

<AI : 과제 #1 Perceptron 구현>



2019학년도 2학기 인공지능

컴퓨터과학부

2015920056

최현호

#1 과제내용

1. Input 차원이 n인 1-layer perceptron 을 구현한다.
2. Weight 값(w[n]와 theta)들을 random 값으로 초기화한다.
3. 다음을 무한 반복한다.
 - AND gate의 입력과 같은 값 $x[n]$ 을 차례로 입력하여 output을 구한다.
 - 네 가지의 input에 대하여 output이 틀린 개수를 구한다.
 - 각 input 별 output 이 모두 맞으면 종료.
 - 임의의 값을 입력 받아 각 weight에 대입한다.

* 여기서 모든 weight 의 값은 강의자료에 있는 것과 다르게 해야 한다.
* 매 루프마다 $w[n]$ 와 theta 중에서 하나만 입력해도 된다.

C, C++언어로 구현

결과물:

- source (주석 포함)
- 결과 보고서 (어떤 값을 입력해야 무한 loop에서 빨리 나올 수 있는지?)

마감: 9월9일 14:00

#2-1 코드설명

dimension(차원)은 2로 입력(가정)

초기 θ , weight ($0 < \theta$, $weight \leq 10$, 소수점 첫 번째 자리까지)

(θ 가 0보다 작거나 같으면 루프를 빠져나올 수 없으므로 양수로 정함)

$net = \sum_{i=0}^n x[i]w[i]$

$net \leq \theta : output=0$

$net > \theta : output=1$

output를 And_Gate Truth_Table 과 비교하여 무한루프 탈출

#2-2 코드

```
/*
```

```
AI assignment #1
```

```
2015920056 computer_science 최현호
```

which has n-dimension.

Input is random weight & theta.

Output is same as and gate.

assumption : 2-dimension.

```
*/  
#include <iostream>  
#include <cstdlib>  
#include <set>  
#include <math.h>  
#include <ctime>
```

```
#define MAX 100000000  
#define D_MAX 10 // dimension max
```

using namespace std;

```
float Dimension; // Dimension  
float x[MAX]; // x  
float w[MAX]; // weight  
float output; // output  
float t[MAX]; // Truth_Table  
float Theta=0; // Theta  
int fail = 0; // num of fail  
int cnt = 0; // Num of Try  
float net; // Sum
```

```
int main()  
{  
    // comment  
    cout << "AI assignment #1\n";  
    cout << "2015920056 computer_science 최현호\n\n";  
  
    cout << "Making 1 - layer perceptron\n";  
    cout << "which has n - dimension.\n\n";  
  
    cout << "Input is random weight & theta.\n";  
    cout << "Output is same as and gate.\n\n";  
  
    cout << "assumption : 2 - dimension.\n\n";  
    cout << "#####\n";  
  
    cout << "Choose Dimension : ";
```

```

cin >> Dimension;

srand((unsigned int)time(0)); // seed
Theta = (rand() % 100)/10.0; // 0 <= theta < 10"
cout << "Theta : " << Theta << "\n\n";
int k = pow(2, Dimension)-1;
t[k] = 1; // And_Gate Truth_Table

// loop
do {
    fail = 0; // initialize Num of Fail
    cnt++; // initialize Num of Try
    cout << "##### " << cnt << "th try!!" << "\n\n";
    // Choose Weight
    if (cnt == 1) { // firtt case
        for (int i = 0; i < Dimension; i++) {
            w[i] = (rand() % 100)/10.0 + 1; // 0 <= weight < 10"
            cout << "Weight " << i + 1 << " : " << w[i] << endl;
        }
    }
    else { // other case
        for (int i = 0; i < Dimension; i++) {
            cout << "Enter the " << i + 1 << "th weigth :";
            cin >> w[i];
        }
    }
    cout << "\n" << endl;

    // test 2^dimension case
    for (int i = 0; i < pow(2,Dimension); i++) {
        // x[n] with bit count
        for (int j = 0; j < Dimension; j++) {
            x[j] = bitset<D_MAX>(i)[j];
            cout << "x" << j << " : " << x[j] << endl;
            cout << "w" << j << " : " << w[j] << endl;

        }
        // sum(net)
        //net = x1*w1 + x2*w2;
        //if(net<Theta) return 0;
        //else return 1;
        net = 0;
        for (int j = 0; j < Dimension; j++) {
            net += x[j] * w[j];
        }
    }
}

```

```

    }

    cout << "net : " << net << "\n";
    // calculate the output
    if (net <= Theta)
        output = 0;
    else
        output = 1;
    cout << "output : " << output << "\n";
    cout << "Truth_table : " << t[i] << "\n";

    // compare with Truth_Table
    if (output != t[i]) {
        fail++;
    }
    cout << "\n";
}

if (fail != 0) {
    cout << "Num of Fail : " << fail << "\n\n";
    cout << "Try again... " << endl;
}
} while (fail!=0);

cout << "\n#####\nSuccess\n\n";
cout << "you tried : " << cnt << "!!!\n\n";
}

```

#2-3 결과

```
AI assignment #1  
2015920056 computer_science 최현호  
  
Making 1 - layer perceptron  
which has n - dimension.  
  
Input is random weight & theta.  
Output is same as and gate.  
  
assumption : 2 - dimension.
```

```
x0 : 1  
w0 : 6.6  
x1 : 1  
w1 : 3.5  
net : 10.1  
output : 1  
Truth_table : 1  
  
Num of Fail : 1  
  
Try again...
```

```
Choose Dimension : 2  
Theta : 4.5
```

```
##### 1th try!!!
```

```
Weight 1 : 6.6  
Weight 2 : 3.5
```

```
x0 : 0  
w0 : 6.6  
x1 : 0  
w1 : 3.5  
net : 0  
output : 0  
Truth_table : 0
```

```
x0 : 1  
w0 : 6.6  
x1 : 0  
w1 : 3.5  
net : 6.6  
output : 1  
Truth_table : 0
```

```
x0 : 0  
w0 : 6.6  
x1 : 1  
w1 : 3.5  
net : 3.5  
output : 0  
Truth_table : 0
```

```
##### 2th try!!!
```

```
Enter the 1th weight :3.0  
Enter the 2th weight :3.3
```

```
x0 : 0  
w0 : 3  
x1 : 0  
w1 : 3.3  
net : 0  
output : 0  
Truth_table : 0
```

```
x0 : 1  
w0 : 3  
x1 : 0  
w1 : 3.3  
net : 3  
output : 0  
Truth_table : 0
```

```
x0 : 0  
w0 : 3  
x1 : 1  
w1 : 3.3  
net : 3.3  
output : 0  
Truth_table : 0
```

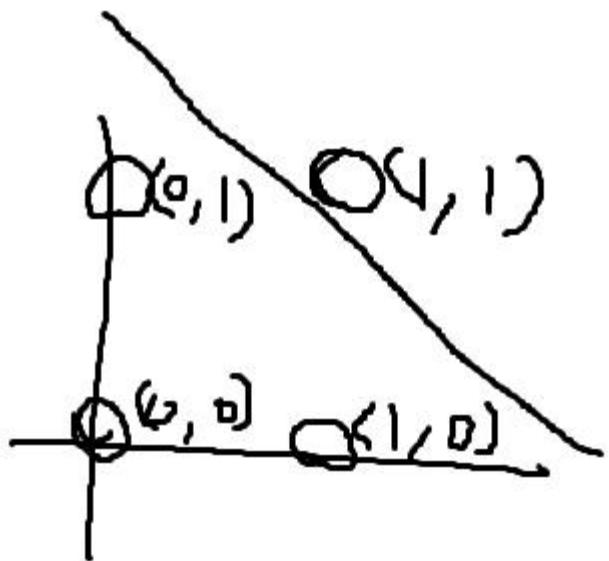
```
x0 : 1  
w0 : 3  
x1 : 1  
w1 : 3.3  
net : 6.3  
output : 1  
Truth_table : 1
```

```
#####
```

```
Success
```

```
you tried : 2!!!
```

#3-1 무한루프 탈출 조건찾기



$(x_1, x_2) = (0,0) (0,1) (1,0)$ 일때 $\text{net} = w_1x_1 + w_2x_2 \leq \theta$

$(x_1, x_2) = (1,1)$ 일때 $\text{net} = w_1x_1 + w_2x_2 > \theta$

이어야 하므로 $w_1 \leq \theta$, $w_2 \leq \theta$ 이고 $w_1 + w_2 > \theta$ 인

weight들을 입력해주면 탈출하였습니다.

#3-2 빨리 탈출하는 조건

1번 조건 : $w_1 \leq \theta$, $w_2 \leq \theta$

2번 조건 : $w_1 + w_2 > \theta$

이라고 했을 때

w_1 과 w_2 의 값이 크게 차이가나서 $w_1 \leq \theta$, $w_2 > \theta$ 라면 1번 조건을 만족하지 못하므로 w_1 과 w_2 는 같거나 비슷한 값일 때 탈출 가능하였고 같은 값일 때 $w_1 \leq \theta$, $w_2 > \theta$ 인 경우가 없기에

$w_1 = w_2$ 일 때 빨리 탈출할 수 있었고

따라서 $w_1 \leq \theta$, $2w_1 > \theta$ 입니다. --> ⑦ 성질

무한루프를 탈출하기 위해서는 4가지 (x_1, x_2) 를 입력하였을 때 output이 And_Gate Truth_Table과 일치하여야 합니다.

$(x_1, x_2) = (0,0) (0,1) (1,0)$ 일때 $\text{net} = w_1x_1 + w_2x_2 \leq \theta$

$(x_1, x_2) = (1,1)$ 일때 $\text{net} = w_1x_1 + w_2x_2 > \theta$

이므로

$$(0,0) \quad 0*w1+0*w2=0 \leq \theta$$

$$(0,1) \quad 0*w1+1*w2=w2 \leq \theta$$

$$(1,0) \quad 1*w1+0*w2=w1 \leq \theta$$

$$(1,1) \quad 1*w1+1*w2=w1+w2 > \theta$$

(0,0)에서 $0 \leq \theta$ 은 항상 성립합니다. (θ 는 양수이므로)

$w1 > \theta$ 이고 $w2 > \theta$ 이며 $w1+w2 \leq \theta$ 일수없으니

매번 케이스마다 틀린 개수는 1개 또는 2개입니다. --> ⑤ 성질

⑦ 성질에 따라 ⑤ 성질을 해석해보면

틀린 개수 1개일 때 : $2*w1 > \theta$ 을 만족하지 못 한다는 것이므로

$w1$ 값을 늘려주어야 합니다.

틀린 개수 2개일 때 : $w1 \leq \theta$ 을 만족하지 못 한다는 것이므로

$w1$ 값을 줄여주어야 합니다.

결론 :

$w1$ 과 $w2$ 의 값을 같게 하고

틀린 개수가 1개일 때는 $w1$ 과 $w2$ 의 값을 늘려주고

틀린 개수가 2개일 때는 $w1$ 과 $w2$ 의 값을 줄여주어야 합니다.