인공신경망 II
Back Propagation

한국공학대학교
전자공학부
채승호 교수

오차 역전파 알고리즘

- 다계층 인공 신경망의 학습
  - 예측값과 실제 출력값의 차이를 최소화하는 매개변수 탐색
    - 시스템 복잡도가 높아 일반적인 해석해 구하는 것이 매우 어려움
    - 비용함수를 설계하고 경사하강법을 적용하여 매개변수 탐색
  - 오차 역전파 알고리즘 (Error Backpropagation Algorithm)
    - 경사하강법에 기반을 둔 매개변수 업데이트 알고리즘
    - 다계층 인공 신경망 학습의 기본 알고리즘

오차 역전파 알고리즘

- 훈련데이터 구성

데이터 번호	입력 $x_0, \cdots, x_m, \cdots, x_{M-1}$	출력 $y_0, \cdots, y_q, \cdots, y_{Q-1}$
0	$x_{00}, \cdots, x_{m0}, \cdots, x_{(M-1)0}$	$y_{00}, \cdots, y_{q0}, \cdots, y_{(Q-1)0}$
$\vdots$	$\vdots$	$\vdots$
n	$x_{0n}, \cdots, x_{mn}, \cdots, x_{(M-1)n}$	$y_{0n}, \cdots, y_{qn}, \cdots, y_{(Q-1)n}$
$\vdots$	$\vdots$	$\vdots$
$N-1$	$x_{0(N-1)}, \cdots, x_{m(N-1)}, \cdots, x_{(M-1)(N-1)}$	$y_{0(N-1)}, \cdots, y_{q(N-1)}, \cdots, y_{(Q-1)(N-1)}$

- N: 훈련 데이터의 개수, $n = 0, 1, \cdots, N-1$
- M: 입력 노드의 개수 (더미노드 제외), $m = 0, 1, \cdots, M-1, M$
- L: 은닉층 노드의 개수 (더미노드 제외), $l = 0, 1, \cdots, L-1, L$
- Q: 출력 클래스의 개수, $\mathrm{q} = 0, 1, \cdots, Q-1$

오차 역전파 알고리즘

- 비용함수
  - 평균제곱오차(MSE) 또는 평균교차엔트로피(CEE) 등 사용 가능
  - 본 강의자료는 평균제곱오차 기준으로 설명
  - 전체 데이터 (N개)에 대한 평균제곱오차

$$\epsilon_{MSE} = \frac{1}{N}\sum_{n=0}^{N-1}\sum_{q=0}^{Q-1}\left(\hat{y}_{qn} - y_{qn}\right)^2 = \frac{1}{N}\sum_{n=0}^{N-1}\epsilon_{MSE}(n)$$

Q: 출력 클래스의 개수 / $\epsilon_{MSE}(n)$: n번째 데이터에 대한 MSE

  - n번째 데이터에 대한 평균제곱오차
    - (주의!) 오차 역전파 알고리즘은 데이터 단위로 동작

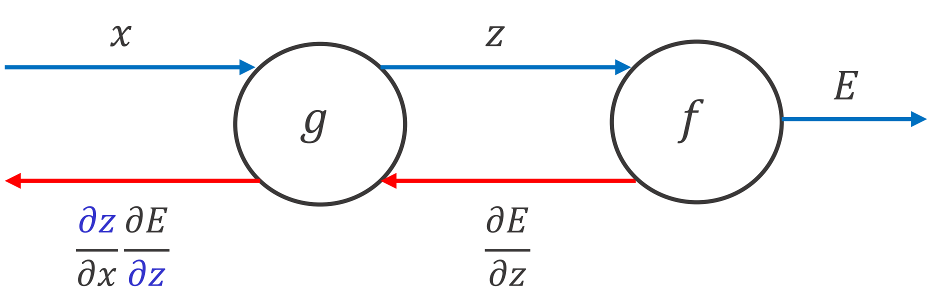
$$\epsilon_{MSE}(n) = \sum_{q=0}^{Q-1}\left(\hat{y}_{qn} - y_{qn}\right)^2$$

오차 역전파 알고리즘

- 오차역전파 기법의 원리
  - 연쇄법칙(Chain rule)을 활용한 효율적인 gradient 계산
    - 미분의 연쇄법칙 $\left(E = f(z), z = g(x)\right)$

$$\frac{dE}{dx} = \frac{d}{dx}f\left(g(x)\right) = f'\left(g(x)\right)g'(x) = \frac{dE}{dz}\frac{dz}{dx}$$

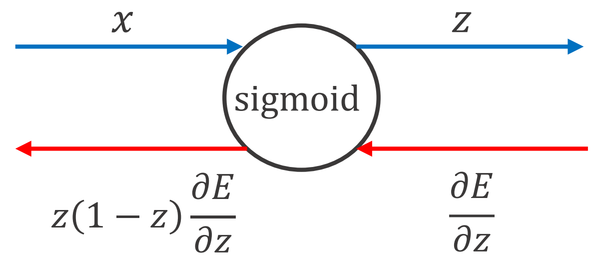
→ E에 대한 x의 영향



오차 역전파 알고리즘

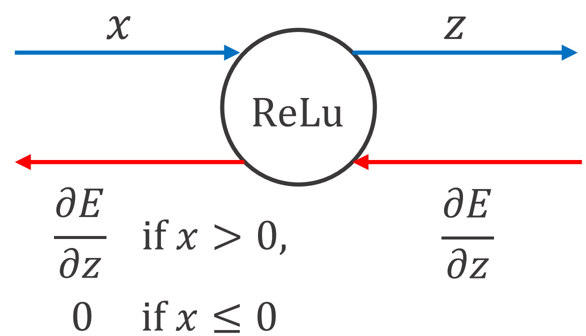
- 활성화 함수의 편미분
  - 시그모이드(sigmoid) 함수

$$z = \frac{1}{1+e^{-x}} \;\Rightarrow\; \frac{\partial z}{\partial x} = \frac{e^{-x}}{(1+e^{-x})^2} = z(1-z)$$



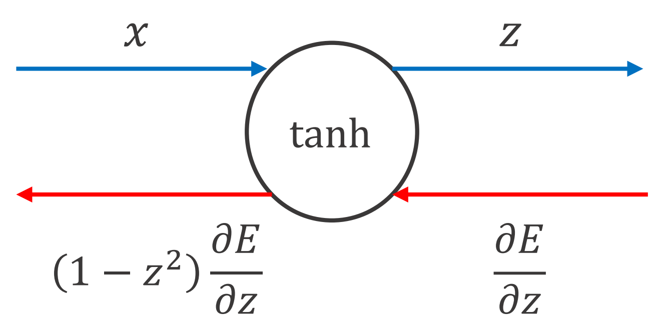
  - Relu 함수

$$z = \begin{cases} x & ,x > 0 \\ 0 & ,x \le 0 \end{cases} \;\Rightarrow\; \frac{\partial z}{\partial x} = \begin{cases} 1 & ,x > 0 \\ 0 & ,x \le 0 \end{cases}$$



  - 하이퍼볼릭탄젠트(tanh) 함수

$$z = \tanh(x) \;\Rightarrow\; \frac{\partial z}{\partial x} = 1 - z^2$$

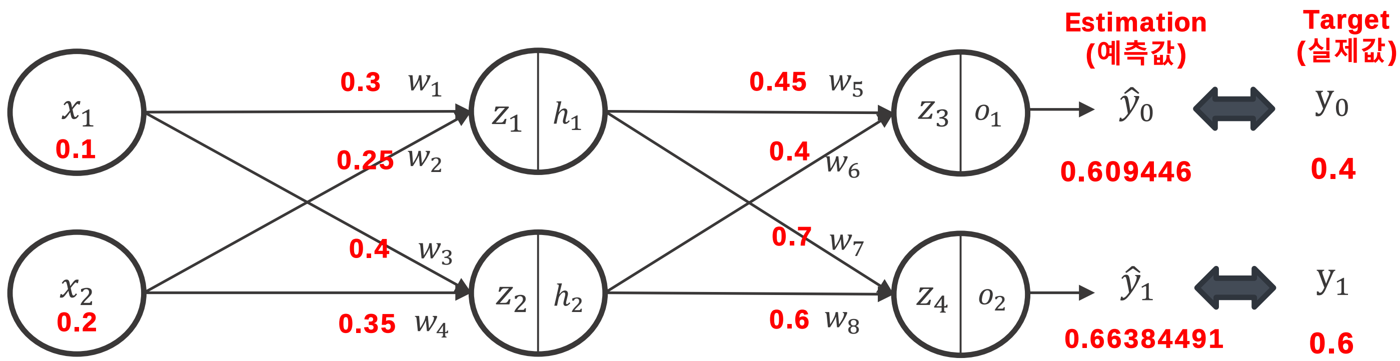


오차 역전파 알고리즘

- 예측값 계산 과정: Forward-propagation (순전파)
- 가중치 업데이트 과정: Back-propagation(역전파)

■ 오차역전파 기법 가중치 업데이트 원리

- (1) 순전파(Forward Propagation)



$$z_1 = w_1x_1 + w_2x_2 = (0.3)(0.1) + (0.25)(0.2) = 0.08$$

$$z_2 = w_3x_1 + w_4x_2 = (0.4)(0.1) + (0.35)(0.2) = 0.11$$

$$h_1 = \text{sigmoid}(z_1) = 0.51998934$$

$$h_2 = \text{sigmoid}(z_2) = 0.52747230$$

$$z_3 = w_5h_1 + w_6h_2 = (0.45)h_1 + (0.4)h_2 = 0.44498412$$

$$z_4 = w_7h_1 + w_8h_2 = (0.7)h_1 + (0.6)h_2 = 0.68047592$$

$$o_1 = \text{sigmoid}(z_3) = 0.609446$$

$$o_2 = \text{sigmoid}(z_4) = 0.66384491$$

$$E_{o1} = (\text{target}_{01} - \text{output}_{o1})^2 = 0.04386763$$

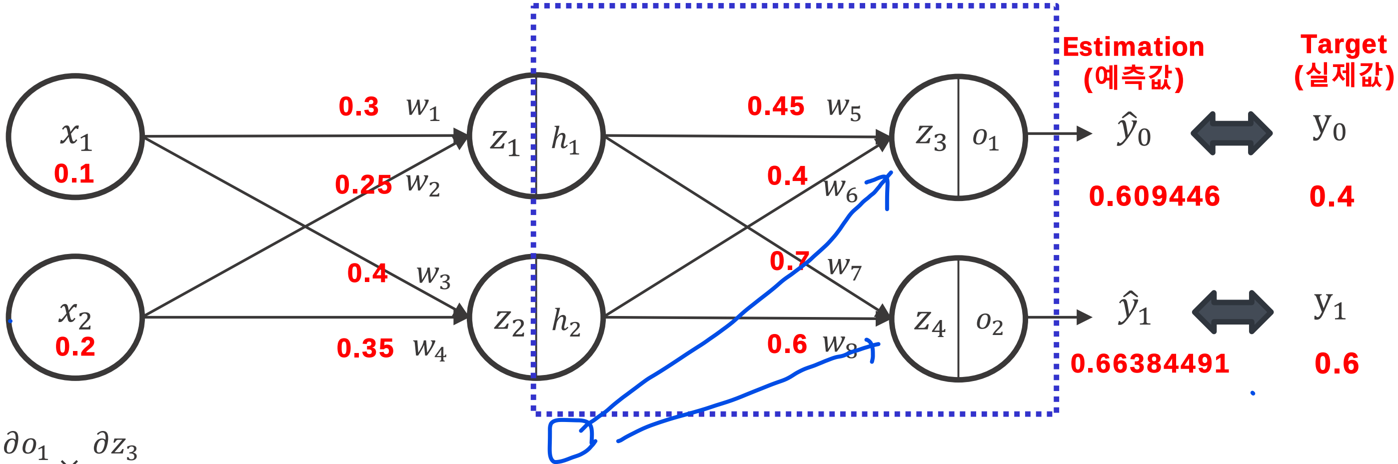
$$E_{o2} = (\text{target}_{02} - \text{output}_{o2})^2 = 0.00407617$$

$$E_{total} = E_{o1} + E_{o2} = 0.04794380$$

오차 역전파 알고리즘

- 예측값 계산 과정: Forward-propagation (순전파)
- 가중치 업데이트 과정: Back-propagation(역전파)

- (2)역전파 1단계



$$\frac{\partial E_{total}}{\partial w_5} = \frac{\partial E_{total}}{\partial o_1} \times \frac{\partial o_1}{\partial z_3} \times \frac{\partial z_3}{\partial w_5}$$

$$E_{total} = (\text{target}_{01} - \text{output}_{o1})^2 + (\text{target}_{02} - \text{output}_{o2})^2$$

$$\frac{\partial E_{total}}{\partial o_1} = 2 \times (\text{target}_{01} - \text{output}_{o1})^{2-1} \times (-1) + 0 = -2(\text{target}_{01} - \text{output}_{o1}) = -2(0.4 - 0.609446) = 0.418892$$

output_{o1}만 o_1의 함수

$$\frac{\partial o_1}{\partial z_3} = o_1 \times (1 - o_1) = 0.609446(1 - 0.609446) = 0.23802157$$

(sigmoid 함수 미분)

$$\frac{\partial z_3}{\partial w_5} = \frac{\partial}{\partial w_5}(h_1 w_5 + h_2 w_6) = h_1 = 0.51998934$$

$$\therefore \frac{\partial E_{total}}{\partial w_5} = 0.418892 \times 0.23802157 \times 0.51998934 = 0.05184571$$

$$w_5 \leftarrow w_5 - \alpha \frac{\partial E_{total}}{\partial w_5} = 0.45 - 0.5 \times 0.05184571 = 0.42407715$$

(동일한 방법으로)

$$\frac{\partial E_{total}}{\partial w_6} = \frac{\partial E_{total}}{\partial o_1} \times \frac{\partial o_1}{\partial z_3} \times \frac{\partial z_3}{\partial w_6} \rightarrow w_6 = 0.3737041$$

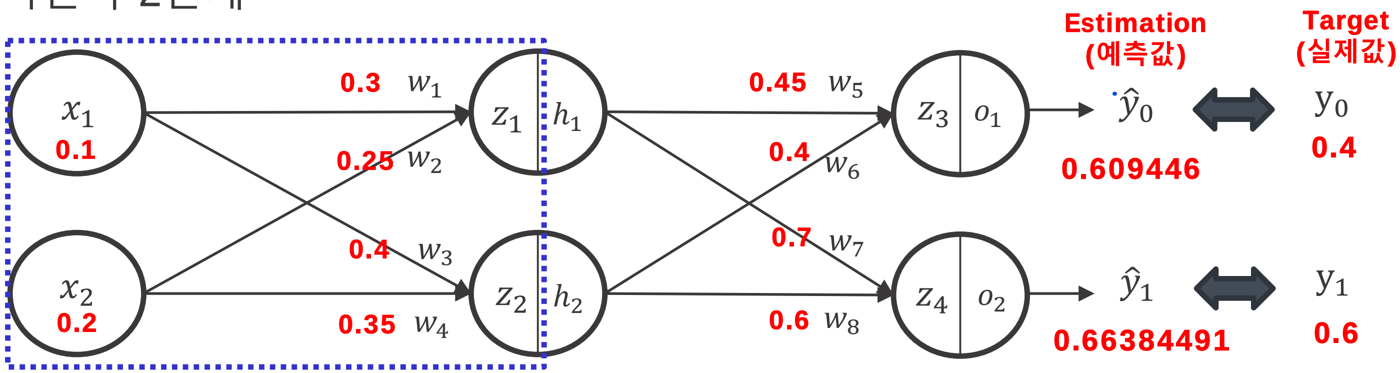
$$\frac{\partial E_{total}}{\partial w_7} = \frac{\partial E_{total}}{\partial o_1} \times \frac{\partial o_2}{\partial z_4} \times \frac{\partial z_4}{\partial w_7} \rightarrow w_7 = 0.69259156$$

$$\frac{\partial E_{total}}{\partial w_8} = \frac{\partial E_{total}}{\partial o_2} \times \frac{\partial o_2}{\partial z_4} \times \frac{\partial z_4}{\partial w_8} \rightarrow w_8 = 0.59248495$$

오차 역전파 알고리즘

- 예측값 계산 과정: Forward-propagation (순전파)
- 가중치 업데이트 과정: Back-propagation(역전파)

- (2)역전파 2단계



$$\frac{\partial E_{total}}{\partial w_1} = \frac{\partial E_{total}}{\partial h_1} \times \frac{\partial h_1}{\partial z_1} \times \frac{\partial z_1}{\partial w_1}$$

$$\frac{\partial E_{total}}{\partial h_1} = \frac{\partial E_{01}}{\partial h_1} + \frac{\partial E_{02}}{\partial h_1}$$

$$f = MSE$$
$$z = g(x) = sigmoid$$
$$y = h(x) = linear\ sum$$

$$f\Big(g\big(h(x)\big)\Big)' = \frac{df}{dz}\frac{dz}{dy}\frac{dy}{dx}$$

$$\frac{\partial E_{01}}{\partial h_1} = \frac{\partial E_{01}}{\partial z_3} \times \frac{\partial z_3}{\partial h_1} = \frac{\partial E_{01}}{\partial o_1} \times \frac{\partial o_1}{\partial z_3} \times \frac{\partial z_3}{\partial h_1} = -2(\text{target}_{01} - \text{output}_{01}) \times o_1 \times (1 - o_1) \times w_5 = 0.418892 \times 0.23802157 \times (0.45) = 0.0448674$$

$$\frac{\partial E_{02}}{\partial h_1} = \frac{\partial E_{02}}{\partial z_4} \times \frac{\partial z_4}{\partial h_1} = \frac{\partial E_{02}}{\partial o_2} \times \frac{\partial o_2}{\partial z_4} \times \frac{\partial z_4}{\partial h_1} = -2(\text{target}_{02} - \text{output}_{02}) \times o_2 \times (1 - o_2) \times w_7 = 0.12768982 \times 0.22231548 \times (0.7) = 0.0198712$$

$$\frac{\partial E_{total}}{\partial h_1} = 0.0448674 + 0.0198712 = 0.0647386$$

$$\frac{\partial h_1}{\partial z_1} = h_1 \times (1 - h_1) = 0.51998934(1 - 0.51998934) = 0.24960043$$ (sigmoid 미분)

$$\frac{\partial z_1}{\partial w_1} = x_1 = 0.1$$

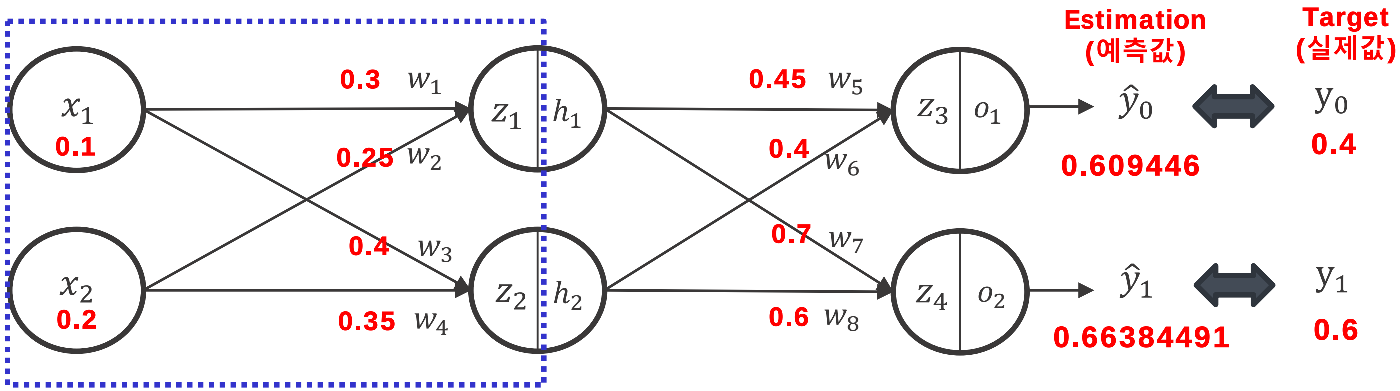
$$\frac{\partial E_{total}}{\partial w_1} = 0.0647386 \times 0.24960043 \times 0.1 = 0.00161588$$

$$w_1 \leftarrow w_1 - \alpha \frac{\partial E_{total}}{\partial w_1} = 0.3 - 0.5 \times 0.00161588 = 0.29919206$$

오차 역전파 알고리즘

- 예측값 계산 과정: Forward-propagation (순전파)
- 가중치 업데이트 과정: Back-propagation(역전파)

- (2)역전파 2단계



$$f = MSE$$
$$z = g(x) = sigmoid$$
$$y = h(x) = linear\ sum$$

$$f\left(g\left(h(x)\right)\right)' = \frac{df}{dz}\frac{dz}{dy}\frac{dy}{dx}$$

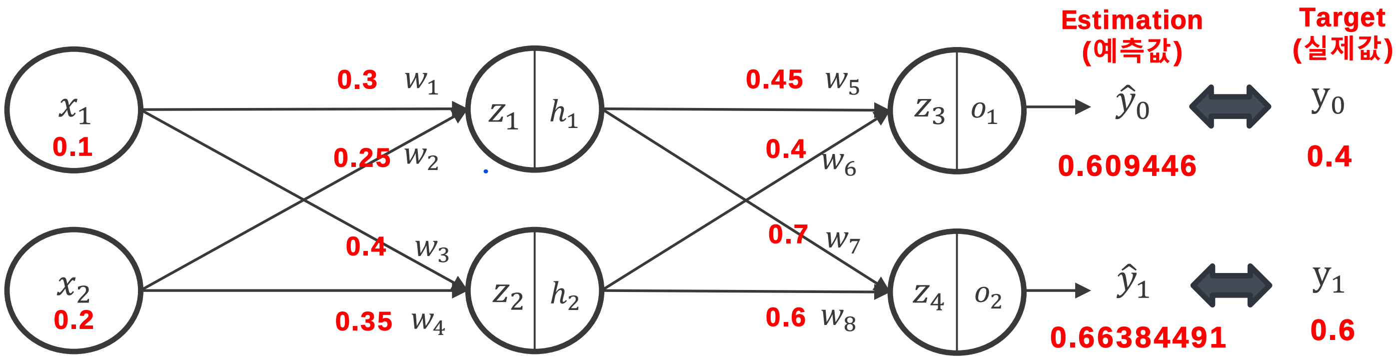
(동일한 방법으로)

$$\frac{\partial E_{total}}{\partial w_2} = \frac{\partial E_{total}}{\partial h_1} \times \frac{\partial h_1}{\partial z_1} \times \frac{\partial z_1}{\partial w_2} \to w_2 = 0.24983841$$

$$\frac{\partial E_{total}}{\partial w_3} = \frac{\partial E_{total}}{\partial h_2} \times \frac{\partial h_2}{\partial z_2} \times \frac{\partial z_2}{\partial w_3} \to w_3 = 0.39928971$$

$$\frac{\partial E_{total}}{\partial w_4} = \frac{\partial E_{total}}{\partial h_2} \times \frac{\partial h_2}{\partial z_2} \times \frac{\partial z_2}{\partial w_4} \to w_4 = 0.34857941$$

오차 역전파 알고리즘

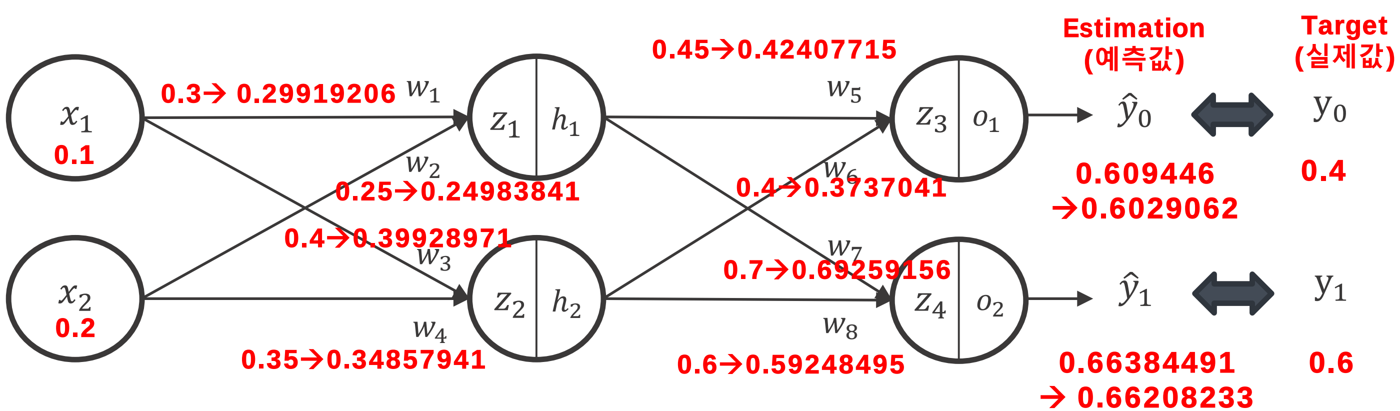


weight	previous	updated
w_1	0.3	0.29919206
w_2	0.25	0.24983841
w_3	0.4	0.39928971
w_4	0.35	0.34857941
w_5	0.45	0.42407715
w_6	0.4	0.3737041
w_7	0.7	0.69259156
w_8	0.6	0.59248495

오차 역전파 알고리즘

weight	previous	updated
w_1	0.3	0.29919206
w_2	0.25	0.24983841
w_3	0.4	0.39928971
w_4	0.35	0.34857941
w_5	0.45	0.42407715
w_6	0.4	0.3737041
w_7	0.7	0.69259156
w_8	0.6	0.59248495

- 결과 확인



$$z_1 = w_1x_1 + w_2x_2 = (0.29919206)(0.1) + (0.24983841)(0.2) = 0.07988689$$

$$z_2 = w_3x_1 + w_4x_2 = (0.39928971)(0.1) + (0.34857941)(0.2) = 0.10964485$$

$$h_1 = \text{sigmoid}(z_1) = 0.5199611$$

$$h_2 = \text{sigmoid}(z_2) = 0.52738378$$

$$z_3 = w_5h_1 + w_6h_2 = (0.42407715)h_1 + (0.3737041)h_2 = 0.4175891$$

$$z_4 = w_7h_1 + w_8h_2 = (0.69259156)h_1 + (0.59248495)h_2 = 0.67258762$$

$$o_1 = \text{sigmoid}(z_3) = 0.6029062$$

$$o_2 = \text{sigmoid}(z_4) = 0.66208233$$

$$E_{o1} = (\text{target}_{01} - \text{output}_{o1})^2 = 0.041171$$

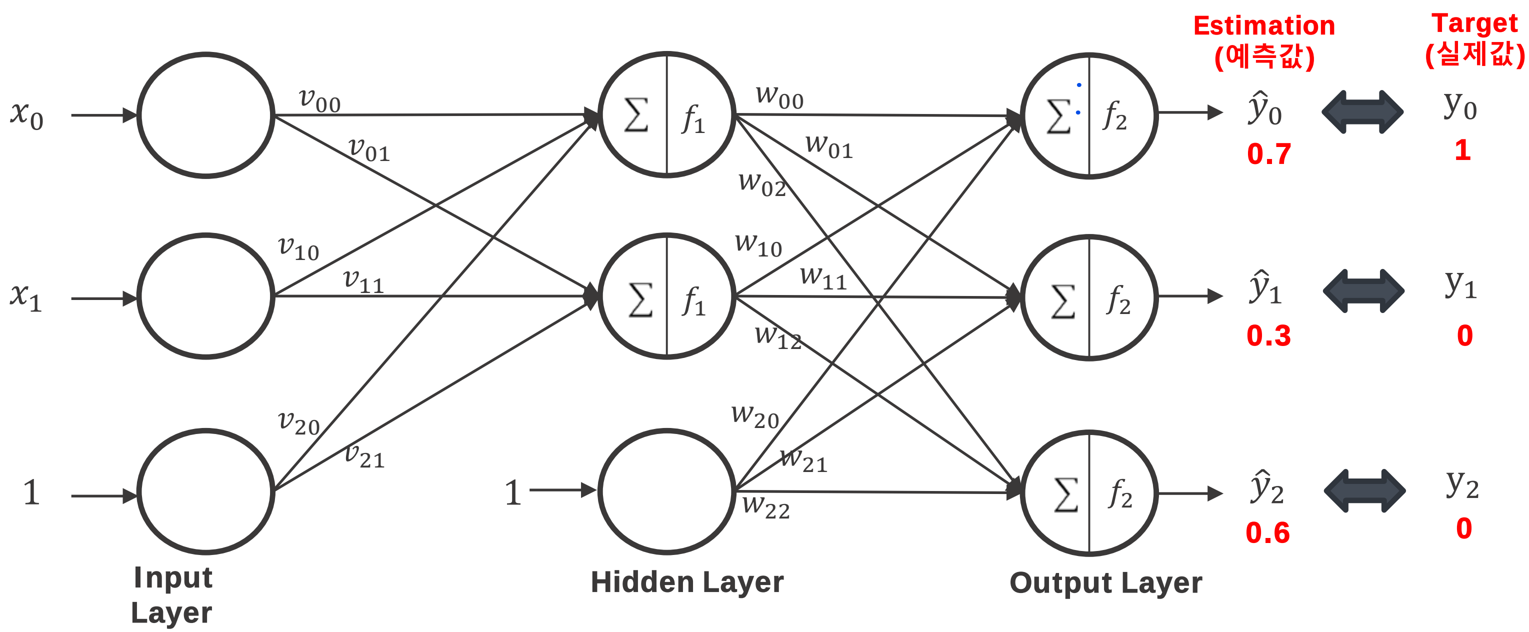
$$E_{o2} = (\text{target}_{02} - \text{output}_{o2})^2 = 0.0058532$$

$$E_{total} = E_{o1} + E_{o2} = 0.0470242$$

v.s.) $E_{total} = E_{o1} + E_{o2} = 0.04794380$

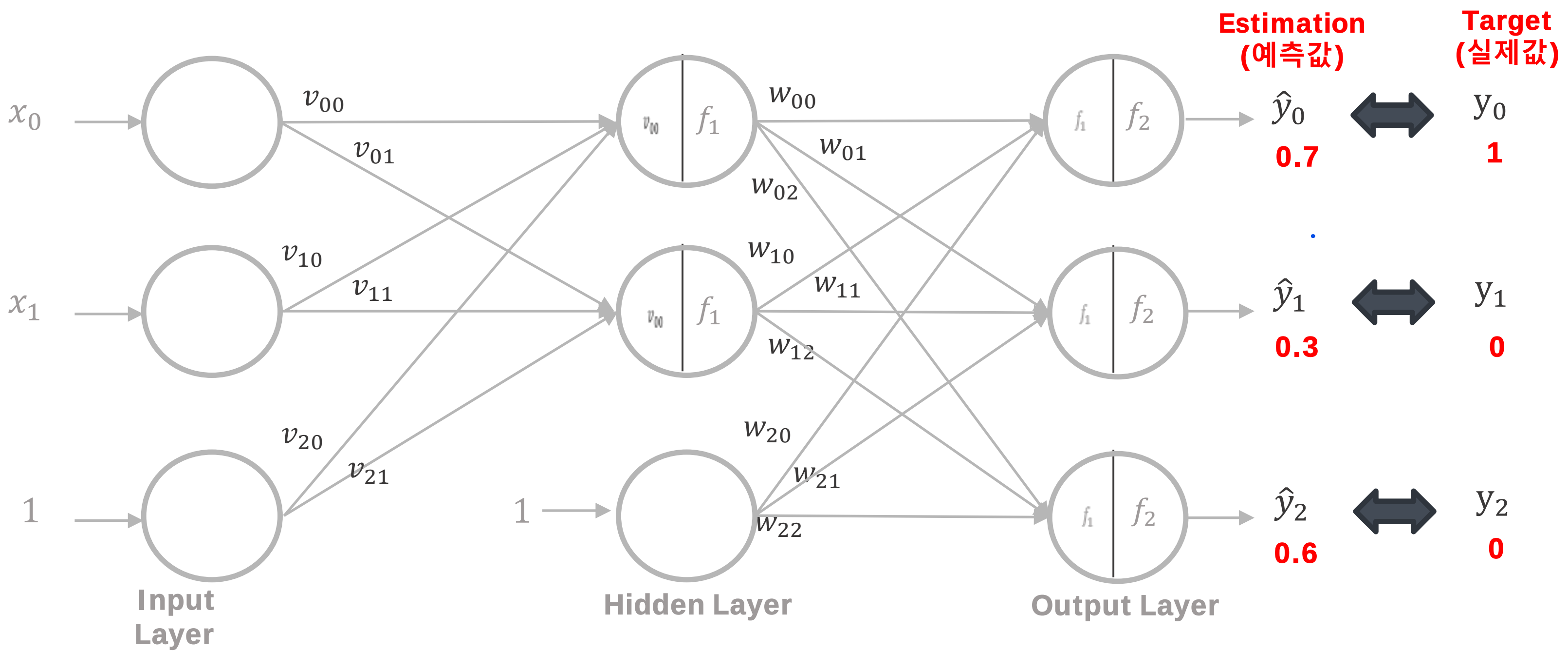
오차 역전파 알고리즘 - 일반화

- Step1) Forward propagation 통한 예측값 도출



오차 역전파 알고리즘 - 일반화

- Step2) 경사하강법을 통한 가중치 업데이트



❖ **n번째 데이터에 대해서 가중치 업데이트**

$$v_{ml} \leftarrow v_{ml} - \eta \frac{\partial}{\partial v_{ml}} \epsilon_{MSE}(n)$$

$$w_{lq} \leftarrow w_{lq} - \eta \frac{\partial}{\partial w_{lq}} \epsilon_{MSE}(n)$$

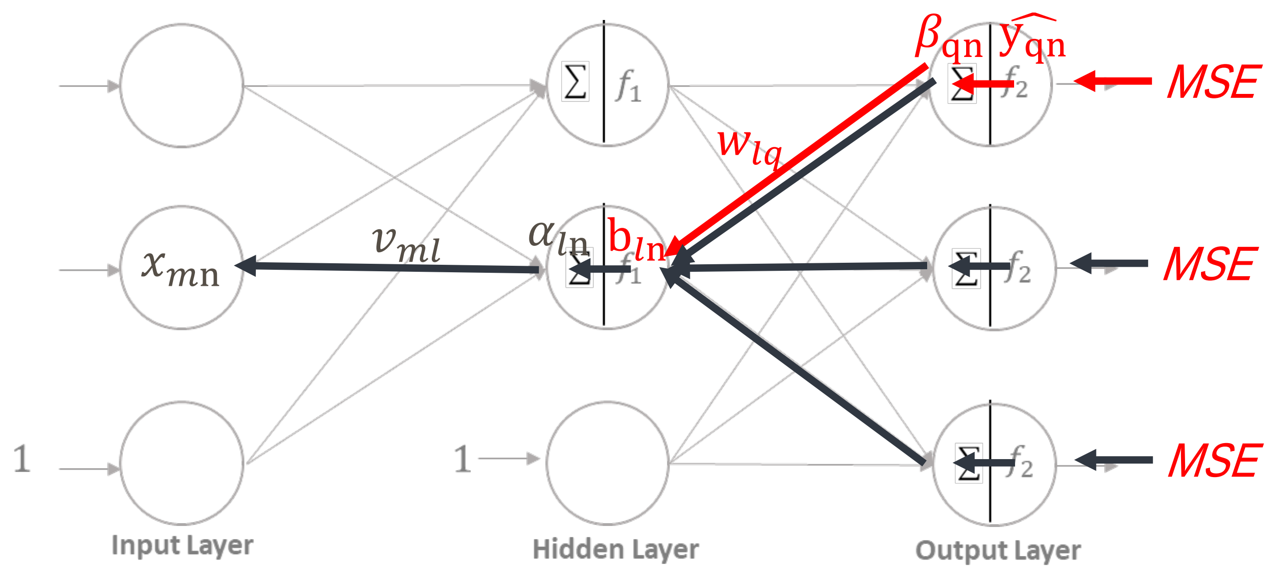
$$\epsilon_{MSE}(n) = \sum_{q=0}^{Q-1} (\hat{y}_{qn} - y_{qn})^2$$

오차 역전파 알고리즘은 데이터 단위로 매개변수를 업데이트

오차 역전파 알고리즘

$m \rightarrow l \rightarrow q$

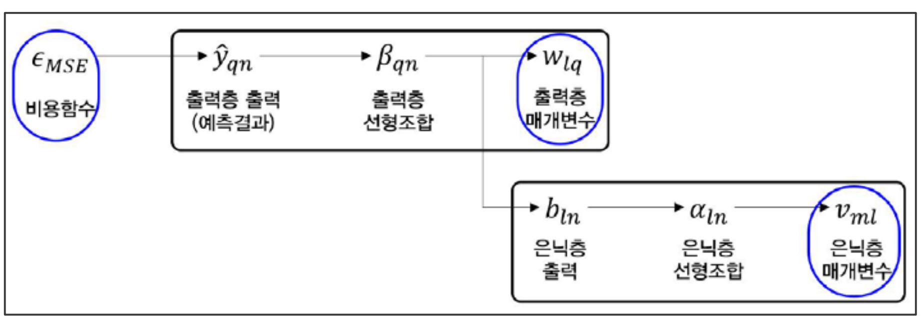
- Step 2-1) 경사하강법에서의 도함수 계산



Forward-propagation에 사용한 weight (Old)

$$v_{ml}^{New} \leftarrow v_{ml}^{Old} - \eta \frac{\partial}{\partial v_{ml}^{Old}} \epsilon_{MSE}(n)$$

$$w_{lq}^{New} \leftarrow w_{lq}^{Old} - \eta \frac{\partial}{\partial w_{lq}^{Old}} \epsilon_{MSE}(n)$$



$$\frac{\partial}{\partial w_{lq}} \epsilon_{MSE}(n) = \frac{\partial \epsilon_{MSE}(n)}{\partial \hat{y}_{qn}} \cdot \frac{\partial \hat{y}_{qn}}{\partial \beta_{qn}} \cdot \frac{\partial \beta_{qn}}{\partial w_{lq}}$$

$$\frac{\partial}{\partial v_{ml}} \epsilon_{MSE}(n) = \frac{\partial \epsilon_{MSE}(n)}{\partial b_{ln}} \cdot \frac{\partial b_{ln}}{\partial \alpha_{ln}} \cdot \frac{\partial \alpha_{ln}}{\partial v_{ml}}$$

오차 역전파 알고리즘

$$\frac{\partial}{\partial w_{lq}}\epsilon_{MSE}(n) = \frac{\partial \epsilon_{MSE}(n)}{\partial \hat{y}_{qn}} \cdot \frac{\partial \hat{y}_{qn}}{\partial \beta_{qn}} \cdot \frac{\partial \beta_{qn}}{\partial w_{lq}}$$

- $\frac{\partial}{\partial w_{lq}}\epsilon_{MSE}(n)$ 구하기
  - 우변 1항

$$\frac{\partial \epsilon_{MSE}(n)}{\partial \hat{y}_{qn}} = \frac{\partial}{\partial \hat{y}_{qn}}\sum_{j=0}^{Q-1}\left(\hat{y}_{jn} - y_{jn}\right)^2$$

$$= \frac{\partial}{\partial \hat{y}_{qn}}\left\{(\hat{y}_{0n} - y_{0n})^2 + \cdots + \left(\hat{y}_{qn} - y_{qn}\right)^2 + \cdots + \left(\hat{y}_{(Q-1)n} - y_{(Q-1)n}\right)^2\right\}$$

$$= 2\left(\hat{y}_{qn} - y_{qn}\right)$$

  - 우변 2항 (sigmoid 함수)

$$\frac{\partial \hat{y}_{qn}}{\partial \beta_{qn}} = \frac{\partial}{\partial \beta_{qn}} f\left(\beta_{qn}\right) = f\left(\beta_{qn}\right)\left(1 - f\left(\beta_{qn}\right)\right) = \hat{y}_{qn}\left(1 - \hat{y}_{qn}\right)$$

(참고) 시그모이드 함수의 미분

$$f'(x) = f(x)\left(1 - f(x)\right)$$

오차 역전파 알고리즘

$$\frac{\partial}{\partial w_{lq}} \epsilon_{MSE}(n) = \frac{\partial \epsilon_{MSE}(n)}{\partial \hat{y}_{qn}} \cdot \frac{\partial \hat{y}_{qn}}{\partial \beta_{qn}} \cdot \frac{\partial \beta_{qn}}{\partial w_{lq}}$$

- 우변 3항

$$\frac{\partial \beta_{qn}}{\partial w_{lq}} = \frac{\partial}{\partial w_{lq}} \sum_{j=0}^{L} w_{jq} b_{jn} = \frac{\partial}{\partial w_{lq}} \left(w_{0q} b_{0n} + \cdots + w_{lq} b_{ln} + \cdots w_{Lq} b_{Ln} \right) = b_{ln}$$

- 최종

$$\frac{\partial \epsilon_{MSE}(n)}{\partial w_{lq}} = \frac{\partial \epsilon_{MSE}(n)}{\partial \hat{y}_{qn}} \cdot \frac{\partial \hat{y}_{qn}}{\partial \beta_{qn}} \cdot \frac{\partial \beta_{qn}}{\partial w_{lq}}$$

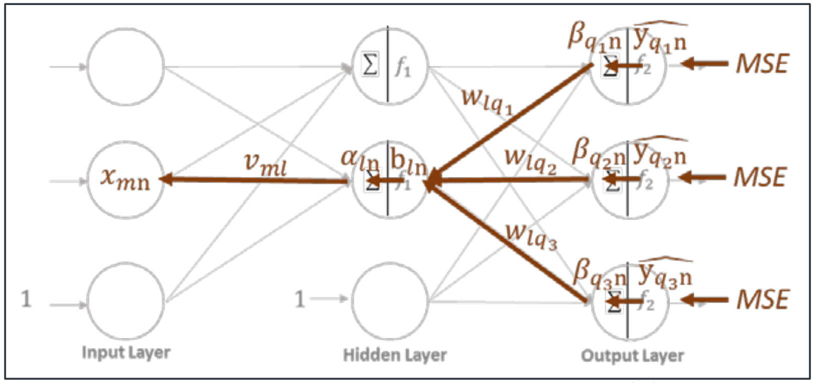
$$= 2\left(\hat{y}_{qn} - y_{qn}\right) \hat{y}_{qn} \left(1 - \hat{y}_{qn}\right) b_{ln}$$

$$= e_{qn} b_{ln}$$

where $e_{qn} \triangleq 2\left(\hat{y}_{qn} - y_{qn}\right) \hat{y}_{qn} \left(1 - \hat{y}_{qn}\right)$

$$w_{lq} \leftarrow w_{lq} - \eta \frac{\partial}{\partial w_{lq}} \epsilon_{MSE}(n)$$

오차 역전파 알고리즘



- $\frac{\partial}{\partial v_{ml}}\epsilon_{MSE}(n)$ 구하기
  - 우변 1항

$$\frac{\partial}{\partial v_{ml}}\epsilon_{MSE}(n) = \frac{\partial \epsilon_{MSE}(n)}{\partial b_{ln}} \cdot \frac{\partial b_{ln}}{\partial \alpha_{ln}} \cdot \frac{\partial \alpha_{ln}}{\partial v_{ml}}$$

n: n번째 data
$m \rightarrow l \rightarrow q$

모든 output $\hat{y}_{qn}$은 b_{ln}의 함수

$$\frac{\partial \epsilon_{MSE}(n)}{\partial b_{ln}} = \frac{\partial}{\partial b_{ln}}\sum_{q=0}^{Q-1}(\hat{y}_{qn} - y_{qn})^2 = 2\sum_{q=0}^{Q-1}(\hat{y}_{qn} - y_{qn})\frac{\partial \hat{y}_{qn}}{\partial b_{ln}}$$

f는 activation function

$$= 2\sum_{q=0}^{Q-1}(\hat{y}_{qn} - y_{qn})\frac{\partial}{\partial b_{ln}}f\left(\boxed{\sum_{j=0}^{L} w_{jq} b_{jn}}\right)$$

L: hidden layer 노드 개수
w_{jq}: hidden layer j번째 노드→특정 q번째 output

q번째 output은 모든 hidden layer 노드로부터 linear sum

$$= 2\sum_{q=0}^{Q-1}(\hat{y}_{qn} - y_{qn})\frac{\partial}{\partial b_{ln}}f(w_{0q}b_{0n} + \cdots + w_{lq}b_{ln} + \cdots + w_{Lq}b_{Ln})$$

$$= 2\sum_{q=0}^{Q-1}(\hat{y}_{qn} - y_{qn})\, f\left(\sum_{j=0}^{L} w_{jq} b_{jn}\right)\left(1 - f\left(\sum_{j=0}^{L} w_{jq} b_{jn}\right)\right) w_{lq}$$

$$= 2\sum_{q=0}^{Q-1}(\hat{y}_{qn} - y_{qn})\, \hat{y}_{qn}(1 - \hat{y}_{qn}) w_{lq}$$

l 번째 hidden node와 연결된 weigh들과 output node를 고려하게 됨

오차 역전파 알고리즘

$$\frac{\partial}{\partial v_{ml}}\epsilon_{MSE}(n) = \frac{\partial \epsilon_{MSE}(n)}{\partial b_{ln}} \cdot \frac{\partial b_{ln}}{\partial \alpha_{ln}} \cdot \frac{\partial \alpha_{ln}}{\partial v_{ml}}$$

- 우변 2항 (sigmoid 함수)

$$\frac{\partial b_{ln}}{\partial \alpha_{ln}} = \frac{\partial}{\partial \alpha_{ln}} f(\alpha_{ln}) = f(\alpha_{ln})\big(1 - f(\alpha_{ln})\big) = b_{ln}(1 - b_{ln})$$

- 우변 3항 (linear sum)

$$\frac{\partial \alpha_{ln}}{\partial v_{ml}} = \frac{\partial}{\partial v_{ml}} \sum_{j=0}^{M} v_{jl} x_{jn} = \frac{\partial}{\partial v_{ml}} (v_{0l}x_{0n} + \cdots + v_{ml}x_{mn} + \cdots + v_{Ml}x_{Mn}) = x_{mn}$$

- 최종

$$\frac{\partial \epsilon_{MSE}(n)}{\partial v_{ml}} = \frac{\partial \epsilon_{MSE}(n)}{\partial b_{ln}} \cdot \frac{\partial b_{ln}}{\partial \alpha_{ln}} \cdot \frac{\partial \alpha_{ln}}{\partial v_{ml}}$$

$$= \left\{2 \sum_{q=0}^{Q-1} (\hat{y}_{qn} - y_{qn})\, \hat{y}_{qn}(1 - \hat{y}_{qn}) w_{lq}\right\} \{b_{ln}(1 - b_{ln})\} x_{mn}$$

$$= b_{ln}(1 - b_{ln}) x_{mn} \sum_{q=0}^{Q-1} e_{qn} w_{lq}$$

$$v_{ml} \leftarrow v_{ml} - \eta \frac{\partial}{\partial v_{ml}} \epsilon_{MSE}(n)$$

오차 역전파 알고리즘

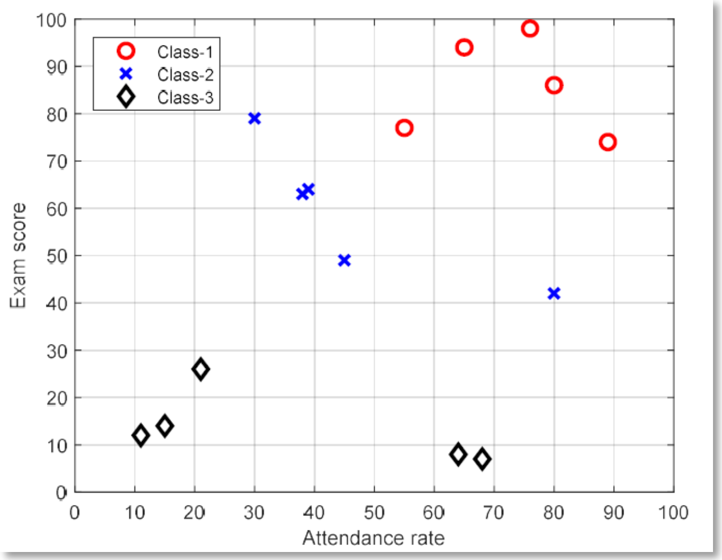
- 전체 알고리즘 적용 순서
  - 1) 신경망 모델 설계(Hidden Layer수, Node수, 활성화 함수, Learning Rate 등)
  - 2) 설계된 모델에 따른 Weight Matrix 생성 및 초기화
  - 3) N개의 훈련 데이터 Shuffle
  - 4) 0부터 N-1번째 데이터에 대해 순차적으로 오차 역전파 알고리즘 적용

    (1 epoch)
    - for n번째 데이터
      - (순전파) n-1에서 업데이트 된 Weight로부터 예측값 $\hat{y}_{qn}$ 계산
      - 0.5를 기준으로 0과 1로 예측값을 변환, 분류 정확도 계산 및 저장
      - (역전파) 실제값(Target)과 예측값(Output) 간의 오차로부터 Weight 업데이트
  - 사용자가 입력한 epoch 수 만큼 3~4) 반복

다계층 인공 신경망을 이용한 분류 시스템 설계

- 데이터
  - 2입력 3클래스 분류 데이터
    - 입력 속성: 출석률, 시험성적
    - 출력 속성: 학점 (A, B, C)

데이터 번호	입력		출력
	출석률	시험성적	학점
0	76	98	A
1	65	94	A
2	80	86	A
3	89	74	A
4	55	77	A
5	30	79	B
6	39	64	B
7	38	63	B
8	45	49	B
9	80	42	B
10	68	7	C
11	64	8	C
12	21	26	C
13	15	14	C
14	11	12	C



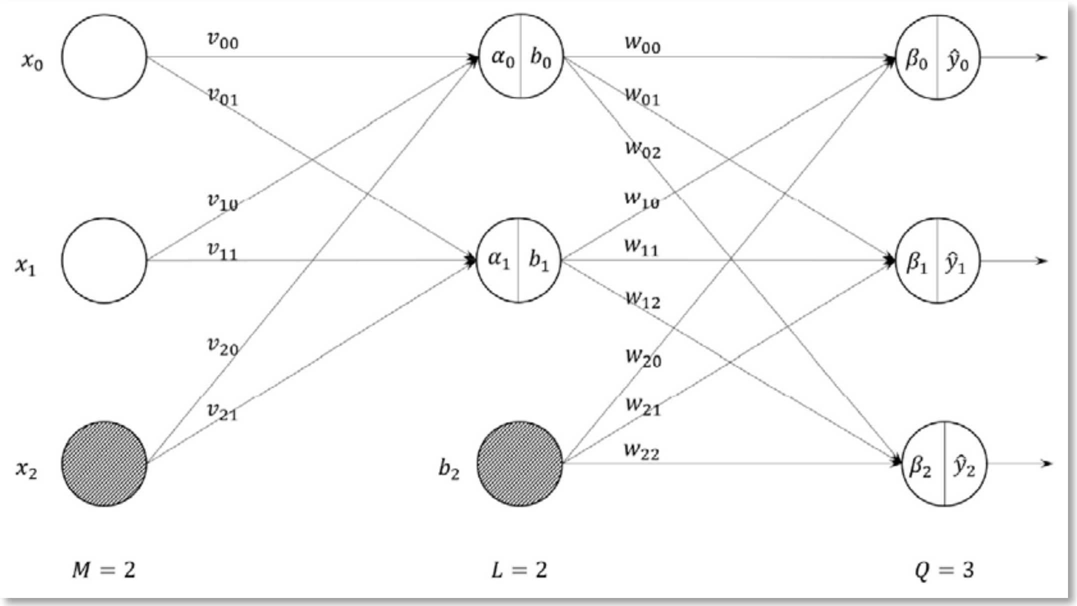
다계층 인공 신경망을 이용한 분류 시스템 설계

- 데이터 가공
  - One-Hot Encoding
    - 출력 변수의 가공
    - 출력 클래스의 수와 같은 출력층 노드 설계를 위한 인코딩
    - y_q: 출력 클래스가 q에 속할 확률(가능성), $q = 0, 1, 2$

데이터 번호	입력		출력		
	출석률	시험성적	y_0	y_1	y_2
0	76	98	1	0	0
1	65	94	1	0	0
2	80	86	1	0	0
3	89	74	1	0	0
4	55	77	1	0	0
5	30	79	0	1	0
6	39	64	0	1	0
7	38	63	0	1	0
8	45	49	0	1	0
9	80	42	0	1	0
10	68	7	0	0	1
11	64	8	0	0	1
12	21	26	0	0	1
13	15	14	0	0	1
14	11	12	0	0	1

다계층 인공 신경망을 이용한 분류 시스템 설계

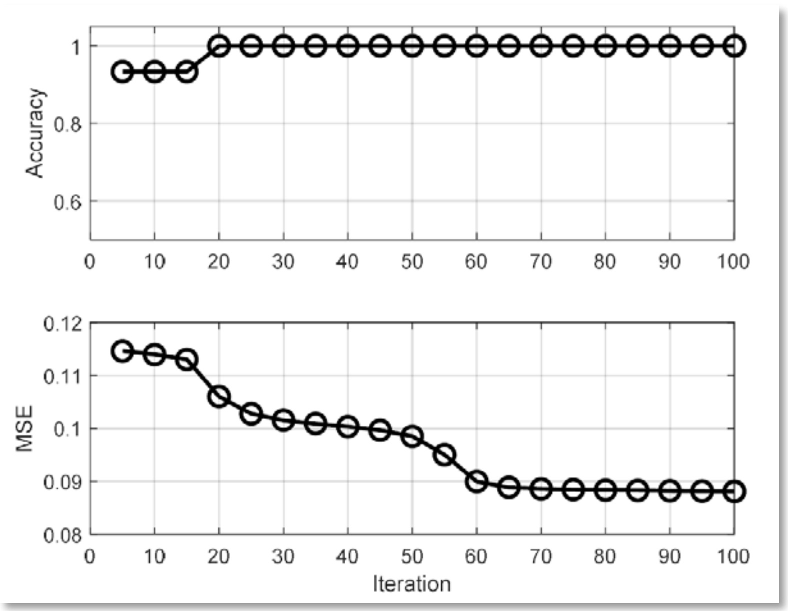
- 인공 신경망 설계
  - 은닉층 추가
    - 임의의 은닉층 추가
    - 각 은닉층의 노드 수 자유롭게 설정 가능
  - 2계층 인공 신경망
    - 모든 계층의 활성함수로 시그모이드 함수 사용

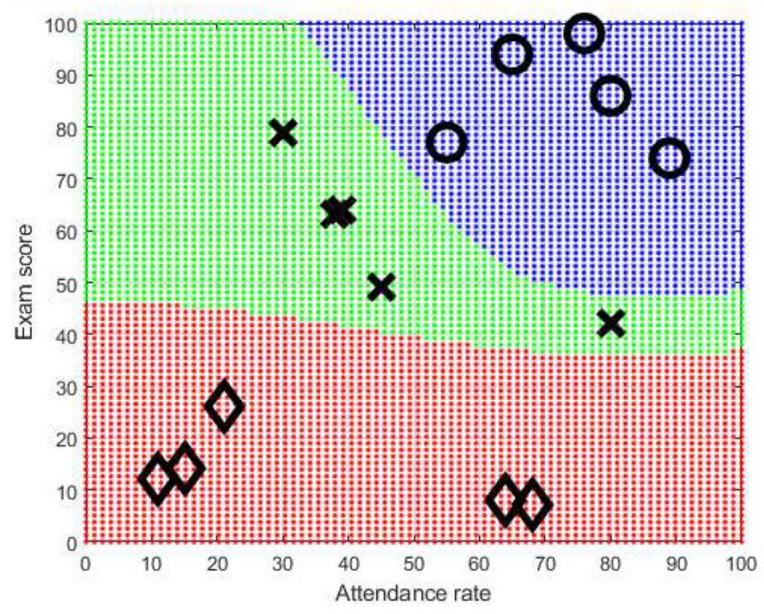


다계층 인공 신경망을 이용한 분류 시스템 설계

- 인공 신경망 학습
  - 비용함수 설계
    - 평균제곱오차 $\epsilon_{MSE}(n) = \sum_{q=0}^{2} \left(\hat{y}_{qn} - y_{qn}\right)^2$
  - 학습률 0.001로 오차 역전파 알고리즘 적용

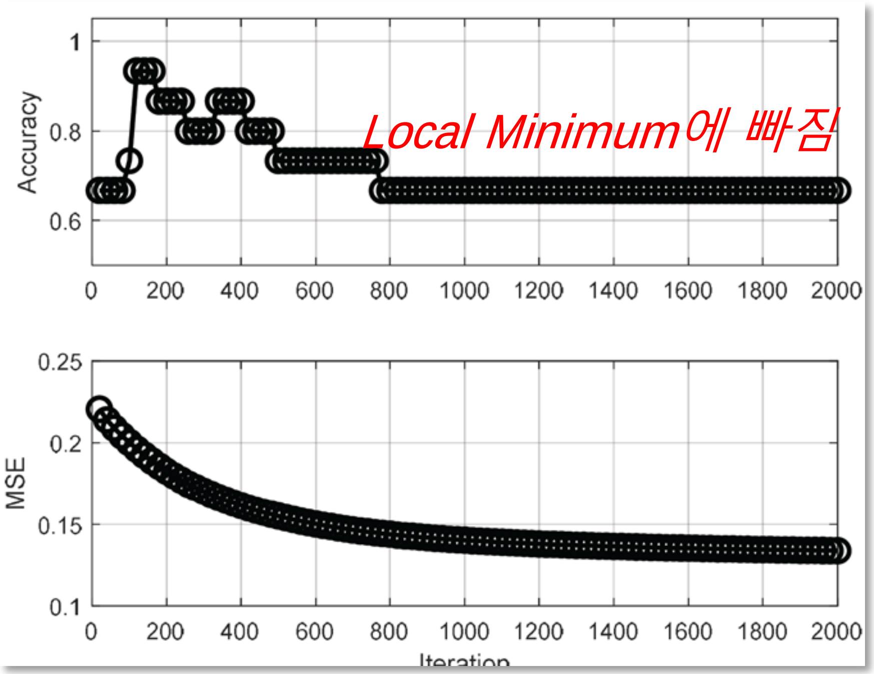
$$V = \begin{bmatrix} -0.0837 & -0.0259 \\ -0.0496 & 0.1757 \\ 6.3349 & -5.1625 \end{bmatrix}, \quad W = \begin{bmatrix} -5.0855 & -0.6823 & 3.9907 \\ 3.5843 & 0.9311 & -5.2163 \\ -2.7844 & -1.0409 & 0.2020 \end{bmatrix}$$

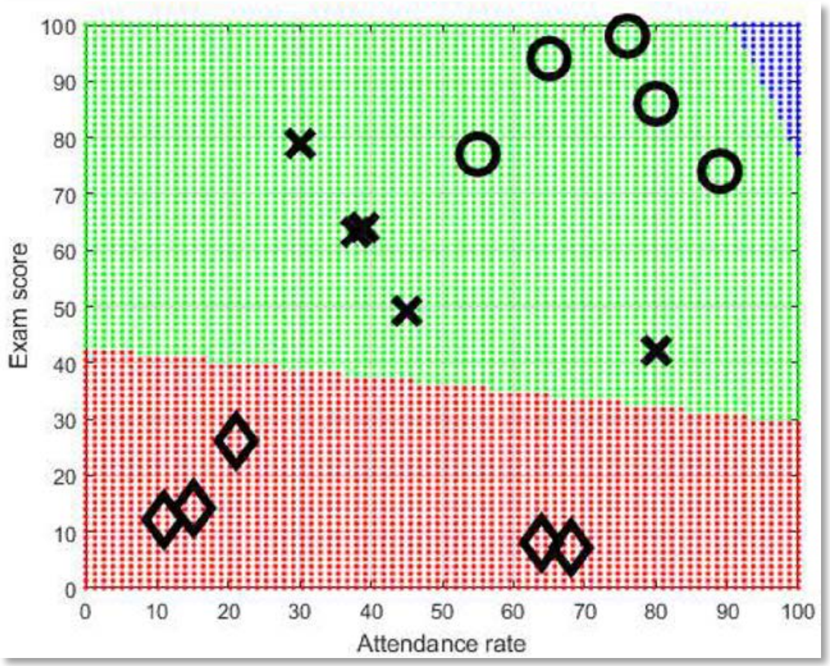




다계층 인공 신경망을 이용한 분류 시스템 설계

- 인공 신경망 학습
  - 학습에 실패한 경우





실습

- Error Back-Propagation 알고리즘 구현
  - 알고리즘을 사용자 지정함수로 구현

- Two-Layer Neural Network “Training”
  - “NN_data.csv”를 7:3 으로 Training set :Test set 분할
  - Training set과 1)에서 구현한 알고리즘으로부터 Two-Layer Neural Network를 Training
  - Epoch에 따른 Accuracy 및 MSE 그래프 도출
  - Hyper-parameter tuning을 통해 최적화

- Two-Layer Neural Network “Test”
  - 2)에서 도출한 Two-Layer Neural Network에 Testing set 삽입