신경회로망의 개괄적 이해

이 정 문

강원대학교 제어계측공학과

1. 서 론

1.1 신경회로망의 정의

인간이 가지고 있는 신체 기능의 대부분은 신경세포(neuron)들의 유기적 결합체인 신경망(neural network)에 의하여 통제된다. 신경망의 중추를 이루고 있는 것이 뇌이다. 인간의 뇌는 대략 150억개의 신경세포들로 구성되며, 각 신경세포는 약 1만개 정도의 연결점(synapse)을 통하여 인접한 다른 신경세포들과 신호를 서로 주고 받는다. 이와 같은 복잡한 구조와 방대한 신호처리 체계는 인간 이외의 다른 생명체에서는 찾아 볼 수 없는 것이다. 물론 인간이 이루어 놓은 그 어떠한 기술문명의 산물도 이와는 견줄 수 없다.

뇌의 구조와 기능에 대하여 구체적으로 알기 위해서 다방면의 연구가 이루어져 왔다. 신경생리학자들은 단일 신경세포나 신경세포들의 집단이 자극에 대하여 어떻게 반응하는지를 연구함으로써 뇌의 생리학적 구조와 기능을 미시적 차원에서 밝히고자 하였다. 또, 심리학자, 언어학자, 인지과학자들은 인간의 언어, 인지능력, 행동 등을 결정하는 뇌의 구조와 기능을 거시적 차원에서 밝히고자 하였다. 이러한 접근 방법들은 생리학적 자극과 반응 사이의 관계, 인지능력, 행동 등에 관한 생물학적 데이타를 설명할 수 있도록 신경망의 구조와 기능을 구체적으로 이해하는 것을 목표로 하고 있다. 그 동안의 연구 결과, 뇌의 구조와 기능에 관한 개괄적인 이해가 가능하게 되었으며, 부분적으로는 상당히 구체적인 성과를 거두기도 하였다. 그러나 우리가 뇌에 대하여 알고 있는 지식은 뇌의 복잡한 구조와 기능에 비추어 볼 때, 아직도 초보적인 수준에 불과하다. 따라서, 뇌에 관하여 완전히 이해하기까지 앞으로도 얼마나 더 많은 시간을 기다려야 할지는 현재로서 미지수이다.

한편, 공학자들의 궁극적인 관심은 뇌의 생리학적 구조와 기능을 밝히는데 있지 않다. 그보다는 뇌의 복잡한 구조적 특징과 이로부터 비롯되는 고도의 신호처리 체계를 연구하여 고난도의 실제 문제에 적용할 수 있는 새로운 연산기법을 개발하는데 그 목표를 두고 있다. 이를 위해서는 보고 듣고 깨닫는 인지능력, 연상기억, 추론, 학습 등의 기능이 뇌에서 어떻게 구현되는가를 이해해야 한다. 이를 바탕으로 하여 von Neumann 방식의 컴퓨터와는 전혀 다른 새로운 개념을 갖는 신경회로 컴퓨터(neural computer)를 개발하고자 하는 것이다. 이는 뇌의 생물학적 구조와 기능을 닮은 시스템을 의미하는 것은 아니다. 단지, 뇌의 구조와 기능을 이해함으로써 뇌가 수행하는 연산기능의 원리로부터 새로운 개념을 추출하여 이를 구현하는 것을 의미한다. 다라서 신경회로 컴퓨터는 뇌의 거시적인 기능을 만들어내는 원리를 응용한 새로운 시스템이라고 볼 수 있다. 이와 같은 시스템을 인공 신경망(artificial neural network)이라고 한다. 공학분야에서 언급되는 신경망은 당연히 인공 신경망을 의미한다. 이후로는 인공 신경망이라는 말 대신에 신경회로망(neural network)이라는 용어를 사용하도록 하겠다.

1.2 신경회로망 연구의 동기

컴퓨터는 여러 부문에서 인간이 가지고 있는 능력의 한계를 확장해 왔다. 행렬을 계산하거나 방정식의 해를 구하고 복잡한 조건을 가진 문제에서 최적의 해를 구하는 등, 산술연산에 있어서 인간의 능력과는 비교할 수 없는 빠른 속도로 매우 정확한 해를 제공한다. 산술논리연산뿐만 아니라 다양하고 방대한 데이타 베이스의 구축과 관리를 통한 정보의 저장 및 제공, 주변기기들을 통한 시스템의 제어 및 자동화, 통신망을 통한 각종 서비스 등, 오늘날의 컴퓨터는 산업

기술 사회에 국한되지 않고 가정이나 개인에 이르기까지 생활 구석구석에 영향을 미치고 있다.

그러나 이와 같이 뛰어난 성능을 가진 컴퓨터라 할지라도 만능일 수는 없다. 서로 다른 거리, 각도, 조명 조건 하에서 얻은 동일한 물체의 영상들로부터 이들이 같은 물체임을 분별해내는 일은 기존의 컴퓨터 비젼 기술로는 매우 어려운 일이 될 것이다. 그러나 인간의 뇌에서 이루어지는 인지능력은 어떤 물체가 놓여있는 거리, 각도, 주위의 밝기 등에 관계없이, 심지어는 장애물에 일부가 가려져 있는 경우에도 별 어려움없이 그 물체를 식별할 수 있게 해 준다. 현재의 패턴인식 시스템으로는 필기 습관이 다른 여러 사람이 쓴 같은 글자, 단어, 또는 문장을 보고 그것이 같은 표현이라는 사실을 인식하기가 쉽지 않다. 그러나 대부분의 사람들은 누가 그것을 썼든지, 심지어 그 일부가 지워졌더라도 그것을 동일한 것으로 인식할 수 있다. 또, 같은 단어나 문장을 서로 다른 사람이 속도, 억양, 크기를 달리 하여 말할 때, 기존의 음성인식 시스템으로 이것을 같은 단어나 문장으로 인식하기란 거의 불가능한 일이다. 그러나 우리들은 주위가 다소 소란할 경우에 조차도 그것들을 같은 단어나 문장으로 인식하는데 그다지 큰 불편을 느끼지 않는다.

우리 주위에는 이와 같이 현재의 디지탈 컴퓨터로는 풀기가 매우 어려운 문제들이 산재해 있다. 따라서 이러한 문제들을 효율적으로 풀 수 있는 강력한 컴퓨터의 필요성이 대두되었으며, 뇌의 구조와 기능을 그 모델로 삼게 된 것이다. 그리고 신경생리학, 생물학, 심리학, 인지과학 등과 같이 뇌의 구조, 기능, 인간의 행동 등에 관한 여러분야의 연구 성과들이 신경회로망의 연구를 뒷받침하고 있다[1-3].

1.3 신경회로망 연구의 발전

신경세포의 기능을 구체적으로 구현한 최초의 모델은 1943년에 McCulloch와 Pitts에 의하여 발표되었다[4]. 그들은 신경세포의 모델인 M-P 신경소자(M-P neuron)가 간단한 논리연산을 수행할 수 있음을 보이고, 뇌의 기능을 이해하는데 있어서 가장 중요하다고 여겨지는 패턴분류 문제를 신경회로망의 적용 예로서 제시하였다[5].

1949년에 Hebb은 학습(learning)이나 훈련(training)이 진행됨에 따라 신경세포들 사이의 결선강도(connection weight)가 변화한다는 생물학적인 학습의 원리를 발견하였다[6]. 즉, 신경세포가 특정한 연결점을 통하여 다른 신경세포의 활동을 반복적으로 자극할 때, 그 연결점의 강도는 점점 강화된다는 것이다. 이와 같은 학습의 원리를 Hebb의 규칙이라고 하며, 신경회로망의 학습에 관한 많은 연구를 촉진시키게 되었다.

학습기능을 갖춘 최초의 신경회로망은 1958년에 Rosenblatt가 제안한 perceptron[7]이라고 볼 수 있다. 이는 perceptron convergence theorem에 의하여 결선강도를 조절할 수 있는 간단한 학습기능을 M-P 신경소자에 부여한 것이다. 뇌의 구조와 기능을 모델링함으로써 기존의 컴퓨터 알고리즘에 의한 연산으로는 해결하기가 곤란했던 문제들에 보다 쉽고 효율적으로 접근할 수 있음을 보였다.

그 후 Widrow와 Hoff는 perceptron을 변형한 Adaline (adaptive linear neuron 또는 adaptive linear element)[8]을 제안하였다. 이것은 적응선형시스템의 일종으로 LMS(Least Mean Squares) 알고리즘, 델타규칙(delta rule), 또는 Widrow-Hoff 규칙이라는 학습규칙에 따라 결선강도가 조정된다. 이와 같이 확실한 수학적 배경을 갖추고 있었기 때문에, 신경회로망에 관한 관심이 식었던 시기에도 적응신호처리 분야에서 지속적인 연구가 이루어질 수 있었다. Adaline은 다수의 연구자들에 의해서 다층 신경회로망인 Madaline(Multiple Adalines)으로 확장 발전되어 오고 있다[9-11].

그러나 1969년에 Minsky와 Papert는 단층 perceptron이 XOR라는 단순한 논리조차도 구현할 수 없다는 근본적인 한계를 수학적으로 규명하였다[12]. 이를 계기로 하여 신경회로망의 연구는 치명적인 영향을 받아서 15년 이상 중단되다시피 하였다. 그 상황에서도 신경세포의 기능을 더 충실히 모델링하고자 하는 소수의 연구는 계속되었다.

절대 다수의 사람들이 신경회로망을 외면하고 있던 1970년대에 Grossberg는 사람의 뇌가 새로운 정보를 지속적으로 학습하면서도 이전에 학습된 정보를 그대로 유지할 수 있는 기능을 가지고 있다는 사실로부터 신경회로망의 학습을 위한 새로운 가설들을 발표하였다[13]. 이러한 개념들은 후에 Carpenter와 Grossberg에 의하여 ART (Adaptive Resonance Theory)[14-16]라고 하는 새로운 신경회로망으로 발전하였다. 한편, Kohonen은 신경세포들이 특정 패턴에 동조되도록 스스로를 조직화해간다는 사실에 근거하여 SOM(self-organizing map)[17]이라는 새로운 신경회로망을 제안하였다.

다층 신경회로망에 관한 연구는 1970년대 초에 Werbos가 역전파(backpropagation) 알고리즘이라는 학습규칙을 발표함으로써 진보의 발판을 마련하였다[18]. 이것은 그 후 Parker[19]에 의해 다시 언급된 후에도 별로 큰 관심을 끌지는 못하였다. 그러다가 1986년에 Rumelhart, Hinton, Williams[20]등이 병렬분산처리에 관한 연구결과의 일부로서 다층 신경회로망과 역전파 알고리즘을 체계적으로 정리하였다. 이를 계기로 하여 역전파 회로망이 신경회로망의 주요 영역으로 확립되었으며, 동시에 침체에 빠져 있던 신경회로망 연구가 다시 활기를 띠게 되었다. 미분가능한 비선형 특성을 가진 다층 신경회로망에 LMS 알고리즘을 확장 적용한 것이 역전파 알고리즘이다. 이를 일반화된 델타규칙(generalized delta rule)이라고도 한다.

한편, 신경소자의 상태(state)를 이산시간변수가 아니라

연속시간변수로 간주하여 신경소자의 동특성을 보다 구체적으로 표현하려는 시도가 이루어졌다. 그리하여 1984년에 Hopfield는 정의된 에너지 함수를 최소화하는 1차 비선형 미분방정식으로 신경소자의 동특성을 모델링하였다. 이것을 Hopfield 회로망[21]이라고 한다.

앞에서 언급한 신경회로망과는 달리 1985년에 Ackley, Hinton, Sejnowski 등은 확률이론과 열역학의 개념에 근거한 Boltzmann machine[22]을 제안하였다. 이 신경회로망은 simulated annealing[23]이라는 확률적 학습규칙에 따라 학습이 이루어진다. 이렇게 함으로써 최적화 과정에서 local mininum에 봉착하지 않고 global minimum에 도달할 수 있게 된다.

기존의 신경회로망들을 기본 구성단위로 하여 새로운 신경회로망을 구성할 수도 있다. 그 중 대표적인 것으로 1987년에 발표된 Hecht-Nielsen의 반전파 회로망(CPN : counterpropagation network)[24]을 들 수 있다. 이것은 instar 및 outstar[25], 경쟁회로망(competitive network)[26] 등을 조합한 것으로서 입력패턴의 특성에 따라 cluster들을 구성한 다음, 각 cluster를 대표하는 출력패턴을 생성한다. 그 외에 잘 알려져 있는 신경회로망으로는 Fukushima의 Neocognitron[27, 28], Kosko의 BAM(Bidirectional Associative Memory)[29-31]등이 있다.

신경회로망의 구현은 주로 컴퓨터 시뮬레이션을 통하여 이루어져 왔다. 그러나 컴퓨터 시뮬레이션에서는 신경회로망의 가장 두드러진 특징인 병렬처리가 불가능하므로 매우 비효율적이다. 따라서 신경회로망을 근간으로 하는 컴퓨터 시스템의 하드웨어 구현에 관해서도 활발한 연구가 진행되고 있다. 최초의 상용 신경회로 컴퓨터는 TRW에서 1985년에 발표한 Mark III라고 볼 수 있다. 이것은 8개의 마이크로프로세서를 이용하여 8,000개의 신경소자와 400,000개의 결선을 구현한 디지탈 신호처리 보드로서 일반 컴퓨터의 주변기기로 동작하도록 되어 있다. 이와 같은 신경회로 컴퓨터로는 TI의 Odyssey, HNC의 Anza 및 Anza Plus, SAIC의 Delta Floating-Point Processor 등이 있다.[32]

이와 더불어 VLSI 설계 및 제조 기술의 발전은 신경회로망의 하드웨어 구현을 위한 연구를 촉진시키게 되었다. 그 대표적인 다수의 연구 결과들은 1992년 5월과 1993년 5월에 IEEE Trans. Neural Networks의 신경회로망 하드웨어 특집에 실려있다.

1.4 신경회로망의 특징 및 기능

신경회로망은 단순한 기능을 가진 무수히 많은 신경소자(neuron) 또는 처리소자(PE : processing element)들이 병렬연결된 연산구조로 되어 있다. 그 특징은 다음과 같다.

(1) 각 신경소자는 다른 신경소자들과 완전히 독립된 기능을 갖는다. 즉, 각 신경소자의 출력은 자신과 연결된 결선을 통하여 직접 전달되는 정보에만 의존할 뿐, 다른 정보들과는 무관하다. 이와 같은 특징으로 인하여 병렬처리가 가능하므로 연산속도가 매우 빠르다. 병렬처리 기능은 신경회로 컴퓨터가 von Neumann 방식의 컴퓨터와 구별되는 가장 두드러진 특징이다.

(2) 신경회로망은 무수히 많은 결선을 가지고 있다. 따라서 정보의 분산표현 및 처리가 가능하다. 또, 중복성(redundancy)이 크므로 fault-tolerant 할 뿐만 아니라 일부의 정보로부터 전체를 얻을 수 있는 연상기억(associative memory) 특성을 갖는다.

(3) 학습이나 훈련을 통해 결선강도를 조정함으로써 새로운 정보를 추가하거나 변경할 수 있는 적응특성을 가지고 있다.

신경회로망은 알고리즘을 프로그램 명령어에 의해 순차적으로 처리하는 von Neumann 방식의 컴퓨터로는 해결하기가 곤란한 다음과 같은 문제들에 적합하다고 알려져 있다.

(1) 패턴분류(pattern classification)-입력패턴들을 그 특성에 따라 분류한다. 이 때, 분류된 집단(class)이나 군(cluster)은 그것을 대표하는 고유한 패턴으로 식별될 수 있다.

(2) 패턴완성(pattern completion)-불완전한 패턴으로부터 완전한 패턴을 생성한다. 잡음제거 문제도 여기에 속한다.

(3) 함수근사(function approximation)-입력패턴에 대응하는 출력패턴을 생성한다. 이 때, 입력과 출력 사이의 대응관계는 임의의 함수 또는 mapping이라도 가능하다. 패턴정합 문제도 넓은 의미에서 여기에 속한다고 볼 수 있다.

이러한 문제들의 공통적인 특징은 다음과 같다. 첫째, 주어진 문제를 수학적 표현으로 정의하기가 곤란하기 때문에 알고리즘에 따라 해를 구하기는 어렵지만, 사람은 그 해법에 관한 know-how를 가지고 있다는 것이다. 둘째, 주어진 문제의 입력패턴과 출력패턴의 예를 충분히 많이 생성해 낼 수 있어서 이들로부터 입력과 출력 사이의 일반적인 관계를 추출하는 것이 가능하다. 이를 위해서는 처리해야 할 정보의 양이 매우 많다. 따라서 정보를 순차적으로 처리하는 컴퓨터로는 해를 구하는 과정이 매우 비효율적이고 시간도 많이 걸린다. 그러므로 병렬구조를 가지고 많은 양의 데이타를 사용하여 학습하는 신경회로망이 이와 같은 문제들을 해결하는데 적합할 것이다. 신경회로망 또는 신경회로 컴퓨터는 실제로 패턴인식, 음성인식, 음성합성, 자동번역, 최적화, 적응제어 등 많은 분야에 응용되고 있다.

2. 신경회로망의 구성요소

생물학적 신경망이 단순한 신경세포들의 대단위의 병렬연결로 이루어져 있듯이, 신경회로망도 단순한 기능을 수행할 수 있는 신경소자들의 수많은 병렬연결로 이루어져 있다. 기능면에 있어서도 생물학적 신경망과 마찬가지로 병렬분산처리를 할 수 있을 뿐만 아니라, 학습이나 훈련을 통해서 결선강도를 조정하여 정보를 추가하거나 변경할 수 있는 적응특성을 가지고 있다. 이와 같이 거시적 차원에서 볼 때, 신경회로망은 생물학적 신경망의 구조와 기능을 닮았다고 할 수 있다. 그러나 신경소자들의 구체적인 기능, 결선을 통한 신호의 전달체계, 결선강도의 조정 메카니즘 등과 같은 미시적 차원의 구조와 기능은 생물학적 신경망과 많은 차이가 있거나, 전혀 다를 수도 있다. 신경회로망을 구성하는 기본적인 요소로는 다음과 같은 세가지를 꼽을 수 있다.

(1) 신경소자(neuron) 또는 처리소자(PE : processing element)
(2) 결선(connection)
(3) 학습규칙(learning rule) 또는 훈련 알고리즘(training algorithm)

2.1 신경소자

구성요소들은 신경회로망의 종류에 따라 매우 다양한 형태를 갖는다. 신경소자는 신경회로망에서 이루어지는 연산을 수행하는 중심 요소이다. 같은 종류의 신경소자들은 보통 층(layer)을 구성한다. 그림 1은 층을 구성하고 있는 i번째 신경소자를 보인 것이다.

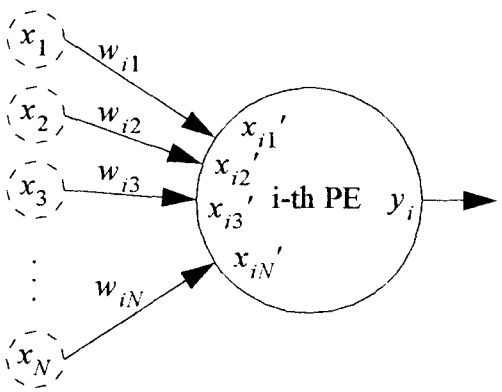


그림 1. 신경소자.

그림에서 $x_1, x_2, \cdots, x_N$은 신경소자에 인가된 입력이다. 이들은 외부로부터 인가된 입력일 수도 있고, 같은 층에 포함된 신경소자들의 출력이거나 다른 층에 있는 신경소자들의 출력, 또는 이들이 섞여있는 것일 수도 있다. 신경소자들간의 결선은 신호가 전파되는 방향의 화살표로 나타낸다. 화살표 위의 w_{ij}는 j번째 입력으로부터 i번째 신경소자에 이르는 결선의 강도를 표시한다. $x_{i1}', x_{i2}', \cdots, x_{iN}'$은 각각 입력 $x_1, x_2, \cdots, x_N$이 결선을 통해 변조되어 i번째 신경소자에 전파된 양을 나타낸다. 이 때, i번째 신경소자의 출력 y_i는 다음과 같이 계산된다.

$$y_i = f\left(\left[\sum_{j=1}^{N} x_{ij}'^{P} \right]^{1/P} - \theta_i \right) \tag{1}$$

여기서 P는 양의 정수로서 보통 1이나 2로 주어진다. θ_i는 i번째 신경소자의 bias 또는 threshold이고, f는 신경소자의 특성을 결정하는 활성함수(activation function)로서 대표적인 것으로는 다음과 같은 것들이 있다.

(1) 선형함수－그림 2(a)

$$f(\lambda) = \begin{cases} \alpha\lambda \; ; \; \lambda \geq 0 \\ -\beta\lambda \; ; \; \lambda < 0 \end{cases} \tag{2.1}$$

α, β는 양수이고, 신경소자의 출력값은 실수 전체범위에 포함된다. 일반적으로 실수 출력값을 생성하는 신경회로망의 출력층 신경소자에 사용된다.

(2) 계단함수－그림 2(b)

$$f(\lambda) = \begin{cases} 1 \; ; \; \lambda \geq 0 \\ 0 \; ; \; \lambda < 0 \end{cases} \tag{2.2a}$$

2진 출력값을 생성한다. Hopfield 회로망이나 BAM 등과 같이 양극성 출력값을 필요로 하는 경우에는 다음과 같은 활성함수를 사용한다.

$$f(\lambda) = \begin{cases} 1 \; ; \; \lambda > 0 \\ 0 \; ; \; \lambda = 0 \\ -1 \; ; \; \lambda < 0 \end{cases} \tag{2.2b}$$

(3) 선형포화함수－그림 2(c)

$$f(\lambda) = \begin{cases} 1 \; ; \; \lambda \geq 1/\alpha \\ \alpha\lambda \; ; \; 0 \leq \lambda < 1/\alpha \\ 0 \; ; \; \lambda < 0 \end{cases} \tag{2.3a}$$

선형함수와 계단함수가 조합된 형태이다. α는 양수이고, 이 값에 따라 직선의 기울기를 조정할 수 있다. 특히 $\alpha \to \infty$이면, 선형포화함수는 (2.2a)와 같은 계단함수가 된다. 양극성 포화값을 필요로 하는 경우에는 다음과 같은 활성함수를 사용하며, 이것은 $\alpha \to \infty$일 때 (2.2b)와 같은 계단함수가 된다.

$$f(\lambda) = \begin{cases} 1 \; ; \; \lambda \geq 1/\alpha \\ \alpha\lambda \; ; \; |\lambda| < 1/\alpha \\ -1 \; ; \; \lambda \leq -1/\alpha \end{cases} \tag{2.3b}$$

(4) sigmoid 함수－그림 2(d)

$$f(\lambda)=\frac{1}{1+exp(-\alpha\lambda)} \tag{2.4a}$$

선형포화함수를 미분가능한 형태로 변형한 것이다. α는 양수이고, 이 값에 따라 곡선의 기울기를 조정할 수 있다. 특히 $\alpha\to\infty$이면, sigmoid 함수는 (2.2a)와 같은 계단함수가 된다. 이 활성함수는 미분가능한 특성을 필요로 하는 신경회로망에 주로 사용된다. 한편, 양극성 포화값을 필요로 하는 경우에는 다음과 같은 활성함수를 사용하며, 이것은 $\alpha\to\infty$일 때 (2.2b)와 같은 계단함수가 된다.

$$f(\lambda)=tanh(-\alpha\lambda) \tag{2.4b}$$

(5) Gaussian 함수－그림 2(e)

$$f(\lambda)=exp\left(-\frac{\lambda^2}{2}\right) \tag{2.5}$$

입력패턴의 특성에 따른 패턴분류를 행할 때, 출력층의 신경소자는 분류된 각 집단에 대응한다. 이 신경소자의 출력값이 특정 집단에 대한 소속도(degree of membership)를 나타내도록 할 경우에 사용한다.

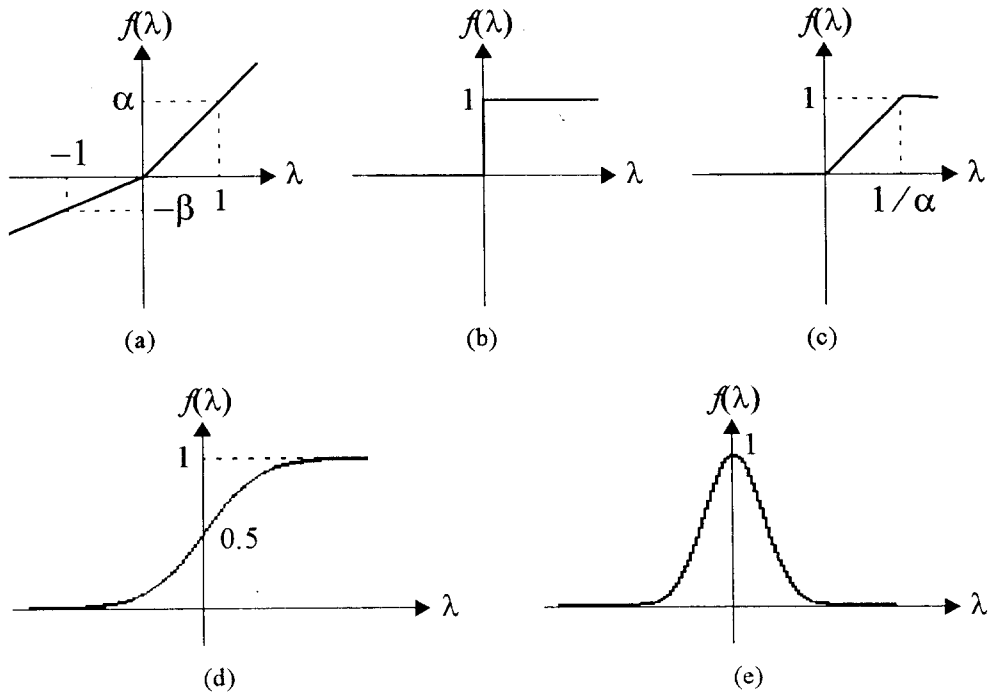


그림 2. 활성함수.

2.2 결　선

신경소자들의 층을 가지고 있는 신경회로망에서는 같은 층에 속한 신경소자들 사이의 층내 결선 (intra－layer connection)과 서로 다른 층에 속한 신경소자들 사이의 층간 결선 (inter－layer connection)이 존재한다. 층내 결선은 신호의 피드백을 야기하므로 신경소자들의 상태는 Hopfield 회로망이나 경쟁회로망 등에서 볼 수 있듯이 동적으로 변화한다. 그러나 층간 결선을 가진 신경소자들의 상태는 일반적으로 정적 변화를 한다고 볼 수 있다.

그림 1에서 각 입력은 결선을 통해 변조되어 i번째 신경소자에 전파된다. 이 변조작용은 결선강도에 의해 결정된다. 그림에서와 같이 결선이 w_{ij}의 결선강도를 갖는 단일 구조로 되어 있는 경우에는 입력패턴이 다음과 같이 변조된다.

$$x_{ij}'=w_{ij}x_j;\quad j=1,2,\cdots,N \tag{3}$$

이 때, $w_{ij}>0$인 것을 여기성 결선(excitatory connection)이라 하고, $w_{ij}<0$인 것을 억제성 결선(inhibitory connection)이라 한다. 단일 결선구조를 갖는 신경회로망에서는 보통 $P=1$로 주어지므로, i번째 신경소자의 출력 y_i는 (1)에 따라

$$y_i=f\left(\sum_{j=1}^{N}w_{ij}x_j-\theta_i\right) \tag{4}$$

이 된다. 신경소자의 bias θ_i는 부가입력 $x_0=1$이 결선강도 $w_{i0}=-\theta_i$인 결선을 통해 i번째 신경소자에 전파되는 것으로

간주할 수 있다. 이 때, 신경소자의 출력은 다음 식과 같이 된다.

$$y_i = f\left(\sum_{j=0}^{N} w_i x_j\right) \tag{5}$$

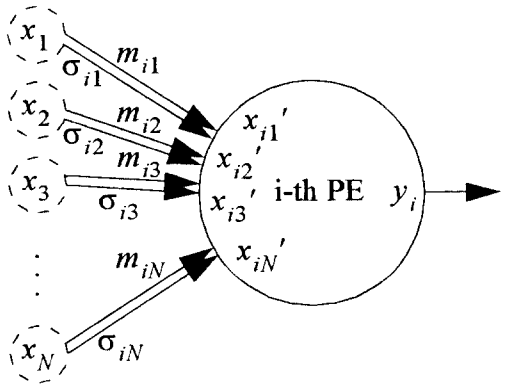


그림 3. 이중 결선을 갖는 신경소자.

현재까지 발표된 대부분의 신경회로망은 이와 같은 단일 결선구조로 되어 있으나, 그림 3에서 보는 바와 같이 결선이 이중 구조로 되어 있을 수도 있다. 이 그림에서는 결선강도가 특정 패턴집단의 평균(mean) m_{ij}와 표준편차(standard deviation) σ_{ij}를 나타낸다. 이 때, 입력 $x_1, x_2, \cdots, x_N$은 각각 다음과 같이 변조된다.

$$x_{ij}' = \frac{x_j - m_{ij}}{\sigma_{ij}} \; ; \; j=1, 2, \cdots, N \tag{6}$$

이중 결선구조를 갖는 신경회로망에서는 보통 $P=2$로 주어지므로, 신경소자의 출력 y_i는 (1)에 의하여

$$y_i = f\left(\sqrt{\sum_{j=1}^{N}\left(\frac{x_j - m_{ij}}{\sigma_{ij}}\right)^2} - \theta_i\right) \tag{7}$$

이 된다. $\theta_i=0$인 경우에 (2.5)와 같은 Gaussian 활성함수를 사용하면, 출력값이

$$y_i = exp\left(-\sum_{j=1}^{N}\left(\frac{x_j - m_{ij}}{\sqrt{2}\sigma_{ij}}\right)^2\right) \tag{8}$$

로 계산된다[33]. 이 신경소자가 입력패턴의 특성에 따라 분류된 패턴의 집단을 대표한다면, y_i는 주어진 입력패턴의 i번째 집단에 대한 소속도라고 볼 수 있다.

이외에도 이중 결선구조를 갖는 신경회로망이 종종 있다. 예로서 입력과 출력 사이의 mapping 관계를 입출력 합성공간에서 정의되는 벡터들로 표현할 수 있다. 이러한 벡터들의 군(cluster)을 RBF (Radial Basis Function)로 나타내는 경우, RBF로 표현되는 군의 중심과 반경이 이중 구조를 갖는 결선의 강도가 된다[34].

3. 신경회로망의 구조

3.1 단일소자 신경회로망

단일 신경소자로 구성된 신경회로망은 그림 4와 같은 구조로 되어 있다. 이것은 perceptron이나 Adaline처럼 독립적으로 사용되기도 하고, instar처럼 CPN(Counterpro-Pagation Network)이나 SOM (Self-Organizing Map)과 같은 다른 신경회로망의 일부가 되기도 한다.

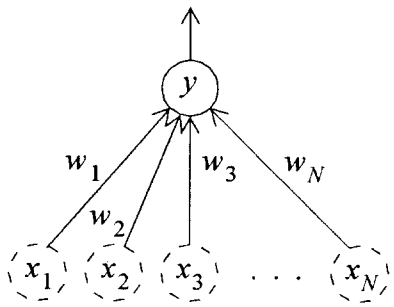


그림 4. 단일소자 신경회로망.

단일소자 신경회로망의 출력값은 다음 식으로 계산된다.

$$y = f\left(\sum_{j=1}^{N} w_j x_j - \theta\right) \tag{9}$$

결선강도 w_j와 bias θ는 회로망에 따라 다른 학습규칙에 의하여 정해진다.

그림 5(a)는 $N=2$인 경우에 단층 perceptron, 혹은 Adaline에 의하여 입력공간이 선형분할(linear separation)된 것을 보여준다. 활성함수 f가 그림 2(b)와 같은 계단함수일 때, ○으로 표시된 입력벡터들에 대해서는 $y=1$이고, ×로 표시된 입력벡터들에 대해서는 $y=0$이다. 이 선형분할의 경계를 결정하는 것이 결선강도와 bias이다. perceptron에서는 perceptron convergence theorem에 의하여 결선강도가 정해지고, Adaline에서는 Widrow-Hoff 규칙에 의하여 결선강도의 학습이 이루어진다.

한편, 그림 5(b)는 역시 $N=2$이고 활성함수가 계단함수인 instar에 의하여 입력패턴들의 집단이 형성된 것을 보여준다. 벡터 A, B, C, D는 2차원 평면에 표시된 입력패턴으로 크기가 1로 정규화되어 있고, 벡터 W는 학습된 결선강도이다. 이 그림에서 ○으로 표시된 벡터 A, B, C의 W방향 성분은 θ보다 커서 $y=1$이 되고, ×로 표시된 벡터 D의 W방향 성분은 θ보다 작기 때문에 $y=0$이 된다. 이것은 이 instar가 W로 대표되는 입력패턴들의 집단을 나타낸다고 할 때, 입력패턴 A, B, C는 이 집단에 속하고, D는 속하지 않음을 의미한다. instar의 결선강도는 그 집단에 속한 입력패턴 A, B, C의 평균치와 같아지도록 학습된다. 이 때, θ가 1에 가까울수록 집단의 범위가 좁게 제한됨을 알 수 있다.

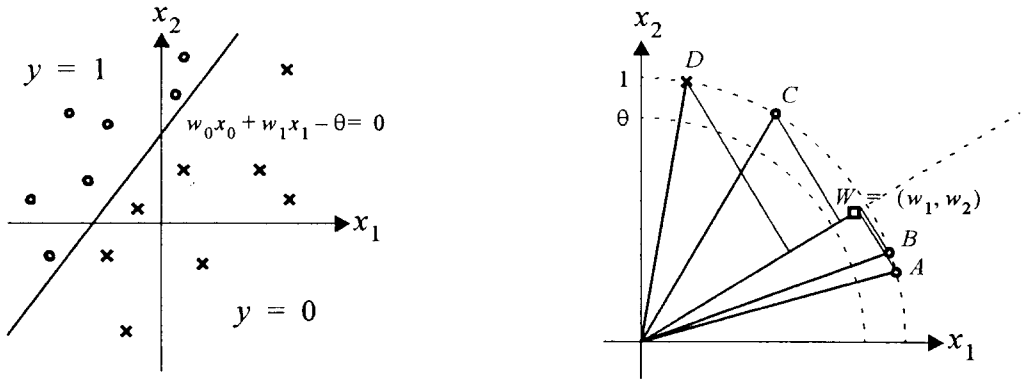


그림 5. 단일소자 신경회로망의 기능.

3.2 단층 신경회로망

가장 간단한 단층 신경회로망은 instar와 쌍대관계에 있는 그림 6(a)의 outstar이다. 이것은 instar와 마찬가지로 보통 다른 신경회로망의 일부가 된다. outstar의 출력값은 선형활성함수를 통하여 다음 식으로 계산된다.

$$y_i = w_i x; \quad i = 1, 2, \cdots, M \tag{10}$$

outstar는 생물학적 조건반사와 유사한 기능을 구현할 수 있는 가장 간단한 신경회로망이다[35]. 결선강도는 $x=1$인 입력이 주어졌을 때, 회로망의 출력이 목표출력과 일치하도록 학습된다.

층간 결선을 갖는 일반적인 단층 신경회로망의 구조는 그림 6(b)와 같다. 이것은 그림 4에 보인 단일 신경소자들이 단순히 하나의 층을 구성하고 있는 형태이다. 이러한 단층 신경회로망은 입력패턴과 출력패턴 사이의 mapping 관계를 제한적으로 구현할 수 있는 기능을 가지고 있다. [36, 37] 이 신경회로망의 출력값은 다음 식으로 계산된다.

$$y_i = f\left(\sum_{j=1}^{N} w_{ij} x_j - \theta_i\right); \quad i = 1, 2, \cdots, M \tag{11}$$

결선강도 w_{ij}와 bias θ_i는 단일소자 신경회로망과 같은 방법으로 정해진다.

층내 결선을 갖는 단층 신경회로망은 그림 6(c)와 같은 구조로 되어 있다. 이것은 연상기억장치를 구현하는 이산 Hopfield 회로망 [38]이나 최적화를 구현하는 연속 Hopfield 회로망 [39]처럼 독립적으로 사용될 수 있다. 또한 Hamming 회로망의 일부인 MAXNET도 이런 구조로 되어 있고, CPN이나 SOM 등의 일부인 경쟁회로망도 층내 결선을 가지고 있다. 층내 결선을 갖는 단층 신경회로망에서는 피드백으로 인하여 동특성이 나타나는 반면, 결선강도는 상수값으로 정해진다.

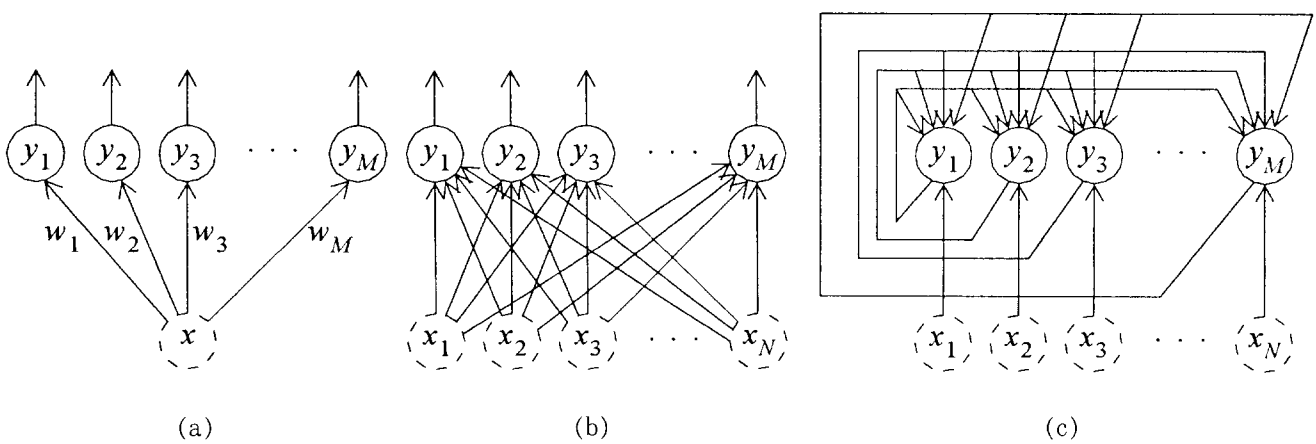


그림 6. 단층 신경회로망.

MAXNET에서의 출력값은 다음과 같은 차분 방정식으로 계산된다.

$$y_i(n+1)=f\left(\sum_{j=1}^{M} w_i y_i(n)-\theta_i\right);$$
$$i=1, 2, \cdots, M \qquad (12)$$

이 때, θ_i는 보통 0이고, 활성함수로는 (2.1)에서 $\alpha=1$, $\beta=0$인 선형함수를 사용한다. 출력의 초기값은 $y_i(0)=x_i$로 주어지며, 결선강도는

$$w_{ij}=\begin{cases} 1; & i=j \\ -\varepsilon; & i\neq j,\ \varepsilon<1/M \end{cases} \qquad (13)$$

와 같이 on-center off-surround 특성을 갖는다. 따라서 MAXNET에서는 입력 x_i가 최대인 신경소자만 양의 출력을 내고, 나머지 출력은 모두 0이 된다. 그림 7은 10개의 신경소자로 구성된 MAXNET의 출력을 나타낸 것이다. 이 예에서는 억제성 결선강도를 $-\varepsilon=-0.08$로 하였다. (a)는 회로망의 입력이고, (b)-(d)는 (12)의 동특성에 따른 출력의 변화이다. $n=7$이후에는 회로망의 상태가 더 이상 변하지 않게 되는데, 최초에 가장 큰 입력이 인가되었던 3번째 신경소자만 출력이 양수로 남아 있음을 보여준다.

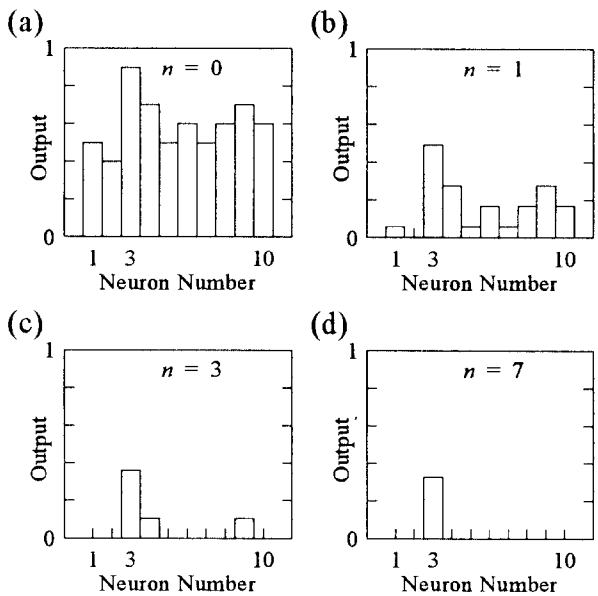


그림 7. MAXNET의 출력.

이산 Hopfield 회로망에서는 다음 식으로 출력값이 계산된다.

$$y_i(n+1)=f\left(\sum_{j=1}^{M} w_{ij} y_i(n)+x_i-\theta_i\right);$$
$$i=1, 2, \cdots, M \qquad (14)$$

이 때, 활성함수 f는 보통 (2.2b)와 같은 양극성 계단함수가 된다. 출력의 초기값은 임의로 주어지고, 결선강도는 다음 식으로 정해진다.

$$w_{ij}=\begin{cases} 0 & ; i=j \\ \sum_S x_i x_j; & i\neq j,\ S\text{: 샘플집합} \end{cases} \qquad (15)$$

$\theta_i=-1$인 8개의 신경소자를 가진 이산 Hopfield 회로망을 예로 들어 본다. 8차원 벡터로 표현되는 2개의 패턴

$$x^1=[1\ -1\ -1\ \ 1\ -1\ \ 1\ \ 1\ -1]^T$$
$$x^2=[1\ \ \ 1\ \ \ 1\ -1\ -1\ -1\ \ 1\ \ \ 1]^T$$

을 저장하면, 결선강도는 다음과 같이 된다.

$$W=[w_{ij}]=\begin{bmatrix} 0 & 0 & 0 & 0 & -2 & 0 & 2 & 0 \\ 0 & 0 & 2 & -2 & 0 & -2 & 0 & 2 \\ 0 & 2 & 0 & -2 & 0 & -2 & 0 & 2 \\ 0 & -2 & -2 & 0 & 0 & 2 & 0 & -2 \\ -2 & 0 & 0 & 0 & 0 & 0 & -2 & 0 \\ 0 & -2 & -2 & 2 & 0 & 0 & 0 & -2 \\ 2 & 0 & 0 & 0 & -2 & 0 & 0 & 0 \\ 0 & 2 & 2 & -2 & 0 & -2 & 0 & 0 \end{bmatrix}$$

결선강도가 결정된 다음, 저장된 2개의 패턴 x, x^2와의 Hamming 거리 [40]가 각각 3과 4인 다음의 입력패턴을 회로망에 인가한다.

$$x=[1\ \ 1\ -1\ \ 1\ \ 1\ -1\ \ 1\ -1]^T$$

출력의 초기값이 모두 0일 경우에 (14)에 의해서 회로망의 출력은 다음과 같이 변화한다.

$$y(1)=[1\ \ \ 1\ -1\ \ 1\ \ \ 1\ -1\ \ 1\ -1]^T$$
$$y(2)=[1\ -1\ -1\ \ 1\ -1\ \ \ 1\ \ 1\ -1]^T$$

$n=2$ 이후에는 저장된 패턴 x^1에 수렴하므로, 자기연상기억(auto-associative memory)이 이루어진 것이다.

한편, 경쟁회로망에 포함된 층내 결선을 갖는 단층 신경회로망이나 연속 Hopfield 회로망의 동특성은 다음과 같은 상태 방정식으로 표현된다.

$$\frac{d}{dt}\xi_i=-\alpha_i\xi_i+\sum_{j=1}^{M} w_i y_i+x_i\ ;\quad i=1, 2, \cdots, M \qquad (16)$$

여기서 α_i는 양의 상수이고, ξ_i는 i번째 신경소자의 내부상태이며, 신경소자의 출력값은 $y_i=f(\xi_i)$로 계산된다. 이 때, 활성함수 f는 보통 sigmoid 함수이다. 경쟁회로망의 층내 결선강도 w_{ij}가 MAXNET과 유사한 on-center off-surround 특성을 갖는 경우에는 입력 x_i가 최대인 신경소자의 출력값만 1이 된다. 그러나 $|i-j| \leq$ C(neighborhood)

에 대해서 결선강도 w_{ij}가 양수인 경우에는 입력 x_i가 최대인 신경소자는 물론, 인접한 신경소자의 출력이 모두 1로 된다. 연속 Hopfield 회로망의 결선강도는 이산 회로망에서와 마찬가지로 (15)로 주어진다.

3.3 다층 신경회로망

다층 신경회로망은 그림 8과 같이 2개 이상의 층을 가지고 있는 신경회로망을 의미한다. 그림 8(a)는 2층 신경회로망인 BAM의 구조이다. BAM의 결선강도는 상수로서 Hopfield 회로망과 유사하게 다음과 같이 정해진다.

$$w_{ij}=\sum_{S} y_i x_j;\quad \text{S: 샘플집합} \tag{17}$$

BAM의 결선은 양방향성이므로 입력층과 출력층의 구별이 없다. 각 신경소자들의 출력값은 다음과 같은 차분 방정식으로 계산된다.

$$y_i(n+1)=f\left(\sum_{j=1}^{N} w_{ij}x_j(n)-\theta_{yi}\right);\quad i=1, 2, \cdots, M$$
$$x_j(n)=f\left(\sum_{i=1}^{M} w_{ij}y_i(n)-\theta_{xj}\right);\quad j=1, 2, \cdots, N \tag{18}$$

신경소자들의 초기값은 $x_j(0)=x_{j0}$ 또는 $y_i(0)=y_{i0}$로 주어진다. 이산 Hopfield 회로망이 자기연상기억 기능을 구현하는데 비해, BAM은 이종연상기억 (hetero-associative memory) 기능을 구현할 수 있다. $\theta_{yi}=0$인 5개의 y층 신경소자와 $\theta_{xj}=0$인 8개의 x층 신경소자들을 가진 BAM을 예로 들어 본다. 8차원 백터로 표현되는 x에 대하여 5차원 패턴 y가 각각 다음과 같이 대응된다고 가정한다.

$$x^1=[1\ -1\ -1\ \ 1\ -1\ \ 1\ \ 1\ -1]^T$$
$$y^1=[1\ -1\ -1\ -1\ -1]^T$$
$$x^2=[1\ \ 1\ \ 1\ -1\ -1\ -1\ \ 1\ \ 1]^T$$
$$y^2=[1\ \ 1\ -1\ \ 1\ -1]^T$$

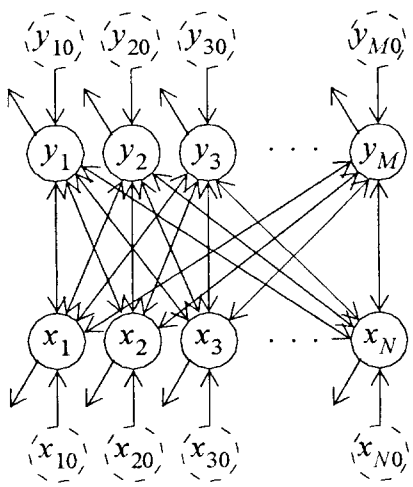
이와 같은 소자와 대응관계를 BAM에 저장하면, (17)에 따라 결선강도가 다음과 같이 정해진다.

$$W=[w_{ij}]=\begin{bmatrix} 2 & 0 & 0 & 0 & -2 & 0 & 2 & 0 \\ 0 & 2 & 2 & -2 & 0 & -2 & 0 & 2 \\ -2 & 0 & 0 & 0 & 2 & 0 & -2 & 0 \\ 0 & 2 & 2 & -2 & 0 & -2 & 0 & 2 \\ -2 & 0 & 0 & 0 & 2 & 0 & -2 & 0 \end{bmatrix}$$

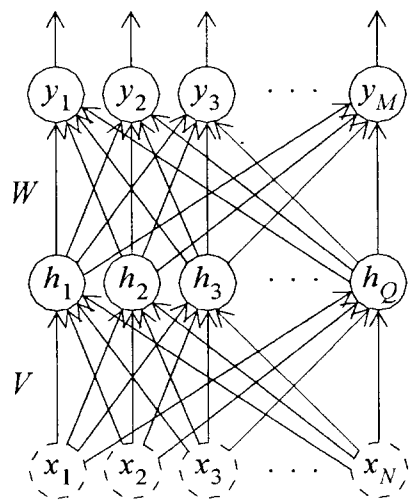
앞절의 예에서와 동일한 입력패턴 x_0를 회로망에 인가하면, (18)에 의해서 회로망의 출력은 다음과 같이 변화한다.

$$y(1)=[1\ -1\ -1\ -1\ -1]^T$$
$$x(1)=[1\ -1\ -1\ \ 1\ -1\ \ 1\ \ 1\ -1]^T$$

$n=1$ 이후에는 회로망의 출력이 저장된 패턴 x^1과 y^1에 바로 수렴하므로, 이종연상기억이 이루어진 것이다.



(a)



(b)

그림 8. 다층 신경회로망.

일반적으로 다층 신경회로망은 그림 8(b)와 같이 입력패턴이 출력층에 전파되기까지 1개 혹은 그 이상의 은닉층 (hidden layer)을 거치도록 되어 있는 신경회로망이다. 다층 신경회로망의 출력값은 다음 식으로 계산된다.

$$y_i=f\left(\sum_{j=1}^{Q} w_{ij}h_j-\theta_{yi}\right);\quad i=1, 2, \cdots, M \tag{19}$$

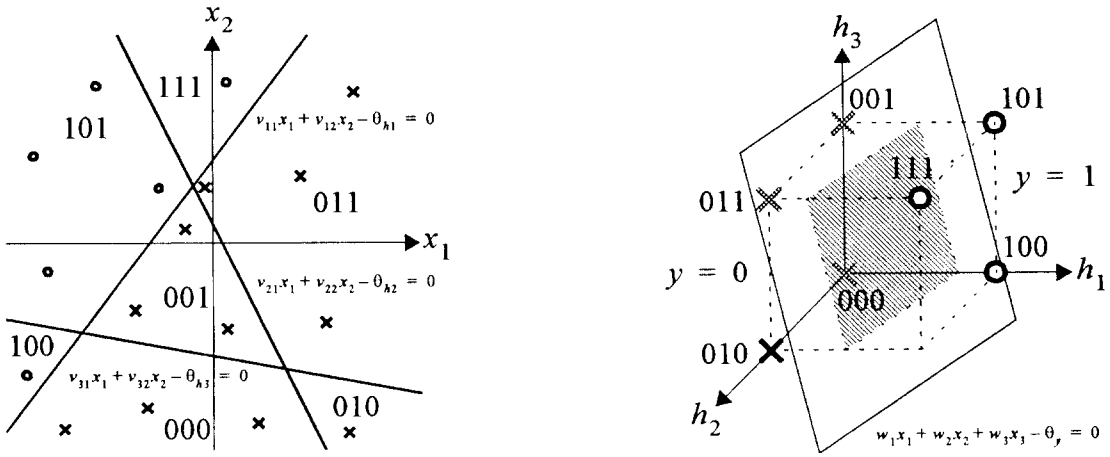
여기서 h_j는 은닉층의 j번째 신경소자의 출력으로서 그림 8(b)와 같이 은닉층이 1개인 경우에는 다음 식으로 계산된

다.

$$h_j = g\left(\sum_{k=1}^{N} v_{jk}x_k - \theta_{hj}\right); \quad j=1, 2, \cdots, Q \tag{20}$$

은닉층은 얼마든지 추가할 수 있으며, 각 은닉층 신경소자의 출력도 역시 (20)와 같은 방법으로 계산할 수 있다.

이와 같은 다층 신경회로망의 가장 전형적인 예는 Madaline, 다층 perceptron, 또는 역전파 회로망으로서 패턴분류, 패턴정합, 함수근사 등에서 매우 다양하게 활용되고 있다. 특히 비선형 활성함수를 사용하면 어떠한 비선형 mapping이라도 구현할 수 있다고 알려져 있다[41]. 이러한 다층 신경회로망은 단층 perceptron을 확장한 형태로서 단층 perceptron의 한계를 극복할 수 있는 우수한 특성을 가지고 있다. 그림 9는 $N=2$인 경우에 3개의 은닉 신경소자를 가진 2층 perceptron이 입력공간을 분할한 모습이다. 먼저 활성함수 g가 그림 2(b)와 같은 계단함수일 때, 입력공간은 그림 9(a)와 같이 7개의 영역으로 나누어진다. 각 영역에 표시된 3 비트의 부호는 해당영역의 입력벡터에 대한 은닉 신경소자의 출력값 h_1 h_2 h_3이다. 따라서 각 영역은 그림 9(b)에 보인 은닉 신경소자의 출력공간에서 7개의 점에 대응한다. 이 때, ○으로 표시된 입력벡터들에 대해서는 $y=1$이고, ×로 표시된 입력벡터들에 대해서는 $y=0$이다. 일반적으로 2층 perceptron은 입력공간을 이 그림에서와 같이 구간 선형경계를 가진 2개의 영역으로 분할할 수 있다. 입력공간을 임의로 분할하기 위해서는 3층 신경회로망이 요구된다[42].



(a) (b)

그림 9. 2층 perceptron에 의한 입력공간의 분할.

결선강도는 단층 perceptron이나 Adaline에서와 같이 주어진 입력에 대한 회로망의 출력과 목표출력 사이의 오차를 최소화하도록 조정되는데, 여기에는 Widrow-Hoff 규칙을 일반화한 역전파 알고리즘이 사용된다. 그 밖에 확률적인 학습규칙인 simulated annealing을 사용하는 Boltzmann machine도 일반적으로 다층 구조를 갖는다.

4. 학습규칙

신경회로망의 학습은 원하는 정보를 회로망에 분산저장하기 위하여 정해진 규칙에 따라 결선강도를 조정해가는 과정이다. 학습을 통하여 결선강도가 정해져야만 신경회로망이 제 기능을 가지게 되는 것이다. 그런데 모든 신경회로망이 학습을 통하여 결선강도를 결정하는 것은 아니다. 일반적으로 instar, outstar, perceptron, Adaline, 역전파 회로망 등과 같이 신호를 한쪽 방향으로 전파하는 신경회로망의 결선들은 학습을 통하여 그 강도가 조정된다. 그러나 MAXNET, Hopfield 회로망 등과 같이 층내 결선 구조를 갖는 경우에는 미리 정해진 고정된 결선강도를 갖는다.

신경회로망의 학습은 크게 지도학습(supervised learning)과 자율학습(unsupervised learning)으로 나누어 볼 수 있다. 지도학습은 입력값과 목표출력으로 이루어진 학습샘플들을 사용하여 샘플입력에 대한 회로망의 출력값과 목표출력 사이의 오차가 최소로 되도록 규칙에 따라 결선강도를 조정하는 방법이다. 따라서 주어진 학습샘플들에 근거하여 입력과 출력 사이의 대응관계를 가장 잘 구현하도록 결선강도가 정해진다. 반면에 자율학습에서는 입력패턴에 대응하는 목표출력이 주어지지 않는다. 이 경우에는 유사한 입력패턴들에 대해서는 각 신경소자들이 유사한 출력을 생성하도록 결선강도를 조정한다.

학습의 종류를 off-line 학습과 on-line 학습으로 나눌 수도 있다. 일반적으로는 주어진 학습샘플들을 사용하여 신경회로망의 성능이 원하는 수준이 될 때까지 결선강도를 조정하는 off-line 학습을 수행하는 것이 보통이다. off-line 학

습에서는 샘플집합 전체에 대한 회로망의 성능을 평가하여 학습종료 시점을 결정하게 되므로 정보를 편중되지 않게 잘 분산저장할 수 있다. 신경회로망의 성능이 일단 원하는 수준에 도달하면, 학습을 종료하여 결선강도를 고정시켜서 사용하게 된다. 따라서 학습종료 후에 새로운 입력패턴이 인가되는 등의 변화가 생기더라도 그것을 결선강도에 반영할 수 없다. 새로운 패턴에 관한 정보를 추가하기 위해서는 그 패턴을 원래의 샘플집합에 포함시켜서 처음부터 다시 학습을 수행해야 한다. 한편, 신경회로망의 성능이 일단 원하는 수준에 도달하더라도 학습을 종료하지 않고 인가되는 입력패턴들에 대해서 결선강도를 계속 조정해가는 on-line 학습은 변화에 적응할 수 있다는 장점이 있다. 그러나 on-line 학습에서는 결선강도가 가장 최근에 인가된 입력패턴에 크게 좌우되므로 이전에 분산저장되어 있던 정보는 점점 약해진다. 이 장에서는 대표적인 몇가지 학습규칙들을 소개하고자 한다.

4.1 Hebb의 규칙

D. O. Hebb이 1949년에 발표한 가설에 근거한 Hebb의 규칙은 신경회로망의 결선강도를 조정하는 가장 단순한 형태의 학습규칙이다. Hebb의 규칙을 정리해 보면 다음과 같다. 즉, 신경소자 A가 활성화되어 그 결과로 이것과 연결되어 있는 신경소자 B가 활성화된다면, 신경소자 A와 B사이의 결선강도는 증가한다는 것이다. 이를 수식으로 표현하면 다음과 같다.

$$W_{BA}^{new} = W_{BA}^{old} + \mu y_A y_B \tag{21}$$

여기서 μ는 학습계수(learning rate)로서 학습의 속도를 결정한다.

이러한 Hebb의 규칙이 신경회로망의 학습규칙으로 그대로 사용되기도 하지만, 여러 가지의 변형된 형태로 사용되기도 한다. 그 대표적인 경우가 instar와 outstar의 학습규칙이다. 그림 4에 보인 instar의 결선강도는 다음과 같은 학습규칙에 의하여 조정된다.

$$w_j(n+1) = w_j(n) + \mu[x_j(n) - w_j(n)]\, y_d(n);$$
$$j = 1, 2, \cdots, N \tag{22}$$

여기서 결선강도의 초기값은 임의로 주어진다. $x_1(n)$, $x_2(n)$, $\cdots$, $x_N(n)$ 및 $y_d(n)$은 주어진 학습샘플 중 n의 시점에서 임의로 선택한 정규화된 입력 및 목표출력값이다. 따라서 (22)는 지도학습을 나타낸다.

그림 10은 $N=2$일 경우에 대하여 활성함수가 계단함수인 instar의 학습과정을 나타낸 것이다. 학습에 사용된 샘플집합은 다음과 같다.

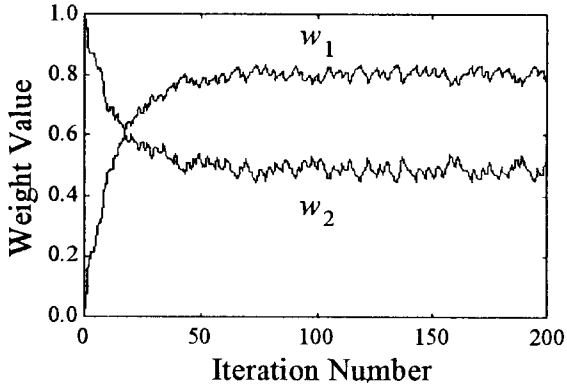


그림 10. Instar의 학습.

$$x^1 = [0.9659 \quad 0.2588]^T \qquad y_d^1 = 1$$
$$x^2 = [0.9397 \quad 0.342\ ]^T \qquad y_d^2 = 1$$
$$x^3 = [0.5 \quad\ 0.866\ ]^T \qquad y_d^3 = 1$$
$$x^4 = [0.1736 \quad 0.9848]^T \qquad y_d^4 = 0$$

초기 결선강도를 $w(0) = [0 \quad 1]^T$로 했을 때, 학습계수를 $\mu = 0.08$로 일정하게 두면 학습이 진행됨에 따라 결선강도가 샘플 x^1, x^2, x^3의 평균치인 $w = [0.8019 \quad 0.4889]^T$를 중심으로 변동하는 것을 볼 수 있다. 이 변동의 폭은 학습계수의 크기에 의해 좌우된다. 따라서 이러한 변동을 줄이기 위해서는 학습이 진행됨에 따라 학습계수를 줄여 주는 것이 바람직하다. 그리고 학습된 instar가 샘플 x^1, x^2, x^3로 이루어진 집합과 샘플 x^4로 이루어진 집합을 분간할 수 있도록 하기 위해서는 bias θ가 다음과 같은 범위 내에 있어야 한다.

$$w^T x^4 < \theta \le \min\{w^T x^1, w^T x^2, w^T x^3\}$$

한편, instar의 결선강도를 다음과 같이 자율학습에 의하여 조정할 수도 있다.

$$w_j(n+1) = w_j(n) + \mu[x_j(n) - w_j(n)]\, y(n);$$
$$j = 1, 2, \cdots, N \tag{23}$$

여기서는 목표출력 $y_d(n)$ 대신 $y(n)$이 사용되는데, 이는 n의 시점에서 주어진 정규화된 입력 $x_1(n)$, $x_2(n)$, $\cdots$, $x_N(n)$에 대한 instar의 출력이다. 또, 결선강도의 초기값은 정규화된 임의의 값으로 주어지고, 출력 $y(n)$은 (9)와 같이

$$y(n) = f\left(\sum_{j=1}^{N} w_j(n)\, x_j(n) - \theta\right) \tag{24}$$

로 계산된다. 이 때, $0 \le \theta \le 1$인 bias 값이 1에 가까울수록 학습결과가 초기 결선강도의 영향을 크게 받는다.

Hebb의 규칙은 지도학습의 형태로 그림 6와 같은 outstar의 학습에도 적용된다. 다음 식으로 표현되는

outstar의 학습규칙은 (22)의 instar 학습규칙과 매우 유사하다.

$$w_i(n+1)=w_i(n)+\mu x(n)[y_{id}(n)-w_i(n)]\ ;\quad i=1, 2, \cdots, M \tag{25}$$

결선강도의 초기값은 역시 임의로 주어진다. 학습샘플 중 n의 시점에서 임의로 선택한 입력 $x(n)$은 보통 1이나 0의 이진값으로 주어지며, $y_{1d}(n), y_{2d}(n), \cdots, y_{Md}(n)$는 그 때의 목표출력값이다.

4.2 경쟁학습

CPN이나 SOM 등과 같은 신경회로망의 일부로 사용되는 경쟁회로망은 다수의 instar들과 층내 결선을 갖는 단층 신경회로망의 조합으로 이루어져 있다. 이것은 그림 5(c)와 같은 단층 신경회로망에서 각 입력들이 instar의 출력으로부터 오는 형태이며, 그림 11과 같이 나타낼 수 있다.

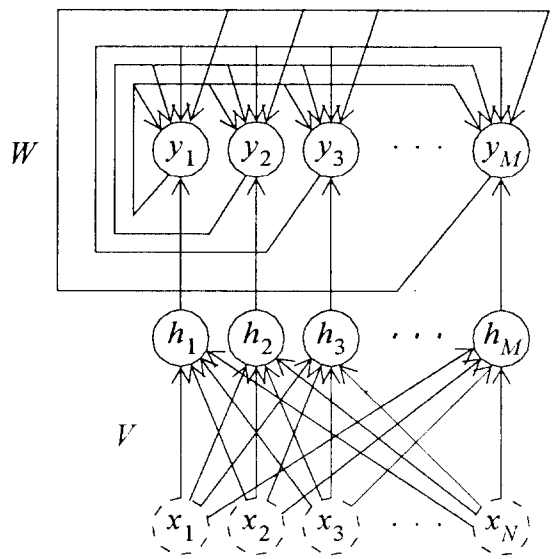


그림 11. 경쟁회로망.

이러한 구조는 그림 6(c)의 단층 신경회로망과 그림 6(b)의 단층 신경회로망을 조합한 형태로도 볼 수 있다.

경쟁회로망에서는 층내 결선의 결선강도가 고정된 값이기 때문에, 학습에 의하여 결선강도가 조정되는 곳은 instar들로 이루어진 부분이다. 따라서 경쟁회로망의 학습규칙은 instar의 학습규칙과 같은 다음 식으로 표현된다.

$$v_{ij}(n+1)=v_{ij}(n)+\mu[x_j(n)-v_{ij}(n)]\,y_i(n);\quad j=1, 2, \cdots, N \tag{26}$$

여기서 결선강도의 초기값은 임의의 값으로 주어지고, 학습계수 μ는 instar의 학습에서와 같이 학습이 진행됨에 따라 줄여가는 것이 좋다. 또, $y_i(n)$은 n의 시점에서 주어진 정규화된 입력 $x_1(n), x_2(n), \cdots, x_N(n)$에 대한 경쟁회로망의 i번째 출력으로서 다음의 미분방정식으로 계산된다.

$$\frac{d}{dt}\xi_i(t)=-\alpha_i\xi_i(t)+\sum_{j=1}^{M} w_{ij}f(\xi_i(t))+h_i(n) \tag{27}$$

$$y_i(n)=f(\lim_{t\to\infty}\xi_i(t)) \qquad ;\quad i=1, 2, \cdots, M$$

여기서 α_i는 양의 상수이고, $\xi_i(t)$는 i번째 신경소자의 내부상태이며, w_{ij}는 층내 결선강도로서 고정된 값이다. CPN의 경우처럼 w_{ij}가 on-center off-surround 특성을 갖는 경우에는 $h_i(n)$이 최대인 신경소자의 출력값만 1이 된다. 결과적으로 보면, c가 $h_c(n)=\max_i\{h_i(n)\}$인 승리소자(winner)를 가리킨다고 할 때, 경쟁회로망의 i번째 출력은

$$y_i(n)=\begin{cases}1; & i=c\\ 0; & i\neq c\end{cases} \tag{28}$$

이다. 그러나 SOM과 같이 $|i-j|\le C$(neighborhood)에 대해서 결선강도 w_{ij}가 양수인 경우에는 $h_i(n)$이 최대인 신경소자를 비롯하여 인접한 신경소자의 출력이 다음과 같이 모두 1로 된다.

$$y_i(n)=\begin{cases}1; & |i-c|\le C\\ 0; & |i-c|>C\end{cases} \tag{29}$$

SOM의 자기조직화 기능은 이 C값에 따라 좌우된다. 학습초기에는 neighborhood의 범위를 넓게 하여 신경소자들의 번호순으로 결선강도가 대강 배정되도록 한다. 이후 학습에 진행됨에 따라 neighborhood의 범위를 좁혀서 결선강도가 미세조정될 수 있게 하고, 결국에는 on-center off-surround 특성을 갖게 하는 것이 바람직하다. 경쟁회로망을 컴퓨터 시뮬레이션하는 경우에는 (27)을 계산할 필요없이, 그 구성요소인 instar의 출력값들만 비교하여 승리소자를 결정한 후 (28)이나 (29)에 의하여 출력값을 구하면 된다. 그리고 승리소자를 결정하는 instar의 출력 $h_i(n)$은 (24)와 마찬가지로

$$h_i(n)=g\left(\sum_{j=1}^{M} v_{ij}(n)\ x_j(n)-\theta_i\right) \tag{30}$$

에 의해서 계산된다. 그리고, 활성함수 g는 일반적으로 단조증가함수이기 때문에, 승리소자를 결정하기 위한 목적이라면, $\theta_i=0$인 선형함수로 충분하다.

그림 12는 $N=2$인 경우에 대하여 20개의 신경소자를 가진 SOM의 학습과정을 보인 것이다. 여기서는 단위원상에 균일하게 분포된 100개의 정규화된 입력벡터를 사용하여, $\mu=0.1$인 일정한 학습계수로 학습하였다. 또, C는 초기값을 10으로 하고, 100회의 반복마다 감소시켜서 $n=1000$ 이후에는 신경소자들의 층내 결선이 on-center off-surround 특성을 갖도록 하였다. 그림에서 결선강도는 ○으로 표시하였

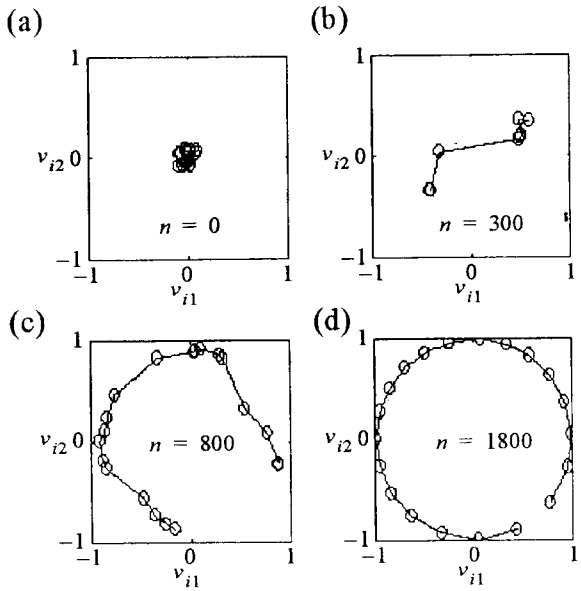


그림 12. SOM의 자기조직화 과정.

고, 이것을 해당 신경소자의 번호순으로 연결하여 상대적인 위치관계를 나타내도록 하였다. (a)는 임의로 주어진 결선강도의 초기값이고, (b)－(d)는 학습의 진행에 따라 결선강도가 신경소자의 순서대로 자기조직화되는 과정을 보여주고 있다.

4.3 Widrow-Hoff 규칙

perceptron이나 Adaline은 그림 4와 같이 단일소자 신경회로망으로 사용되기도 하지만, 보통은 그림 6(b)처럼 층을 이룬 단층 신경회로망의 형태로 사용된다. 특히 적응 신호처리 등의 분야에서는 활성함수가 선형함수인 ALC (Adaptive Linear Combiner)의 형태로 흔히 사용된다. ALC는 선형 신경회로망으로서 그 결선강도를 조정하는 학습규칙은 LMS에 근거한 Widrow-Hoff 규칙이다. 그리고 이를 gradient 방법 혹은 최대경사법(steepest descent method)에 따라 순환형(recursive) 알고리즘으로 구현한 것이 델타규칙이다.

Widrow-Hoff 규칙에 따라 ALC와 같은 단층 신경회로망의 결선강도를 결정하기 위해서는 입력패턴 $x_1, x_2, \cdots, x_N$과 그에 대응하는 목표출력 $y_{1d}, y_{2d}, \cdots, y_{Md}$로 구성되는 학습 샘플들이 필요하다. 입력패턴이 주어지면 신경회로망의 출력은 (11)로 계산되는데, bias를 부가입력으로 대체한 (5)를 사용하는 것이 편리하다. 활성함수가 선형함수인 경우의 출력은 다음 식으로 계산된다.

$$y_i=\sum_{j=0}^{N} w_{ij}x_j \tag{31}$$

이 때, 주어진 입력패턴에 대한 신경회로망의 출력과 목표출력 사이의 제곱오차를 다음 식으로 정의한다.

$$E=\frac{1}{2}\sum_{i=1}^{M}(y_{id}-y_i)^2 \tag{32}$$

또, 샘플집합 전체의 평균제곱오차 (MSE : Mean Square Error)는 다음과 같이 정의한다.

$$E_T=\langle E\rangle_S=\frac{1}{2}\left\langle\sum_{i=1}^{M}(y_{id}-y_i)^2\right\rangle_S \tag{33}$$

여기서 $\langle\cdot\rangle_S$는 샘플집합 S에 관한 평균을 의미한다.

조정하고자 하는 결선강도 w_{ij}에 관한 오차의 gradient는 (33)으로부터 다음과 같이 구해진다.

$$\begin{aligned}\frac{\partial E_T}{\partial w_{ij}}&=\left\langle\frac{\partial E}{\partial w_{ij}}\right\rangle_S\\&=\frac{1}{2}\left\langle\frac{\partial}{\partial w_{ij}}\sum_{i=1}^{M}(y_{id}-\sum_{j=0}^{M}w_{ij}x_j)^2\right\rangle_S\\&=-\left\langle(y_{id}-\sum_{j=0}^{N}w_{ij}x_j)x_j\right\rangle_S\\&=-\langle(y_{id}-y_i)x_j\rangle_S\end{aligned} \tag{34}$$

따라서 결선강도는 다음과 같은 순환형 gradient 알고리즘에 의하여 계산된다.

$$\begin{aligned}w_{ij}(n+1)&=w_{ij}(n)-\eta\frac{\partial E_T}{\partial w_{ij}}\\&=w_{ij}(n)+\eta\langle(y_{id}-y_i)x_j\rangle_S\end{aligned} \tag{35}$$

여기서 η는 학습계수 (learning rate)로서 학습의 속도를 결정한다. (34)는 매번 샘플집합 전체를 다 사용하는 off-line 알고리즘이다.

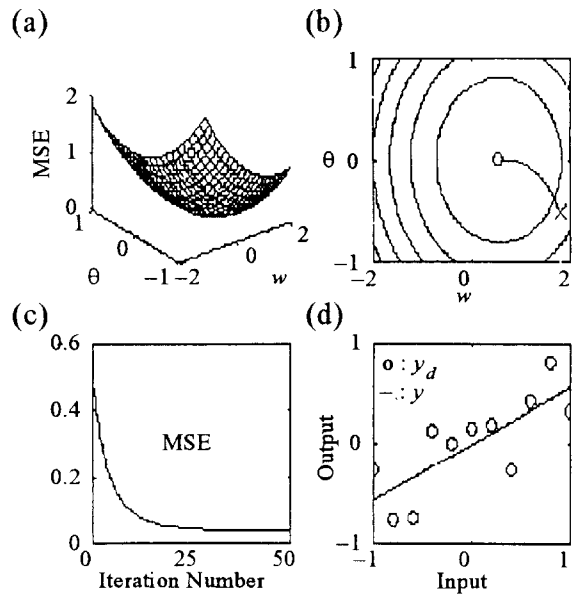


그림 13. ALC의 학습.

그림 13은 단일소자로 구성된 ALC를 사용하여 다음과 같은 11개의 샘플들의 입출력 관계를 표현한 예이다.

$$S=\{(x, y_d) \mid \ (-1, -0.259), (-0.8, -0.7547), (-0.6, -0.743), (-0.4, 0.1226), (-0.2, -0.0073), (0, 0.1534), (0.2, 0.1815), (0.4, -0.2673), (0.6, 0.4216), (0.8, 0.8047), (1, 0.3227)\}$$

그림에서 (a)는 구간 [−2, 2]에서의 결선강도와 구간 [−1, 1]에서의 bias에 대한 회로망의 평균제곱오차를 나타낸 곡면이고, (b)는 이에 대응하는 contour이다. 여기서 ×표시는 초기 결선강도 및 bias이고, ○ 표시는 해석적으로 구한 LMS 해의 위치, 즉 최소오차 지점의 결선강도 및 bias이다. 이 두 지점을 연결한 곡선은 Widrow-Hoff 규칙에 따른 학습에 의해서 결선강도와 bias값이 초기위치에서 최소오차 지점으로 이동해가는 과정을 나타낸 것이다. 이 때, $\eta=0.02$인 학습계수를 사용하였는데, (c)의 반복회수에 대한 MSE의 감소를 참조하면, 약 20회 정도의 반복학습으로 최적해에 거의 도달한 것을 알 수 있다. 50회의 반복학습 후에 회로망이 표현하는 입출력 관계가 (d)에 나와 있다. 여기서 ○로 표시된 점들은 학습샘플이고, ALC에서 학습한 입출력 관계는 실선으로 표시되어 있다.

샘플집합 전체에 대한 평균제곱오차의 gradient 대신, 임의로 선택한 샘플 한 개에 대한 오차의 gradient를 사용하여 근사화하면 다음과 같은 학습규칙이 얻어진다.

$$w_i(n+1)=w_i(n)+\eta[y_{id}(n)-y_i(n)]\,x_i(n) \qquad (36)$$

$x_1(n)$, $x_2(n)$, ⋯, $x_N(n)$ 및 $y_{1d}(n)$, $y_{2d}(n)$, ⋯, $y_{Md}(n)$는 학습샘플 중 n의 시점에서 임의로 선택한 입력 및 목표출력 값이다. 이 때, 신경회로망의 출력은 다음과 같이 계산된다.

$$y_i(n)=\sum_{j=0}^{N} w_{ij}(n)\,x_j(n); \quad i=1, 2, \cdots, M \qquad (37)$$

(36)과 같은 학습규칙은 매번 한 개의 학습샘플만을 사용해서 결선강도를 조정해 나가므로 on−line 학습에 사용할 수 있다. 동일한 샘플집합을 사용해서 학습하는 경우에 (35)보다 계산량이 훨씬 적지만, 성능면에서는 별로 뒤지지 않기 때문에 매우 효율적인 학습규칙이라고 할 수 있다.

4.4 역전파 학습규칙

그림 8(b)와 같은 역전파 회로망 (backpropagation network)은 단층 perceptron을 확장한 형태로서 단층 perceptron의 한계를 극복할 수 있는 우수한 특성을 가지고 있다. 이와 같은 특성은 신경소자의 활성함수가 가지고 있는 비선형성에 기인한다. 그러나 이러한 비선형성으로 인하여 단층 perceptron에 비해 학습하기가 훨씬 어렵다. 역전파 회로망과 같은 다층 perceptron을 학습하는 가장 대표적인 학습규칙은 역전파 알고리즘이다. 이것은 미분가능한 비선형 활성함수를 가진 다층 perceptron에 LMS 방법을 확장하여 적용한 것이다. 이것을 일반화된 델타규칙(generalized delta rule)이라고도 한다.

역전파 알고리즘에 따라 다층 신경회로망의 결선강도를 결정하기 위해서는 입력패턴과 그에 대응하는 목표출력들로 구성되는 학습샘플들이 필요하다. 입력패턴에 대한 신경회로망의 출력을 다음과 같이 표현할 수 있다.

$$y_i=f(\psi_i); \ \psi_i=\sum_{j=0}^{Q} w_{ij}h_j; \ i=1, 2, \cdots, M \qquad (38)$$

여기서 h_j는 은닉층의 j번째 신경소자의 출력으로서 은닉층이 1개인 경우에는 다음과 같이 계산된다.

$$h_j=g(\phi_j); \ \phi_j=\sum_{k=0}^{N} v_{jk}\,x_k; \ j=1, 2, \cdots, Q \qquad (39)$$

회로망의 출력과 목표출력 사이의 제곱오차와 샘플집합 전체의 평균제곱오차는 단층 perceptron에서와 마찬가지로 (32) 및 (33)으로 각각 정의된다.

출력층 결선강도 w_{ij}에 관한 전체 오차의 gradient는 다음과 같이 구해진다.

$$\begin{aligned}\frac{\partial E_T}{\partial w_{ij}} &= \left\langle \frac{\partial E}{\partial w_{ij}} \right\rangle_S \\ &= \frac{1}{2} \left\langle \frac{\partial}{\partial w_{ij}} \sum_{i=1}^{M}(y_{id}-y_i)^2 \right\rangle_S \\ &= -\left\langle (y_{id}-y_i)\frac{\partial y_i}{\partial \psi_i}\frac{\partial \psi_i}{\partial w_{ij}} \right\rangle_S\end{aligned} \qquad (40)$$

(38)로부터

$$\begin{aligned}\frac{\partial y_i}{\partial \psi_i} &= \frac{\partial f(\psi_i)}{\partial \psi_i} \\ \frac{\partial \psi_i}{\partial w_{ij}} &= \frac{\partial}{\partial w_{ij}} \sum_{j=0}^{Q} w_{ij}h_j = h_j\end{aligned} \qquad (41)$$

이 되므로 이를 (40)에 대입하면 다음 식과 같이 된다.

$$\frac{\partial E_T}{\partial w_{ij}} = -\left\langle (y_{id}-y_i)\frac{\partial f(\psi_i)}{\partial \psi_i}\,h_j \right\rangle_S = -\langle \delta_{yi}h_j \rangle_S \qquad (42)$$

여기서 δ_{yi}는 출력층 i번째 신경소자의 일반화된 오차로서 다음과 같이 정의된다.

$$\delta_{yi}=(y_{id}-y_i)\,\frac{\partial f(\psi_i)}{\partial \psi_i} \qquad (43)$$

마찬가지로 은닉층 결선강도 v_{jk}에 관한 전체 오차의 gradient는 다음과 같이 구해진다.

$$\begin{aligned}\frac{\partial E_T}{\partial w_{jk}} &= \left\langle \frac{\partial E}{\partial v_{jk}} \right\rangle_S \\ &= \frac{1}{2} \left\langle \frac{\partial}{\partial v_{jk}} \sum_{i=1}^{M}(y_{id}-y_i)^2 \right\rangle_S\end{aligned}$$

$$= - \left\langle \sum_{i=1}^{M} (y_{id} - y_i) \frac{\partial y_i}{\partial \psi_i} \frac{\partial \psi_i}{\partial h_j} \frac{\partial h_j}{\partial \phi_j} \frac{\partial \phi_j}{\partial v_{jk}} \right\rangle_S \quad (44)$$
$$= - \left\langle \sum_{i=1}^{M} \delta_{yi} \frac{\partial \psi_i}{\partial h_j} \frac{\partial h_j}{\partial \phi_j} \frac{\partial \phi_j}{\partial v_{jk}} \right\rangle_S$$

(38)과 (39)로부터

$$\frac{\partial \psi_i}{\partial h_j} = \frac{\partial}{\partial h_j} \sum_{j=0}^{Q} w_{ij} h_j = w_{ij}$$
$$\frac{\partial h_j}{\partial \phi_j} = \frac{\partial g(\phi_j)}{\partial \phi_j} \quad (45)$$
$$\frac{\partial \phi_j}{\partial v_{jk}} = \frac{\partial}{\partial v_{jk}} \sum_{k=0}^{N} v_{jk} x_k = x_k$$

이 되고, 이로부터 (44)를 다음과 같이 쓸 수 있다.

$$\frac{\partial E_T}{\partial v_{jk}} = - \left\langle \sum_{i=1}^{M} \delta_{yi} w_{ij} \frac{\partial g(\phi_j)}{\partial \phi_j} x_k \right\rangle_S = - \langle \delta_{hj} x_k \rangle \quad (46)$$

여기서 δ_{hj}는 은닉층 j번째 신경소자의 일반화된 오차로서

$$\delta_{hj} = \sum_{i=1}^{M} \delta_{yi} w_{ij} \frac{\partial g(\phi_j)}{\partial \phi_j} \quad (47)$$

로 정의되는데, 우변의 끝에 있는 편미분항을 제외한 나머지 부분은 출력층의 일반화된 오차가 은닉층의 j번째 신경소자에 역으로 전파되어 온 양이다. (42)와 (46)을 이용하여 결선강도의 조정을 위한 계산 식을 다음과 같이 쓸 수 있다.

$$w_{ij}(n+1) = w_{ij}(n) - \eta \frac{\partial E_T}{\partial w_{ij}}$$
$$= w_{ij}(n) + \eta \langle \delta_{yi} h_j \rangle_S \quad (48)$$
$$v_{jk}(n+1) = v_{jk}(n) - \eta \frac{\partial E_T}{\partial v_{jk}}$$
$$= v_{jk}(n) + \eta \langle \delta_{hj} x_k \rangle_S$$

이 식은 매번 샘플집합 전체를 다 사용하는 off-line 알고리즘이다. Widrow-Hoff 학습규칙과 마찬가지로 샘플집합 전체에 대한 오차의 gradient를 학습샘플 한 개에 대한 오차의 gradient로 근사화하면 다음과 같은 on-line 학습규칙을 얻을 수 있다.

$$w_{ij}(n+1) = w_{ij}(n) + \eta \delta_{yi}(n) h_j(n) \quad (49)$$
$$v_{jk}(n+1) = v_{ik}(n) + \eta \delta_{hj}(n) x_k(n)$$

(48)과 (49)는 은닉층이 하나인 경우의 학습규칙이지만, 동일한 방법을 사용하여 여러개의 은닉층을 가진 경우로 확장이 가능하다.

그림 14는 앞절의 예에서와 동일한 샘플집합이 주어졌을 때, 하나의 은닉층을 가진 역전파 회로망을 사용하여 입출력 관계를 표현하는 함수근사의 예이다. 여기서 은닉층은 (2.4b)와 같은 활성함수를 갖는 5개의 신경소자로 되어있고, 학습계수는 $\eta = 0.2$로 하였다. (a)는 반복학습에 따른 MSE의 감소를 보여주는데, 학습속도가 느리고 일정하지도 않음을 알 수 있다. 그러나 상당히 오랜 반복학습을 성공적으로 마치면, (d)에서 보듯이 비선형성이 매우 강한 함수라도 정확하게 일치시킬 수 있다. 일반적으로 역전파 알고리즘은 속도가 느리고, 결선강도 및 bias의 초기값이 학습결과에 큰 영향을 끼친다. 뿐만 아니라 학습계수, 은닉층 신경소자의 수 등도 매우 중요한 문제가 된다. 따라서 이들에 관한 연구가 많이 이루어지고 있다.

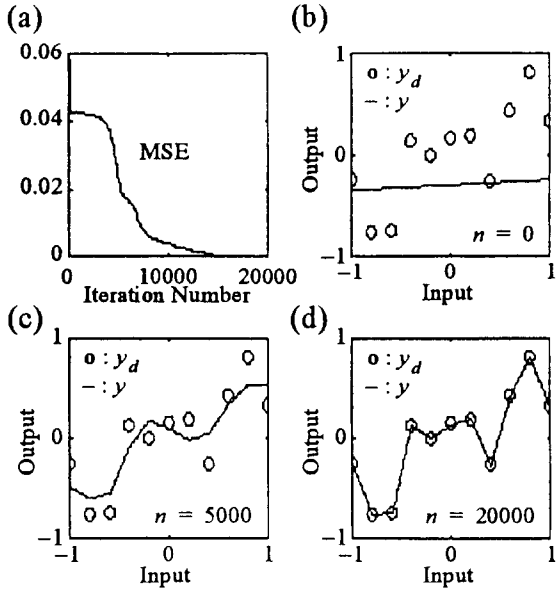


그림 14. 역전파 회로망의 학습.

5. 맺는 말

신경회로망은 기존의 순차처리형 컴퓨터가 해결하기 곤란한 많은 문제들을 효율적으로 해결해 주는 강력한 연산도구임에는 틀림없다. 하지만 이것이 순차처리형 컴퓨터를 완전히 대체할 수 있는 것은 아니다. 현재까지의 신경회로망은 앞에서 살펴본 바와 같이 패턴인식 등의 기능은 매우 뛰어나지만 산술논리연산 기능은 그렇지 못하다. 수학적으로 명확하게 정의된 문제를 푸는데 필요한 산술논리연산 기능에 있어서는 순차처리형 컴퓨터가 신경회로망에 비해서 훨씬 우수하다. 따라서 양자는 상호배타적인 관계가 아니라 상호보완적인 관계에 있는 것이다. 그리고, 신경회로망의 가장 큰 잠재력은 병렬구조에 기인한 고속 연산이 가능하다는 것이다. 그러므로 신경회로망의 기능이 충분히 활용되기 위해서는 하드웨어의 개발이 뒷받침되어야 한다.

현재까지 연구되어 온 신경회로망의 종류와 그 응용에 관

한 학술논문은 전세계적으로 수천편에 달하며, 그 증가율은 매년 높아지고 있다. 이들은 IEEE와 INNS의 학술지와 학술발표회 등에서 발표되고 있다. 또한 대표적인 연구 결과들을 정리해 놓은 자료들[43-45]도 참고가 될 만하다.

참 고 문 헌

[1] S. Grossberg, "How does the brain build a cognitive code?," *Psychol. Rev.*, vol. 87, pp. 1-51, 1980.

[2] G. Shepherd, *Neurobiology*, Oxford Univ. Press, New York, NY, 1983.

[3] D. E. Rumelhart and J. L. McClelland, *Parallel Distributed Processing : Explorations in the Microstructure of Cognition*, MIT Press, Cambridge, MA, 1986.

[4] W. S. McCulloch and W. H. Pitts, "A logical calculus for the ideas immanent in nervous activity," *Bulletin of Mathematical Biophysics*, vol. 5, pp. 115-133, 1943.

[5] W. H. Pitts and W. S. McCulloch, "How we know universals : The perception of auditory and visual forms," *Bulletin of Mathematical Biophysics*, vol. 9, pp. 127-147, 1947.

[6] D. O. Hebb, *The Organization of Behavior*, John Wiley, New York, NY, 1949.

[7] F. Rosenblatt, "The perceptron, a probabilistic model for information storage and organization in the brain," *Psychol. Rev.*, vol. 65, pp. 386-408, 1958.

[8] B. Widrow and M. E. Hoff, "Adaptive switching circuits," *1960 IRE WESCON Convention Record*, New York, NY, pp. 96-104, 1960.

[9] B. Widrow, "Generalization and information storage in networks of adaline neurons", *Self-Organizing Systems*, Spartan Books, Washington DC, pp. 435-461, 1962.

[10] R. Winter and B. Widrow, "MADALINE RULE II : A training algorithm for neural networks," *Proc. ICNN*, San Diego, CA, pp. I401-I408, July 1988.

[11] D. Andes, B. Widrow, M. Lehr, and E. Wan, "MRIII : A robust algorithm for training alalog neural networks," *Proc. IJCNN*, pp. I533-I536, Jan. 1990.

[12] M. Minsky and S. Papert, *Perceptron : An Introduction to Computational Geometry*, MIT Press, Cambridge, MA, 1969.

[13] S. Grossberg, "Adaptive pattern classification and universal recoding, II : Feedback, expectation, olfaction, and illusions," *Biolog. Cybernetics*, vol. 23, pp. 187-202, 1976.

[14] G. A. Carpenter and S. Grossberg, "A massively parallel architecture for a self-organizing neural pattern recognition machine," *Computer Vision, Graphics, and Image Processing*, vol. 37, pp. 54-115, 1983.

[15] G. A. Carpenter and S. Grossberg, "ART2 : Self-organization of stable category recognition codes for analog output patterns," *Applied Optics*, vol. 26, pp. 4919-4930, Dec. 1, 1987.

[16] G. A. Carpenter and S. Grossberg, "ART3 hierarchical search : Chemical transmitters in self-organizing pattern recognition architectures," *Proc. IJCNN*, Washington DC, vol. 2, pp. 30-33, 1990.

[17] T. Kohonen, *Self-Organization and Associative Memory*, Springer-Verlag, New York, NY, 1984.

[18] P. Werbos, *Beyond Regression : New Tools for Prediction and Analysis in the Behavioral Sciences*, Ph. D. dissertation, Harvard Univ., Cambridge, MA, Aug. 1974.

[19] D. Parker, "Learning-logic," Invention Report S81-64, File 1, Office of Technology Licensing, Stanford Univ., Stanford, CA, Oct. 1982.

[20] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," *Parallel Distributed Precessing*, D. E. Rumelhart and J. L. McClelland, Eds., vol. 1. ch. 8, MIT Press, Cambridge, MA, 1986.

[21] J. J. Hopfield, "Neurons with graded response have collective computational properties like those of two-state neurons," *Proc. Nat'l. Acad. Sci.*, vol. 81, pp. 3088-3092, 1984.

[22] D. H. Ackley, G. E. Hinton, and T. J. Sejnowski, "A learning algorithm for Boltzmann machines," *Cognitive Science*, vol. 9, pp. 147-169, 1985.

[23] S. Kirkpatrick, Jr., C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, pp. 671-680, 1983.

[24] R. Hecht-Nielsen, "Counterpropagation networks," *Proc. ICNN*, Piscataway, NJ, pp. II19-II32, June 1987.

[25] S. Grossberg, *Studies of the Mind and Brain*, D. Reidel Publishing Co., Boston, MA, 1982.

[26] D. E. Rumelhart and D. Zipser, "Feature discovery of competitive learning," *Parallel Distributed Process-*

ing : Explorations in the Microstructure of Cognition, D. E. Rumelhart and J. L. McClelland, Eds., MIT Press, Cambridge, MA, 1986.

[27] K. Fukushima, "Neocognitron : A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biol. Cybern.*, vol. 36, pp. 193-202, 1980.

[28] K. Fukushima, S. Miyake, and T. Ito, "Neocognitron : A neural network model for a mechanism of visual pattern recognition," *IEEE Trans. Syst., Man, Cybern.*, vol. 13, no. 5, pp. 826-834, 1983.

[29] B. Kosko, "Constructing an associative memory," *Byte*, vol. 12, no. 10, pp. 137-144, 1987.

[30] B. Kosko, "Adaptive bidirectional associative memories," *Appl. Opt.*, vol. 26, no. 23, pp. 4947-4960, 1987.

[31] B. Kosko, "Bidirectional associative memories," *IEEE Trans. Syst., Man, Cybern.*, vol. 18, no. 1, pp. 49-60, 1988.

[32] R. Hecht-Nielsen, "Neurocomputing : Picking the human brain." *IEEE Spectrum*, vol. 25, no. 3, pp. 36-41, 1988.

[33] S. Lee and R. M. Kil, "A Gaussian potential function network with hierarchically self-organizing learning," *Neural Networks*, vol. 4, no. 2, pp. 207-224, 1991.

[34] S. Lee and J. M. Lee, "Nonlinear system control based on multi-resolution radial-basis competitive and cooperative networks," *Proc. ICONIP '94*, Seoul, Korea, vol. 1, pp. 281-286, 1994.

[35] S. Grossberg, "Learning by neural networks," *Studies of Mind and Brain*, S. Grossberg, Ed., D. Reidel Publishing Co., Boston, MA, pp. 65-156, 1982.

[36] B. Widrow and R. Winter, "Neural nets for adaptive filtering and adaptive pattern recognition," *IEEE Computer Mag.*, pp. 25-39, Mar. 1988.

[37] Y. Pao, *Adaptive Pattern Recognition and Neural Networks*, Addison-Wesley, Reading, MA, 1989.

[38] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proc. Nat'l. Acad. Sci.*, vol. 79, pp. 2554-2558, 1982.

[39] J. J. Hopfield and D. W. Tank, "Neural computation of decisions in optimization problems," *Biol. Cybern.*, vol. 52, pp. 141-152, 1985.

[40] J. A. Freeman and D. M. Skapura, *Neural Networks; Algorithms, Applications, and Programming Techniques*, Addison-Wesley, Reading, MA, 1992.

[41] H. White, "Learning in neural networks : A statistical perspective," *Neural Computation*, vol. 1, pp. 425-464, 1989.

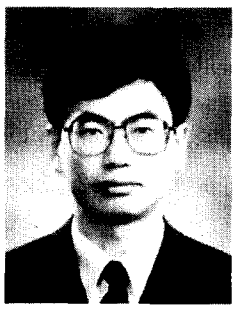
[42] R. P. Lippmann, "An introduction to computing with neural nets," *IEEE ASSP Mag.*, pp. 4-22, Apr. 1987.

[43] V. Vemuri, *Artificial Neural Networks : Theoretical Concepts*, Computer Society Press of the IEEE, Washington DC, 1988.

[44] C. Lau, *Neural Networks : Theoretical Foundations and Analysis*, IEEE Press, New York, NY, 1992.

[45] E. Sanchez-Sinencio and C. Lau, *Artificial Neural Networks : Paradigms, Applications, and Hardware Implementations*, IEEE Press, New York, NY, 1992.

저 자 소 개



이 정 문

1981 서울대학교 전자공학과 졸업(학사)
1983 서울대학교 전자공학과 졸업(석사)
1988 서울대학교 전자공학과 졸업(박사)
경 력 : 1993-1994 미국 USC visiting scholar
1988-현재 강원대학교 제어계측공학과 조교수/부교수
신경회로망 응용, 퍼지 시스템, 지능제어
(200-701) 강원도 춘천시 효자동 192-1
TEL) (0361) 50-6342 / FAX) (0361) 242-2059