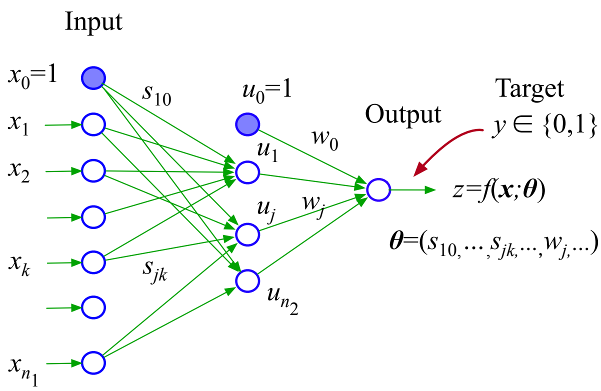
23 Apr 2021

YRM71, YS591 – Mathematical Neuroscience
Project 1: Backpropagation

A short oral presentation is required in addition to turn in a short writeup that describes the problem you investigated, why it is interesting, and results.

Key words: learning from examples, feed-forward neural networks, error backpropagation



Assignment

1. XOR problem. $n_1 = 2, n_2 = 2$.

   Data $\mathcal{D} = \{(\boldsymbol{x}^\alpha, y^\alpha)\} = \{((0,0),0),((0,1),1),((1,0),1),((1,1),0)\}$

2. Mirror Symmetry Detection Problem (Rumelhart *et al.* 1986), $n_1 = 6, n_2 = 2$. Among $2^6 = 64$ possible inputs, the network is trained to output 1 for only $2^3 = 8$ inputs.

   $\mathcal{D} = \{(\boldsymbol{x}^\alpha, y^\alpha)\} = \{((0,1,0,0,1,0),1),((0,1,1,1,1,0),1),((1,1,0,1,1,0),0),\cdots\}$

Learning Procedure

1. Generate a set of input-output pairs, $(\boldsymbol{x}^\alpha, y^\alpha), \alpha = 1,2,\cdots$.

2. Set weights connections,w_j, and $s_{jk}, j = 0,1,\cdots,n_2, k = 0,1,2,\cdots,n_1$, random values according to the normal distribution $\mathcal{N}(0, 0.1^2)$, mean 0, variance 0.1^2.

3. Set a randomly selected $\boldsymbol{x}^\alpha$ as an input to the network.

4. **Activity dynamics (Fast):** Calculate $u_j, j = 1,2,\cdots,n_2$, output z, and E.

$$u_j = f\left(\sum_{k=0}^{n_1} s_{jk}x_k\right),\ z = f\left(\sum_{j=0}^{n_2} w_j u_j\right),\ E = \frac{1}{2}\Big(z-y\Big)^2$$

5. **Weight dynamics (Slow):** Update parameter values w_j, and $s_{jk}, j = 0, 1, \cdots, n_2, k = 0, 1, 2, \cdots, n_1$.

$$\Delta w_j = -\mu \frac{\partial E}{\partial w_j} = \mu r u_j, \qquad r = (y - z) \cdot z(1 - z)$$

$$\Delta s_{jk} = -\mu \frac{\partial E}{\partial s_{jk}} = \mu r_j^* x_k, \qquad r_j^* = r w_j \cdot u_j(1 - u_j)$$

where r and r_j^*'s are learning signals, and $\mu = 0.8$ is learning coefficient.

6. Repeat this procedure from 3. until ΔE is small enough.

Evaluation of the network (* Must)

1. * Draw a graph with number of iterations in horizontal axis and E in vertical axis.

2. * Every 10 iterations, calculate the percentage of correct responses. Draw a graph with number of iterations in horizontal axis and the percentage correct responses (decide 1 if the weighted sum input to the output unit is larger than 0, 0 otherise) in vertical axis.

3. * Try several different initial parameter values in learning.

4. * Try several cases of number of hidden units, n_2. Interpret the trained network, and explain how the network solves the XOR and Mirror Symmetry problems.

5. In mirror symmetry detection problem, only $m = 20$ (for example) examples are used to train the network. Evaluate network performance $64 - m = 44$ untrained examples. Try various m.

6. Compare the results (e,g., the learning speed) when output unit z and target signal y take $(-1, 1)$ instead of $(0, 1)$.

7. Compare the learning speed when the target signal y takes $(0.1, 0.9)$ and $(-0.9, 0.9)$ instead of $(0, 1)$ and $(-1, 1)$.

8. Try different activity functions. Try ReLU as hidden units activity functions

$$f_5(u) = \begin{cases} u, & u \geqq 0 \\ 0, & u < 0 \end{cases}$$

9. Try batch learning. Instead of adjusting parameter for each input, accumulate 10 $\Delta w_j, \Delta s_{jk}$'s for 10 inputs, then change these values to the means. Compare the results.

10. Draw the learning trajectory of connection weights. In Mirror Symmetry Detection Problem, $n_1 = 6, n_2 = 2$, there are 17 parameters $\boldsymbol{\theta} = (s_{10}, s_{20}, \cdots, s_{62}, w_0, w_1, \cdots, w_6)$. Collect many $\boldsymbol{\theta}$s during learning and represent these in 2 dimensional plane by dimensional reduction method, like PCA or t-SNE.