

Neural Networks

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

- 1 Introduction: Neurons and Brains
- 2 Mathematical model of neurons
- 3 Perceptron
- 4 3-layer perceptron

Introduction: Neurons and Brains

- Neural network (神経回路網)
 - Generate collective responses.
- Single cell organism (単細胞生物)
 - No neurons
 - Respond to external stimuli (刺激).
- Plants (植物)
 - No neurons
- multi-cell animals (多細胞動物)
 - Neurons by cell differentiation (細胞分化)

Neural systems in multi-cell animals

- Poriferan (海綿動物) and Placozoa (平板動物)
 - No neurons
 - Almost no cell differentiation
- Animals with scattered neural system (散在神経系)
 - Neural network on body surface
 - No central neural system (中枢神経)
 - Coelenterate (腔腸動物): jellyfish (くらげ), coral (さんご) etc.

Animals with cage-shaped neural system (かご型神経系をもつ動物)

- Ganglion (神経節) as a center of neural system at head
- Cage-shaped neural network on body surface
- flatworm (扁形動物): Planaria (プラナリア) etc.

Animals with ladder-shaped neural system (はしご型神経系をもつ動物)

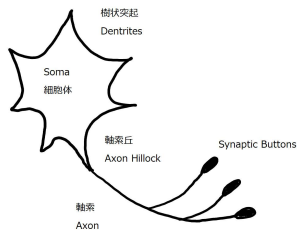
- At head, brain evolved through ganglion
- Central neural system at abdomens (腹面)
- Ganglions at each somites (体節)
- Arthropod (節足動物): insects (昆虫) etc.
- Annelid (環形動物): earthworm (ミミズ) etc.

Animals with tubular neural system (管状神経系をもつ動物)

- At head, brain evolved through ganglion
- Central neural system at body center
- Chordate (脊索動物)
- Vertebrate (脊椎動物)

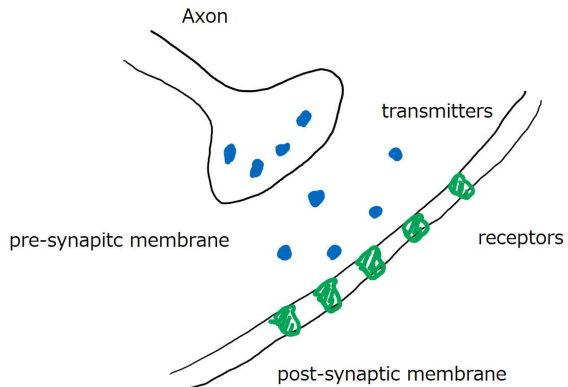
Neurons (神経細胞)

- Receive pulses from other neurons through synapses
- Electric pulses using ions
- Coding scheme are not clearly understood
- Two state: fire and rest
- Neuron fires if pulses from other neurons exceed some threshold



- Soma (細胞体)
 - Keep living activities such as normal cells
- Dendrite (樹状突起)
 - Receive pulses from other neurons
- Axon (軸索)
 - Send pulses to other neurons
 - Tip divided into 10,000 synaptic buttons for human neurons
- Cell division almost finished during early childhood
- Body cells keep dividing lifelong

Synapse



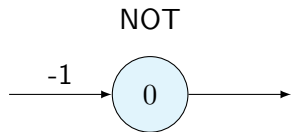
McCulloch-Pitts model

- Stimuli for neuron j from neuron i

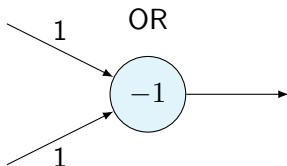
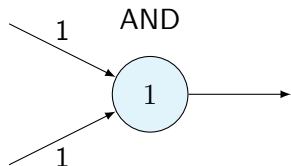
$$x_j = \phi \left(\sum_i w_{ji} x_i - h_j \right) \quad (1)$$

- x_i : output from neuron i
- w_{ji} : synaptic connection
- h_j : threshold
- ϕ : response function, usually sigmoidal shape

McCulloch-Pitts neuron as a logical gate



- Encode $\{T, F\}$ by $\{1, -1\}$
- Threshold are shown as numerical values in nodes
- Step response function.

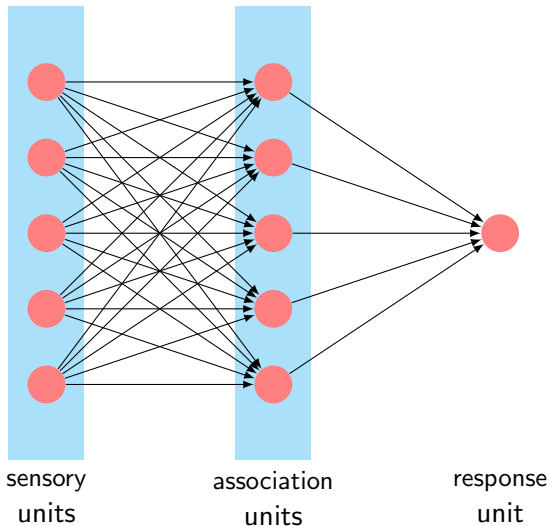


Classes in model package

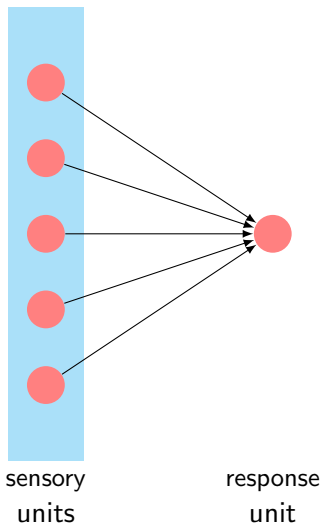
- Neuron Class
 - Constructor with weight and response function
 - response() with input
- McCullochPitts class
 - Show response of AND, OR, and NOT gates.
- CorrectResponse class
- AbstractMultiLayer class

Perceptron : Rosenblatt (1966)

Learning and recognition by neural networks



Two-layer perceptron



Two-layer perceptron

- Response unit receives $\{a_i\}$ from sensory units
- Response unit outputs η

$$\eta = \theta \left(\sum_j \xi_j a_j \right) \quad (2)$$

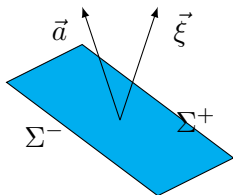
- ξ_j : weight for input
- The threshold is placed as the last element of the weight. And the last element of input is constant.
- θ : Step function

$$\theta \left(\vec{\xi} \cdot \vec{a} \right) = \begin{cases} 1 & \vec{\xi} \cdot \vec{a} \geq 0 \\ -1 & \vec{\xi} \cdot \vec{a} < 0 \end{cases} \quad (3)$$

Linearly Separable

- Consider a input space spanned by \vec{a}
- Divide the space by a hyper space normal to the weight vector $\vec{\xi}$

$$\begin{cases} \vec{\xi} \cdot \vec{a} \geq 0 & \vec{a} \in \Sigma^+ \\ \vec{\xi} \cdot \vec{a} < 0 & \vec{a} \in \Sigma^- \end{cases} \quad (4)$$



Learning by error-correction

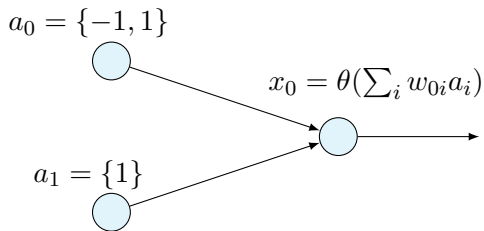
$$\begin{cases} \vec{\xi} \rightarrow \vec{\xi} + c\vec{a} & \text{if } \eta = -1 \text{ for } \vec{a} \in \Sigma^+ \\ \vec{\xi} \rightarrow \vec{\xi} - c\vec{a} & \text{if } \eta = 1 \text{ for } \vec{a} \in \Sigma^- \end{cases} \quad (5)$$

- Correct response η_{correct}

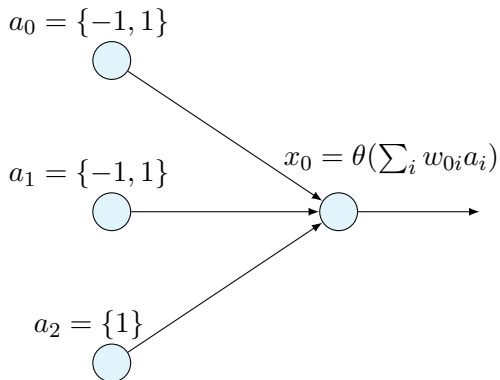
$$\eta_{\text{correct}} = \begin{cases} 1 & \vec{a} \in \Sigma^+ \\ -1 & \vec{a} \in \Sigma^- \end{cases} \quad (6)$$

$$\vec{\xi} \rightarrow \vec{\xi} - \frac{c}{2} (\eta - \eta_{\text{correct}}) \vec{a} \quad (7)$$

Example: Perceptron learning NOT gate



Example: Perceptron learning AND and OR gates

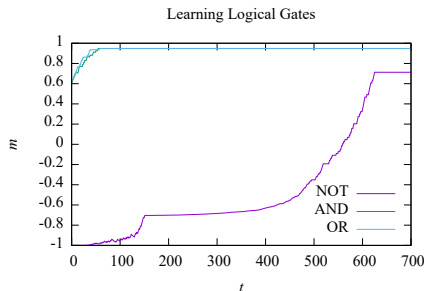


Classes in twoLayer package

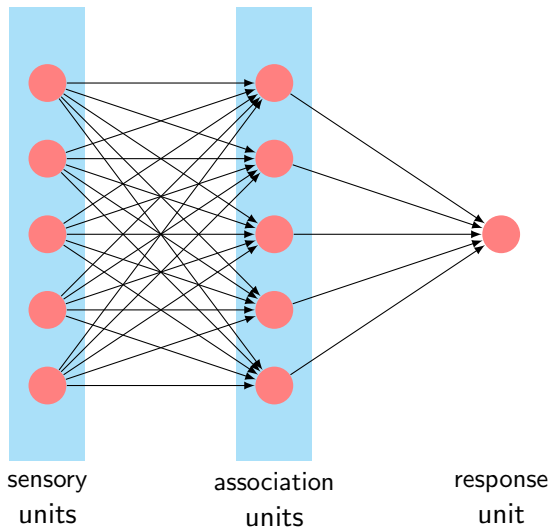
- TwoLayer class: general two layer model
- LearningLogicGate class: general learning model
- LearningAndGate, LearningOrGate, and LearningNotGate

Experiments

- ① $\vec{\xi}_0$: Correct answers
- ② $\vec{\xi}$: Initial random vectors
- ③ At each learning step
 - ① learning
 - ② normalize vector
 - ③ evaluation: $m = \vec{\xi}_0 \cdot \vec{\xi}$



3-layer perceptron



3-layer perceptron

- Outputs from sensory units: $\{a_i\}$
- Outputs from association units: $\{x_i\}$

$$x_i = f \left(\sum_j w_{ij} a_j \right) \quad (8)$$

w_{ij} : weight

- Output from response unit: η

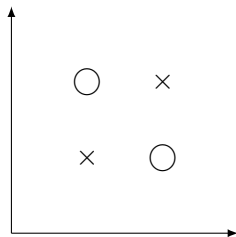
$$\eta = g \left(\sum_k s_k x_i \right) \quad (9)$$

s_k : weight

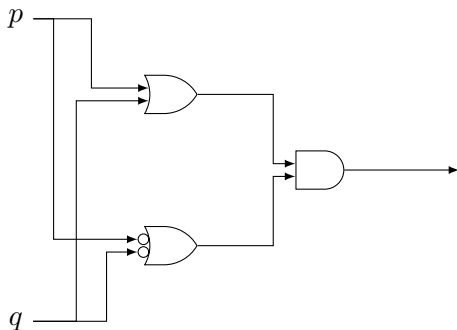
Example XOR

Note that XOR is not linearly separable

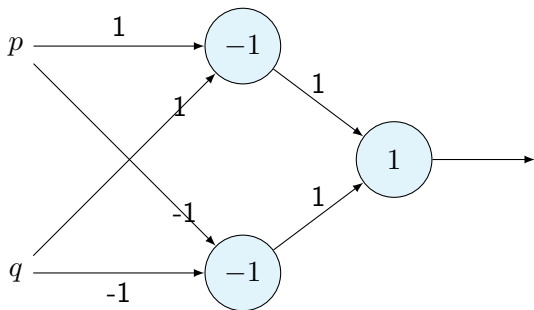
p	q	$p \text{ XOR } q$
0	0	0
0	1	1
1	0	1
1	1	0



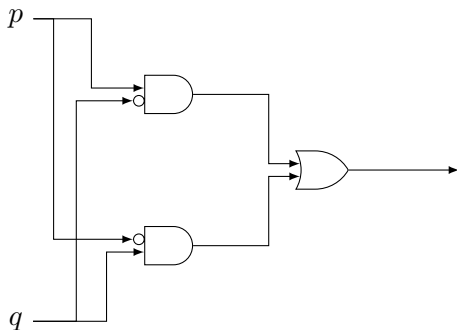
Example: logical circuit



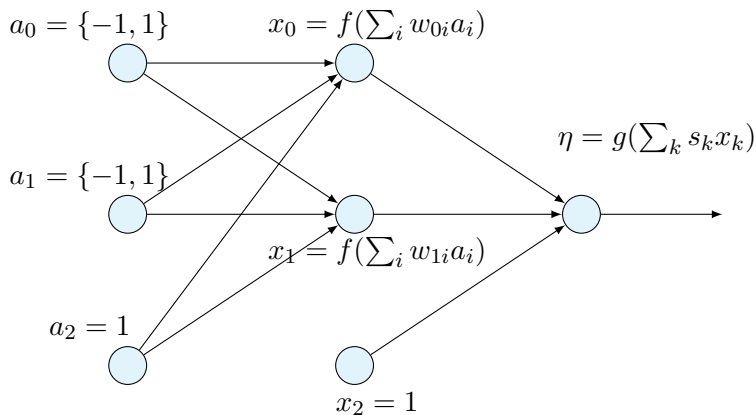
Implement as 3-layer perceptron



Example: logical circuit: another version



Real implementation



Back propagation method

- Minimize square error

$$E = \frac{1}{2} (\eta - \eta_{\text{correct}})^2 \quad (10)$$

- Continuous output from any elements

$$x_i = f \left(\sum_j w_{ij} a_j \right) \quad (11)$$

$$\eta = g \left(\sum_k s_k x_k \right) \quad (12)$$

Update weights in response unit

$$\begin{aligned}\frac{\partial E}{\partial s_k} &= (\eta - \eta_{\text{correct}}) g' \left(\sum_j s_j x_j \right) x_k \\ &\equiv r x_k\end{aligned}\tag{13}$$

$$s_k \rightarrow s_k - c r x_k\tag{14}$$

Update weights in association unit

$$\begin{aligned} \frac{\partial E}{\partial w_{ij}} &= (\eta - \eta_{\text{correct}}) g' \left(\sum_k s_k x_k \right) s_i f' \left(\sum_{\ell} w_{i\ell} a_{\ell} \right) a_j \\ &\equiv \tilde{r}_i a_j \end{aligned} \quad (15)$$

$$w_{ij} \rightarrow w_{ij} - c \tilde{r}_i a_j \quad (16)$$

- Errors seem to propagate backwardly from response unit to association unit
- Possibly trapped at local minima

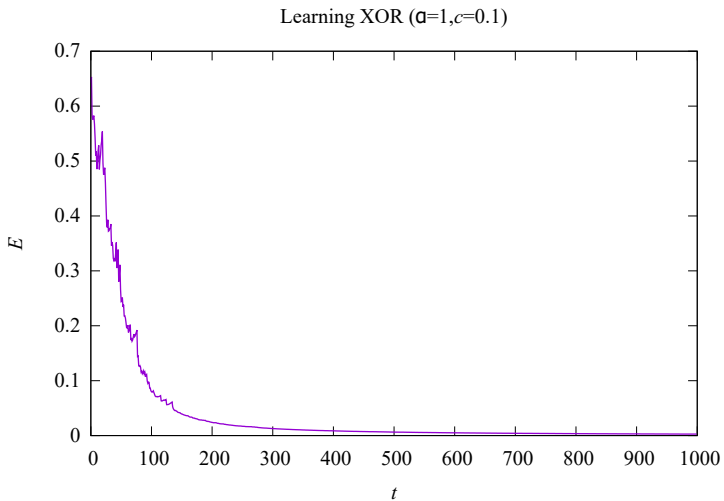
Simulation setups

- Response function

$$f(x) = g(x) = \tanh(\alpha x) \quad (17)$$

- random initial value: $\{w_{ij}\}$ 、 $\{s_k\}$
- Observe error at every step

$$\bar{E} = \frac{1}{4} \sum_{a_0=\{-1,1\}} \sum_{a_1=\{-1,1\}} \frac{1}{2} (\eta - \eta_{\text{correct}})^2 \quad (18)$$



Response of output unit

p	q	o
-1	-1	-1
-1	1	-0.94
1	-1	-0.94
1	1	0.98

Output unit works like AND gate.

Responses of associative units

p	q	o_1	o_2
-1	-1	-0.92	1
-1	1	0.85	0.84
1	-1	0.86	0.85
1	1	1	-0.92

The 1st unit works like $p \vee q$. and 2nd one $\neg p \vee \neg q$.