AI Application Programming

by M. Tim Jones

ISBN:1584502789

Charles River Media © 2003

The purpose of this book is to demystify the techniques associated with the field of artificial intelligence. It covers both the theory and the practical applications to teach developers how to apply AI techniques in their own designs.

Table of Contents

Chapter 5: Introduction to Neural Networks and the Backpropagation Algorithm

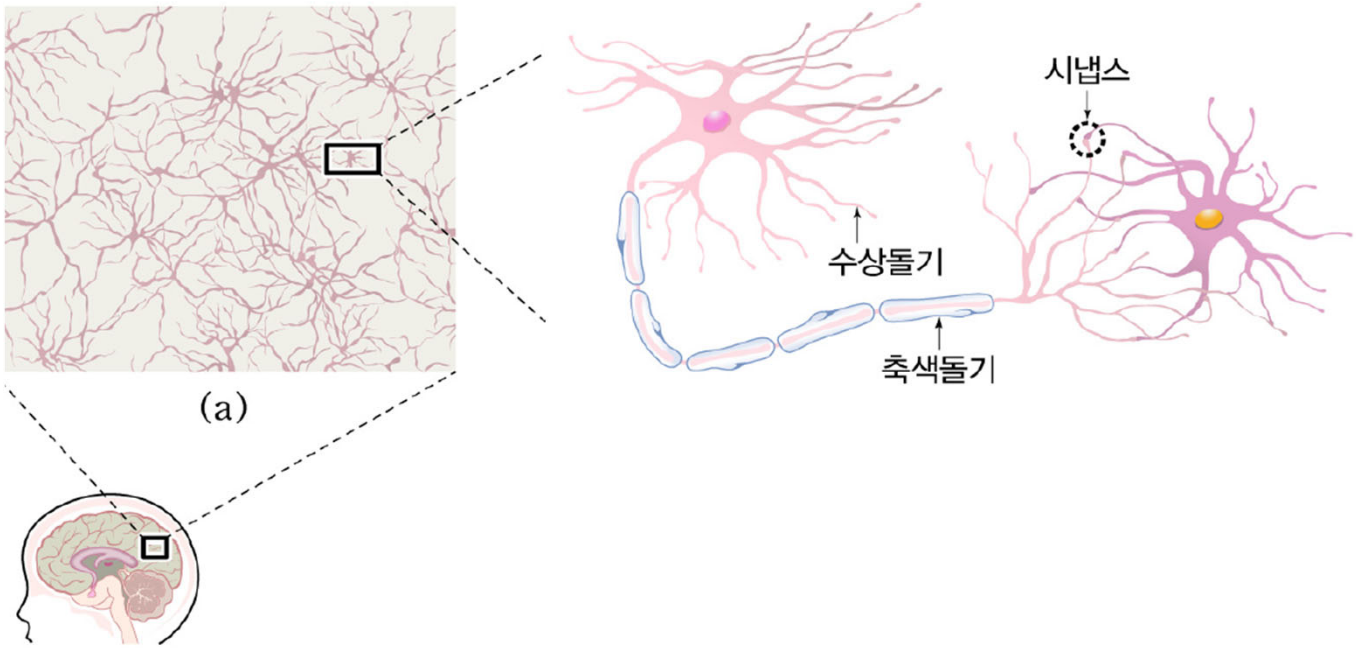
- Download CD Content

This chapter introduces feed-forward, multi-layer neural network architectures with learning provided by the backpropagation algorithm. Backpropagation is likely the most important learning algorithm for neural networks and has contributed to a resurgence of biologically-inspired methods for computation. After detailing neural networks and backpropagation, we'll look at the resulting network's application in game character AI.

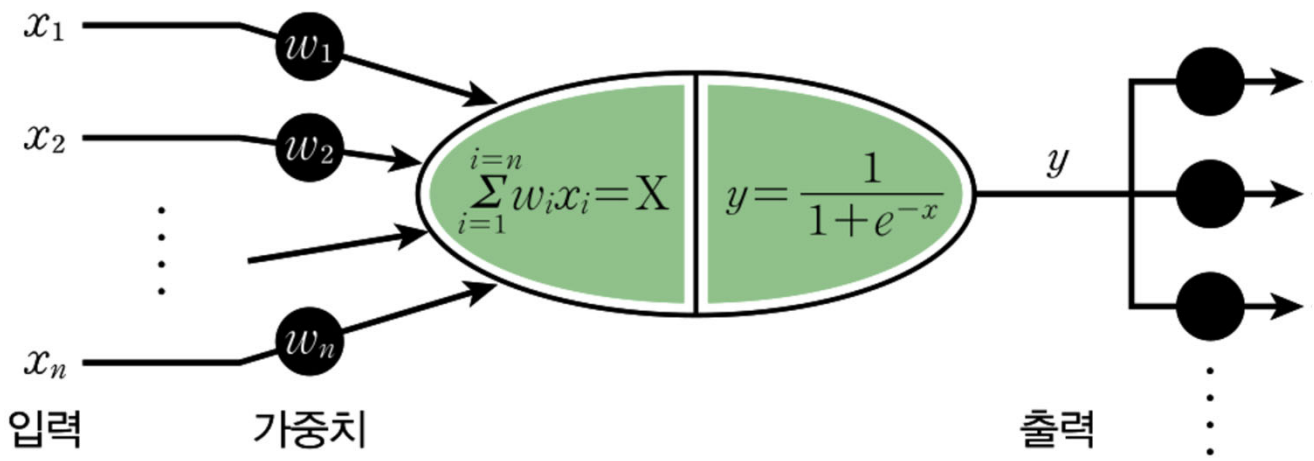
5장 신경망과 역전파 알고리즘

While a large variety of neural network topologies and learning algorithms exist, this chapter will focus on feed-forward, multi-layer networks using backpropagation learning. We'll begin with a simple introduction of neural networks and their components; discuss the learning algorithm and then some problems that can arise during backpropagation learning. We'll look at an example of a simple network and walk through the backpropagation algorithm to understand its properties. Finally, we'll look at simple neural networks as a way to give life to characters within game environments.

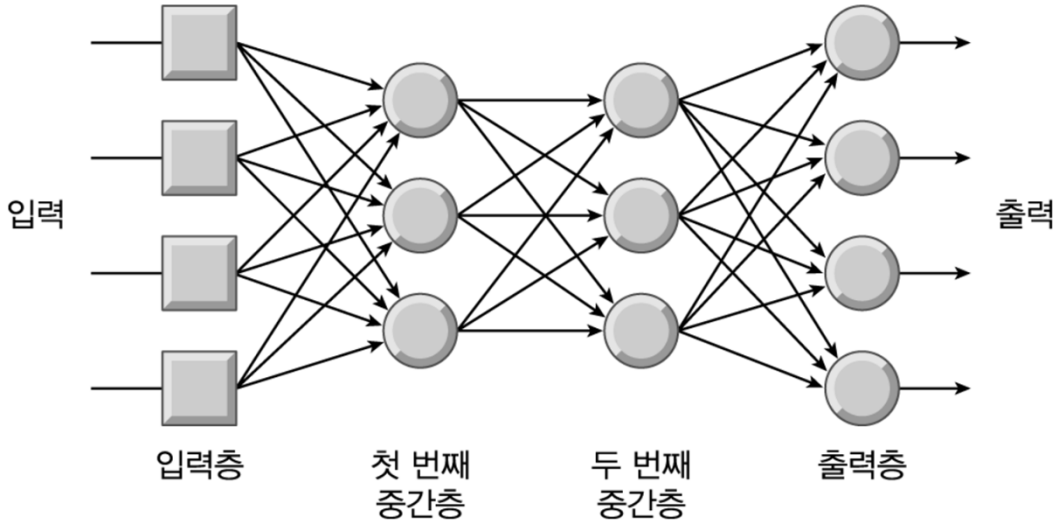
5장 신경망과 역전파 알고리즘; 뉴런 neuron



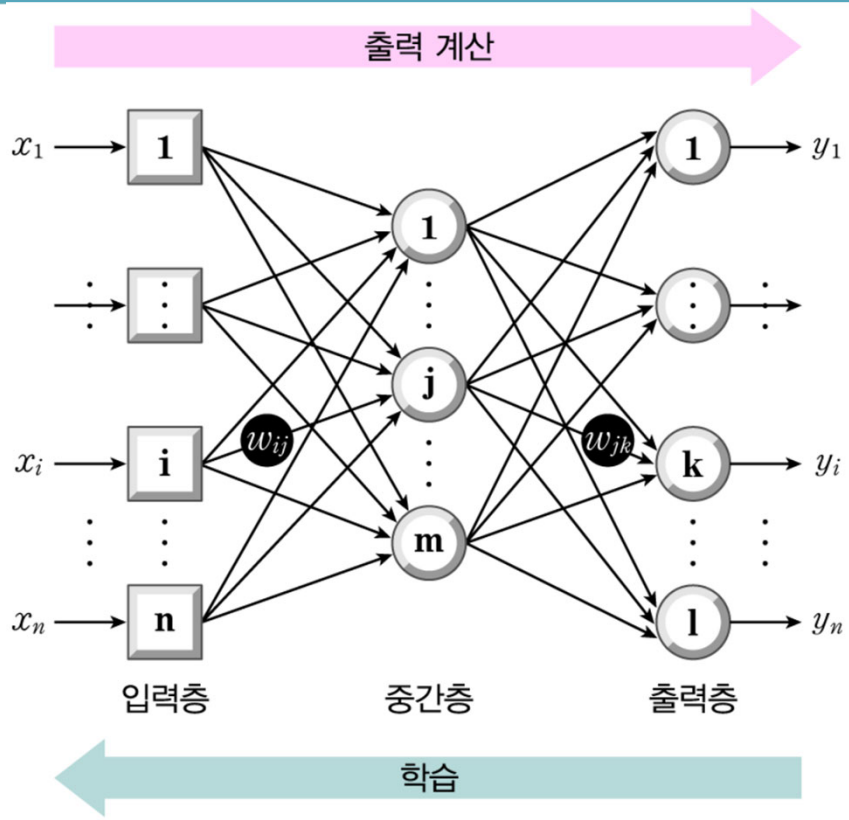
5장 신경망과 역전파 알고리즘; 인공 뉴런 artificial neuron



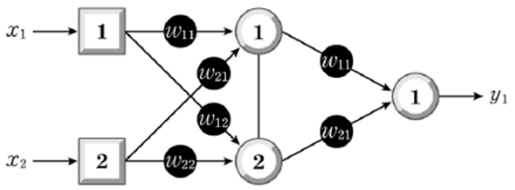
5장 신경망과 역전파 알고리즘; 뉴럴 네트워크



5장 신경망과 역전파 알고리즘; 뉴럴 네트워크



다음과 같은 3층 신경망의 출력을 계산해보자.



입력–중간층 가중치

$w_{11} = 0.5,\ w_{12} = 0.9$

$w_{21} = 0.4,\ w_{22} = 1.0$

중간–출력층 가중치

$w_{11} = -1.2,\ w_{21} = 1.1$

입력

$x_1 = x_2 = 1.0$

출력층 뉴런은 중간층 뉴런들의 출력을 입력으로 받아 자신의 출력을 계산하므로 먼저 중간층 첫 번째 뉴런의 출력 y_1과 두 번째 뉴런의 출력 y_2를 계산한다.

$$X_1 = x_1 w_{11} + x_2 w_{21} = 1.0 \times 0.5 + 1.0 \times 0.4,\ y_1 = \frac{1}{1+e^{-X_1}} \text{이므로}$$

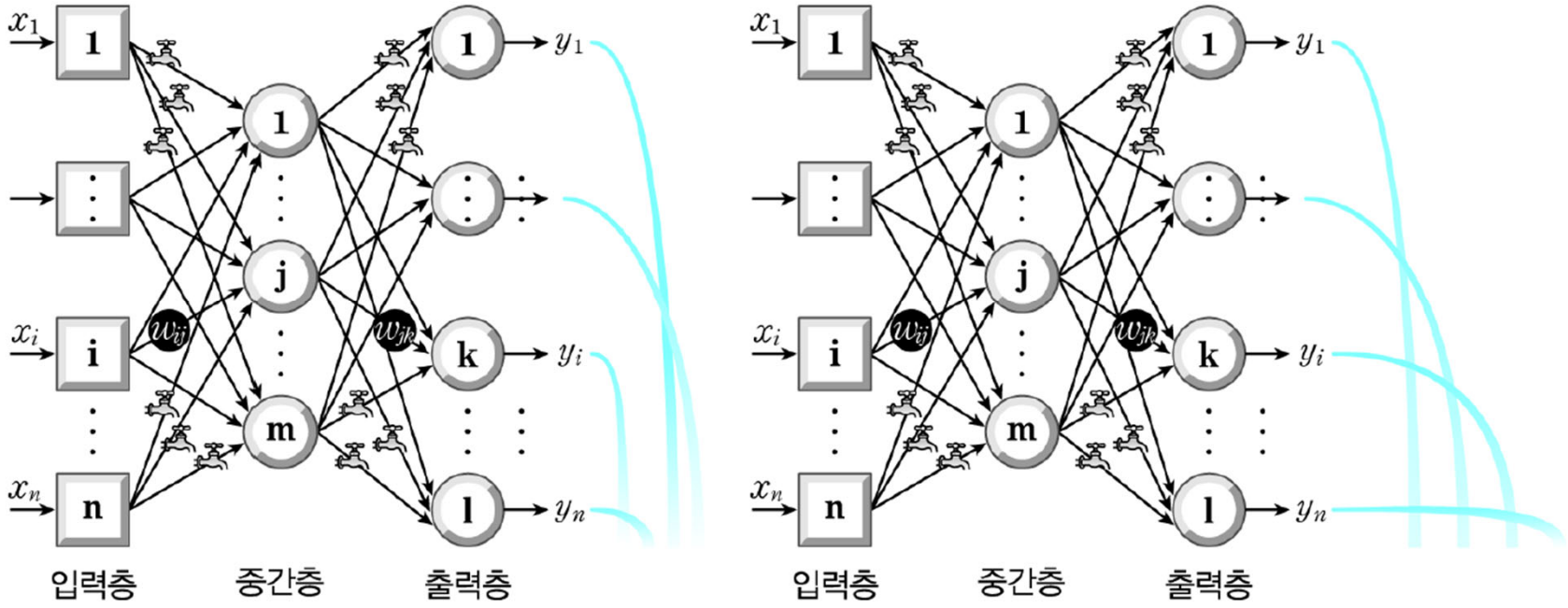
$$y_1 = 1/(1+e^{-(1.0\times 0.5 + 1.0\times 0.4)}) = 0.7 \text{이고}$$

$$X_2 = x_1 w_{12} + x_2 w_{22} = 1.0 \times 0.9 + 1.0 \times 1.0,\ y_2 = \frac{1}{1+e^{-X_2}} \text{이므로}$$

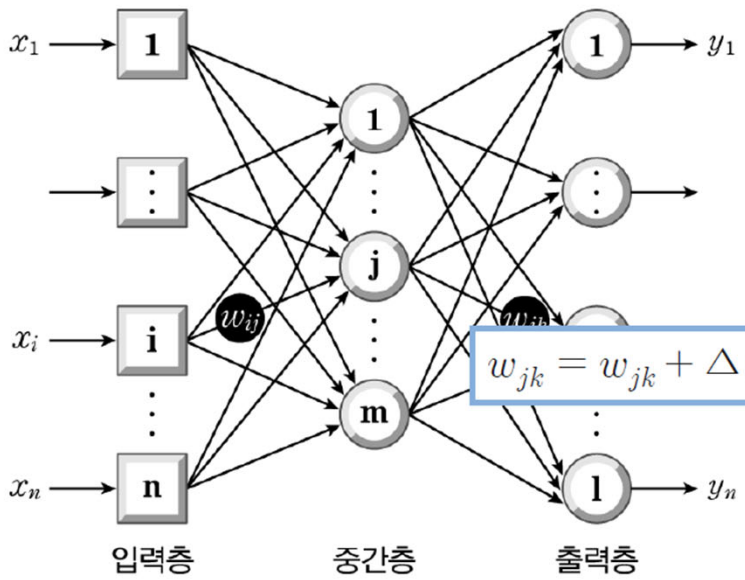
$$y_2 = 1/(1+e^{-(1.0\times 0.9 + 1.0\times 1.0)}) = 0.9 \text{이다.}$$

출력층의 뉴런은 1개뿐이다. 이 뉴런의 출력 y_1을 계산한다.

5장 신경망과 역전파 알고리즘; 학습 training



5장 신경망과 역전파 알고리즘; (오류) 역전파 (학습) 알고리즘 backpropagation alg.



$$e_k = y_{d,k} - y_{a,k}$$

$$\delta_k = y_k \times (1 - y_k) \times e_k$$

$$\Delta w_{jk} = \alpha \times y_j \times \delta_k$$

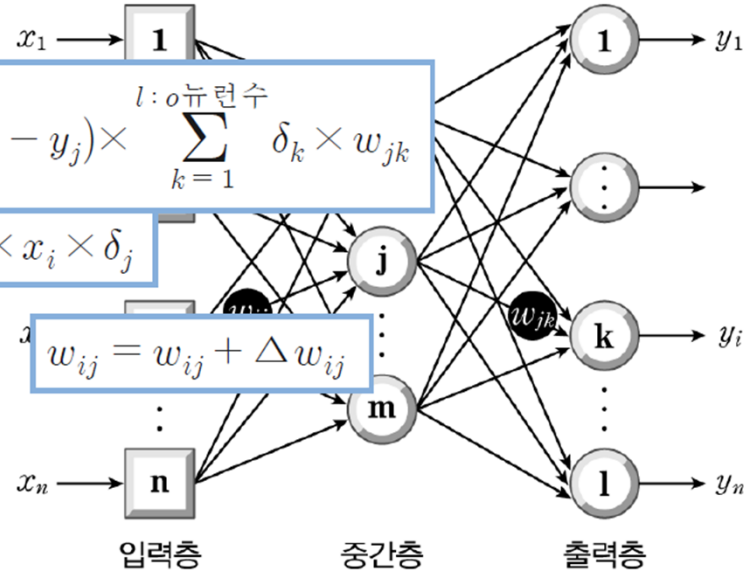
$$w_{jk} = w_{jk} + \Delta w_{jk}$$

5장 신경망과 역전파 알고리즘; (오류) 역전파 (학습) 알고리즘 backpropagation alg.

$$\delta_j = y_j \times (1 - y_j) \times \sum_{k=1}^{l:o\text{뉴런수}} \delta_k \times w_{jk}$$

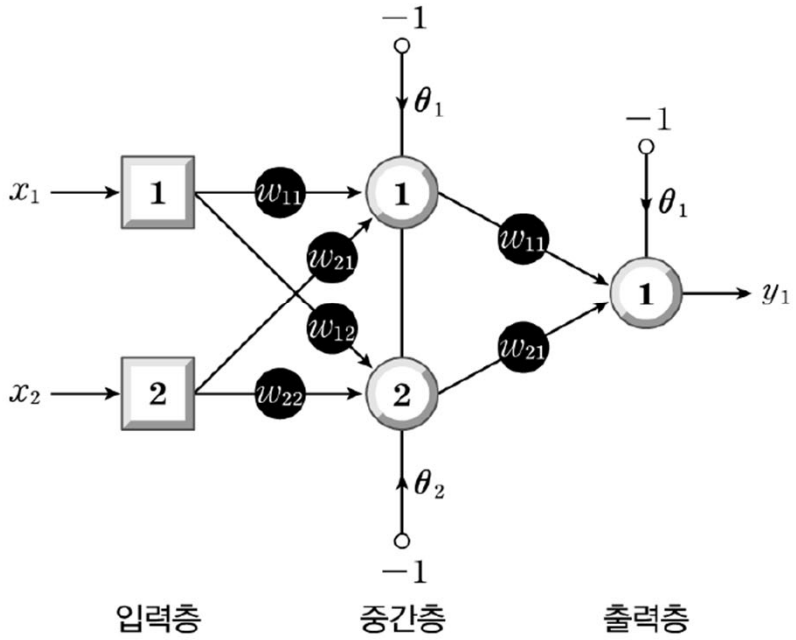
$$\Delta w_{ij} = \alpha \times x_i \times \delta_j$$

$$w_{ij} = w_{ij} + \Delta w_{ij}$$



다층 신경망 학습 예제

하나의 중간층을 포함하고, 중간층과 출력층에 바이어스 입력이 있는 다층 신경망 학습과정을 자세하게 살펴보자. 초기 가중치는 알 수 없으므로 다음과 같이 임의로 설정한다.



- 입력-중간층 가중치

  $w_{11} = 0.5,\ w_{12} = 0.9$

  $w_{21} = 0.4,\ w_{22} = 1.0$

  $\theta_1 = 0.8,\ \theta_2 = 0.1$

- 중간-출력층 가중치

  $w_{11} = -1.2,\ w_{21} = 1.1$

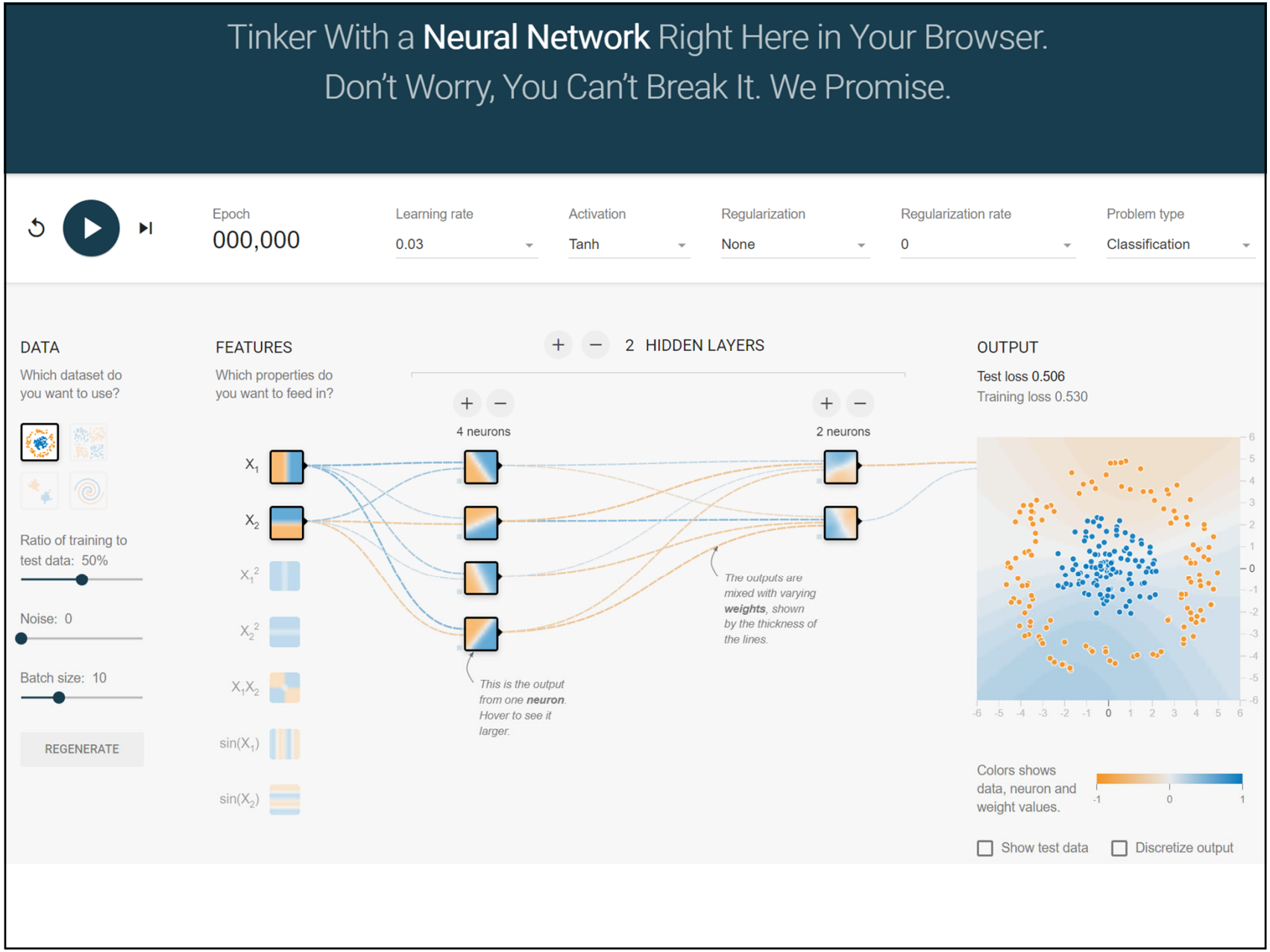
  $\theta_1 = 0.3$

- 입력

  $x_1 = x_2 = 1.0$

- 목표출력 $y_{d,1} = 0.0$

- 학습계수

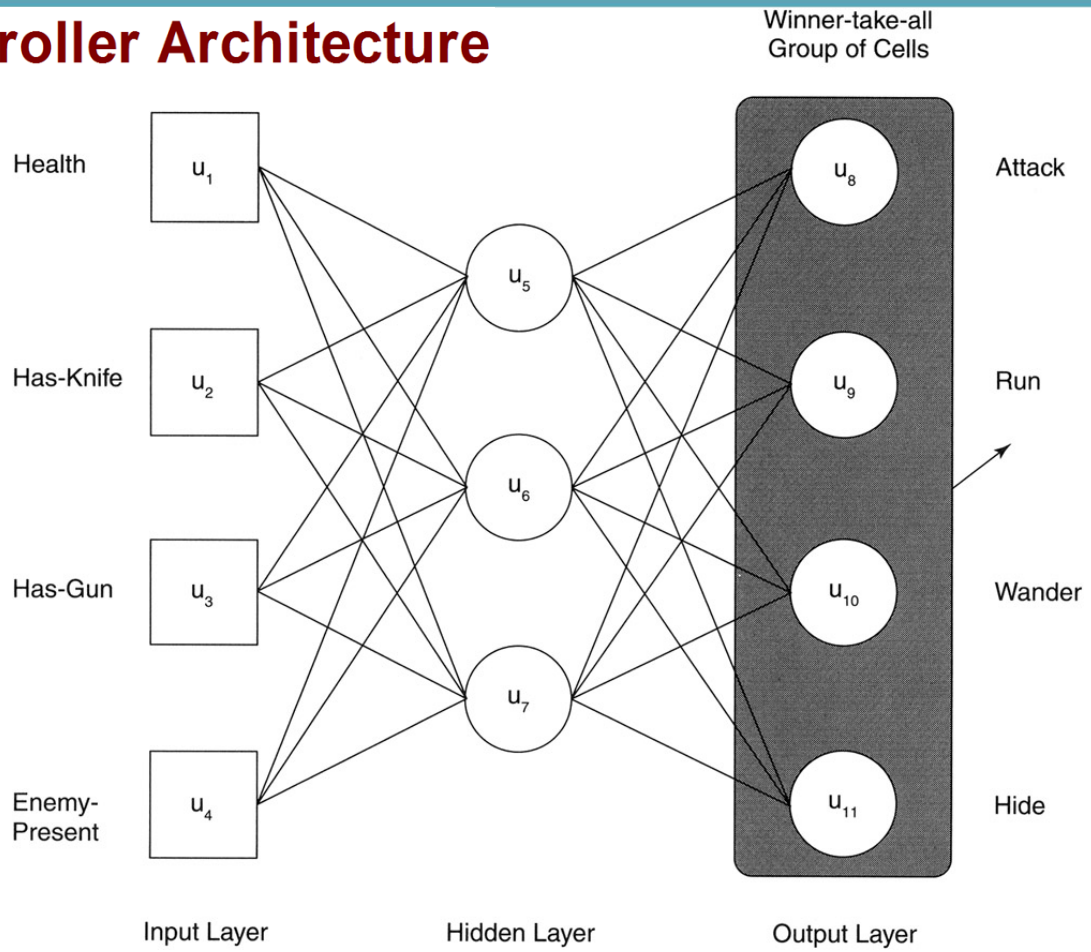


인공지능 및 로봇 실습, 강태원, 컴퓨터공학과, 강릉원주대학교

Evolving Game AI Behaviors

The application that we'll use for backpropagation is the creation of neurocontrollers for game AI characters. A neurocontroller is the name commonly given to neural networks that are used in control applications. In this application, we'll use the neural network to select an action from an available set based upon the current environment perceived by the character. The usage of "character" and "agent" in the following discussion are synonymous.

Neurocontroller Architecture



Training the Neurocontroller

Health	Has-Knife	Has-Gun	Enemies	Behavior
2	0	0	0	Wander
`2	0	0	1	Wander
2	0	1	1	Attack
2	0	1	2	Attack
2	1	0	2	Hide
2	1	0	1	Attack
1	0	0	0	Wander
1	0	0	1	Hide
1	0	1	1	Attack
1	0	1	2	Hide
1	1	0	2	Hide
1	1	0	1	Hide
0	0	0	0	Wander
0	0	0	1	Hide
0	0	1	1	Hide

5장 신경망과 역전파 알고리즘; 응용

Health	*Has-Knife*	*Has-Gun*	*Enemies*	*Behavior*
Good (2)	Yes	Yes	1	Attack
OK (1)	Yes	Yes	2	Hide
Poor (0)	No	No	0	Wander
Poor (0)	Yes	Yes	1	Hide
Good (2)	No	Yes	3	Hide
Good (2)	Yes	No	3	Hide
Poor (0)	Yes	No	3	Run

Source Discussion

Let's now look at the source code, which implements both the backpropagation algorithm for a configurable network topology as well as training and testing for the neurocontroller example.

The global variables are shown in Listing 5.1.

Listing 5.1: Backpropagation Neural Network Globals.

```
#define INPUT_NEURONS                4
#define HIDDEN_NEURONS               3
#define OUTPUT_NEURONS               4

/* Input to Hidden Weights (with Biases) */
double wih[INPUT_NEURONS+1][HIDDEN_NEURONS];

/* Hidden to Output Weights (with Biases) */
double who[HIDDEN_NEURONS+1][OUTPUT_NEURONS];
```