Hopfield model

モデル化とシミュレーション特論
2021 年度前期

佐賀大学理工学研究科　只木進一

Introduction

- neural network with mutual interaction
  - memory and retrieval
- Non-deterministic motion
- Hopfield model
- Boltzmann machine

Hopfield model

- N neurons: state $s_i = \{-1, 1\}$
- Interaction (symmetric because of theoretical reason)

$$w_{ij} = w_{ji}, \quad w_{ii} = 0 \tag{1}$$

- Asynchronous update
  - Select one neuron randomly and update its state
  - One time step consists of N updates (one Monte Carlo step)

$$s_i = \begin{cases} 1 & \text{if } h_i = \sum_j w_{ij} s_j \geq 0 \\ -1 & \text{otherwise} \end{cases} \tag{2}$$

Energy

- Energy

$$E = -\frac{1}{2}\sum_i \sum_j w_{ij} s_i s_j \tag{3}$$

- Energy varies by update of one neuron

$$\delta E = -\sum_j w_{ij} s_j \delta s_i \leq 0 \tag{4}$$

- δs_i always has the same sign of $h_i = \sum_j w_{ij} s_j$

- Energy monotonously decreases
  - Monotonously degreasing functions are called Lyapunov
- For $h_i = \sum_j w_{ij} s_j \geq 0$

$$\delta s_i = \begin{cases} 2 & \text{if} s_i = -1 \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

- For $h_i = \sum_j w_{ij} s_j \leq 0$

$$\delta s_i = \begin{cases} -2 & \text{if} s_i = 1 \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

Hebb's learning rule

- For learning pattern $\vec{\xi}$

$$w_{ij} = \lambda \xi_i \xi_j, \qquad i \neq j \tag{7}$$

- λ is $O(N^{-1})$ quantity
  - Energy must be extensive (**示量変数**): proportional to system size
  - Energy is extensive for ordinary physical systems
  - cf: intensive variables (**示強変数**)(independent of system size) temperature, pressure, etc.

$$E = -\frac{1}{2}\sum_i \sum_{j \neq i} \lambda \xi_i \xi_j s_i s_j$$
$$= -\frac{\lambda}{2}\left[\left(\sum_i \xi_i x_i\right)^2 - \sum_i \xi_i^2 s_i^2\right] = -\frac{\lambda}{2}\left[\left(\sum_i \xi_i x_i\right)^2 - N\right] \quad (8)$$

- Two energy minima: $\vec{s} = \pm\vec{\xi}$
- Same as ferromagnetic (強磁性) systems

P patterns

- Hebb' rule

$$w_{ij} = \lambda \sum_{\mu=0}^{P-1} \xi_i^\mu \xi_j^\mu \tag{9}$$

- Overlapping with μ-th pattern

$$m_\mu = \frac{1}{N} \sum_i \xi_i^\mu s_i \tag{10}$$

$$E = -\frac{\lambda}{2} N^2 \sum_{\mu} (m_\mu)^2 + \frac{\lambda}{2} NP \qquad (11)$$

- Each pattern is indicated by energy minima, if they are orthogonal
- There may be energy minima not indicating memory patterns

Dynamics at finite temperature

At finite temperature T, transition probability is given by

$$P\left(\delta s_i = \pm 2\right) = \frac{1}{1 + e^{\mp 2\beta h_i}} \tag{12}$$

$$h_i = \sum_j w_{ij} s_j, \quad \beta = 1/T \tag{13}$$

- probability of $\delta s_i = -2$ ($s_i = 1$)
  - low temperature limit($\beta \to \infty$)

$$P(\delta s_i = -2) \to \begin{cases} 1 & h_i > 0 \\ 0 & h_i < 0 \end{cases} \tag{14}$$

  - high temperature limit ($\beta \to 0$)

$$P(\delta s_i = -2) \to 1/2 \tag{15}$$

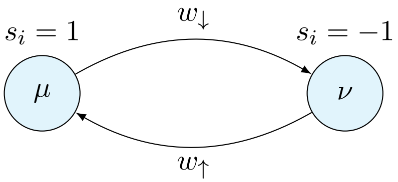
- probability of $\delta s_i = 2$ ($s_i = -1$)
  - low temperature limit($\beta \to \infty$)

$$P(\delta s_i = 2) \to \begin{cases} 1 & h_i < 0 \\ 0 & h_i > 0 \end{cases} \tag{16}$$

  - high temperature limit

$$P(\delta s_i = 2) \to 1/2 \tag{17}$$

Equilibrium



$$\frac{P_\mu}{P_\nu} = \frac{w_\uparrow}{w_\downarrow} = \frac{1+e^{-2\beta h_i}}{1+e^{2\beta h_i}} = e^{-2\beta h_i} = \frac{e^{-\beta h_i}}{e^{\beta h_i}}$$

$$= \frac{e^{-\beta E(\mu)}}{e^{-\beta E(\nu)}} \tag{18}$$

$$P_\mu \propto e^{-\beta E(\mu)} \tag{19}$$

$$
\begin{aligned}
h_i &= \sum_{j \neq i} w_{ij} 1 \times s_j \\
&= \frac{1}{2} \sum_{j \neq i} w_{ij} 1 \times s_j + \frac{1}{2} \sum_{j \neq i} w_{ij} s_j \times 1 + \frac{1}{2} \sum_{j \neq i} \sum_{k \neq i} w_{jk} s_j s_k \\
&= E\left(\mu\right) && (20) \\
-h_i &= \sum_{j \neq i} w_{ij} \left(-1\right) \times s_j \\
&= \frac{1}{2} \sum_{j \neq i} w_{ij} \left(-1\right) \times s_j + \frac{1}{2} \sum_{j \neq i} w_{ij} s_j \times \left(-1\right) + \frac{1}{2} \sum_{j \neq i} \sum_{k \neq i} w_{jk} s_j s_k \\
&= E\left(\nu\right) && (21)
\end{aligned}
$$

Simulation

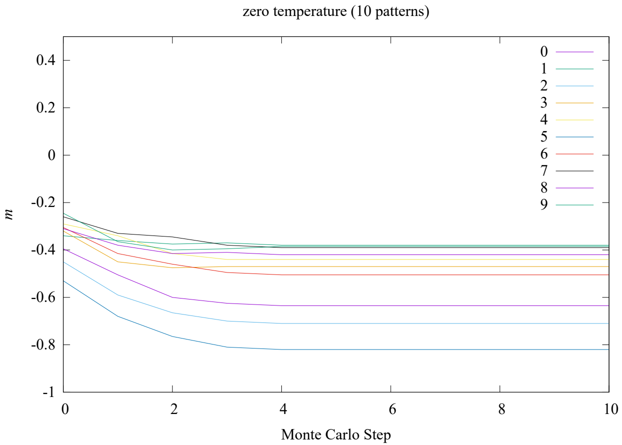
- Setting 10 Kanji patterns: not orthogonal!
- Change the number of patterns
- Zero or finite temperature
- Observe how the system find one of patterns

model package

- Neuron class:states of a neuron
- Hopfield class
  - generate weight vectors from patterns
  - update states at zero and finite temperature
  - evaluate overlapping with patterns
  - evaluate the energy
- AbstractPatterns

Zero temperature

Overlapping with patterns



Temporal variation of energy at $T = 0$