

MATLAB Problem of the Fourth Practice Day

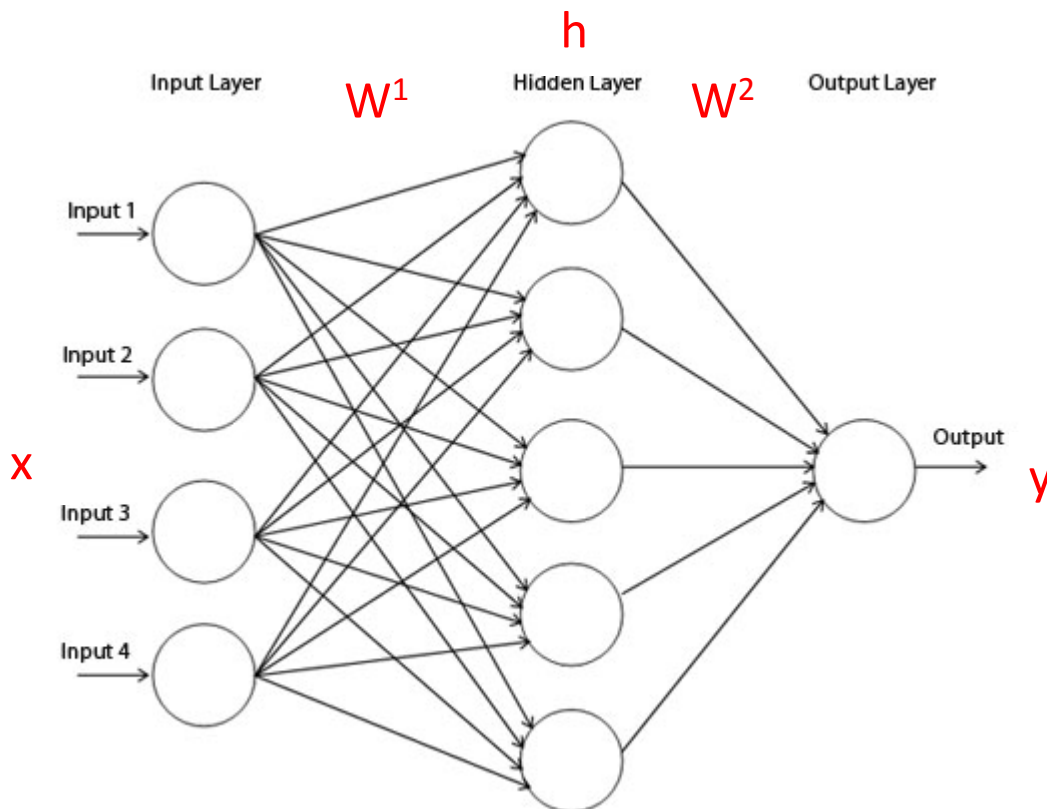
Multi-Layer Perceptrons

이번 시간에는 기초적인 Neural Network 인 Multi-Layer Perceptron (MLP)을 구현하는 것을 목표로 한다.

MLP 는 하나 이상의 적은 수의 hidden layers 를 가진 neural network 이다. 이때, 각 hidden layer 는 sigmoid function 등의 nonlinear transform 을 수행하게 된다. MLP 는 기본적으로 target value 가 continuous 한 regression 에서 쓰일 수 있고, discrete 또는 categorical 한 경우 여러 output node 들의 soft-max function 을 통해 classification 에서 쓰일 수 있다. 흔히 Deep Learning 이라 불리는 neural network 계열 모델들도 기본적으로 이러한 MLP 를 근간으로 하여 구성된다.

이번 시간에는 MLP 의 weight matrix 를 학습하는 알고리즘을 직접 구현해보도록 한다. Iteration (10 회)에 따라서 output y 값이 어떻게 변하는지 5 회 반복 수행 후, 그 결과를 메일 본문에 기재하여 source code 와 함께 제출하도록 한다.

MLP 의 구조는 다음과 같다.



$$h = \sigma(W_1 x),$$
$$y = \sigma(W_2 h)$$

MLP 의 학습을 위한 cost function E 와 관련 partial derivatives 는 다음과 같다.

$$E = -(t \log y + (1 - t) \log(1 - y))$$
$$\frac{\partial E}{\partial y} = -\frac{t - y}{y(1 - y)}$$
$$\frac{\partial y}{\partial w_j^2} = h_j y(1 - y),$$
$$\frac{\partial y}{\partial h_j} = w_j y(1 - y),$$
$$\frac{\partial h_j}{\partial w_{ij}^1} = x_i h_j(1 - h_j).$$

MLP 의 학습을 구현하기 위한 코드 블록은 아래와 같다.

```
INPUT_SIZE = 784;
HIDDEN_SIZE = 300;
OUTPUT_SIZE = 1;

% dummy single data
x = ones(1, INPUT_SIZE); % row input vector
t = 0; % true output

% random weight
W1 = normrnd(0, 1, INPUT_SIZE, HIDDEN_SIZE); % weight matrix for first
layer
W2 = normrnd(0, 1, HIDDEN_SIZE, OUTPUT_SIZE); % weight matrix for second
layer
for iter=1:10

    % forward
    h = sigmoid(x*W1); % hidden vector is also row vector
    y = sigmoid(h*W2); % output is scalar

    fprintf('y:%f\n', y);

    % back propagation computing gradient descent
    W2 = W2 + r1*(t - y)*h';
    % W1 = W1 + (...); % Fill the update equation in the parentheses
end
```

Matlab Code: Learning Multi-Layer Perceptron (To be implemented)

시간이 남을 경우 다음 사항 중 일부를 구현하고, 구현한 내용을 메일 본문에 기재하도록 한다.

Soft-max function 을 통한 (nonbinary) n-ary classification 모델 구현

MNIST dataset 을 통한 학습 (학습 데이터 1000 개)

Mini-batch 방식 학습

Test dataset 에 대한 성능 보고 (테스트 데이터 100 개)

Hidden layer 수를 임의로 조정