

단순신경회로망의 설계 및 구현

우원택

경산대학교 자연과학대학 정보과학부

초록

본 연구에서는 단순신경망의 구조와 특성을 이해하기 위해 신경회로망의 알고리즘을 이론적으로 분석하고 이를 토대로 프로그램을 설계 실행하여 신경망의 학습과정을 실험하였다. 본 연구에서 채택한 학습알고리즘은 3계층구조의 역전파알고리즘이며 신경망의 모형은 단순의료전문가시스템모형을 입력치로 채택하였다. 계층수, 노드수, 학습사이클 수, 학습율, 모멘텀항등의 모수를 입력한 실험의 결과는 입력치에 대한 출력이 기대목표와 거의 일치하였다.

I. 서언

최근에 자연과학 분야에 여러 수학적 이론이 난립하고 있어 많은 사람들이 개념상 혼돈에 빠져 있는 것 같다. 이 중 대표적인 이론들을 들면 퍼지이론, 신경망, 시뮬레이티드 어닐링, 그리고 유전알고리즘이라 할 수 있을 것이다.¹⁾ 퍼지이론은 원래 확률과 이치(Binary) 논리를 결합한 새로운 아이디어로 애매모호한 표현을 나타내는데 있어서 긍정이나 부정밖에 할 수 없는 이치논리에서 불확실한 정도를 수치로 표현할 수 있도록 발전시킨 것이다. 따라서 퍼지는 집합의 관점에서 보았을 때 원소들이 자기 자신에게 어느정도로 포함되는지를 표현하며 원소의 관점에서 자기 자신이 어떤 집합에 어느 정도로 포함되는지를 표현하고자 하는 것이기 때문에 확률이론에서와 같이 원소의 확률의 합이 반드시 1이 되어야 할 필요가 없다. 신경망연구는 1950년대 맥클로와 피츠가 신경망의 구조와 매커니즘을 연구하여 생체의 지능시스템을 수학적으로 해석하려는 데서 유래한 것이며 시뮬레이티드 어닐링은 통계역학에서 물체구조의 상태를 연구하다가 발명하였는데 유리나 철과 같이 고온에서 액체상태로 있는 물질을 어떤 속도로 식히느냐에 따라 각기 다른 에너지를 갖는 최종의 결정상태를 알수있는데 1983년 패트릭은 이 현상으로부터 함수의 최소값을 결정할 수 있는 최적화 알고리즘을 발명한 것이다. 한편 홀란드는 1975년에 생물의 진화 메커니즘을 연구하다 유전알고리즘을 고안해 내게 되었다. 그는 진화는 생물조직의 설계도인 염색체에서 일어나는 것으로 이 염색체의 코딩과 디코딩과정을 유전자에 의한 진화 메커니즘의 최적화에 응용함으로써 진화 메커니즘을 수학적으로 해석할 수 있게 하였다. 이상의 수학적이론은 인공지능, 신경망, 유전자알고리즘, 퍼지신경망등으로 프로그래밍되어 현실적으로 이용되고 있다. 본 연구에서는 이들제연구중 신경망이론에 대하여 이론적인 배경을 살펴보고 이를 바탕으로 신경망을 구현하여 보고자 한다. 신경망의 구현은 기존의 응용소프트웨어를 이용하지않고 제임스 노

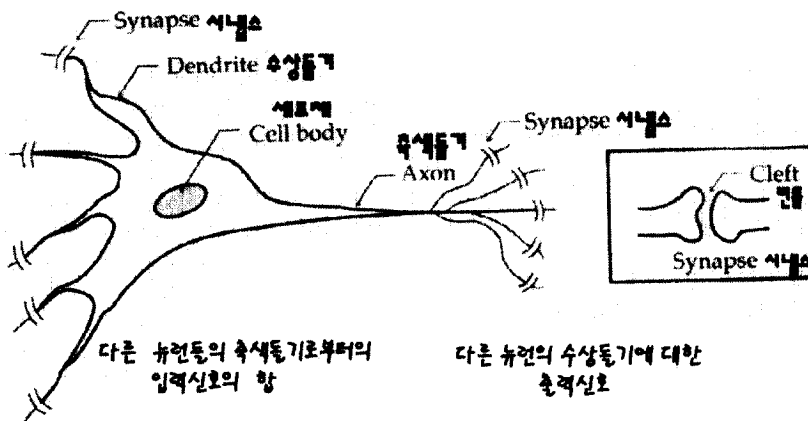
1) 김명원, 방승양, 이수영, 이종호, 전홍태, 정호선, 정홍공저, 알기쉬운 신경망 컴퓨터, 전자신문사, 1992, p.79

이즈(James L. Noyes)의 PASCAL모형을 분석하고 이를 TURBO PASCAL로 구현해 봄으로써 신경망의 내재적 구조와 이의 응용분야를 실제 체험함으로써 향후의 더욱 복잡한 모형을 개발하는데 도움이 되고자 하는 것이 본연구의 목적이다.

II. 인공신경망의 이론적인 배경

1. 뉴런의 특징과 유니트모델

신경망에 관한 연구는 인간의 두뇌와 신경세포 모델에 대한 연구에서부터 시작된다. <그림 1>과 같이 가장 기본적인 단위인 뉴런(neuron)이라는 세포는 신경시스템에서 여러 기능을 수행한다. 뉴런은 세포체(Cell Body)로부터 나온 축삭돌기(Axon)라는 한 개의 가느다란 섬유와 수상돌기(Dendrite)라 불리는 가지들처럼 넓게 퍼진 다수의 돌기들로 구성되어 있다. 뉴런은 뉴런에 의하여 발생된 펄스(pulse)를 다른 뉴런들에게 축삭을 통하여 전달하며, 수상돌기는 다른 뉴런과 연결되어 입력신호를 받아서 연산을 수행한 후에 세포체에 이를 다시 내보낸다. 뉴런은 다른 뉴런의 수상돌기와 자신의 세포체를 연결할 때 아래 그림에서와 같은 시냅스(synapse)라 불리는 접합점을 통하여 신호를 내 보내게 된다.²⁾



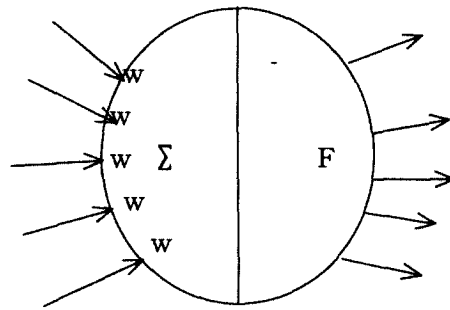
< 그림 1 > 뉴런의 단순생물적구조

자료: James L. Noyes, *Artificial Intelligence With Common Lisp: Fundamentals of Symbolic and Numeric Processing*, D.C. Heath and Company, 1992, p.422

인간의 신경세포(Neuron)의 모델을 이용하여 인공시스템을 구축한 것을 총칭하여 뉴로(Neuro)라고 부르며 뉴로를 구성하는 한 뉴런세포의 인공적인 모델을 유니트(unit)라 하거나 처리요소(PE: processing Element)라고 한다. 한 뉴런의 유니트구조는 <그림 2>와 같

2) 권용석, 인공신경망을 이용한 최초공모주식의 최초균형시장가격의 예측, 한국과학기술원 테크노경영대학원 석사학위논문, 1996, p. 28

며 그림과 같이 다입력 1출력의 소자도 있고 다입력 다출력의 소자도 있다. 그밖에 뉴런(유니트)모델의 작동원리를 간략히 보면 아래 그림에서와 같이 처리요소(Processing Element: 세포의 역할)가 수상돌기에 입력된 자극들을 일정한 규칙에 의해 처리(summation function)하고 그것을 다시 다른 처리요소(Processing Element)에 전달(transfer function)하는 기능을 하게 되며, 이러한 신호는 어떤 무게치(웨이트 또는 결합하중: w_i)가 걸려서, 결합된 유니트에 입력된다. 이 무게치에 의해서 유니트간의 결합 강도가 나타내어 지며 무게치가 걸린 각각의 입력치는 총합이 얻어져 응답함수(f)에 의해 일정 변형을 받은 후 출력된다. 뉴로의 특징을 보면 첫째 뉴로의 본보기는 뇌이며 둘째 뉴로는 고도의 병렬 분산처리 시스템이며 셋째 뉴로는 학습하는 특징을 가지고 있다. 이하에서 각 유니트(뉴로)가 상호 연결된 신경망의 네트워크 구조와 작동원리를 살펴보자.



다른 뉴런으로부터의
가중 입력치의 합

다른 뉴런에 대한 출력

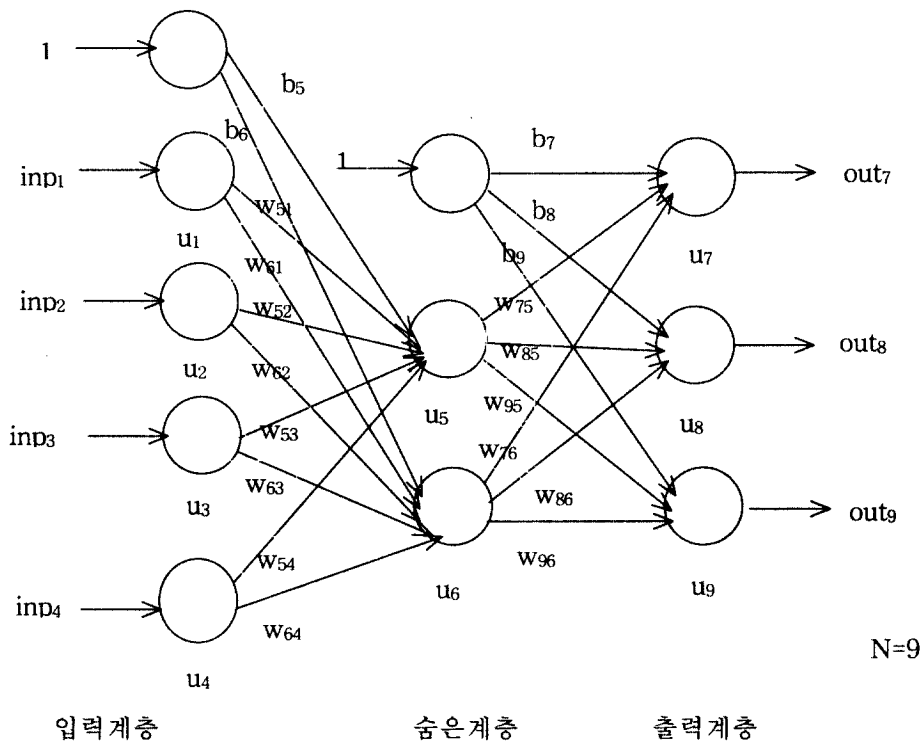
<그림 2> 뉴런의 인공적가상구조

자료: 박문용, 최항식역 中野馨, 飯沼一元, ニューロン、ネシトグループ*, 桐谷滋著, 뉴로컴퓨터, 대영사, 1991, p.4

2. 네트워크모델의 구조와 원리

뉴로의 네트워크 구조를 신경망이라 하는데 이는 전술한 유니트를 시냅스로 상호 연결 접속한 것으로 상호 결합형 네트워크 구조와 계층형 네트워크구조로 나눌 수 있다. 여기서 유니트는 유니트를 그물망 모양으로 연결한 뉴로네트워크상의 교점(node)에 해당한다. 또 유니트간은 신경섬유에 대응하는 선으로 연결되나 실제의 뉴런과의 시냅스(Synapse)결합이라고 불리워지는 결합과 같이 신호는 한 방향으로만 전달되고 어떤 무게치가 걸려서 결합된 유니트에 입력된다. 무게치는 흥분성의 결합일때는 정의 수를 억제성결합일 때는 부의 수를 취한다. 또 시냅스결합 결합도의 가변성에³⁾ 대응하는 무게치는 학습에 의해서 변화시

킬 수 있다. <그림 3>과 같이 유닛이 층모양으로 그룹화되어 있고 신호가 그룹간의 특정 방향으로만 전해지는 구조를 계층형네트워크구조라 한다. 뉴로의 학습이란 정보처리시스템의 목적에 맞도록 유닛간의 결합하중을 조절하는 것이다. 뉴로의 유닛간 결합하중은⁴⁾ 행렬로써 표시할 수 있다. 뉴로의 학습은 일반적으로 우선 무게치의 초기값을 설정함으로써 시작된다. 처음 무게치에 관한 무언가의 지식이 없으면 무작위(random)값으로 해둔다. 그 다음 이 무게치에 대해서 학습데이터를 입력하고 평가기준을 참조하여 평가한다. 그리고 평가결과를 바탕으로 무게치를 조정하고 재차 평가를 행한다. 이와같은 프로세스를 반복하여 차츰 최적치에 가까이 가는 것이 학습이다. 어떤 뉴로에 있어서의 최적학습알고리즘은 적용하려는 문제의 성질이나 이용되는 네트워크의 종류에 의해 달라진다. 학습방식의 학습방법으로는 교사달린 학습과 교사없는 학습방법이 있으며 학습을 행하기 위해서는 얼마간의 평가기준이 필요하다. 이 평가 기준에 의해 평가된 결과를 피드백해서 유닛간의 결합의 무게치 W_{ij} 를 조정한다. 평가를 행할때에는 외부에서 교사신호로서 입력신호에 대한 이상적인 출력을 부여하는 학습방식을 교사달린 학습방식이라고 한다. 한편 평가기준을 내장해서 외부에서 하나하나 교사 신호를 부여하지 않는 학습방식을 교사없는 학습방식이라고 한다. 본 연구는 교사달린 학습방식(Supervised Learning)과 단순 3계층구조의 신경망을 중심으로 분석하고자 한다.



<그림 3> 3계층신경망의 예

자료: James L. Noyes, *Artificial Intelligence With Common Lisp: Fundamentals of*

- 3) 가역성이라고도 한다.
- 4) 무게치라 한다.

본 연구의 대상인 상기의 3계층구조망을 살펴보면 우선 상기 모델의 각셀(유니트)은 뉴런을 구성하는 자동처리요소(autonomous processing element)이며 각셀은 매우 단순한 하나의 컴퓨터로 생각할 수 있으며 기호로는 U_1, U_2, \dots, U_N 으로 표기한다. 단 N 은 네트워크에 있는 셀의 총수를 나타낸다. 이하에서 신경망을 구성하는 노드와 변수를 보면 다음과 같다.

1)무게치(Weights)

무게치 W_{ij} 는 셀 U_j 가 셀 U_i 에 미치는 영향을 표시한다. 만약 $W_{ij} = 0$ 라면 셀 i 와 셀 j 간에 어떤 연결 (connection)도 없음을 나타낸다. 정의 무게치는 강화(Reinforcement)를 부의 무게치는 억제(Inhibition)를 나타내며 무게치는 행렬 \mathbf{W} 로 나타낼 수 있다. 이 무게치는 0를 주어진 값으로 또는 무작위값으로 초기화되며 경험(학습)에 의해 변경된다.

2)전파율(Propagation Rule)

전파율은 각셀로부터 입력된 값들이 어떻게 결합되어 셀 u_i 에 입력되는 지를 나타내는 네트워크 규칙(식)이다. 본연구는 가장 통상적인 규칙의 하나인 아래의 가중총합규칙(Weighted-sum rule)을 사용한다. 전파율은 각 입력치와 그의 가중치(무게치)와의 곱의 합에다 입력셀 u_i 의 편차(bias)를 더 한 것이다.

$$net_i = b_i + \sum_j w_{ij} \cdot inp_j$$

상기식에서 j 는 셀 u_i 에 정보를 보내는 모든 셀의 수를 나타내고 inp_j 는 셀 u_j 로부터 셀 u_i 에 대한 입력값을 나타낸다. 그리고 b_i 는 셀 u_i 와 관계된 편차를 나타낸다. 셀편차는 보통 1의 상수 입력값을 가지며 보통 필요시 고정 상수값을 가지는 하나 이상의 특수셀을 각 계층에 추가하여 이를 나타낸다. 상기식을 행렬형태로 다시 쓰면 $net = b + w \cdot inp$ 로 쓸 수 있으며 편차를 $b = 0$ 로 가정할 경우 $net = w \cdot inp$ 로 쓸 수 있다. 그리고 신경망 처리를 위한 기호표기법으로는 네트워크표기법(network notation)과 계층표기법(layer notation)이 있다. 계층표기법은 계층별로 셀의 번호를 독립적으로 줄 수 있어 인접계층과 필요한 행렬처리에 편리하며 네트워크표기법은 입출력을 나타내는 전체 네트워크를 취급하는데 편리하다.

3) 활성화(Activations)

셀의 상태를 나타내는 셀 u_i 의 활성을 a_i 로 나타내며 이 값은 $\{0,1\}$ 또는 $\{-1,0,+1\}$ 과 같은 이산값이 되거나 $[0,1]$ 또는 $[aLi, aHi]$ 과 같은 구간값인 연속값이 될 수 있다. 그리고 활성값 a_i 는 셀의 활성규칙에 의해 결정된다.

4) 활성화규칙(Activation Rule)

활성규칙은 활성함수 $F(x, a)$ 에 의해 주어지며 대부분 모든 셀에 대해 똑같은 함수가 사용되며 입력값 x 가 새로운 활성값을 산출하기 위해 어떻게 현재의 활성값 a 와 결합되는지를 나타낸다. 즉 $a_i = F(net_i)$ 로 통상 계산하며 신경망시뮬레이션에 사용하는 함수들을보

면 아래와 같다.

i) 항등 함수(Identity Function): 활성값은 결합입력값과 같다.

$$F(x) = x$$

ii) 선형 함수(Linear Function): 활성값은 결합입력치의 선형중속이다.

$$F(x) = mx + c$$

iii) 역치 함수(Threshold Function): 기준이 되는 역치값 p 를 사용하는 함수로 통상 세가지 유형이 있다.

$$F(x) = \begin{cases} 0 & \text{for } x < p \\ 1 & \text{for } x \geq p \end{cases}$$

$$F(x) = \begin{cases} 0 & \text{for } x < p \\ Ai & \text{for } x \geq p \end{cases}$$

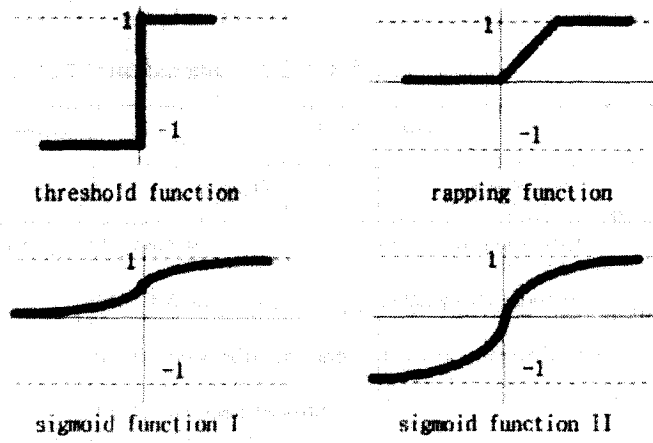
$$F(x) = \begin{cases} 0 & \text{for } x < p \\ x & \text{for } x \geq p \end{cases}$$

역치함수의 역할은 활성값이 역치(threshold) p 의 기준에 미달하면 계속 0의 값이 출력되며 그 이상이면 1 또는 지정값이 출력된다. 역치함수를 사용할 경우 보통 관계된 셀안에 상수 기준값 p 를 입력하여 평가한다. 상기 세 함수중 앞의 두 함수를 단계함수(step function)라고도 한다.

vi) 로지스틱 함수(Logistic Function)

$$F(x) = \frac{1}{1 + e^{-\frac{x}{c}}}$$

로지스틱함수는 3계층신경망과 같은 숨은 계층모델에서 사용되며 S-커브 모양을 가진 시그모이드(sigmoid)함수이다. 시그모이드함수는 $x \rightarrow -\infty \Rightarrow F(x) \rightarrow 0$, $F(0) = 1/2$, $x \rightarrow +\infty \Rightarrow F(x) \rightarrow 1$ 과 같이 단조증가함수이며 미분가능한 함수를 말하며 이때의 도함수는 $F'(x) = F(x)[1 - F(x)]/c$ 과 같이 된다. 즉 $F'(x)$ 는 단일 최대값함수(unimodal curve)로 $x \rightarrow -\infty \Rightarrow F'(x) \rightarrow 0$, $F'(0) = 1/(4c)$ (F' 의 최대값), $x \rightarrow +\infty \Rightarrow F'(x) \rightarrow 0$ 가 된다. 상기식에서 c 는 0보다 큰 상수로 결과 값을 비례화 시킨다.



<그림 4> 활성화함수의 보기

자료: 권용석, “인공신경망을 이용한 최초공모주식의 최초균형시장가격의 예측”, 한국과학기술원 테크노경영대학원 석사학위논문, 1996, p.34

시그모이드함수와 같이 초기의 퍼셉트론신경망에 사용된 활성화함수의 예는 < 그림 4 >와 같으며 활성화함수를 전이함수(transfer function)라고도 한다.

5) 출력(Outputs)

한 셀의 출력값 out_i 는 그 셀의 활성화수준의 또 다른 함수로 $out_i = f(a_i)$ 로 나타내며 이식을 출력함수(output function)라 한다. 전술한 활성화함수와 같이 출력함수는 항등함수(identity function), 선형함수(linear function) 또는 역치함수(threshold)로 나타낼 수 있으며 활성화된 결과값은 수정하는데 사용된다. 일반적으로 수정은 실수 활성화값을 이산치의 출력값으로 변환하는 것을 말하며 또 다른 통상의 수정은 아래와 같은 경계함수를 출력함수로 이용한다.

$$\text{경계함수 } f(x) = \begin{cases} 0 & \text{for } x < p \\ x & \text{for } p \leq x \leq q \\ 1 & \text{for } x > q \end{cases}$$

경계함수는 통상적으로 x 가 0 이상이고 1 이하인 경우에는 적은 p 값을 취하며 q 값은 $q = 1 - p$ 로 계산된다.⁵⁾ [0, 1]사이의 경계값을 산출하기 위한 경계함수는 일반적으로 전술한 로지스틱활동함수와 잘 작용한다. 경계함수는 출력층에서만 사용될 수 있다.

6) 학습규칙(Learning Rule)

학습규칙은 모든 연결노드에 적용되고 무게치 w_{ij} 가 경험에 의해 어떻게 갱신되는가

5) 예를들면 $p = 0.1$ 을 취하면 q 는 0.9 가 된다.

를 명시하는 규칙이다. 학습은 전술한 바와같이 교사달린학습(Supervised Learning)과 교사 없는학습(Unsupervised Learning)이 있으며 교사달린학습에서는 목표(goal) 또는 표적(target)이라는 값을 제공하는 교사(teacher)가 있고 교사없는학습에서는 목표값을 알지 못하고 다른 유형으로 자기기준이 강화(reinforcement)된다. 신경망에서 자주 사용되는 세가지 유형의 학습규칙을 보면 다음과 같다.

i) 단순헵규칙(Simple Hebbian Rule)

$$d_{ij} = L \cdot out_i \cdot inp_j$$

단순헵규칙은 상기식에서 보는 바와 같이 out_i 와 inp_j 의 부호가 같으면 학습의 힘(strength)은 증가하고 부호가 서로 다르면 감소한다. 이 규칙은 교사달린 학습이나 교사없는학습 양쪽 모두에 사용할 수 있으며 2계층- 교사달린학습 시스템에서는 out_i 를 $goal_i$ 로 대체할 수 있다. 효율적인 학습을 위해 입력벡터는 직교라야 하며 그렇지 않으면 상호 간섭이 일어날 수 있다.

ii) 그로쓰버그규칙(Grossberg Rule)

$$d_{ij} = L \cdot out_i \cdot (inp_j - w_{ij})$$

그로쓰버그규칙은 상기식이 보이는 바와 같이 학습(개선)이 현재의 무게치와의 차이에 비례적인 함수를 의미한다.

iii) 위드로우호프델타규칙(Widrow-hoff Delta Rule)

$$d_{ij} = L \cdot (goal_i - out_i) \cdot inp_j$$

위드로우호프델타규칙은 교사달린학습방법이며 학습의 개선은 교사가 세운 $goal_i$ 와의 차이에 비례한다고 가정한다. 최소자승평균법(LMS: least mean squares)이라고도 하며 입력벡터가 선형독립인 2계층시스템에 사용될 수 있으며 입력은 정규화되어 입력되기도 한다.

상기식들에서 L을 학습률이라 하며 학습률(Learning Rate) L은 정의 승수이며 통상 0 와 1 사이의 실수값으로 통상 전과정에서 고정값을 취한다. 일반적으로 학습데이터가 비교적 예러가 없으면 1 에 가까운 값을 취하고 6) 학습데이터의 신뢰성이 낮으면 0 에 가까운 L 값을 사용하는 것이 안전하다. 데이터의 신뢰성에 관한 것이 알려져 있지 않은 경우는 0.5 의 L 값을 택하는 것이 합리적 선택이다. 7) L = 0 는 학습을 통한 개선이 없음을 나타내며 L = 1 은 무게치가 수렴하지 않는 비 안전적 시스템을 유도한다. L > 1 이면 수렴을 가속화

6) 보다 빠른 학습을 의미한다.

7) James L. Noyes, *Artificial Intelligence With Common Lisp: Fundamentals of Symbolic and Numeric Processing*, D.C. Heath and Company, 1992, p.428

한다. 학습과정에서 훈련집합(training set)과정의 각 패스를 기간(Epoch)이라 하며 과정의 초기에 모든 무게치 w_{ij} 는 적당한 값으로 초기화된다. 실제로 보면 -1.0 과 +1.0 과 같은 두 극한값 사이의 무작위값이 사용된다. 전술한 바와 같이 무게치의 값이 0 이면 상호 셀간의 연결성이 없다는 것을 의미한다. 이러한 무게치를 갱신하는데는 케이스갱신(case updating)과 기간갱신(epoch updating)의 두가지 방법이 있다. 보기갱신(exemplar updating)이라고도 하는 케이스 갱신은 각각의 보기마다 무게치 w_{ij} 가 $w_{ij} + d_{ij}$ 로 모두 갱신되는 것으로 기간갱신보다 간단한 방법이다. 두 번째의 기간갱신방법은 학습집단의 첫 번째 케이스가 검토되기전에 즉 각각의 w_{ij} 에 대해 t_{ij} 가 존재하며 각 보기마다 모든 t_{ij} 값이 갱신되어 $t_{ij} + d_{ij}$ 가 된다.⁸⁾ 학습집합의 마지막 케이스가 처리된 후 무게치는 최종적으로 $w_{ij} + t_{ij}$ 로 갱신된다. 이 갱신들은 훈련집합의 각 패스의 마지막마다 수행된다. 이 방법은 교사달린 학습에서 주로 사용된다. 이 방법은 무게치가 실제로 갱신되기전에 심한 변화를 주는 갱신을 합산에서 제외시키는 효과를 가지고 있다. 학습과정은 모든 d_{ij} (또는 t_{ij}) 값이 무시할 정도가 되었을 때⁹⁾ 끝난다. 교사달린학습의 경우에는 학습과정은 출력계층에서의 에러 자승의 합이 데이터 모두에 대해 충분히 작아지면 멈추게 된다.¹⁰⁾ 여기서 각 에러는 단순히 $goal_i - out_i$ 이다.¹¹⁾

3. 신경망의 학습규칙

신경망에 대하여 코딜(Caudill)과 버틀러(Butler)는 신경회로망을 비알고리즘적이며 아날로그방식이며 다수의 처리장치를 이용한 동시 작업을 특징으로 하는 정보처리시스템이라고 정의하고 있으며 헤흐트-닐슨(Hecht-Nielsen)은 신경회로망의 구조적 측면을 중시하여 신경회로망을 방향성을 갖는 신호채널로서 상호 연결된 처리단위들로 구성되어 있는 병렬, 분산정보처리구조로 정의하고 있다. 맥클로(McCulloch)와 피츠(Pitts)는 1943년의 논문에서 인간의 두뇌를 수많은 신경세포들로 이루어진 잘 정의된 컴퓨터라고 여기고 단순한 논리적 업무를 수행하는 모델을 보여주었으며 패턴분류 문제가 인간의 지능적인 행위를 규정하는 이론에 매우 중요하다는 것을 인식하였다. 이 논문이 신경망 모델의 시초로 알려져 있다.¹²⁾ 신경망의 학습규칙을 약술하면 웹(Hebbian: 1949)에 의하여 제시된 웹학습규칙(Habbian Learning Rule)에 의하면 두 뉴런 사이의 가중치를 조정할 수 있는 최초의 규칙으로 여기서의 학습방식은 처리요소로의 입력변수중에서 영향이 큰 쪽을 더욱 가중치를 두는 방향으로 학습을 진행하여 나아가는 방법이다. 로젠브라(Rosenblatt: 1957)은 퍼셉트론(Perceptron)이라는 최초의 신경망모델을 발표하였으며 학습과정에서 알파강화규칙을 사용하였다. 이 방법은 어떤 타입의 패턴이 입력층에 주어졌을 때 이 모델이 반응하게 되는 연결강도의 집합을 스스로 발견하는 자동적인 절차이다. 그러나 이러한 퍼셉트론은 민스키(Minsky)와 페퍼트(Papert: 1969)가 'PERCEPTRONS' 라는 저서에서 퍼셉트론 모델을 수학적으로 분석하여 그 모델의 단점을 밝혀내었는데 그것은 퍼셉트론 모델로는 XOR함수와 같이 단순한 비선

8) 이들의 합은 무게치 대신에 갱신된다.
 9) 모든 무게치의 값이 안정되었음을 의미한다.
 10) 모든 계산된 값이 목표값들에 충분히 가까워 짐을 의미한다.
 11) 주어진 값과 계산된 값과의 차이를 의미한다.
 12) 권용석, 인공지능망을 이용한 최초공모주식의 최초균형시장가격의 예측, 『한국과학기술원 테크노경영대학원 석사학위논문』, 1996, p.29

형 분리문제도 풀 수 없다는 것을 밝혔다. 퍼셉트론은 단일의 조정층(single-adjustable layer)으로 구성되어 있는 모델의 한계점을 개선하기 위하여 입력계층(Input layer)과 출력계층(Output layer)사이에 하나의 숨은계층(hidden layer)을 사용하여 문제를 해결하려고 하는 시도가 1980년대에 제안되었으며 특히 루멜하트(Rumelhart: 1986)는 역전파학습 알고리즘(backpropagation algorithm)을 개발하였다. 이것은 숨은계층(hidden layer)을 가진 다계층퍼셉션(multi-layer perceptron)에 역전파학습알고리즘을 사용함으로써 선형분리 문제등을 해결할 수 있게 하였다. 이러한 다층퍼셉트론의 구조(multi-layer perceptron)는 입력계층, 숨은구층, 출력계층 방향으로 연결되며 각 층내의 연결과 출력층에서 입력층으로서의 직접적인 연결은 존재하지 않는 전방향(feedforward)의 구조이다. 이러한 다층퍼셉트론은 단층 퍼셉트론과 유사한 구조를 가지고 있지만 중간계층과 각 유닛의 입출력 특성을 비선형으로 함으로써 단층 퍼셉트론의 단점을 극복하여 능력을 향상시켰다. 여기서 사용되는 역전파 알고리즘은 최소평균자승(Least Mean Square)알고리즘을 비선형적으로 확장으로, 미분의 반복규칙(Chain Rule)을 여러번 반복적으로 적용(Generalized Delta Rule)하여 확률적근사모형(Stochastic Approximation Framework)과 관련지어서 유도한다. 전방향네트워크 역전파학습 신경망은 계산을 전방향으로 진행하면서 오류에 대한 수정은 역전파학습에 의하여 조정해나가는 신경망이다. 전술한 바와 같이 학습은 크게 교사달린학습(Supervised Learning)과 교사없는학습(Unsupervised Learning) 및 강화학습(Reinforcement Learning)으로 구분할 수 있다. 교사달린학습은 입력값에 대하여 출력값이 정해져 있기 때문에 학습에서 피드백을 줄 수 있는 학습인 반면에 교사없는학습은 코호넨(Kohonen)의 경쟁적학습(Competitive Learning)과 같이 입력값에 대한 출력값이 정해져 있는 것이 아니라 같은 계층내에서 처리요소(Processing Element)간의 상호작용에 의하여 군별분류(Clustering)하면서 유사한 변수끼리 묶는 것과 같은 학습이다. 그리고 강화학습은 입력값에 대한 결과에 대하여 좋은지 나쁜지를 알려주는 학습방식이다. 그리고 입력값에 대한 출력값을 확인하는 방식으로 회상역학(Recall Dynamic)의 형태에 따라서 학습형태를 달리한다. 즉 전방(Feedforward), 후방(Feedbackward)의 형태로 나눌 수 있다. 심슨(Simpson:1992)의 분류에 의한 학습형태를 보면 < 표 1 >과 같다.

학습(Learning)	회상역학(Recall Dynamic)	
	후방향(Feedback)	전방향(Feedforward)
교사없는학습(Unsupervised)	홉필드네트워크(Hopfield Networks) 아트1과 아트2(ART1 & ART2) 양방향연상기억(Bidirectional Associative Memory) 주성분네트워크(Principal Component Networks)	선형연상기억(Linear Associative Memory) 학습벡터수량화(Learning Vector Quantization) 운전방향강화학습(Drive Reinforcement Learning)
교사달린학습(Supervised)	상자뇌상태(Brain-State-In-A-Box) 신경최적화(Neural Machine Optimazation) 회귀역전파학습네트워크(Recurrent Back Propagation Network)	볼츠만머신(Boltzman Machine) 역전파(Backpropagation) 퍼셉트론(Perceptron) 아달라인(ADALINE)

< 표 1 > 학습방식에 따른 신경망분류

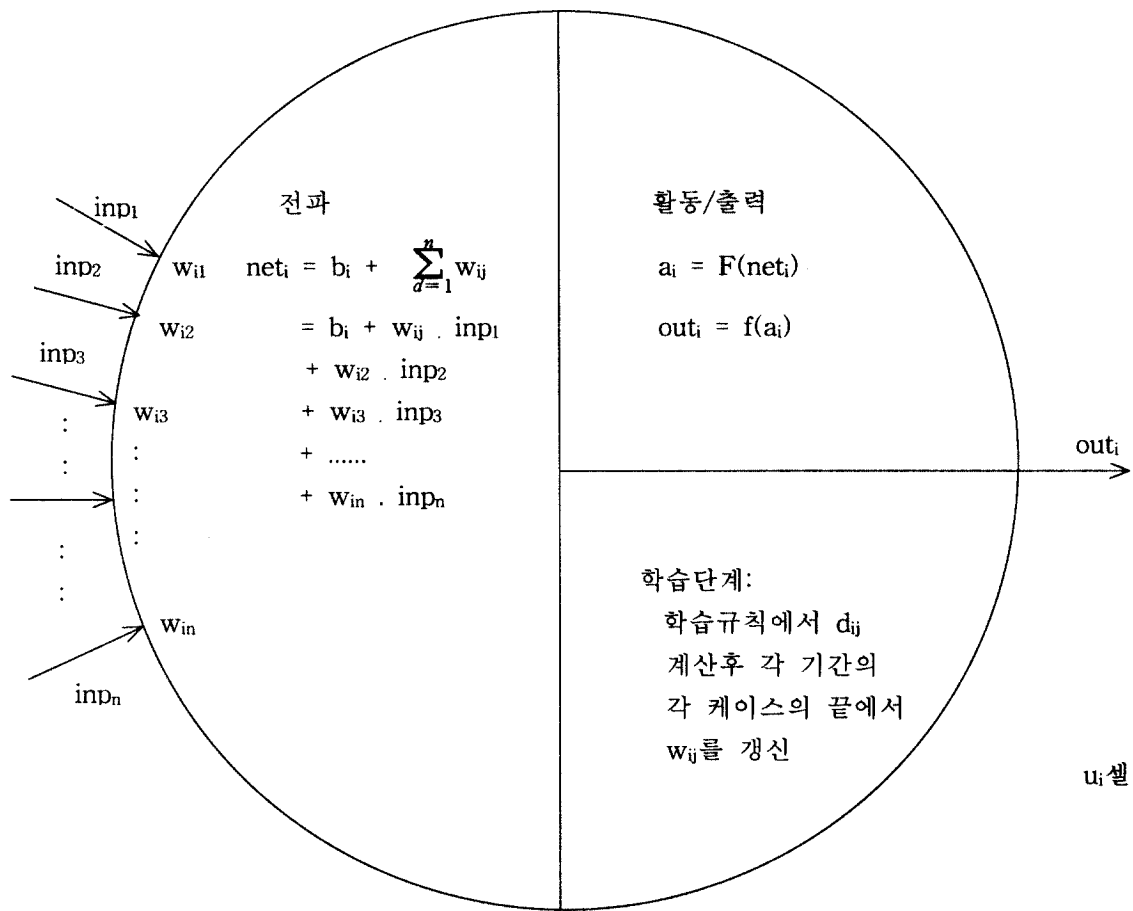
자료: 권용석, "인공신경망을 이용한 최초공모주식의 최초균형시장가격의 예측", 한국과학기술원 테크노경영대학원 석사학위논문, 1996, p.35

III. 신경망알고리즘의 분석

1. 신경망의 알고리즘

신경망의 프로그래밍은 알고리즘을 실행하는 것과 같은 전통적 방법으로 프로그래밍하지 않고 대신에 프로그래머가 전술한 구조를 명시하는 신경망구조를 정의하여 프로그래밍한 후 프로그래머나 사용자가 적절한 입력¹³⁾을 적용하여 신경망이 이에 반응하도록 하는 형태를 가진다. 이러한 과정은 <그림 5>와 같이 간단히 모형화 할 수 있다.

13) 이를 통상 목표라 한다.



< 그림 5 > 단순셀(처리구조)의 구조

상기 그림에서 유니트의 입력은 여러개이고 출력은 하나이나 이 하나의 출력은 여러개의 다른 셀(유니트)로 갈 수있다. 상기식을 시간을 고려하여 다시 표기할 경우 입력은 $inp_i(t)$ 로 무게치는 $w_{ij}(t)$ 로 전파율은 $net_i(t)$ 로 고쳐 쓸 수 있으며 활성규칙, 출력함수, 무게치는 다음과 같이 표기할 수 있다.

$$a_i(t + 1) = F(\text{net}_i(t))$$

$$\text{out}_i(t + 1) = f(a_i(t + 1))$$

$$w_{ij}(t + 1) = w_{ij}(t) + d_{ij}(t)$$

현실적으로 유니트의 입출력처리는 시간종속적이며 싱기표기가 더 정확한 표기에 해당한다. 인공신경망의 모델은 숨은계층의 수, 숨은계층내에서의 노드의 수, 학습사이클(training cycle)의 수, 기간(epoch)의 크기, 학습을 및 모멘텀항(momentum term)과 같은 모수(parameter)들로 구성하므로써 구성된다. 이러한 모수들을 어떻게 선택하는가에 따라서 인공신경망의 성과에 영향을 크게 미치고 있으므로 매우 중요한 문제이다. 그러나 이러한 선택의 기준이 될만한 모델 파라미터나 네트워크 위상(network topology)에 대한 이론적인 기준은 거의 존재하지 않지만 여러 가지 휴리스틱한 방법에 의한 지침들이 제시되고 있다.¹⁴⁾ 3계층 네트워크모델에서 숨은 계층은 공간을 여러 가지 방법으로 분할하는 역할을 한다. 일반적으로 숨은 계층은 n 개의 입력과 하나의 출력에 대해 n 차원 단위 평면을 초평면(hyperplane)으로 분할하는 역할을 한다. 주어진 문제를 효율적으로 풀기 위해서는 각 층안에 있는 셀의 수를 작게 그리고 숨은 계층의 수를 작게 선택하는 것이 좋다. 그러나 전술한 바와 같이 아직까지 셀과 계층 수에 대한 최선의 조합은 알려져 있지 않다. 만약 숨은 계층이 너무 많은 셀을 가진다면 시스템은 입력을 하나의 특징을 나타내는 현상으로 일반화하기보다 입력패턴을 단지 기억하기만 하는 경향을 띤다. 반면에 셀의 수가 너무 적으면 시스템은 더 긴 학습시간을 필요로 하거나 더 낮은 회상정확도, 또는 전혀 학습하지 않는 경향을 띤다. ¹⁵⁾ 숨은계층에서 무게치를 개선하는 하나의 방법으로 역전파(Backpropagation)법이 제시되고 있다. 역전파법(Backpropagation method)은 신경망의 학습방법의 하나로 각 계층에서 에러를 계산하고 이 에러를 거꾸로 전파함으로써 무게치를 적당하게 개선시킬 수 있다. 하나의 단순한 입력패턴에 대해 시스템은 출력계층에 도달할 때까지 입력을 각 계층마다 앞으로 전파한다. 이를 위해 로지스틱활성함수가 주로 사용된다. 역전파법은 활동함수가 x에서 미분가능하고 비감소함수이어야 한다. 그리고 시스템은 마지막 계층에 있는 출력을 목표와 비교하고 에러를 계산한 후 이를 전파하여 각 계층마다 에러를 거꾸로 계산함으로써 무게치에 대한 어떠한 개선이 요청되는지를 결정한다. 역방향 에러전파가 끝나면 이러한 개선을 더함으로써 모든 무게치는 갱신된다. 전과같이 무게치는 각케이스나 각 기간후 마다 갱신된다. 이를 위해 위도우호프델타규칙(Widrow-Hoff Delta Rule)이 확장되어 사용된다. 이 일반화된 델타규칙(Generalized Delta Rule)은 활용함수의 도함수 F'(x)를 활용하여 다음과 같이 작용한다. 결합입력 $\text{net}_{i,i}$ 와 출력 $o_{i,i}$ 을 가지는 각 셀 u_i 가 계층 l 에 있고 마지막 출력계층에 도달할 때까지 셀 u_i 에 대한 에러는 다음과 같이 계산된다. 이것은 실제로는 비례적 에러이다.

$$\text{error}_{i,i} = F'(\text{net}_{i,i})[\text{goal}_i - o_{i,i}]$$

14) 권용석, 인공신경망을 이용한 최초공모주식의 최초균형시장가격의 예측, 『한국과학기술원 테크노경영대학원 석사학위논문』, 1996, p.60
 15) James L. Noyes, 『Artificial Intelligence With Common Lisp: Fundamentals of Symbolic and Numeric Processing』, D.C. Heath and Company, 1992, p.445

각각의 숨은 계층에 대해 거꾸로 작용하면¹⁶⁾ 각 계층에 있는 각 셀 u_i 에 대한 에러는 다음과 같이 계산된다.

$$\text{error}_{i,i} = F'(\text{net}_{i,i}) \sum_j w_{i+1,j,i} \cdot \text{error}_{i+1,j}$$

색인 j 는 원래 셀 u_i 와 관계된 무게치에 의해 야기된 셀 에러를 합하기 위해 사용된다. 즉 이식은 원래의 무게치를 사용하여 에러를 수정하며 보는바와 같이 양쪽 모두에서 도함수 $F'(x)$ 가 사용된다. 이 도함수는 기 사용된 활동함수의 비율을 표시한다. 모든 숨은 계층에서의 각 셀에 대한 에러가 결정되면 케이스갱신학습(Case Updating Learning)의 무게치 변화는 아래 식중의 하나에 의해 결정된다.

$$d_{i,i,j} = L \cdot \text{error}_{i,i} \cdot o_{i-1,j}$$

또는

$$d_{i,i,j} = L \cdot \text{error}_{i,i} \cdot o_{i-1,j} + M \cdot Z_{i-1,j}$$

단 $Z_{i,i,j}$ 는 전술한 학습방식¹⁷⁾ 위해 만들어진 개선된 $d_{i,i,j}$ 이다. 상기의 두 번째 식은 승수로 사용되는 상수 M 을 가진 모멘텀항을 포함한다. $M = 0.9$ 라는 값은 많은 경우에 성공적으로 적용되어 왔다. 어떤식이 사용되더라도 무게치는 전과같이 아래 식으로 갱신된다.

$$w_{i,i,j} = w_{i,i,j} + d_{i,i,j}$$

기간갱신에 대한 무게변화는 똑같은 모양으로 수행된다. 즉 무게치를 갱신하기 전에 학습세트에 있는 모든 경우에 대한 개선을 합한다. 전술한 각 단계를 프로그래밍을 위한 학습알고리즘으로 기술하면 < 표 2 > 와 같다.

16) 첫 번째 계층은 원래 입력계층이므로 개선시킬 무게치가 없는 셈이다.

17) 만약 이것이 첫 번째 입력이라면 $Z_{i,i,j} = 0$ 이다.

1. 네트워크를 규정하고 초기화한다.
 - a) 계층수 NL을 선택한다.
 - b) 각 계층에 포함될 셀의 수 $nc_l, l = 1 \dots NL$ 을 선택한다.
 - c) 전파규칙을 사용한다.
 - d) 활동함수 $F(x)$ 와 이와 관련된 모두를 선택한다. 단 활동함수는 x 에서 비감소이며 미분가능하여야 한다.
 - e) 출력함수와 이와 관련된 모수를 선택한다. 이 함수로 d)에서 전술한 조건을 만족해야 한다. 전통적으로 항등함수 $f(x) = x$ 가 사용된다.
 - f) 학습율과 모멘텀 모수 M 을 선택한다.
 - g) 기간갱신(Epoch Updating)이나 케이스갱신(Case Updating)이나를 선택하고 만약 기간갱신이 사용된다면 합산행렬 $T_l = 0, l = 2 \dots NL$ 로 초기화 한다.
 - h) 무작위 또는 선택된 값을 사용하여 모든 무게치 행렬 $w_l, l = 2 \dots NL$ 을 초기화한다. 그리고 셀편차값을 초기화한다.
2. 다음의 학습사례를 선택한다.
 - a) nc_1 - 요소입력벡터 o_1 을 받아들인다.
 - b) nc_{NL} - 요소벡터 goal을 받아들인다.
3. 전방전파를 $l = 2 \dots NL$ 의 순으로 이를 수행한다.
 - a) w_l 과 o_{l-1} 을 사용하여 nc_l - 요소 net_l 벡터를 계산한다.
 - b) net_l 을 사용하여 nc_l - 요소 a_l 벡터를 계산한다.
 - c) a_l 을 사용하여 nc_l - 요소 o_l 벡터를 계산한다.
4. 후방전파(Backward Propagation)를 $l = NL \dots 2$ 의 순으로 이를 수행한다.
 - a) nc_l - 요소 $error_l$ 벡터를 계산한다.
 - b) nc_l 의 개선행렬 D_l 을 사용하여 nc_l 을 계산한다.
5. 갱신을 수행한다.
 - a) 케이스갱신(Case Updating)이 사용되면 무게치행렬 $w_l = w_l + d_l$, 단 $l = NL \dots 2$ 를 즉각 갱신한다.
 - b) 기간갱신(Epoch Updating)이 사용되면, 개선치를 $T_l = T_l + D_l$, 단 $l = NL \dots 2$ 로 합산하고 훈련집합의 제일 마지막 케이스가 되면 $w_l = w_l + t_l$, 단 $l = NL \dots 2$ 로 w_l 을 갱신한다.
6. 학습기간(training session)이 끝이 난지를 결정하라. 만약 학습기간이 끝이라면 즉 D_l 값이 작다면 w_l 행렬의 무게치가 바람직하게 정해졌으므로 학습과정을 멈추어라. 그렇지않으면 단계 2 로 돌아가 다시 시도하라.

< 표 2 > 역전파학습알고리즘의 순서

2. 신경망의 프로그래밍

전술한 알고리즘을 가지는 상호작용적 신경망 프로그램과 이의 응용을 수행하기 위해 Noyse 의 모델을 이용하여 TURBO PASCAL 로 시행하였다. 이 프로그램은 케이스갱신(case updating)과 기간갱신(epoch updates) 모두가 가능하며 또한 셀편차도 다룰수 있다.

그리고 전술한 모델과 이의 유사모델을 풀기위해 서로 다른 모수값을 가진 여러 가지의 활동과 출력함수를 사용하여 역전파알고리즘을 실행하는데 매우 유용하다. 프로그램에 대해서 간단히 설명하면 이 프로그램은 샘플데이터를 파일의 형태로 입력하도록 되어 있으며 네트워크의 구조도 계층수는 4 개까지 각 계층의 유니트 수는 26 개까지 자유로이 설정할 수 있도록 되어 있다. 이 프로그램은 로지스틱활성함수와 항등출력함수가 사용될 때 3계층이상에서의 역전파를 수행하는 프로그램이다. 이 프로그램은 항등활동함수와 다른 출력함수가 사용될 때에도 사용가능하며 2계층문제에서는 정규델타규칙을 사용할 수 있다. 그리고 다른 조합의 사용도 가능하다. 먼저 주요함수로는 벡터의 유클리드거리를 계산하는 `vnorm()`, 행렬의 무한거리를 계산하는 `mlnorm()` 등이 있다. 프로시저함수로는 주어진 벡터를 출력하는 `displayvec()` 와 주어진 행렬을 출력하는 `displaymat()`, 네트워크학습의 진행을 보여주는 `working()`, 난수함수 `random()`, 활성함수 `F()`, 학습기간동안 사용할 최초의 편차와 무계치를 결정하는 `getwgt()`, 각 셀에 네트워크입력치를 계산하는 `propagation()`, 첫 번째 계층의 입력치와 마지막계층 셀들의 훈련목표치를 읽는 `training()`, 선택한 출력함수를 현재의 활성결과에 적용하는 `results()`, 현재의 에러를 결정하고 델타개선을 계산하는 `learning()`, 현재의 계층에 있는 셀로부터의 출력을 보여주는 `writeout()` 와 프로그램을 실행하면 메뉴가 표시되는 메인프로그램으로 구성되어 있다. 그리고 지금까지의 계산을 좀더 쉽게 보여주기 위해 $\mathbf{w} = \mathbf{0}$ 가 되도록 \mathbf{w} 를 초기화하였다. 그러나 실제로 이렇게 되면 어떤 문제들은 풀기가 불가능하게 된다. 실제 어떤 범위내의 무작위 값이 \mathbf{w} (또는 각 w_i)를 초기화하기 위해 사용된다. 그러면 학습이 매우 늦어지거나 무계치가 전혀 안정화되지 않을 수도 있다. 만약 이러한 현상이 일어나면 다른 초기화 무계치가 사용되거나 활동함수가 변하게된다. 이하에서 본 프로그램의 실행을 하여 보면 다음과 같다.

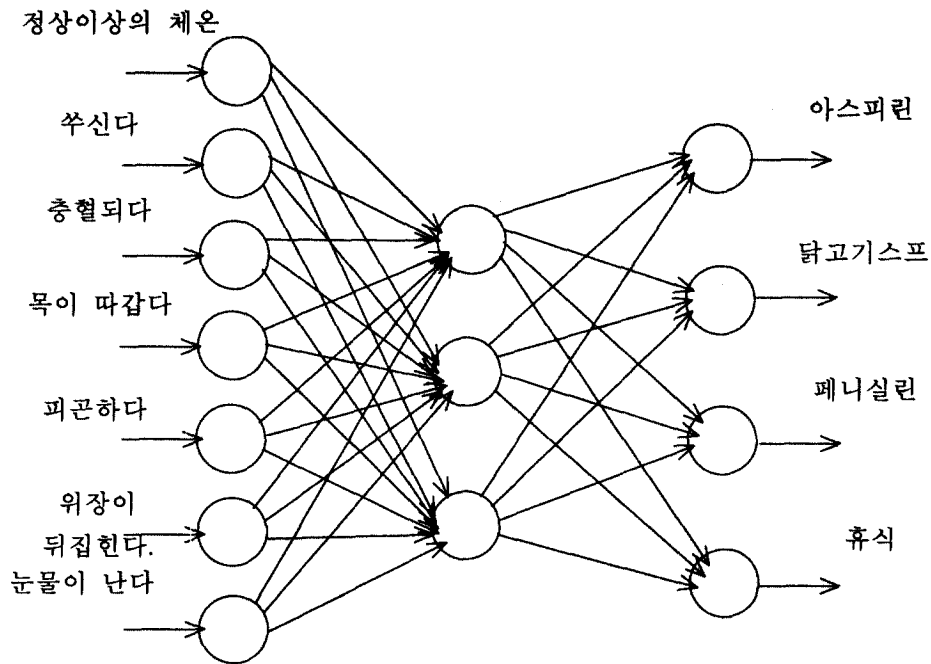
IV. 신경망프로그램의 실행과 평가

본 연구의 프로그램을 실행하기 위해서는 맨먼저 구체적으로 각 층의 유니트의 수나 출력함수의 타입을 정해야 한다. 본연구는 3계층 7입력(입력층), 3셀(숨은층), 4출력(출력층)으로 구성되어 있다. 출력함수는 선형함수(Linear)와 시그모이드함수(Sigmoid)의 2 종류가 있으며 어느 것인가를 선택하는 형태로 되어있다. 보통 입력층은 선형함수로 하고 중간층은 시그모이드함수로 한다.¹⁸⁾ 네트워크의 구조가 정해지면 다음은 샘플데이터에 관한 입력을 행한다. 즉 샘플데이터의 수와¹⁹⁾ 그 샘플 데이터를 받아들여 어떤 이름의 입력파일을 만든다. 예를들면 본 연구의 경우 샘플데이터의 수는 7개이며 데이터파일명은 `traning.pas` 이다. 이상과 같은 설정이 끝나면 드디어 학습이 개시된다. 본 프로그램의 실험을 위해 매우 단순한 의료전문가시스템을 개발하여 훈련해 보고자 한다. 아래 < 그림 6 >과 같은 3 계층모델(NL = 3)을 가정하고 아래와 같이 이를 정의한다.

- $nc_1 = 7$ 가능한 증상의 수(입력)
- $nc_2 = 3$ 가능한 병의 수(숨은)
- $nc_3 = 7$ 가능한 치료의 수(출력)

18) 김용수역, C로 만드는 뇌의 정보시스템, 생능출판사, 1996, p.135

19) 네트워크에 학습시킨 입출력 데이터가 몇 개가 있을까이다.



< 그림 6 > 단순의료진단모델

모수 1.0을 가진 로지스틱함수를 활동함수로 사용하였고 항등함수를 출력함수로 사용하였다.²⁰⁾ 각 셀 편차는 0로 세트하고 학습율은 $L = 0.5$ 로 모멘텀은 0.9를 사용한다.²¹⁾ 초기 무게치의 값들은 최종무게치와 필요한 훈련기간의 수 모두인 최종해의 값에 상당한 영향을 끼칠 수 있기 때문에²²⁾ 초기무게치는 무작위로 선택하였다. 전술한 바와같이 무게치 w_{ij} 는 셀 u_j 와 u_i 간의 관계(connection)에 대한 가중치이다. 예를들면 $w_{8,1} = -0.0995$ 는 셀 u_1 에서 u_8 에 대한 부의 가중치를 나타낸다. 무게치갱신 방법은 기간갱신과 케이스갱신중 케이스갱신을 선택한다. 이제까지 설명한 입력내용을 TURBO PASCAL로 실행한 후 이를 입력한 결과는 다음과 같다.

20) 역전파가 사용될 때 이 함수들은 미분가능하여야 한다.

21) 이것은 이전의 개선을 아주 높게 활용한다는 것을 의미한다.

22) 일반적으로 이산적학습데이터에 대해서는 최종무게치는 유일할 필요는 없다.

Interactive Neutral Network Program

Please enter the title for the model:

medical

Please enter the external training filename: A:training.pas

Please enter the total number of layers (2-4): 3

Please enter the number of cells in layer 1 (1-26): 7

Please enter the number of cells in layer 2 (1-26): 3

Please enter the number of cells in layer 3 (1-26): 4

Please enter the activation type (I or L) and parameter: L 1.0

Please enter the output type (I, S or B) and parameter: I 0.1

Please enter the positive lRate Learning rate parameter: 0.5

Please enter the non-negative alpha parameter for the momentum: 0.9

Please enter the cell bias types (C, S, R or F): C

--- are the initial cell biases to be fixed (Y/N): Y

Plz enter the initial connection weight type (C,S,R or F): R

Enter a single bias for all cells:0.0

Enter the positive Pseudo Random Number seed: 0.1

Enter the lower and upper PRN limits with rmin < max: -1.0 1.0

Update biases/weights after each case or epoch (C or E): C

Normalize the output for learning? (Y/N): Y

Enter the number of training periods (epochs) to be used: 40

Enter the non-negative convergence tolerance for this training: 0.1

Do you wish intermediate output from the training period? (Y/N): Y

40 training periods(epochs) have been used so far

The (sum-of-squares)/2 of last layer error is : 1.5420993911E+00

Do you wish to apply the network?(Y/N): Y

< 표 3 > 단순의료진단신경망의 입력예

그리고 아래와 같이 3개의 훈련케이스가 데이터파일(training.pas)로 입력되어 신경망의 학습을 위해 각기간(epoch)마다 사용되었다. 이기간별 학습과정을 보면 다음과 같다.

케이스	입력치	목표
1	0 1 1 1 1 0 1	1 1 0 1
2	1 1 0 0 0 1 0	1 0 0 1
3	1 0 0 1 1 1 0	1 0 1 1

< 표 4 > 입력케이스의 예

첫 번째 학습기간(epoch)은 케이스 1 에 대한 입력으로 시작되고 다음으로 케이스 2 가 입력되고 케이스 1 과 같은 계산과정이 반복된다. 그리고 케이스 3 이처리되면 첫 번째 훈련 기간이 끝난다. 이 과정은 각 출력이 0 또는 1 에 접근할 때 까지 계속된다. 본 실험에서는 40번째 기간(epoch)에서 모든 출력값은 각각의 목표치의 0.1 범위내에 있게되었다. 즉 각 출력값은 상기의 목표치의 0.1 범위내의 값에 가까워짐으로써 여기에서 그 처리를 멈추게 된다.²³⁾ 이와같이 일단 시스템이 훈련이 되면 다른 관계된 입력이 사용될 수 있다. 출력의 적정성은 입력의 유사성에 달려있다. 예를들면 정상 이상의 체온에다 목이 따갑다라는 증상이 입력되면 이것은 입력이 1 0 0 1 0 0 0 와 같이 됨을 의미하며 이것에 의해 신경망은 다음과 같은 처방을 추천하게 된다. 즉 아스피린(0.9840), 닭고기스프(0.3247), 페니실린(0.3334), 휴식(0.9891)을 아래 표와 같이 출력하게 된다.

```

Input to (and Output from) Cell(s) u[1] - u[7]:
1 0 0 1 0 0 0

Layer 1:

Output from cell(s) u[8 ] - u[ 10 ] :
    0.5000    0.5000    0.5000

Layer 1:

Output from cell(s) u[11 ] - u[ 14 ] :
    0.9840    0.3247    0.3334    0.9891

DO U wish to continue the applications? (Y/N): N
40training period (epochs) have been used so far
the (sum-of-squares)/2 of last layers error is: 6.7882439180E-01
Is additional training still required? (Y/N): N

```

23) 이 값들은 반드시 매 각 기간의 끝에서 더 좋아지는 방향으로 개선될 필요는 없다.

< 표 5 > 출력의 보기

```
Network Model Summary:

The file used for training was: A:training.pas

There were 3 layers in this model
Layer 1 had 7 cell(s)
Layer 1 had 3 cell(s)
Layer 1 had 4 cell(s)
The activation type L was used with a parameter of 1.0000
The output type I was used with a parameter of 0.1000

The learning rate was 0.5000
The alpha parameter for the momentum term was 0.9000
Normalization of o-vector was used
The cell bias initialization type was C
--- using a constant of 0.0000

The cell biases were held at the initial value
The connection weight initialization type was R
--- using a PRN seed of 0.1000 and range of (-1.0000,1.0000)

Update of weights/biases performed after each training case (example)

There were 40 training periods (epochs) used

there were 3 training case in A:training.pas

The convergence tolerance of the last training set was: 1.0000000000E-01

The (sum-of-squares)/2 of the last layer errors was: 6.7882439180E-01

End of Backpropagation Program
```

< 표 6 > 네트워크실행의 출력요약

이상의 실행결과에서 볼 때 입력이 1 0 0 1 0 0 0 과 같이 유사할 경우 아스피린과 휴식이 확실히 추천된다. 본 프로그램에서 선택한 무게치 갱신방법으로 케이스갱신을 선택하였다. 기간갱신을 택하여도 비슷한 결과가 기대된다. 통상적으로 시스템을 적당하게 훈련시키기

위해 수십에서 수천번의 많은 훈련기간이 필요하다. 학습시간은 모수들, 최초무게치행렬, 숨은계층에서의 셀의 수와 같은 여러 가지 요소에 달려있다. 본 연구의 보기는 2 계층모델에서 더 빨리 수행될 수 있다. 그러나 계층수가 많은 네트워크는 좀더 유연하고 더 폭 넓은 문제들을 해결할 수가 있다. 신경망시스템을 훈련할 때 항상 작동하는 단순한 절차가 있는 것은 아니다. 단지 경험만이 빠르게 훈련시킬 수 있는 시스템을 설계할 수 있다. 본 연구에서 실행한 역전파는 숨은계층신경망을 훈련하는데 중요한 돌파구를 제공한다. 그러나 이의 사용에는 몇가지 문제점도 있다. 첫째 지역적최소량(Local Minima) $E(w)$ 의 광역적 최소값을 산출할 수 없는 w 가 있을수 있다. 이러한 경우 w 에 대한 다른 초기값을 추정하여 이를 다시 시도해야한다. 아래에서 지역적최소량 $E(w)$ 을 보면 다음과 같다.

$$E(w) = 1/2 \sum_{i=1}^c \sum_{j=1}^m [goal_{i,j} - o_{i,j}(w)]^2$$

상기식에서 단 c 는 학습세트에서 케이스의 수 m 은 출력계층에서의 셀의 수 $o_{i,j}(w)$ 는 $goal_{i,j}$ 에 대응하는 출력계층출력치이다. 최소자승에러(Least Square Error) $E(w)$ 는 목표와 이에 대응하는 출력값 간의 차이에 대한 자승합을 나타내며 이의 기울기(gradient)는 무계치의 변화량과 일치한다. 둘째 느린학습의 문제가 있다. 때때로 무계치 변화는 무시할 수 있을 정도로 극히 작다. 이 경우 w 에 대한 다른 시작값을 사용하거나 활동함수를 변경할 수 있다. 그리고 신경망 학습을 가속화하기 위해 다른 기법이 개발되고 있다.

V. 결론

본연구에서는 단순신경망의 구조와 특성을 이해하기 위해 신경회로망의 알고리즘을 이론적으로 분석하고 이를 토대로 프로그램을 설계 실행하여 신경망의 학습과정을 실험하였다. 본연구에서 채택한 학습알고리즘은 3계층구조의 역전파알고리즘이며 신경망의 모형은 단순의료전문가시스템모형을 입력치로 채택하였다. 계층수, 노드수, 학습사이클 수, 학습율, 모멘텀항등의 모수를 입력한 실험의 결과는 입력치에 대한 출력이 기대목표와 거의 일치하였다. 신경네트워크는 맥컬로(McColloch W. S.)와 피츠(Pitts W.)에 의해 1940년대 초기에 개발되었다. 맥컬로는 정신과의사와 신경해부학자였고 피츠는 수학천재였다.²⁴⁾ 신경망 또는 뉴럴네트워크는 고정도 계산처리보다는 빠른 기억연상능력으로 더 잘 알려져 있으며 많은 양의 데이터를 빠르고 효율적으로 처리한다. 이 시스템은 어떤 데이터들이 없거나 잘못되더라도 잘 수행될 수 있는 특징을 가지는 강건한 시스템이다. 뉴럴네트는 특별한 센서들을 포함하는 다양한 원천들로부터 받은 입력들을 통일된 표현으로 표시가 가능하게 만들어준다. 뉴럴네트는 비전이나 음성처리등과 같은 센서와 관련된 패턴인식에서 특히 우수한 것으로 판명되고 있다. 그리고 여러 가지의 패턴매칭문제에서도 잘 돌아간다. 뉴럴네트는 병렬분산처리(PDP: Parallel Distributed Processing) 또는 연결모델(Connectionist)이라고도 한다. 본 연구의 단순신경망은 전술한 각분야에서 응용될 수 있는 기초를 제공한다는 면에서 그 의의가 있다고 본다. 그리고 신경망을 개발할 때 사용하는

24) Abhijit S. Panadya and Robert B. Macy, *Pattern Recognition with Neural Networks in C++*, CRC Press, 1996, p.35

프로그램 언어는 LISP 보다는 사실 C, PASCAL, FORTRAN 으로 개발 실행되거나 NETWARE 라고 불리는 소프트웨어 틀에 의해 수행된다. 신경망을 수행할 수 있는 하드웨어로 구현된 특수컴퓨터 칩도 개발되어 있다.²⁵⁾ 본 신경망 연구는 패턴인식, 전문가시스템 또는 제어등의 분야로 그 응용을 넓히거나 학습효율을 떨어뜨리지 않는 백프로파게이션의 이론적 모형개발 등의 향후 개발과제를 안고 있다 하겠다.

참고문헌

1. 김명원, 방승양, 이수영, 이종호, 전홍태, 정호선, 정홍공저, 「알기쉬운 신경망컴퓨터」, 전자신문사, 1992
2. 김응수역, 「C로 만드는 뇌의 정보시스템」, 생능출판사, 1996
3. 권용석, “인공신경망을 이용한 최초공모주식의 최초균형시장가격의 예측”, 한국과학기술원 테크노경영대학원 석사학위논문, 1996
4. 박문용, 최항식역, 中野肇, 飯沼一元, ニューロン、ネシトグループ’, 桐谷滋著, 「뉴로컴퓨터」, 대영사, 1991
5. 위남숙, “A Neural Network Model for Assessing Expert Judgements”, 「한양대학교논문집」, 1992, pp.451-463
6. 이재현, “Neural Networks를 이용한 한글문자인식시스템설계 및 성능분석”, 「원주대학학술논총」 제22집, pp.101-103
7. 임영도, 이상부공저, 「퍼지.신경망.유전진화」, 도서출판 영과 일, 1997
8. Soon Ho Jung, “Extendibility of Neural Networks”, 「부경대학교논문집」 제2권 제2호, 1997, pp.19-24
9. 정호선, 여진경공역, 「뉴로.퍼지.카오스:신세대 아날로그 컴퓨터 입문」, 대광서림, 1994
10. 조성배, “CAM-Brain: 셀룰라 오토마타 기반의 진화하는 신경망”, 「한국정보과학회봄학술발표논문집」, Vol27, No.1, 2000, pp.459-465
11. Abhijit S. Panadya and Robert B. Macy, *Pattern Recognition with Neural Networks in C++*, CRC Press, 1996
12. A.J. Maren, “Feed-Forward Networks”, *Journal Of Neural Network Computing*, Fall 1989, pp.54-58
13. Curt A. Lovey, Cameron W. Brennan and James A. Anderson, A Neural Network Knowledge Base For Medical Diagnosis, *Journal Of Neural Network Computing*, summer 1990, pp.6-17
14. Derek F. Stubbs, “Six Ways To Improve Back-Propagation Results”, *Journal Of Neural Network Computing*, spring 1990, pp.64-67
15. E. Michael Azoff, *Neural Network: Time Series Forecasting of Financial Markets*, John Wiley & Sons, 1994
16. G. Allen Pugh, “Neural Works Professional II”, *Journal Of Neural Network Computing*, Fall 1989, pp.61-64
17. James L. Noyes, *Artificial Intelligence With Common Lisp: Fundamentals of Symbolic and Numeric Processing*, D.C. Heath and Company, 1992
18. Joey Rogers, *Object-Oriented Neural Networks in C++*, Academic Press, 1997

25) James L. Noyes, *Artificial Intelligence with common LISP: Fundamental of Symbolic and Numeric Processing*, 1992, p.420

19. K. Fukumizu and S. Amari, Local Minima and Plateaus In Hierarchical Structures Of Multilayer Perceptions, *Neural Networks*, 13(2000) pp.317-327
20. M.A. Kon and L. Plaskota, "Information Complexity Of Neural Networks", *Neural Networks*, 13(2000) pp.365-375
21. Martin T. Hagen, Howard B. Demuth and Mark Beale, *Neural Network Design*, PWS Publishing Company, 1996
22. Philip D. Wasserman, "A Combined Back Propagation/Cauchy Machine Network", *Journal Of Neural Network Computing*, winter 1990, pp.34-38
23. Robert J. Schalkoff, *Artificial Neural Networks*, McGraw-Hill International Editions, 1997
24. Robert R. Trippi and Efraim Turban, *Neural Networks in Finance and Investing*, Probus Publishing Company, 1993
25. S. He, H. Rief and R. Unbehauen, "Multilayer Neural Networks For Solving A Class Of Partial Differential Equations", *Neural Networks*, 13(2000), pp.385-396
26. Simon Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice Hall, 1999
27. Sorin Draghic, "Neural Networks In Analog Hardware Design and Implementation Issues", *International Journal Of Neural System*, Vol.10, No.1(February, 2000) pp.19-42
28. Susan Eberlein, "Developing A Decision-Making Network For A Rover", *Journal Of Neural Network Computing*, Fall 1989, pp.5-15
29. W.E. Dietz, E.L. Kiech and M. Ali, "Jet and Rocket Engine Fault Diagnosis In Real Time", *Journal Of Neural Network Computing*, summer 1989, pp.5-18
30. William H. Verduin, "Neural Networks For Diagnosis and Control", *Journal Of Neural Network Computing*, winter 1990, pp.46-52
31. Young kiu Choi, Min Jung Lee and Sung Shin Kim, "An Adaptive Neural Network Compensator Control Systems With Application To Robot Manipulator", *Journal Of Electrical Engineering and Information Science*, Vol.5, No.2, 2000, pp.61-65
32. Young Ohc Yoon, Robert W. Brobst, Paul R. Bergstresser and Lynn L. Peterson, "A Desktop Neural Network For Dermatology Diagnosis", *Journal Of Neural Network Computing*, summer 1989, pp.43-51
33. Yuri Shestov, " A Neural Network Enhancement To Von Neumann Computer Architecture", *Journal Of Neural Network Computing*, winter 1990, pp.41-45

A Design And Implementation Of Simple Neural Networks System In Turbo Pascal

Woo, Won Tack

Faculty of Information Science, College of Natural Science, Kyungsan University

The field of neural networks has been a recent surge in activity as a result of progress in developments of efficient training algorithms. For this reason, and coupled with the widespread availability of powerful personal computer hardware for running simulations of networks, there is increasing focus on the potential benefits this field can offer. The neural network may be viewed as an advanced pattern recognition technique and can be applied in many areas such as financial time series forecasting, medical diagnostic expert system and etc.. The intention of this study is to build and implement one simple artificial neural networks hereinafter called ANN. For this purpose, some literature survey was undertaken to understand the structures and algorithms of ANN theoretically. Based on the review of theories about ANN, the system adopted 3-layer back propagation algorithms as its learning algorithm to simulate one case of medical diagnostic model. The adopted ANN algorithm was performed in PC by using turbo PASCAL and many input parameters such as the numbers of layers, the numbers of nodes, the number of cycles for learning, learning rate and momentum term. The system output more or less successful results which nearly agree with goals we assumed. However, the system has some limitations such as the simplicity of the programming structure and the range of parameters it can dealing with. But, this study is useful for understanding general algorithms and applications of ANN system and can be expanded for further refinement for more complex ANN algorithms.