

논문 2004-41C1-2-8

# 임팩트 팩터를 이용한 신경 회로망의 연결 소거 알고리즘

## (A Pruning Algorithm of Neural Networks Using Impact Factors)

이 하 준\*, 정 승 범\*\*, 박 철 훈\*\*\*

(Ha-joon Lee, Seung-Beom Chung, and Cheol-Hoon Park)

### 요 약

일반적으로 작은 구조의 신경 회로망은 좋은 일반화 성능을 나타내지만 원하는 학습 목표까지 학습하기가 어려운 경향이 있다. 반면에 큰 구조의 신경 회로망은 학습 데이터는 쉽게 배우지만 일반화 성능이 좋지 않은 경향이 있다. 따라서 좋은 일반화 성능을 얻기 위한 일반적인 방법은 학습이 되는 한도 내에서 최소 구조의 신경 회로망 즉 최적 구조 신경 회로망을 찾는 것이다. 본 논문에서는 가중치의 제곱과 뉴런 출력의 분산의 곱으로 정의되는 임팩트 팩터(ImF: Impact Factor)를 이용한 새로운 연결 소거 알고리즘을 제안한다. 그리고 함수 근사화 문제에 적용하여 제안된 방법이 효율적임을 보인다.

### Abstract

In general, small-sized neural networks, even though they show good generalization performance, tend to fail to learn the training data within a given error bound, whereas large-sized ones learn the training data easily but yield poor generalization. Therefore, a way of achieving good generalization is to find the smallest network that can learn the data, called the optimal-sized neural network. This paper proposes a new scheme for network pruning with 'impact factor' which is defined as a multiplication of the variance of a neuron output and the square of its outgoing weight. Simulation results of function approximation problems show that the proposed method is effective in regression.

**Keywords :** pruning algorithm, Impact Factor, Artificial Neural Networks, regression, structure optimization

## I. 서론

\* 학생회원, 한국과학기술원 전자전산학과  
(Dept. of Electrical Engineering and Computer Science, Korea Advanced Institute of Science and Technology)

\*\* 정회원, 삼성전자(주) 기술총괄 소프트웨어센터  
(Samsung Electronics, Software Center)

\*\*\* 정회원, 한국과학기술원 전자전산학과  
(Dept. of Electrical Engineering and Computer Science, Korea Advanced Institute of Science and Technology)

※ 본 연구는 한국과학재단 목적기초연구(R01-2003-000-10829-0) 지원으로 수행되었습니다.

접수일자 : 2003년2월3일, 수정완료일 : 2004년2월16일

폰 노이만(von Neumann) 방식을 기본으로 하는 현재의 컴퓨터는 자료 저장, 자료 관리 및 검색, 수치적 자료의 처리 면에서 탁월한 성능을 보이고 있으나 영상이나 음성 인식과 같이 병렬성이 내재한 실시간 처리나 제어 등의 효율적인 수행에 있어서는 인간에 크게 못 미치는 결과를 내고 있다. 이러한 점들을 극복하고자 인공 신경 회로망(Artificial Neural Networks)이 등장하게 되었다<sup>[1]</sup>.

인공 신경 회로망은 인간 두뇌의 생물학적 신경 회로

망을 수학적으로 모델링한 것인데 실제 응용 분야에 가장 많이 사용되는 신경 회로망 모델 중의 하나는 시그모이드 비선형 활성화 함수를 가진 다층 구조 퍼셉트론 (MLPs: Multilayer Perceptrons)이다. 신경 회로망의 실제 응용에 있어서 고려되어야 할 여러 가지 능력들 중, 주어진 학습 데이터에 대한 학습 능력과 학습되지 않은 데이터에 어떻게 대응하는지에 관한 일반화 성능은 매우 중요하다. 문제의 복잡도보다 신경 회로망의 복잡도가 더 크게 되면 주어진 데이터에 대해서는 학습이 잘 되지만 학습되지 않은 데이터에 대해서는 과도학습 (overfitting) 현상이 나타나고 그 반대의 경우에는 학습이 잘 이루어지지 않는 부족학습 (underfitting) 현상이 나타난다<sup>[1, 2]</sup>. 이 문제를 해결하기 위하여 여러가지 방법들이 제안되었는데 이를 크게 구조 증가 알고리즘 (construction algorithm)<sup>[3]</sup>과 연결 소거 알고리즘 (pruning algorithm)<sup>[2]</sup>으로 나누어 볼 수 있다. 구조 증가 알고리즘은 작은 네트워크에서부터 학습을 시작하여 학습이 될 때까지 네트워크의 크기를 증가시키는 방법으로 사전에 문제의 복잡도를 알지 못해도 사용 가능하다는 장점이 있다. 하지만 과도학습 현상이 발생할 수 있고 학습 시간이 너무 오래 걸릴 수도 있다는 단점이 있다<sup>[4]</sup>. 연결 소거 알고리즘은 주어진 문제에 필요한 것보다 충분히 큰 네트워크로 학습한 다음 필요 없는 뉴런과 가중치들을 소거하여 네트워크의 크기를 줄여가는 방법이다. 이 방법은 큰 네트워크가 가중치의 초기값에 덜 민감하기 때문에 학습이 쉽게 되는 장점과 작은 네트워크의 좋은 일반화 성능을 함께 가지고 있는 방법이다<sup>[2,5]</sup>. 여러 가중치들 중에서 어떤 가중치를 소거할 것인지 즉 가중치들의 중요도에 대한 기준을 어떻게 정의하는가에 따라 여러 가지 방법들이 제안되었다<sup>[2]</sup>. 물론 구조 증가 알고리즘을 적용하여 신경 회로망을 구성한 다음 연결 소거 알고리즘을 적용할 수도 있다<sup>[6]</sup>. 연결 소거 알고리즘으로는 다음 장에서 설명하는 다양한 방법들이 제안되었는데, 수학적인 뒷받침이 부족하거나 오차 함수에 대한 가정이 존재하여서 실제 사용하는데 어려움이 있거나 정확성이 떨어지는 경향이 있다. 그래서 우리는 첫째, 최소 구조 신경 회로망을 찾는데 효율적이고 우수한 일반화 성능을 지니며 둘째, 계산량이 적으며 셋째, 연결 소거 후 바이어스를 보상하여 오차 증가를 감소시키며 넷째 물리적인 의미를 지닌 임팩트 팩터 연결 소거 알고리즘을 제안한다.

II장에서는 기존에 제안된 연결 소거 알고리즘과 그

문제점에 대하여 살펴보고 III장에서 새로운 연결 소거 알고리즘을 제안한다. 그리고 IV장에서 함수 근사화 문제에 대하여 기존의 알고리즘과 비교하였다. 마지막으로 V장에서 결론을 맺는다.

## II. 기존의 연결 소거 알고리즘

### 1. 가중치 크기 기반 연결 소거 알고리즘

(MB: Magnitude-Based Pruning Algorithm)

가중치의 중요도로 가장 쉽게 생각해 볼 수 있는 것은 작은 값의 가중치들이 큰 값의 가중치보다 덜 중요할 것이라는 생각에서 가중치의 크기를 사용하는 것이다. 이 방법은 가장 간단하지만 수학적인 뒷받침이 약하다는 단점이 있다<sup>[7]</sup>.

### 2. 최적 두뇌 손실(OBD: Optimal Brain Damage)

Le Cun *et al.*<sup>[8]</sup>은 오차 함수의 2차 미분을 사용하여 가중치의 중요한 정도를 나타내는 최적 두뇌 손실 알고리즘을 제안하였다. 신경 회로망이 충분히 학습되어 최소점에 도달했고 그래서 최소점에서 오차 표면이 2차 함수의 형태에 가깝다고 가정하자. 신경 회로망의 가중치  $w_i$ 가  $w_i + \delta w_i$ 로 변한다면 그에 따른 오차 함수  $E$ 의 변화  $\delta E$ 는 다음과 같이 나타낼 수 있다.

$$\delta E = \sum_i \frac{\partial E}{\partial w_i} \delta w_i + \frac{1}{2} \sum_i \sum_j H_{ij} \delta w_i \delta w_j + O(\|\delta w\|^3) \quad (1)$$

여기에서  $H_{ij}$ 는 다음과 같은 헤시안 행렬의 원소이다.

$$H_{ij} = \frac{\partial^2 E}{\partial w_i \partial w_j} \quad (2)$$

학습 과정이 수렴하였다고 가정하였기 때문에 식 (1)의 첫 번째 항은 무시할 수 있다. 헤시안 행렬을 대각 행렬로 근사화하고 고차항을 무시하면 오차 함수의 변화  $\delta E$ 는 다음과 같이 표현된다.

$$\delta E \approx \frac{1}{2} \sum_i H_{ii} \delta w_i^2 \quad (3)$$

만약 현재 값이  $w_i$ 인 가중치를 0으로 만든다면  $\delta w_i = -w_i$  이므로 오차의 증가는 식 (3)에 의하여 근

사적으로  $\frac{1}{2} H_{ii} w_i^2$ 으로 나타낼 수 있다. 따라서 가중치들의 중요 정도를 나타내는  $S_i$ 를 식 (4)로 정의하고  $S_i$ 가 작은 가중치부터 제거하고 재학습을 하게 된다.

$$S_i = \frac{1}{2} H_{ii} w_i^2 \quad (4)$$

### 3. 최적 두뇌 의사(OBS: Optimal Brain Surgeon)

최적 두뇌 손실 알고리즘은 헤시안 행렬을 대각행렬로 근사화하였다. 하지만 헤시안 행렬은 실제로는 대각행렬이 아니다. Hassibi와 Stork<sup>[9]</sup>는 최적 두뇌 손실 알고리즘에서 헤시안 행렬이 대각 행렬이라는 가정을 제거하여 최적 두뇌 손실 알고리즘의 일반적인 형태인 최적 두뇌 의사 알고리즘을 제안하였다. 최적 두뇌 의사 알고리즘은 연결 소거 후 남은 가중치들을 보상해 주는 과정(식 (6))이 있기 때문에 연결 소거 후 오차의 증가는 최소화되고 재학습에 걸리는 시간은 줄어든다. 가중치들의 중요도를 나타내는  $S_i$ 는 식 (5)로 정의된다.

$$S_i = \frac{w_i^2}{2 [H^{-1}]_{ii}} \quad (5)$$

$$\delta \mathbf{w} = - \frac{w_i}{[H^{-1}]_{ii}} \mathbf{H}^{-1} \mathbf{e}_i \quad (6)$$

여기에서  $\mathbf{e}_i$ 는 가중치 공간에서  $w_i$ 축에 평행한 단위 벡터이다. 하지만 최적 두뇌 의사 알고리즘은 헤시안 행렬의 역행렬을 구해야 하는 계산상의 복잡성이 있다<sup>[1]</sup>. 그리고 연결 소거할 가중치의 크기가 클 경우 식 (1)의 2차 근사화가 성립하지 않아서 실험 결과에서 보이듯이 연결 소거 후 오차를 보상해 주는 과정이 오히려 오차를 증가시키는 경우도 발생한다.

### III. 임팩트 팩터 연결 소거 알고리즘

본 논문에서 제안하는 연결 소거 알고리즘의 핵심 아이디어는 어느 특정한 가중치의 중요도를 그 가중치가 연결된 다음 층의 뉴런에 얼마나 중요한 역할을 하는가를 데이터 공간에서 직접 측정하는 것이다. 가중치의 중요도를 단순히 가중치의 크기만의 함수로 정의하는 것은 불충분하다는 생각을 해 볼 수 있으며 신호가 전방향으로 전달됨에 있어서 가중치의 중요도와 그것이 연결되어 있는 뉴런의 중요도를 한꺼번에 고려한다면 더

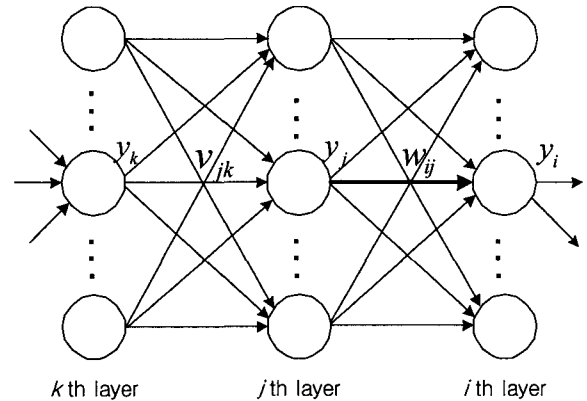


그림 1. 다층 퍼셉트론의 일부분.  
Fig. 1. A part of MLPs.

욱 효과적인 측정량이 될 것이다. 가중치의 중요도는 가중치의 크기로 나타낼 수 있고, 뉴런의 중요도는 그 뉴런의 출력 값의 변화량이라고 할 수 있다. 어느 뉴런이 다른 것들에 비해서 값이 더 큰 폭으로 변한다면 다음 층으로 연결된 가중치가 모두 같다고 가정할 때 출력에 더 큰 영향을 미치는 것은 당연하다. 또, 어느 뉴런의 출력이 들어오는 입력에 대해서 상수처럼 거의 변화하지 않는다면 여기에 연결된 모든 가중치들은 다음 층의 바이어스로 영향을 미치는 것으로 해석할 수 있다.

그림 1과 같이 다층 퍼셉트론의 일부만 생각해 보자.  $m$ 번째 입력 데이터에 대한 뉴런  $y_i$ 의 입력  $x_i^m$ 은 다음과 같이 나타난다.

$$\begin{aligned} x_i^m &= \sum_j w_{ij} y_j^m + w_{i0} \\ &= \sum_j w_{ij} (y_j^m - \bar{y}_j) + \sum_j w_{ij} \bar{y}_j + w_{i0} \end{aligned} \quad (7)$$

$$y_i^m = f(x_i^m), \quad y_j^m = f(x_j^m), \quad y_k^m = f(x_k^m) \quad (8)$$

여기에서  $f$ 는 활성화 함수이고  $w_{ij}$ 는  $i$ 번째 층의 뉴런  $y_i$ 와  $j$ 번째 층의 뉴런  $y_j$ 를 연결하는 가중치이고  $w_{i0}$ 는 뉴런  $y_i$ 의 입력으로 들어오는 바이어스이다. 그리고  $\bar{y}_j = \frac{1}{M} \sum_{m=1}^M y_j^m$ 이고  $M$ 은 학습 데이터의 개수이다. 식 (7)의 세 항 중 두 번째와 세 번째 항은 상수이므로 뉴런  $y_i$ 의 값은 식 (7)의 첫 번째 항에 의해 정해진다. 이와 같은 이유로, 특정 가중치  $w_{ij}$ 의 중요도 임팩트 팩터(ImF: Impact Factor)는 다음과 같이 정의한다.

$$\text{ImF } w_{ij} = w_{ij}^2 \text{Var}(y_j) \quad (9)$$

여기에서  $\text{Var}(y_j)$  는 입력 데이터에 대한 뉴런 출력  $y_j$ 의 분산이다. 임팩트 팩터가 가장 작은 가중치  $w_{ij}$ 를 제거한 후에는 식 (7)의 세항 중 두 번째 항 즉 뉴런 출력의 평균과 가중치의 곱을 다음 층의 바이어스에 더해 줌으로써 연결 소거 후 오차의 증가를 감소시킨다.

$$w_{i0new} = w_{i0old} + w_{ij}y_j \quad (10)$$

가중치의 중요도를 나타내는 가중치의 임팩트 팩터와 비슷한 방법으로 뉴런의 임팩트 팩터도 정의할 수 있는데 뉴런  $y_j$ 의 임팩트 팩터는 자신과 다음 층의 뉴런들과 연결된 가중치들의 임팩트 팩터의 합으로 정의한다.

$$\text{ImF } y_j = \sum_i w_{ij}^2 \text{Var}(y_j) \quad (11)$$

뉴런  $y_j$ 를 제거한 후에는 다음 식에 따라 바이어스를 보상해 준다.

$$w_{0new} = w_{0old} + w_j y_j \quad (12)$$

여기에서  $w_0 = [w_{10} \ w_{20} \ \dots \ w_{D0}]^T$ ,  $w_j = [w_{1j}$

$w_{2j} \ \dots \ w_{Ij}]^T$ 이고  $I$ 는  $i$  번째 층의 뉴런의 개수이다. 어떤 특정한 뉴런 혹은 가중치의 임팩트 팩터가 작다는 것은 그 뉴런 혹은 가중치가 다음 층에 기여하는 정도가 작다는 것을 의미하므로 각층마다 임팩트 팩터가 작은 뉴런이나 가중치부터 제거하고 재학습한다. 임팩트 팩터는 다음의 세 가지 경우에 대하여 작은 값을 가지는데 그 물리적인 의미는 다음과 같다. 첫째, 가중치의 크기  $|w_{ij}|$ 가 작을 때 임팩트 팩터가 작다. 이 경우는 다음 층으로의 연결이 약하다는 것을 의미한다. 둘째, 뉴런의 분산  $\text{Var}(y_j)$ 이 작을 때 임팩트 팩터가 작다. 이 경우는 이 뉴런이 포화 영역(saturation region)에서 동작한다는 것을 의미한다. 셋째,  $\text{Var}(y_j)$ 와

$\sum_k |w_{kj}|$  (이전 층과 현재 층 사이의 가중치, 그림 1)가 작을 때이다. 이 경우는 이전 층으로부터의 연결이 약하다는 것을 의미한다. 이런 이유로 임팩트 팩터가 작은 가중치나 뉴런을 소거한다.

표 1. 출력 층의 뉴런이 하나인 2층 구조 신경 회로망에 대한 임팩트 팩터를 이용한 연결 소거 알고리즘 전체과정.

Table 1. The overall procedure of the pruning algorithm using Impact Factor for a two-layer neural network with one output neuron.

```

num_fail←0
1. 신경 회로망을 학습한다.
2. 뉴런 소거 과정
  2.1 현재 신경 회로망 구조를 저장한다.
  2.2 만약 num_fail=0 이면 현재 남아있는 은닉 층 뉴런들의 ImF를 계산한다.
  2.3 (num_fail+1)번째로 작은 ImF를 가진 뉴런을 제거한다.
  2.4 바이어스를 보상하고 재학습한다.
  2.5 만약 training_rmse≤GOAL 이면 num_fail←0으로 설정하고 2.1번으로 간다.
  2.6 num_fail←num_fail+1
  2.7 저장된 신경 회로망 구조를 불러온다.
  2.8 만약 num_fail<NUM_TRIAL 이면 2.1번으로 간다.
3. num_fail←0
4. 가중치 소거 과정
  4.1 현재 신경 회로망 구조를 저장한다.
  4.2 만약 num_fail=0 이면 현재 남아있는 은닉 층과 입력 층 사이의 가중치들의 ImF를 계산한다.
  4.3 (num_fail+1)번째로 작은 ImF를 가진 가중치를 제거한다.
  4.4 바이어스를 보상하고 재학습한다.
  4.5 만약 training_rmse≤GOAL 이면 num_fail←0으로 설정하고 4.1번으로 간다.
  4.6 num_fail←num_fail+1
  4.7 저장된 신경 회로망 구조를 불러온다.
  4.8 만약 num_fail<NUM_TRIAL 이면 4.1번으로 간다.
5. 연결 소거 과정을 끝낸다.

```

표 1은 출력 층의 뉴런이 하나인 2층 구조 신경 회로망의 전체 연결 소거 과정을 나타낸 것이다. 연결 소거 과정은 크게 뉴런 소거와 가중치 소거 과정으로 나눌 수 있다. 표 1에서 2번은 뉴런 소거 과정이고 4번은 가중치 소거 과정이다. 표 1을 보면 은닉 층의 뉴런 소거를 먼저하고 가중치 소거를 나중에 하였다. 그 이유는 뉴런 소거를 하면 소거된 뉴런과 관련되는 가중치들이 한꺼번에 제거됨으로 속도 면에서 장점이 있기 때문에 뉴런 소거를 먼저하고 뉴런 소거가 더 이상 안 될 경우 가중치 소거를 하였다. 표 2는 출력 층의 뉴런이 하나인 3층 구조 신경 회로망의 전체 연결 소거 과정을 나타낸 것이다. 3층 구조 신경 회로망의 연결 소거도 2층 구조 신경 회로망과 마찬가지로 크게 뉴런 소거와 가중치 소

표 2. 출력 층의 뉴런이 하나인 3층 구조 신경 회로망에 대한 임팩트 팩터를 이용한 연결 소거 알고리즘 전체과정.

Table 2. The overall procedure of the pruning algorithm using Impact Factor for a three-layer neural network with one output neuron.

<p>num_fail_1←0, num_fail_2←0</p> <p>1. 신경 회로망을 학습한다.</p> <p>2. 뉴런 소거 과정</p> <p>2.1 두 번째 은닉 층의 뉴런 소거 과정</p> <p>2.1.1 현재 신경 회로망 구조를 저장한다.</p> <p>2.1.2 만약 num_fail_2=0 이면 현재 남아있는 두 번째 은닉 층 뉴런들의 ImF를 계산한다.</p> <p>2.1.3 두 번째 은닉 층의 뉴런들 중 (num_fail_2+1)번째로 작은 ImF를 가진 뉴런을 제거하고 바이어스를 보상하고 재학습한다.</p> <p>2.1.4 만약 training_rmse≤GOAL 이면 num_fail_1←0, num_fail_2←0으로 설정하고, 그렇지 않으면 num_fail_2 ← num_fail_2+1 로 설정하고 저장된 신경 회로망 구조를 불러온다.</p> <p>2.1.5 만약 num_fail_1+num_fail_2≥NUM_TRIAL 이면 3번으로 간다.</p> <p>2.2 첫 번째 은닉층의 뉴런 소거 과정</p> <p>2.2.1 현재 신경 회로망 구조를 저장한다.</p> <p>2.2.2 만약 num_fail_1=0 이면 현재 남아있는 첫 번째 은닉 층 뉴런들의 ImF를 계산한다.</p> <p>2.2.3 첫 번째 은닉 층의 뉴런들 중 (num_fail_1+1)번째로 작은 ImF를 가진 뉴런을 제거하고 바이어스를 보상하고 재학습한다.</p> <p>2.2.4 만약 training_rmse≤GOAL 이면 num_fail_1←0, num_fail_2←0으로 설정하고, 그렇지 않으면 num_fail_1←num_fail_1+1 로 설정하고 저장된 신경 회로망 구조를 불러온다.</p> <p>2.2.5 만약 num_fail_1+num_fail_2≥NUM_TRIAL 이면 2.1.1번으로 간다.</p> <p>3. num_fail_1←0, num_fail_2←0</p> <p>4. 가중치 소거 과정.</p> <p>4.1 두 번째 은닉 층과 첫 번째 은닉 층 사이의 가중치 소거 과정 (두 번째 은닉층의 뉴런 소거 과정과 비슷 방법으로 가중치 소거)</p> <p>4.2 첫 번째 은닉 층과 입력 층 사이의 가중치 소거 과정 (첫 번째 은닉 층의 뉴런 소거 과정과 비슷한 방법으로 가중치 소거)</p> <p>5. 연결 소거 과정을 끝낸다.</p>
---

거 과정으로 나눌 수 있다. 표 2에서 2번은 뉴런 소거 과정이고 4번은 가중치 소거 과정이다. 어느 특정 뉴런 혹은 가중치의 임팩트 팩터는 다음 층에 얼마나 중요한 역할을 하는지를 나타내기 때문에 층이 다른 뉴런 혹은 가중치의 임팩트 팩터는 같이 평가할 수 없기 때문에 뉴런 소거와 가중치 소거 모두 층을 번갈아 가면서 연결 소거를 하였다. 표 1과 2에서 NUM\_TRIAL은 뉴런

소거와 가중치 소거 과정에서 연속해서 소거를 실패할 때 허용 횟수를 나타낸다. 예를 들어 표 2의 3층 구조 신경 회로망에서 NUM\_TRIAL을 2로 설정하면 뉴런 소거와 가중치 소거에서 각 층에서 모두 연결 소거를 실패할 경우 연결 소거 과정을 마치게 된다.

#### IV. 실험 결과 및 고찰

제안된 연결 소거 알고리즘을 4가지 함수 근사화 문제에 적용하여 기존의 방법들과 비교하였다. 모의실험에 사용한 함수는 그림 2에 나타나 있고 그 식은 다음과 같다.

Sine function<sup>[10]</sup>:

$$f_1(x_1, x_2) = \sin\left(\frac{1}{4}x_1x_2\right) \quad (13)$$

Complicated interaction function<sup>[11]</sup>:

$$f_2(x_1, x_2) = 1.9[1.35 + e^{2x_1-1}\sin(13(2x_1-1.6)^2) \times e^{-2x_2+1}\sin(14x_2-7)] \quad (14)$$

Harmonic function<sup>[12]</sup>:

$$f_3(x_1, x_2) = \sin(2\pi\sqrt{x_1^2 + x_2^2}) \quad (15)$$

Hybrid function<sup>[5, 6]</sup>:

$$f_4(x_1, x_2) = \sin(3x_1 + \sin(3x_2)) + 0.25[3(1-3x_1)^2e^{-9x_2^2-3(x_2+1)^2} - 10(1.5x_1-27x_1^3-81x_2^4)e^{-9x_1^2-9x_2^2} - \frac{1}{3}e^{-3(x_1+1)^2-9x_2^2}] \quad (16)$$

학습 데이터는 다음의 원칙에 따라 229개의 데이터를 선택했다.

1. 학습 데이터의 정의역  $[-1, 1]^2$ 을  $15 \times 15$ 의 사각형으로 나눈다. 각각의 사각형에서 한 개의 점을 임의로 선택하여 학습 데이터 225개를 만든다.
2.  $(x_1, x_2) = (-1, -1)$ ,  $(x_1, x_2) = (-1, 1)$ ,  $(x_1, x_2) = (1, -1)$ ,  $(x_1, x_2) = (1, 1)$ 을 선택하여 학습 데이터 4개를 만든다.

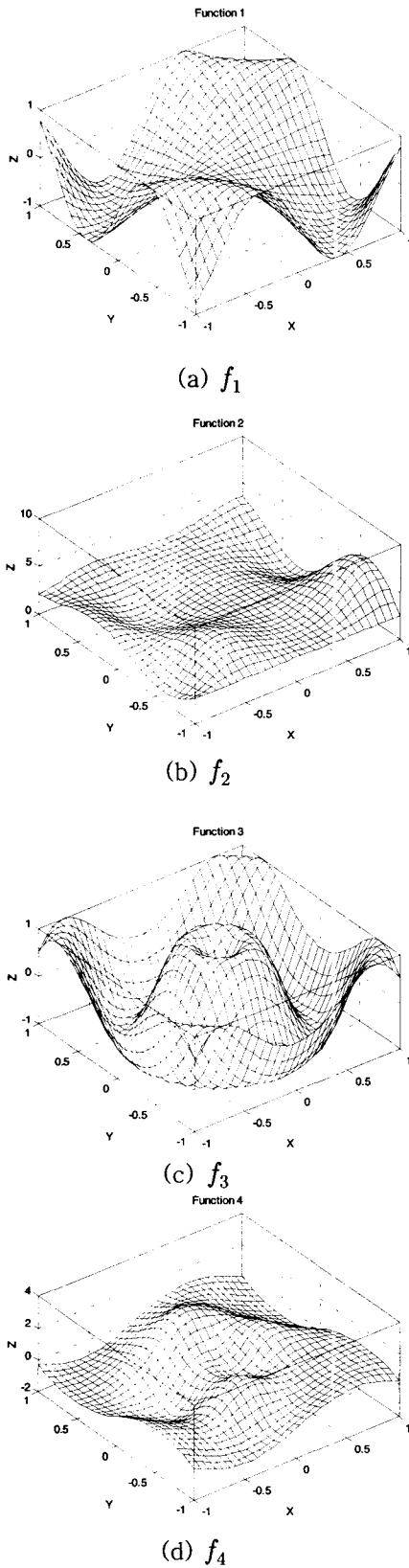


그림 2. 모의실험에 사용한 함수들.  
Fig. 2. Test functions for simulation.

일반화 성능 테스트를 위한 데이터는 10,000개의 점을 균일하게 선택했다. 신경 회로망의 성능 비교 지표로는 RMSE(Root Mean Squared Error)를 사용했다.

$$RMSE = \sqrt{\frac{1}{M} \sum_{m=1}^M (e^m)^2} \quad (17)$$

여기에서  $e^m$ 은  $m$ 번째 학습 데이터  $x^m$ 에 대한 학습 오차이다. 학습과 연결 소거 과정에서 재학습은 다음의 사항에 해당하면 종료하였다.

1. 최대 반복 횟수(max epochs) : 1000
2. 학습 RMSE: 0.02 이하

신경 회로망의 초기화는 모든 신경 회로망의 가중치들을  $[-0.05, 0.05]$ 에서 임의의 값으로 초기화하였고 학습 알고리즘은 Levenberg-Marquardt 알고리즘<sup>[13]</sup>을 사용하였다. 성능 비교 지표로는 다음을 사용하였다.

1. NGP(Normalized Generalization Performance):  
Generalization RMSE / Training RMSE
2. 자유도(DoFs: Degrees of Freedom): 신경 회로망의 가중치의 개수

단순히 일반화 RMSE만 비교를 하면 학습이 잘 되어서 일반화 성능이 좋은 것인지 아니면 학습 목표치보다 학습이 훨씬 많이 되어서 성능이 좋은 것처럼 나타난 것인지 구분이 되지 않는다. 이를 공정하게 비교하기 위하여 일반화 RMSE를 학습 RMSE로 나눈 값을 NGP로 정의하고 이를 첫 번째 성능비교 지표로 삼았다. 두 번째 지표는 신경 회로망의 가중치의 개수 즉 신경 회로망의 복잡도로 잡았다. 가중치의 개수가 적을수록 연산 속도가 빠르고 메모리 측면에서 보면 비용이 적게 드는 장점이 있다. 대체로 작은 신경 회로망 즉 자유도가 작을수록 일반화 성능이 좋지만 반드시 그런 것은 아니기 때문에 모의실험 결과는 NGP, 자유도 이 두 가지를 함께 고려하여 살펴보았다. 모든 결과는 20번 반복 실험을 해서 평균을 구했다. 반복 실험의 각각은 동일한 학습된 신경 회로망에 대하여 연결 소거 알고리즘들을 적용하여 비교하였다. 실험은 2층 구조 신경 회로망과 3층 구조 신경 회로망의 경우에 대하여 기존의 연결 소거 알고리즘들과 비교하였다. 초기 신경 회로망의 크기는 학습이 충분히 잘되는 크기로 정하였는데  $f_1, f_2, f_3, f_4$ 에 대하여 2층 구조 신경 회로망의 경우에는 각각 2-20-1, 2-30-1, 2-40-1, 2-40-1을 사용하였고 3층 구조 신경 회

표 3. 2층 구조 신경 회로망에 대한 제안된 연결 소거 방법과 기존의 방법들의 성능 비교.  
Table 3. Comparison of the performance of the proposed method with the previous ones for two-layer neural networks.

\* NGP(DoFs)

함수	연결소거전	MB	OBD	OBS	ImF
$f_1$	<b>1.2273</b> (81)	1.2188 (41.60)	1.2774 (41.53)	<b>1.2105</b> (41.63)	1.2399 <b>(41.07)</b>
$f_2$	1.6409 (121)	1.4281 (62.13)	1.4852 (61.57)	1.5503 <b>(61.50)</b>	<b>1.3942</b> (61.63)
$f_3$	<b>1.5889</b> (161)	1.6702 (92.57)	1.7795 (94.20)	<b>1.6577</b> <b>(87.47)</b>	1.6866 (89.00)
$f_4$	<b>1.6240</b> (161)	1.8163 (90.50)	1.8640 (91.07)	1.6533 <b>(86.67)</b>	<b>1.6477</b> (87.07)

표 4. 3층 구조 신경 회로망에 대한 제안된 연결 소거 방법과 기존의 방법들의 성능 비교.  
Table 4. Comparison of the performance of the proposed method with the previous ones for three-layer neural networks.

\* NGP(DoFs)

함수	연결소거전	MB	OBD	OBS	ImF
$f_1$	1.2102 (151)	1.1971 (38.05)	1.1100 (38.15)	<b>1.0772</b> (32.65)	1.0880 <b>(28.80)</b>
$f_2$	2.1696 (151)	1.4485 (53.45)	1.3763 (53.30)	1.4928 (53.05)	<b>1.3554</b> <b>(49.95)</b>
$f_3$	1.5469 (151)	1.7561 (48.75)	1.4155 (51.55)	1.8501 (57.40)	<b>1.2490</b> <b>(42.55)</b>
$f_4$	1.9015 (151)	1.5089 (63.40)	1.4310 (62.00)	1.5040 (65.40)	<b>1.4100</b> <b>(61.55)</b>

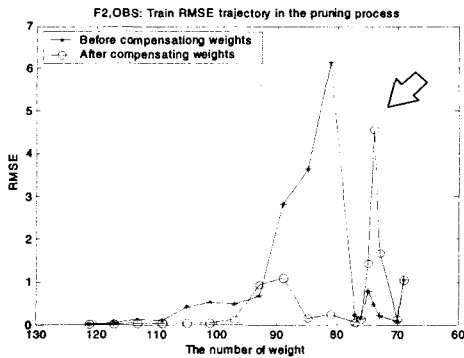
로망의 경우에는 모두 2-10-10-1을 사용하였다. 기존의 연결 소거 알고리즘과 비교를 위하여 크기 기반 연결 소거 알고리즘(MB), 최적 두뇌 손실(OBD), 최적 두뇌 의사(OBS)을 사용하여 제안된 방법인 임팩트 팩터 연결 소거 알고리즘(ImF)과 비교하였다. 2층 구조 신경 회로망의 실험에서는 뉴런 소거와 가중치 소거 과정에서 각각 한 번 실패하면 연결 소거 과정을 마쳤다. 임팩트 팩터 연결 소거 알고리즘은 NUM\_TRIAL을 1로 설정했다. 그리고 3층 구조 신경 회로망의 실험에서는 뉴런 소거와 가중치 소거 과정에서 각각 두 번 연속해서 실패하면 연결 소거 과정을 마쳤다. 임팩트 팩터 연결 소거 알고리즘은 NUM\_TRIAL을 2로 설정했다.

표 3과 4는 각각 2층 구조 신경 회로망과 3층 구조 신

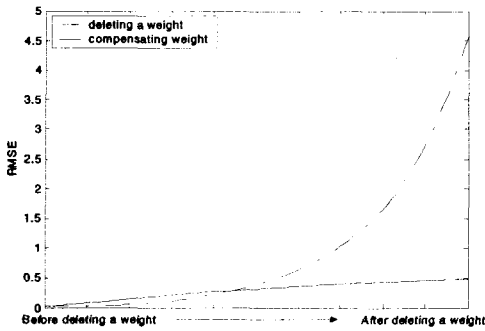
경 회로망에 대한 제안된 연결 소거 방법과 기존의 연결 소거 방법들의 일반화 성능과 자유도를 비교 실험한 결과이다. 2층 구조 신경 회로망에 대한 실험에서는 최적 두뇌 의사 연결 소거 알고리즘이 함수  $f_2, f_3, f_4$ 의 경우 최소 구조를 찾은 데 반해 제안한 임팩트 팩터를 이용한 연결 소거 알고리즘은  $f_1$ 의 경우에만 최소 구조를 찾았다. 그리고 NGP는 연결 소거하기 전의 신경 회로망의 일반화 성능이 좋으므로 연결 소거 후 성능 향상이 없었다. 하지만 이보다 복잡한 구조인 3층 구조 신경 회로망의 경우는 본 논문에서 제안한 임팩트 팩터를 이용한 연결 소거 알고리즘이 모든 테스트 함수에 대하여 최소 구조를 찾았고  $f_1$ 의 경우는 두 번째로 좋은 일반화 성능을 나타내었다.

최적 두뇌 의사 연결 소거 알고리즘이 3층 구조 신경 회로망의 경우에 좋지 않은 결과를 나타낸 것은 알고리즘 유도 과정에서 오차 함수를 2차로 근사화하기 때문으로 생각된다. 신경 회로망의 구조가 비교적 간단한 2층 구조인 경우에는 제대로 동작했지만 보다 복잡한 3층 구조 신경 회로망의 경우에는 다음에서 설명하는 것과 같이 오차 함수를 2차로 근사화하는 것이 제대로 맞지 않아서 좋지 않은 결과를 나타낸 것으로 생각된다.

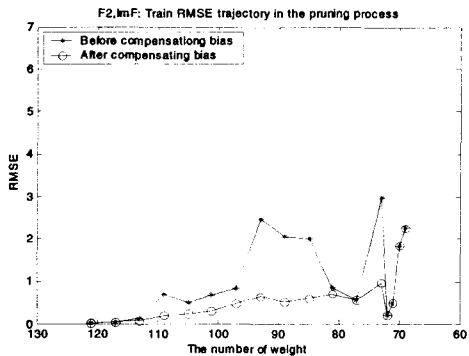
제안된 임팩트 팩터를 이용한 연결 소거 방법은 성능 측면 이외에도 몇 가지 장점을 가지고 있다. 첫째, 헤시안 행렬의 역행렬을 구해야 하는 최적 두뇌 손실, 최적 두뇌 의사 연결 소거 알고리즘에 비해 계산량이 적다. 둘째, 연결 소거 과정에서 오차 증가를 보상해 주는 과정이 있다. 이것은 물론 최적 두뇌 의사 방법에도 있지만 최적 두뇌 의사 방법은 오차 함수가 2차 근사화가 가능한 경우에는 오차 증가를 보상해 주지만 2차 근사화가 안되는 경우에는 연결 소거 후 남아있는 가중치를 보상하기 전보다 보상 후 오차가 오히려 증가하는 현상을 나타낸다. 그림 3이 그 예를 보여준다. 그림 3의 (a)는 함수  $f_2$  문제에서 2층 구조 신경 회로망에 대하여 최적 두뇌 의사 알고리즘으로 연결 소거할 때 현재 남아있는 가중치의 수(자유도)에 대한 학습 오차를 나타낸 것이다. 가중치의 수가 97에서 93으로 뉴런을 소거했을 때와 77에서 76, 76에서 75, 75에서 74로 가중치를 연결 소거했을 때 오차를 보정한 후 오히려 오차가 커지는 현상을 볼 수 있다. 그림 3의 (b)는 이중 가중치의 수가 77개에서 76개로 가중치를 서서히 제거함에 따른 오차의 변화(실선)와 남아있는 가중치를 서서히 보상함에 따른



(a)



(b)



(c)

그림 3. 오차 보상 과정 비교. (a) 최적 두뇌 의사로 연결 소거할 때 가중치 보상전과 보상후의 학습 오차. (b) (a) 그림에서 가중치를 76개에서 75개(화살표 표시)로 서서히 제거함에 따른 학습 오차의 변화. (c) 임팩트 팩터로 연결 소거 할 때 가중치 보상전과 보상후의 학습 오차.

Fig. 3. The effect of error compensation. (a) Training error before and after the weight compensation in OBS. (b) Variation of training error when a weight is gradually deleted. (c) Training error before and after the weight compensation in lmF.

오차의 변화(점선)를 나타낸다. 처음에는 보상후의 오차가 더 작지만 나중에는 보상후의 오차가 더욱 증가하는 현상을 나타낸다. 이것은 오차 함수를 2차로 근사화한 것이 정확하지 않기 때문이다. 하지만 임팩트 팩터의 경우는 아무런 제약 조건 없이 바이어스만을 보상하기 때문에 그림 3의 (c)에서 보는 것과 같이 보상 과정에서 오차를 더 증가시키는 현상은 일어나지 않는다. 셋째, 임팩트 팩터는 III장에서 언급했듯이 물리적인 의미를 가지는 장점이 있다.

### V. 결론

본 논문에서는 주어진 문제에 알맞게 다층 구조 신경 회로망의 구조를 최적화할 수 있는 임팩트 팩터를 이용한 연결 소거 알고리즘을 제안하였다. 제안된 알고리즘의 장점은 다음과 같다. 첫째, 복잡한 3층 구조 신경 회로망의 경우 기존의 연결 소거 알고리즘과 비교했을 때 최소의 자유도(DoFs)를 가지면서 우수한 일반화 성능을 나타낸다. 둘째, 뉴런 혹은 가중치의 중요도를 계산할 때 최적 두뇌 손실, 최적 두뇌 의사 연결 소거 알고리즘에 비하여 계산 량이 적다. 셋째, 연결 소거 후 바이어스를 보상해 줌으로써 연결 소거로 인한 오차 증가를 감소시킨다. 넷째, 연결 소거 과정에서 어떤 가중치 혹은 뉴런이 연결 소거되는지에 대한 물리적인 의미를 가지고 있다. 추후 과제로 제안된 알고리즘을 패턴 인식 문제에 적용해 볼 예정이다.

### 참고 문헌

- [1] S. Haykin, *Neural Networks: A Comprehensive Foundation*, NJ: Prentice Hall, 1999.
- [2] R. Read, "Pruning algorithm - a survey," *IEE Transactions on Neural Networks*, vol. 4, no. 5, pp. 740-747, Sep. 1993.
- [3] T. Y. Kwok, and D. Y. Yeung, "Constructive algorithm for structure learning in feedforward neural networks for regression problems," *IEEE Transactions on Neural Networks*, vol. 8, no. 3, pp. 630-645, May 1997.
- [4] A. P. Engelbrecht, "A new pruning heuristic based on variance analysis of sensitivity in formation," *IEEE Transactions on Neural*



- Networks*, vol. 12, no. 6, pp. 1386-1399, Nov. 2001.
- [5] H. Lee, and C. H. Park, "A pruning algorithm of neural networks using impact factor regularization," *Proceedings of the 9th International Conference of Neural Information Processing*, vol. 5, pp. 2605-2609, Singapore, Nov. 2002.
- [6] J.-S. Lee, and C. H. Park, "Self-organizing neural networks using adaptive neurons," *Proceedings of the 9th International Conference of Neural Information Processing*, vol. 2, pp. 935-939, Singapore, Nov. 2002.
- [7] C. M. Bishop, *Neural Networks for Pattern Recognition*, Oxford, U.K.: Oxford Univ. Press, 1995.
- [8] Y. LeCun, J. S. Denker, and S. A. Solla, "Optimal brain damage," *Advances in Neural Information Processing Systems*, vol. 2, pp. 598-605, San Mateo, CA:Morgan Kaufmann, 1990.
- [9] B. Hassabi, and D. G. Stork, "Second order derivative for network pruning: optimal brain surgeon," *Advances in Neural Information Processing Systems*, vol. 5, pp. 164-171, 1993.
- [10] L. Breiman, "The pi method for estimating multivariate functions from noisy data," *Technometrics*, vol. 3, no. 2, pp. 125-160, 1991.
- [11] M. Maechler, D. Martin, J. Schimert, M. Csooppensky, and J. N. Hwang, "Project pursuit learning networks for regression," *Proceedings of International Conference on Tools for AI*, Washington D.C., pp. 350-358, Nov. 1990.
- [12] V. Cherkassky, and H. Lari-Najafi, "Constrained topological mapping for nonparametric regression analysis," *Neural Networks*, vol. 4, pp. 27-40, 1991.
- [13] M. T. Hagan, and M. B. Menhaj, "Training feedforward networks with the Marquardt algorithm," *IEEE Transactions on Neural Networks*, vol. 5, no. 6, pp. 989-993, Nov. 1994.

---

 저 자 소 개
 

---



이 하 준(학생회원)

2000년 경북대학교  
전자전기공학부 졸업  
2002년 한국과학기술원  
전자전산학과 석사졸업  
현재 한국과학기술원  
전자전산학과 박사과정.

<주관심분야 : 지능제어, 신경회로망, 진화 연산>



정 승 범(정회원)

1995년 한국과학기술원 전기 및  
전자공학과 졸업  
1998년 동 대학 석사 졸업  
현재 삼성전자 기술총괄  
소프트웨어센터 연구원.

<주관심분야 : 디지털신호처리>



박 철 훈(정회원)

1984년 서울대학교 전자공학과  
졸업  
1985년 캘리포니아 공과대학교  
석사  
1990년 캘리포니아 공과대학교  
박사

1991년~1992년 한국과학기술원 전기 및  
전자공학과 대우교수/연수 연구원.

현재 한국과학기술원 전자전산학과 교수.

미국 전기전자공학회 준석학회원

<주관심분야 : 신경회로망, 지능시스템>