0.37
0.1	0.61

increasing blackout in retrieval

- Stable memorized states are
  - true attractors
  - reverse configurations
  - spurious states (undesirable) – due to existence of a null space

Stochastic Hopfield model

- How to get rid of spurious attractors?
- Introduction of noise into the model
  - more biologically plausible
  - narrows down basins of attraction of spurious attractors
- Interpretation from statistical physics: noise ↔ inverse temperature
- stable config: $P(\mathbf{S}) = 1/Z \exp(-\beta\, E(\mathbf{S}))$, $\quad Z = \sum_{\mathbf{S}'} \exp(-\beta E(\mathbf{S}'))$, $\beta=1/T$.
- $P(\mathbf{S} \to \mathbf{S}') = 1/(1+\exp(\beta\Delta E))$ where $\Delta E = E(\mathbf{S}') - E(\mathbf{S})$
- i.e. non-zero probability of transition to a state with a higher *E*
- For $T \to 0$ we get a deterministic model
- For $T \to \infty$ we get an ergodic model, i.e. $P(S_m = +1) = 0.5$, $\quad m = 1..n$
  - no stable memories

Stochastic Hopfield model (ctd)

- Stochastic rule that a unit *m* changes its state

  $P(S_m = \pm 1) = 1/(1+\exp(-2\beta h_m S_m))$ (m-th bit was changed), since

  $$\Delta E = E' - E = (S_m - S'_m)\, h_m = \begin{cases} -2h_m S'_m & \text{if } S'_m = -S_m \\ 0 & \text{if } S'_m = S_m \end{cases}$$

- Then $P(S_m=+1) = 1/(1+\exp(-2\beta h_m))$

  and $P(S_m=-1) = 1 - P(S_m=+1)$

- probabilities of transitions depend on $\beta = 1/T$
- All spurious attractors can become destabilized by a suitable setting of β (Amit el al., 1985).
- lower accuracy of retrieved memorized patterns (prob. distrib.): overlap $m^{(p)} = 1/N \sum_j x_j^{(p)}.S_j^{(p)}$ with one of patterns > 0.

Applications

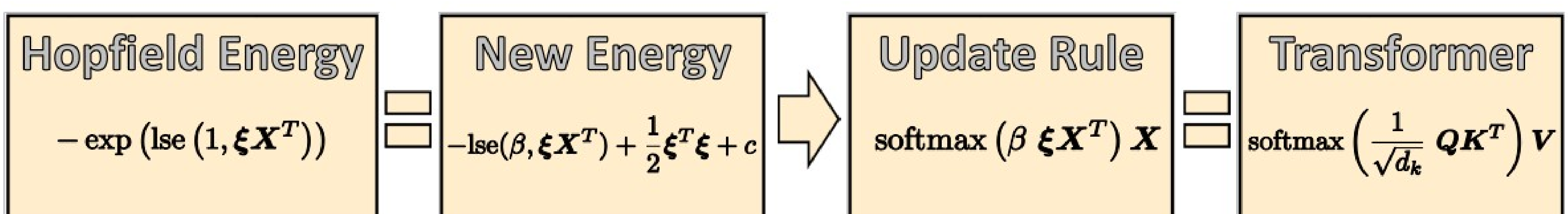
- Modeling neurobiological and psychological effects:
  - autoassociative pattern recall (given a cue)
  - recognition of sequences (relaxation times in cognitive modeling)
  - generation of sequences (e.g. melodies)
- Optimization problems:
  - combinatorial (e.g. TSP)
  - image processing (filtering) – reconstruction of an image from its noisy version

Modern Hopfield networks

- Krotov and Hopfield (2016): the binary two-layer model with new energy function and update rule, has a much higher capacity and reduced stability of spurious states.
- Ramsauer et al (2020): Hopfield two-layer model with continuous states (CHN)
- update rule in new CHN = attention mechanism in transformers
- can store exponentially (with dimension) many patterns, converges with one update, and has exponentially small retrieval errors.
- trade-off b/w number of stored patterns and convergence speed + retrieval error
- 3 types of energy minima (fixed points): (1) global fixed point averaging over all patterns, (2) meta-stable states averaging over a subset of patterns, (3) fixed points which store a single pattern.

Hopfield net with continuous states

- CHN: Assume contin. patterns $\boldsymbol{X} = \{\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(N)}\},\ \boldsymbol{x}^{(i)} \in \Re^n$
- Log-sum-exp function used: $LSE(\beta, \boldsymbol{y}) = \beta^{-1} \ln\left(\sum_{i=1}^{N} \exp(\beta y_i)\right)$
- LSE represents free energy (in statistical thermodynamics)
- Assume (continuous) state (query) $\boldsymbol{s} \in \Re^n$, $M = \max_i \|\boldsymbol{x}^{(i)}\|$
- Define new energy: $E = LSE(\beta, \boldsymbol{X}^T \boldsymbol{s}) + ½\, \boldsymbol{s}^T \boldsymbol{s} + \beta^{-1} \ln(N) + ½\, M^2$
- Update rule: $\boldsymbol{s} \leftarrow \boldsymbol{X}.\mathrm{softmax}(\beta \boldsymbol{X}^T \boldsymbol{s})$
- A new interpretation of NLP models with attention (e.g. BERT)
- Hopfield layer added in PyTorch



Summary

- Hopfield's (1982) work has had a significant impact on NNs
  - modern generative models build on it
- symmetric weights – novel useful feature introduced
- Defining network state as energy (to be minimized)
- Hopfield model shows that it is possible for a structured behavior to emergent from evolution of a complex, nonlinear dynamic system over time.
- In autoassociative memory, stochastic model overcomes limitations of deterministic version, by properly destabilizing spurious attractors.
- modern versions of Hopfield model with enhanced memory capacity