

## 신경회로망을 이용한 인간공학적 접근

제 종 식 \*

## An Ergonomics Approach by Neural Network

Jong-Sik Je \*

### 요 약

인체 계측을 실시함에 있어서 현재 사용하는 방법은 피험자를 실측하여 실측치를 분석하는 방법을 사용하고 있다.

여기에 Neural Networks를 이용하여 기존 데이터를 분석하고 비교함으로써 향후 여러 가지 분석 자료로 활용할 수 있도록 함에 연구의 목적을 둠.

### Abstract

In measuring human body, the present method involves actual measuring of the subject and analysis of the measured value.

In this paper study set up the purpose of analysing & comparing the exiting data by means of Neural Networks, so that can be virtually used as various analysing materials hence forth.

---

\* 동부산대학 사무자동화과 조교수  
논문접수:98.8.28. 심사완료:98.10.28.

## I. 서론

인간의 신체부위는 시간이 지나면서 변화를 하게 된다. 이에 따른 변화의 추이를 Neural Networks를 이용하여 사전 데이터를 분석할 수 있다면 상당한 효율성이 있으리라 본다. 이에 따라 본 연구에서는 신경회로망을 이용한 데이터를 산출하여 향후 데이터를 활용할 수 있는 방법을 강구하고자 함에 그 목적을 둔다.

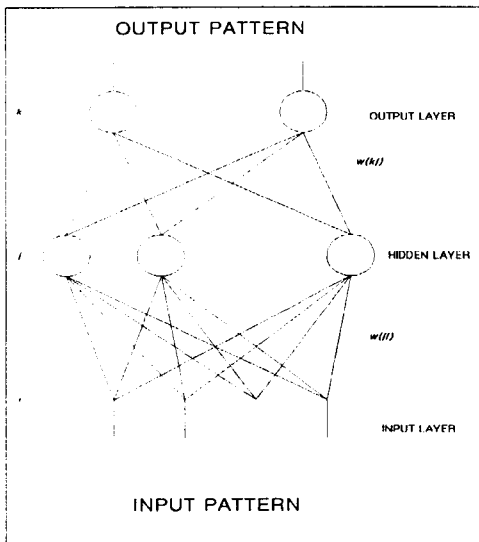


그림. 1. The Structure of Neural Network

## II. 신경회로망

### 2.1 신경회로망의 구조

신경회로망은 Fig. 1과 같이 입력층, 은닉층, 출력층으로 구성되고, 입력층과 출력층의 데이터는 최대값을 1로 정규화 하였다.

신경회로망의 Node는 사람의 뇌 신경세포에 해당하며 j층의 노드입력을  $net_j$  는 식 (1)과 같다.

$$net_j = \sum W_{ji} O_i \dots\dots\dots (1)$$

j층의 Node 출력  $O_j$ 는 식 (2)와 같다.

$$O_j = f(net_j) \dots\dots\dots (2)$$

여기서  $f$ 는 활성화 함수(Activation function)이다.

식(2)를 Sigmoid 활성화 함수로 나타내면 식 (3)과 같다.

$$O_j = f(net_j) = \frac{1}{1 + e^{-(net_j + \theta_j)/\theta_0}} \dots\dots\dots (3)$$

여기에서  $\theta_j$ 는 Bias이고  $\theta_0$ 의 값에 따라 Sigmoid 형태가 변화한다.

### 2.2 역전파 학습 알고리즘

역전파 학습 알고리즘은 모든 연결 Weight를 초기화 한 후 학습 패턴 집합에서 다음 학습 패턴을 선택하여 신경 회로망에 입력하여 실제 출력을 계산한다.

이러한 과정은 다음과 같다.

출력  $\{o_{pk}\}$ 는 목표값 또는 원하는 값  $\{t_{pk}\}$ 과 일치 하지 않으며, 이들간의 오차  $E_p$ 는 다음과 같다.

$$E_p = \frac{1}{2} \sum (t_{pk} - o_{pk})^2 \dots\dots\dots (4)$$

이고, 각 층의 Node에 대한 평균오차는 식 (5)와 같다.

$$E_p = \frac{1}{2} \sum_p \sum_k (t_{pk} - o_{pk})^2 \dots\dots\dots (5)$$

또한 Weight의 변화량  $\Delta w_{kj}$ ,  $\Delta w_{ji}$  는 다음과 같다.

$$\Delta W_{kj} = \eta \delta_k o_j = \eta (t_k - o_k) o_k (1 - o_k) o_j \dots\dots (6)$$

$$\Delta W_{ji} = \eta \delta_j o_i = \eta o_j (1 - o_j) \sum \delta_k w_{kj} o_i \dots\dots (7)$$

여기에서  $\eta$  는 학습율이다.

그리고 반복 학습회수가  $(t+1)$ 회 때의 Weight 값은 다음과 같다.

$$\Delta W_{kj}(t+1) = \alpha \Delta W_{kj}(t) + \Delta W_{kj} \dots\dots\dots (8)$$

$$\Delta W_{ji}(t+1) = \alpha \Delta W_{ji}(t) + \Delta W_{ji} \dots\dots\dots (9)$$

여기서  $\alpha$  는 모멘트량이다.

학습률  $\eta$  는 실험으로 적당한 값을 선택하여야 하며

일반적으로 큰 값은 빠른 수렴 속도를 얻을 수 있으나 진동할 수 있고 작은 경우는 느린 수렴 속도를 가진다.

### III. 데이터 변환

구현한 신경회로망을 기존의 데이터와 변형된 데이터를 Fig. 1과 같은 비선형 시스템으로 구성하였다.

구현하고자 하는 데이터 시뮬레이션을 위한 프로그램은 다음과 같다.

```

/* threshold is treated as weight of link
   from a virtual node whose output
   value is unity */
int rumelhart(int from_snum, int
to_snum)
{
    int i,j,k,m,n,p,offset, index;
    float out;

    /* long cnt: */
    char *err_file = "criter.dat";

    nsold = 0;
    cnt = 0;
    result = CONTNE;

    if(fplot10==1)
        if((fp3=fopen(err_file,"w"))==
        NULL) {
            perror("Cannot open error
            file");
            exit(0);
        }

    do {
        err_cur = 0.0;

```

```

/* for each pattern */
for(i=from_snum; i<to_snum;i++)
{

/* bottom_up calculation */
forward(i);

/* top_down error propagation */

/* output_level error */
for(m=0; m<nunit[nhlayer+1];
m++) {
    out= *(outptr[nhlayer+1]+m);

*(errptr[nhlayer+1]+m)=(target[i][
m]-out) * (1-out) * out;
}

/* hidden &input layer errors */
for(m= nhlayer+1; m>=1; m--) {
    for(n=0; n<nunit[m-1]+1; n++)
    {
*(errptr[m-1]+n)= 0.0; for (p=0;
p<nunit[m]; p++) {
        offset = (nunit[m-1]+1) * p + n;
        *(delw[m-1]+offset)=eta
        *(errptr[m]+p))*(*(outptr[m-1]+n))
        +alpha * (*(delw[m-1]+ offset));

*(errptr[m-1]+n)+=*(errptr[m]+p)*(*(
wtptr[m-1]+ offset));
}
*(errptr[m-1]+n)=*(errptr[m-1]+n)*(1-
*(outptr[m-1]+n))
*(outptr[m-1]+n);
}
}

/* weight changes */
for(m=1; m<nhlayer+2; m++) {
    for(n=0; n<nunit[m]; n++) {

```

```

    for(p=0; p<nunit[m-1]+1; p++) {
        offset= (nunit[m-1]+1) * n + p;
        *(wtptr[m-1]+offset) +=
*(delw[m-1]+offset);
    }
}
ep[i] = 0.0;
for(m=0; m<nunit[nhlayer+1];
m++) {
    ep[i] += fabs((target[i][m] -
*(outptr[nhlayer+1]+m)));
}
err_curr += ep[i] * ep[i];
}

/* normalized ysystem error */
err_curr = 0.5 * err_curr/ninput;

/* save errors in file to draw the
system error with plot10 */
if(fpplot10==1)

fprintf(fp3, "%d, %2.9f\n", cnt, err_curr)
;
cnt++;
/* check condition for terminating
learning */
result=introspective(from_snum, to_snum
);
/* printf("cnt=%d\n", cnt); */
} while(result == CONTNE);
/* update output with changed weights
*/
for(i=from_snum; i < to_snum; i++)
forward(i);
for(i=0; i < nhlayer+1; i++) {
index = 0;
for(j=0; j < nunit[i+1]; j++) {
printf("\n\nWeights between unit
%d of layer %d", j, i+1);
printf(" and units of layer %d\n

```

```

", i);
for(k=0; k<nunit[i]; k++)
printf(" %f ", *(wtptr[i] +
index++));
printf("\n Threshold of unit %d of
layer %d is %f",
j, i+1, *(wtptr[i]+index++));
}
}
for(i=0; i<ninput; i++)
for(j=0; j<noutattr; j++)
printf("\n\n sample %d output
%d = %f target %d = %f",
i, j, outpt[i][j], j, target[i][j]);

printf("\n\nTotal number of
iteration is %d", cnt);
printf("\n\nNormalized system error
is %f\n\n", err_curr);
return(result);
}

```

#### IV. 실험 및 고찰

본 실험에서는 입력층 데이터를 5비트로 변환하여 15개의 데이터를 사용하였으며, 은닉층 1개와 은닉층 노드 3개를 가진 신경회로망을 설정하였다. 반복회수는 2,000번~200,000번 까지 주었다.

또한 입력데이터와 출력데이터는 0과 1 사이의 정수로 변환하여 학습한 결과를 나타내었다.

여기에 사용된 1986년, 1992년, 1998년 데이터는 실측치를 그대로 사용한 것이다.

(키 [Stature] : 바닥에서 머리마루점까지의 수직거리의 평균값)

4.1 입력데이터 및 출력데이터

연도 나이	1986년	1992년	1998년	학습결과
6	117.0	120.4	120.4	123
7	121.5	123.4	124.8	126
8	126.5	128.9	128.4	130
9	131.3	133.0	134.8	134
10	135.8	137.7	139.7	138
11	140.4	142.7	145.4	142
12	147.0	149.4	151.9	149
13	153.4	155.8	158.8	155
14	160.1	162.3	165.7	162
15	164.2	165.3	168.4	166
16	166.5	168.5	170.1	168
17	167.0	170.5	170.7	169
18	167.9	169.9	171.6	169
19	168.3	170.4	171.6	170
20	168.6	169.6	170.8	170

표 1. Input and Output Data

V. 결론

상기의 학습 결과 1998년도 실측치와 비교한 데이터를 위의 Table 1.에 나타내었다. 입력데이터와 출력데이터는 은닉층 데이터에서 결정되는데, 주어진 입력데이터와 출력데이터의 분석으로 은닉층 노드수와 반복회수 결정에 따라 다소의 데이터 값의 변화를 볼 수 있다.

학습결과 실측치와의 관계에서 어떤 상관관계는 파악되지 않는다. 이는 여러 가지 이유가 있겠으나 인체계측치는 매년 변화를 하고 있는 반면 학습모델은 일정한 범위내에서 학습을 하기 때문에 계속적인 변화에 대처하기 위한 알고리즘 변화 및 학습 방법 등이 변화하여야 하겠다.

향후 이와 같은 방법으로 데이터를 분석하여 어떤 목적에 이용하고자 할 경우 데이터에 최대한 접근할 수

있는 방법을 강구하여 활용하는 것이 과제이다.

참고문헌

- [1] Duda, R.O. and P.E. Hart, 1973. Pattern Classification and Scene Analysis, Wiley, New York.
- [2] Giles, C.L. and T. Maxwell, 1987. Learning, invariance, and generalization in high-order neural networks, Applied Optics, Vol. 26, pp. 4972-4978.
- [3] Kohonen, T., 1982. Self-organized formatting of topologically correct maps, Biological Cybernetics, Vol. 43, pp.59-69
- [4] Lippman, R.P., 1987. An introduction to computing with neural nets, IEEE ASSP Magazine, Vol. 4, pp. 4-22.
- [5] Minsky, M. and S. Papert, 1969. Perception: An Introduction to Computational Geometry, MIT Press, Cambridge, MA.
- [6] Rumelhart, D.E., G.E. Hinton, and R.J. Williams, 1986. Learning internal representations by error propagation. In D.E. Rumelhart and J.L. McClelland (Eds.), Parallel Distributed Processing: Explorations in the Microstructures of Cognition, Vol. 1: Foundations, pp. 318-362, MIT Press, Cambridge, MA.

저자소개

제종식  
1987. - 1991 한국산업경제 정책 연구원  
현재 동부산대학 사무자동화과 조교수