

Machine Learning

10강

신경망 (2)

컴퓨터과학과 이관용 교수

학습목차

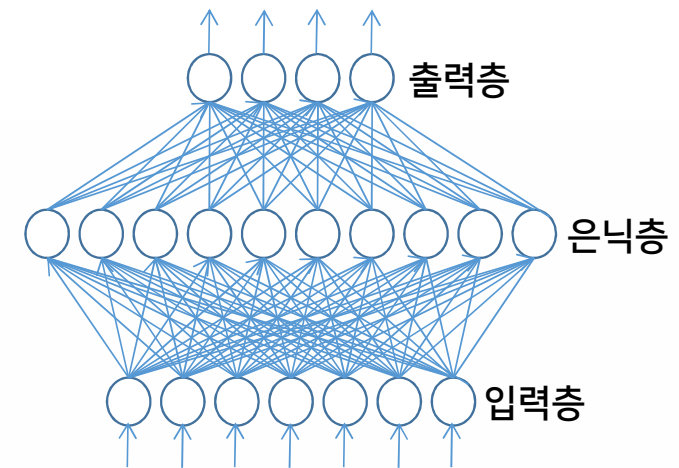
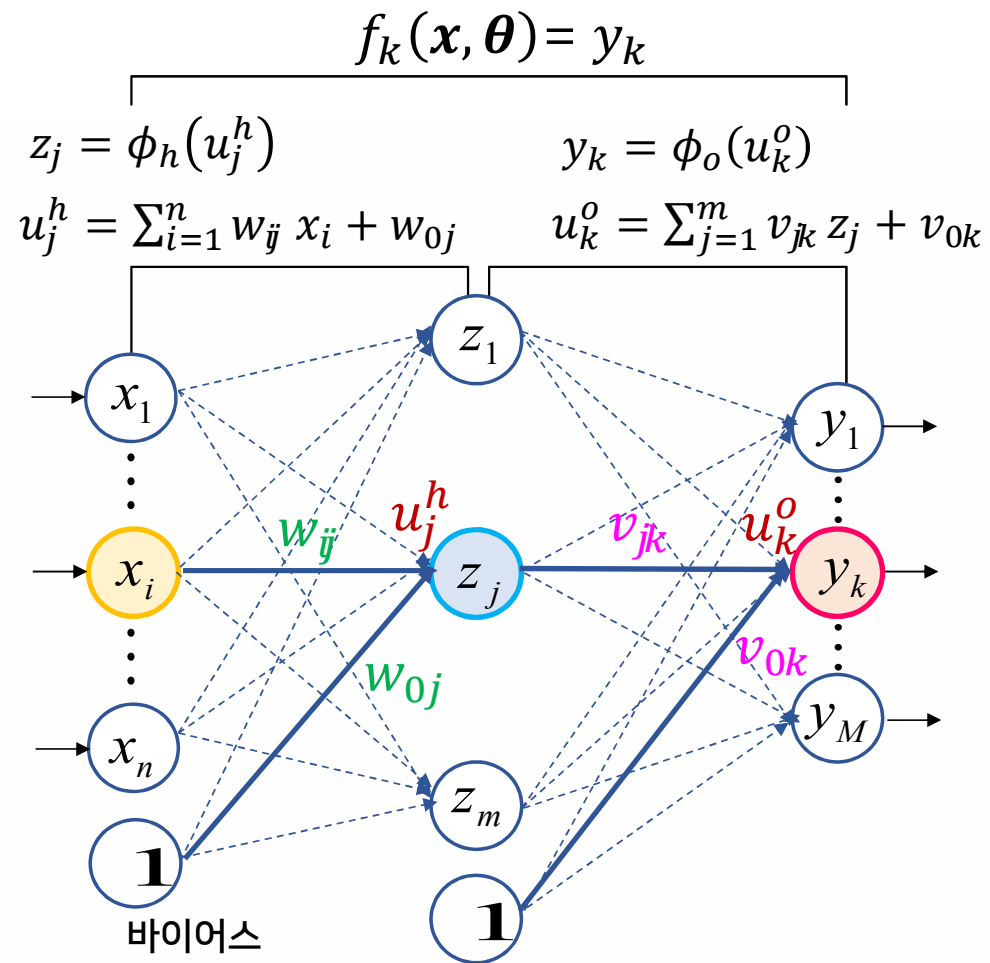
01 다층 퍼셉트론의 학습

02 응용: 숫자인식

1

다층 퍼셉트론의 학습

MLP의 수학적 표현



다층 퍼셉트론의 학습

○ 신경망의 학습이란?

- 원하는 함수를 나타내는 가중치 weight를 찾는 것

○ MLP의 학습

- 지도학습 → 학습 데이터 집합 $X = \{(x_i, t_i)\} (i = 1, \dots, N)$

- 학습의 목적

✓ 입력 x_i 에 대한 신경망의 출력 y_i 와 목표 출력 t_i 의 차이를 최소화

- 오차함수 → 평균 제곱 오차

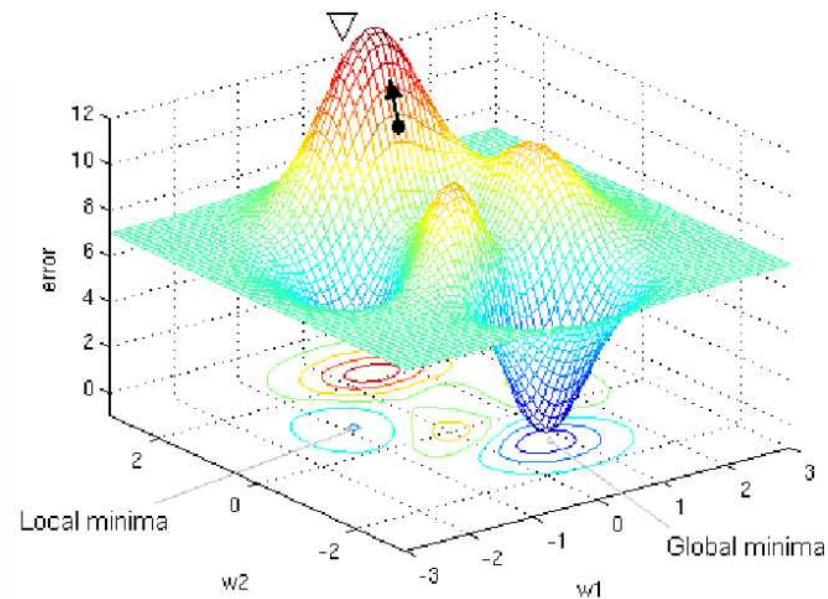
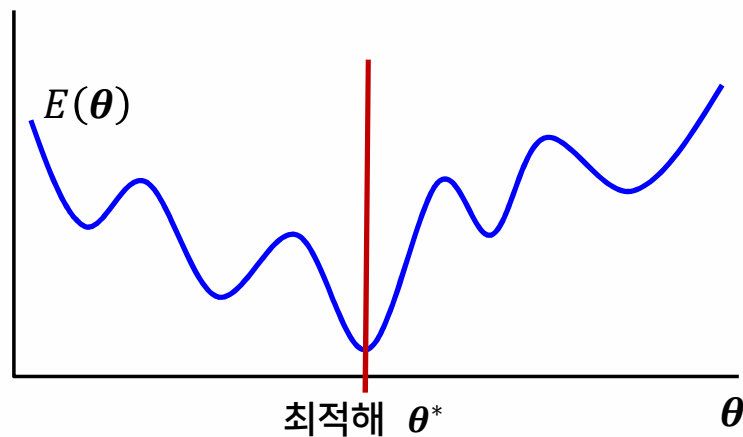
$$E(X, \theta) = \frac{1}{2N} \sum_{i=1}^N \|t_i - f(x_i, \theta)\|^2$$

✓ 최적의 가중치 θ^* → 오차함수 $E(X, \theta)$ 를 최소화하는 θ

다층 퍼셉트론의 학습

○ 오차함수의 일반적 형태

□ 매우 복잡한 형태의 비선형함수



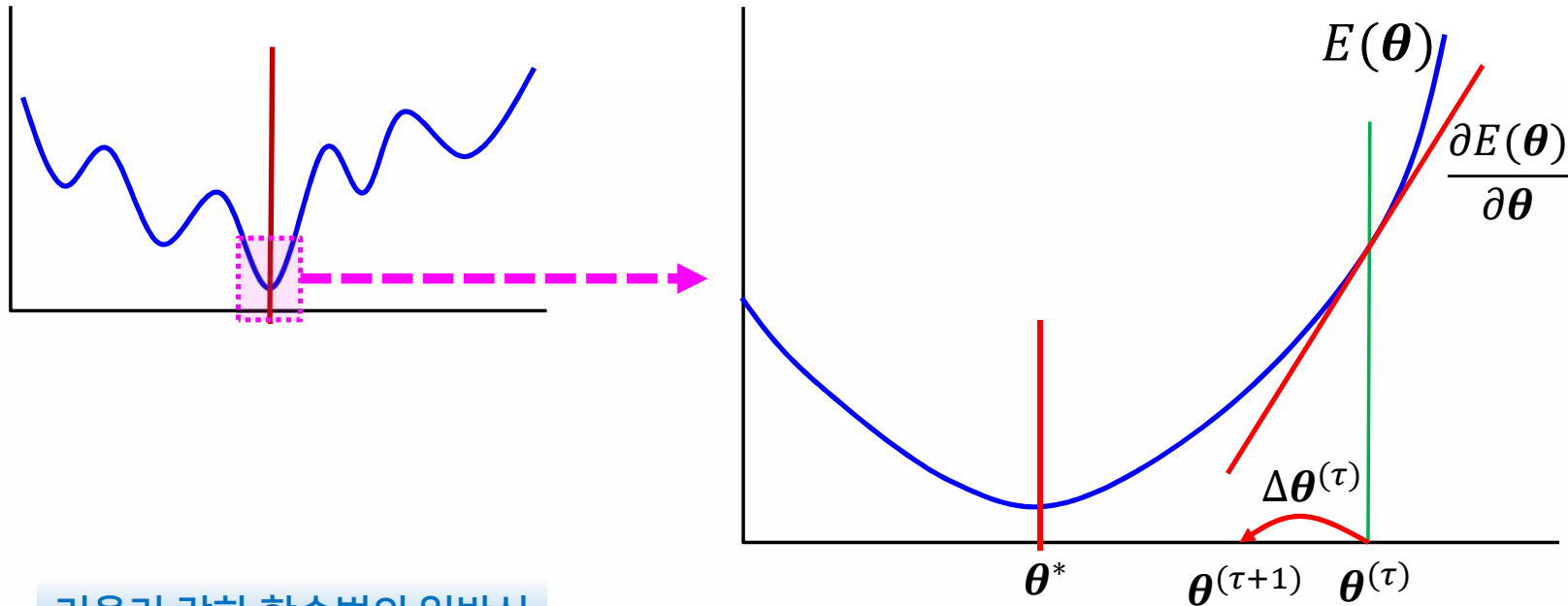
기울기 강하 학습법
gradient descent learning method

비선형적인 함수의 최소값을 찾아가는 반복적인 방법

오차 함수 $E(\theta)$ 의 기울기를 이용하여 감소하는 방향으로 조금씩 이동

오류 역전파 학습 알고리즘
error backpropagation learning algorithm

기울기 강하 학습법



기울기 강하 학습법의 일반식

$$\theta^{(\tau+1)} = \theta^{(\tau)} + \Delta\theta^{(\tau)} = \theta^{(\tau)} - \eta \frac{\partial E(\theta^{(\tau)})}{\partial \theta}$$

η : 학습률 \rightarrow 변화 폭 결정
 $\frac{\partial E(\theta^{(\tau)})}{\partial \theta}$: 오류 역전파 알고리즘으로 계산

오류 역전파 학습

현재 입력 x 에 대한 오차 함수 (온라인 학습 모드)

$$E(x, \theta) = \frac{1}{2} (t_k - y_k)^2 = \frac{1}{2} (t_k - f_k(x, \theta))^2$$

$$= \frac{1}{2} (t_k - \phi_o(u_k^o))^2$$

은닉층에서 출력층으로의 가중치 v_{jk} 의 수정식

$$\frac{\partial E}{\partial v_{jk}} = \frac{\partial E}{\partial u_k^o} \frac{\partial u_k^o}{\partial v_{jk}} = \delta_k z_j$$

$$\delta_k = -\phi_o'(u_k^o)(t_k - y_k)$$

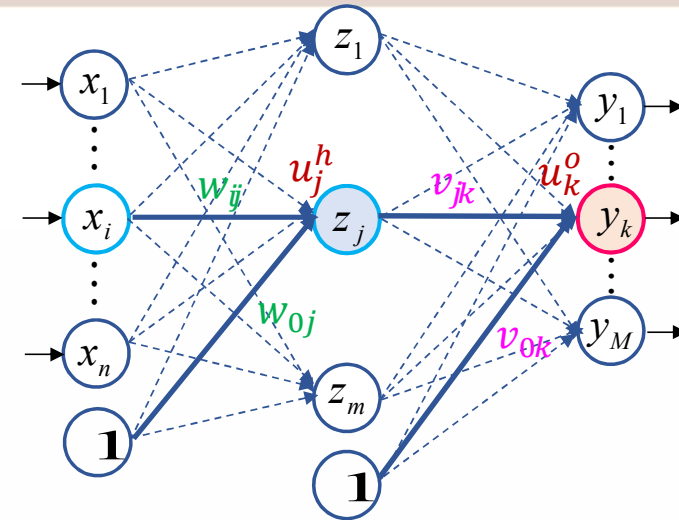
시그모이드 함수

$$\phi_o'(u_k^o) = (1 - y_k)y_k$$

하이퍼탄젠트 함수

$$\phi_o'(u_k^o) = (1 - y_k)(1 + y_k)$$

1. 다층 퍼셉트론의 학습



$$y_k = \phi_o(u_k^o) \quad u_k^o = \sum_{j=1}^m v_{jk} z_j + v_{0k}$$

$$z_j = \phi_h(u_j^h) \quad u_j^h = \sum_{i=1}^n w_{ij} x_i + w_{0j}$$

오류 역전파 학습

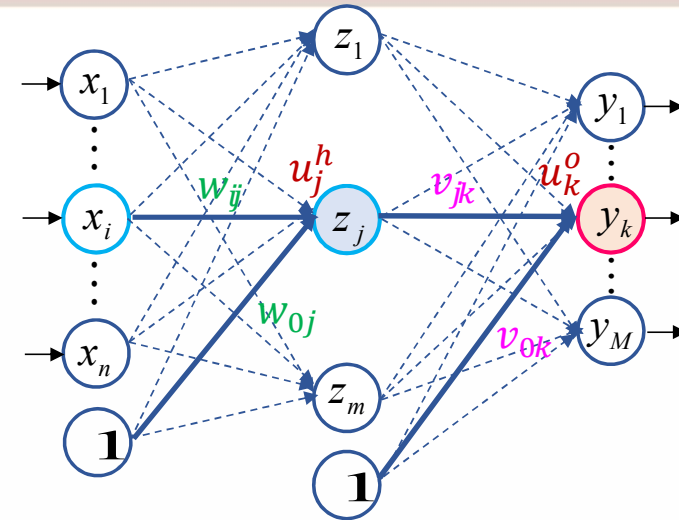
입력층에서 은닉층으로의 가중치 w_{ij} 의 수정식

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial u_j^h} \frac{\partial u_j^h}{\partial w_{ij}} = \delta_j x_i$$

$$\delta_j = \frac{\partial E}{\partial u_j^h} = \sum_{k=1}^M \frac{\partial E}{\partial u_k^o} \frac{\partial u_k^o}{\partial u_j^h} = \phi_h'(u_j^h) \sum_{k=1}^M \delta_k v_{jk}$$

$$u_k^o = \sum_{j=1}^m v_{jk} \phi_h(u_j^h) + v_{0k}$$

$$\frac{\partial u_k^o}{\partial u_j^h} = \phi_h'(u_j^h) v_{jk}$$



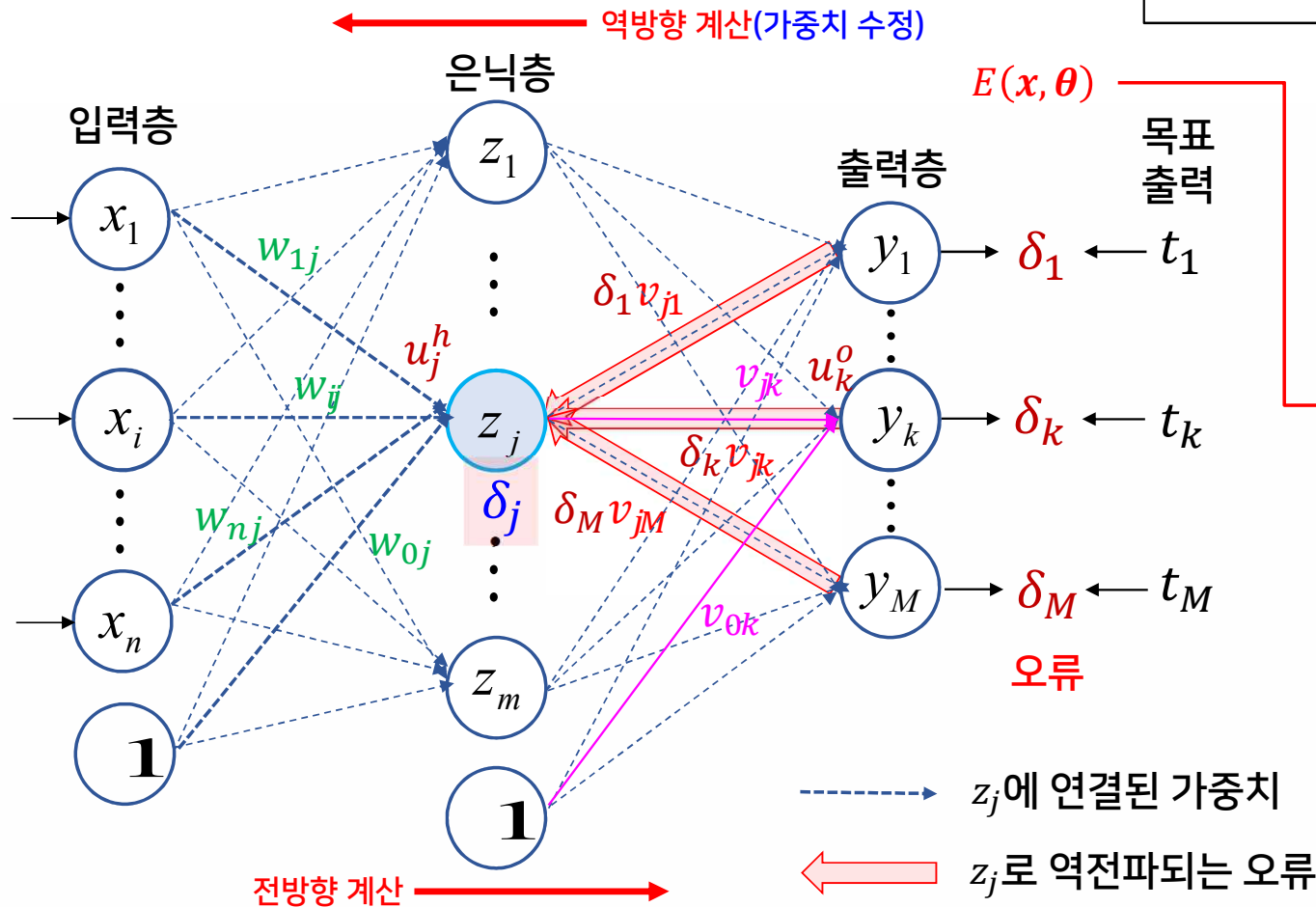
$$y_k = \phi_o(u_k^o) \quad u_k^o = \sum_{j=1}^m v_{jk} z_j + v_{0k}$$

$$z_j = \phi_h(u_j^h) \quad u_j^h = \sum_{i=1}^n w_{ij} x_i + w_{0j}$$

출력 뉴런의 오차가 은닉 뉴런에 거꾸로 전파되는 형태

오류 역전파 학습

$$\theta^{(\tau+1)} = \theta^{(\tau)} + \Delta\theta^{(\tau)} = \theta^{(\tau)} - \eta \frac{\partial E(\theta^{(\tau)})}{\partial \theta}$$



$$v_{jk}^{(\tau+1)} = v_{jk}^{(\tau)} + \Delta v_{jk}$$

$$w_{ij}^{(\tau+1)} = w_{ij}^{(\tau)} + \Delta w_{ij}$$

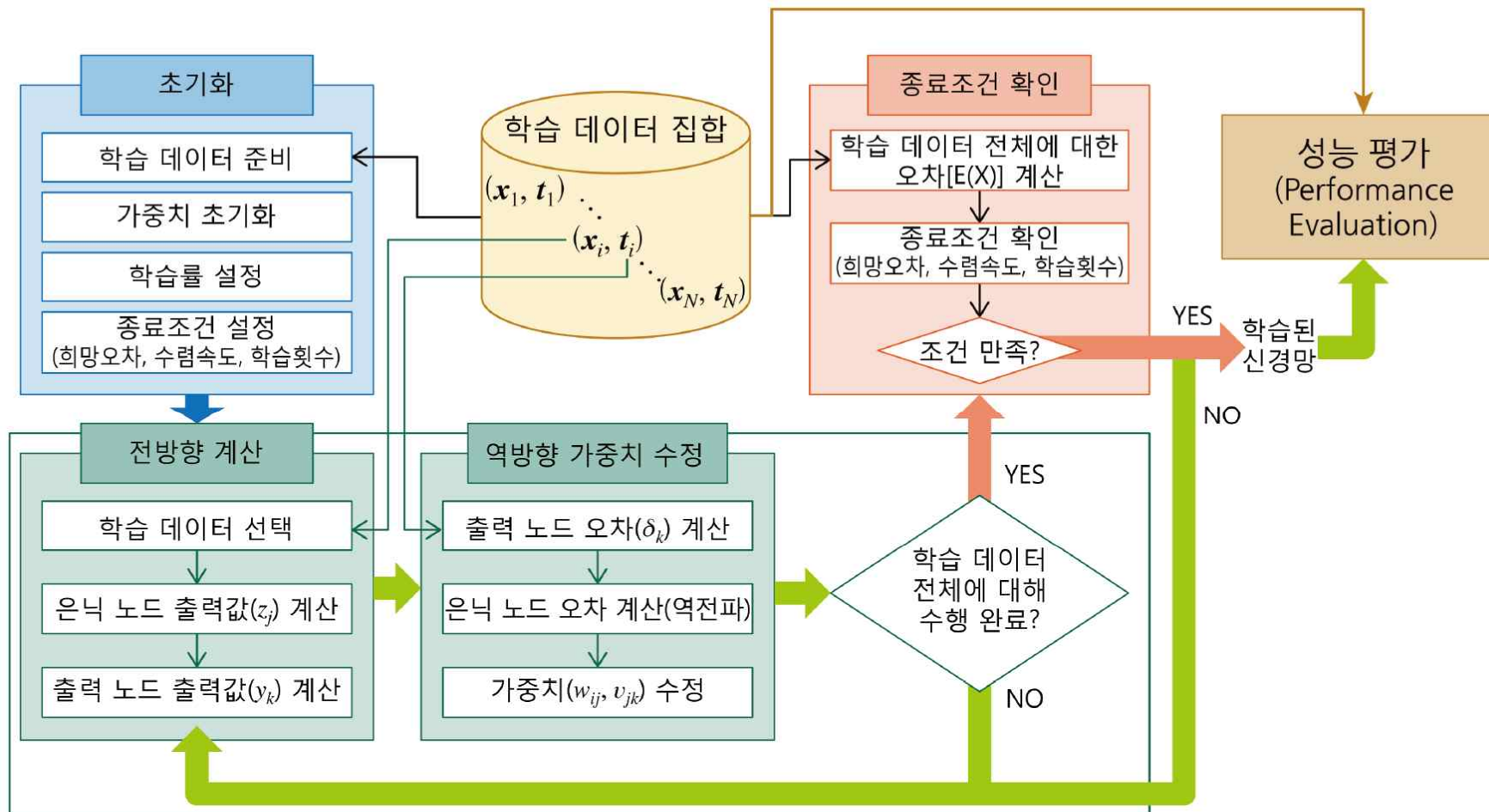
$$\Delta v_{jk} = -\eta \frac{\partial E}{\partial v_{jk}} = -\eta \delta_k z_j$$

$$\delta_k = -\phi'_o(u_k^o)(t_k - y_k)$$

$$\Delta w_{ij} = -\eta \frac{\partial E}{\partial w_{ij}} = -\eta \delta_j x_i$$

$$\delta_j = \phi'_h(u_j^h) \sum_{k=1}^M \delta_k v_{jk}$$

MLP의 전체적인 학습 과정



MLP 학습 - 오류 역전파 학습 알고리즘

- ① 임의의 값으로 가중치 파라미터를 초기화하고, 학습률 η 과 원하는 오차함수의 목표값을 설정
- ② 모든 학습 데이터에 대해, 한 번에 하나의 데이터를 추출하여 다음 단계에 따라 가중치를 수정하는 과정을 반복

[전방향 계산] 현재의 가중치 w_{ij} , v_{jk} 를 이용하여 **출력값**을 계산

- ②-1 하나의 입력 데이터 $x = [x_1, \dots, x_n]^T$ 에 대하여 각 은닉 뉴런의 출력값 z_j ($j = 1, \dots, m$)를 계산

$$u_j^h = \sum_{i=1}^n w_{ij} x_i + w_{0j} \quad z_j = \phi_h(u_j^h)$$

- ②-2 이어서 출력 뉴런의 출력값 y_k ($k = 1, \dots, M$)를 계산

$$u_k^o = \sum_{j=1}^m v_{jk} z_j + v_{0k} \quad y_k = \phi_o(u_k^o)$$

MLP 학습 - 오류 역전파 학습 알고리즘

[역방향 계산] 현재의 가중치 w_{ij} , v_{jk} 를 이용하여 **오류**를 계산

②-3 각 출력 노드의 출력값 y_k 와 목표 출력값 t_k 를 비교하여 다음 식을 이용하여 출력 뉴런으로의 가중치 수정항을 계산 ($k = 1, \dots, M$, $j = 0, \dots, m$, $z_0 = 1$)

$$\frac{\partial E}{\partial v_{jk}} = \frac{\partial E}{\partial u_k^o} \frac{\partial u_k^o}{\partial v_{jk}} = \delta_k z_j \quad \delta_k = -\phi_o'(u_k^o)(t_k - y_k)$$

②-4 계산된 δ_k 를 이용하여 각 은닉 뉴런으로의 가중치 수정항을 계산 ($j = 1, \dots, m$, $i = 0, \dots, n$, $x_0 = 1$)

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial u_j^h} \frac{\partial u_j^h}{\partial w_{ij}} = \delta_j x_i \quad \delta_j = \frac{\partial E}{\partial u_j^h} = \sum_{k=1}^M \frac{\partial E}{\partial u_k^o} \frac{\partial u_k^o}{\partial u_j^h} = \phi_h'(u_j^h) \sum_{k=1}^M \delta_k v_{jk}$$

②-5 계산된 수정항과 학습률을 이용하여 가중치 파라미터를 수정

$$w_{ij}^{(\tau+1)} = w_{ij}^{(\tau)} - \eta \frac{\partial E}{\partial w_{ij}} \quad v_{jk}^{(\tau+1)} = v_{jk}^{(\tau)} - \eta \frac{\partial E}{\partial v_{jk}}$$

MLP 학습 - 오류 역전파 학습 알고리즘

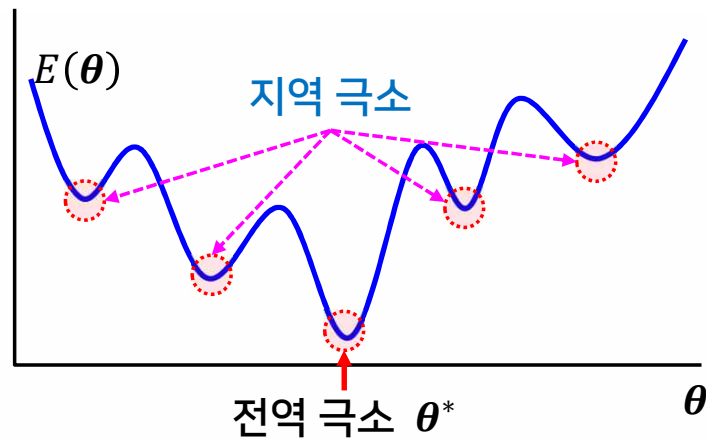
- ③ 전체 학습 데이터에 대하여 ② 과정이 완료된 후,
학습 데이터 전체 집합 X 에 대한 평균 제곱 오차를 다음과 같이 계산

$$E(X, \theta) = \frac{1}{2N} \sum_{i=1}^N \|t_i - f(x_i, \theta)\|^2$$

- ④ ③에서 계산된 오차값이 원하는 목표값보다 작으면 학습을 마치고,
그렇지 않으면 ②~③ 과정을 반복

MLP 학습의 고려사항

(1) 지역 극소의 문제 local minima problem



- ☐ 학습률의 적응적 조정, 시뮬레이티드 어닐링 기법 등
- ☐ 현실적인 해결책
 - ✓ 탐색의 시작점을 결정하는 초기치를 변화시키면서 여러 번 학습을 시도
 - ✓ 충분히 많은 수의 은닉 노드를 사용

MLP 학습의 고려사항

(2) 수렴 속도의 문제

- 원하는 해에 수렴하기까지 긴 학습 시간이 필요

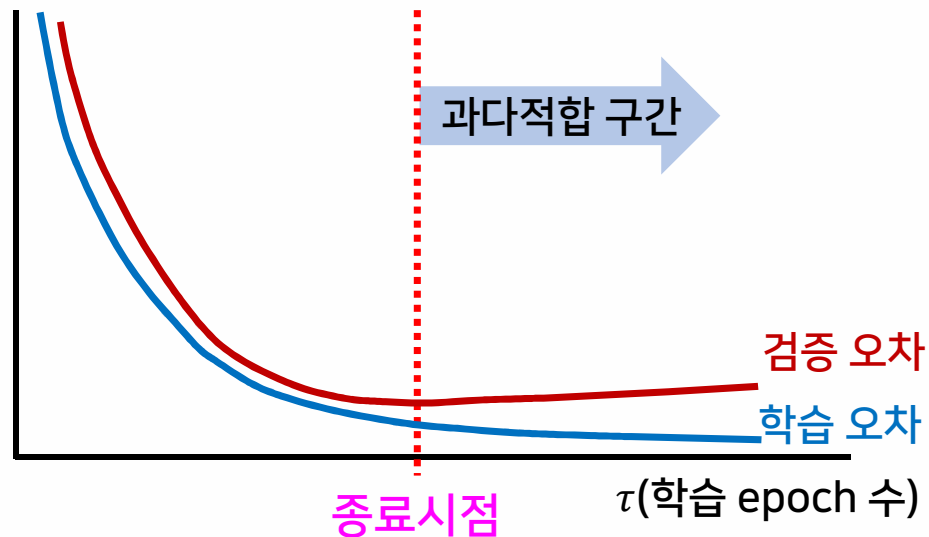


- 다양한 가속화 방법들이 제안됨
 - ✓ 모멘텀 방법, 뉴턴 방법, 자연 기울기 방법 등

MLP 학습의 고려사항

(3) 학습 종료점의 문제

- 과다적합을 피할 수 있는 적절한 학습 종료 시점의 결정이 필요
- 검증 데이터 집합을 사용하는 방법



MLP 학습의 고려사항

(4) 은닉 뉴런의 수

- ☐ 실제 문제에서 학습의 속도와 찾아지는 해의 성능을 좌우
- ☐ 문제 의존적
 - ✓ 입력 데이터 차원과 데이터의 개수 등에 비례하여 개수를 조정

MLP의 학습 전략

○ 학습 모드의 설정

- 온라인 모드 → 각 데이터에 대해 가중치 수정
 - ✓ N 개의 데이터 → N 번의 가중치 수정
 - ✓ 오차 감소 속도는 빠르나 학습이 불안정적
- 배치 모드 → 모든 데이터에 대해 전방향 계산 후 오차를 모두 더하여 가중치 수정
 - ✓ N 개의 데이터 → 1번의 가중치 수정
 - ✓ 오차 감소 속도는 느리나 학습이 안정적
- 미니 배치 모드 → 모든 데이터를 작은 부분집합으로 나누어 가중치 수정
 - ✓ N 개의 데이터를 m 개의 그룹으로 나눔 → m 번의 가중치 수정
 - ✓ 데이터 규모가 큰 경우에 적합

MLP의 학습 전략

○ 모델 설정

□ 은닉 노드의 수

- ✓ 많을수록 표현 가능한 함수가 다양하고 복잡해짐
- ✓ 계산 비용과 일반화 성능을 고려해서 문제에 맞게 적절히 결정

□ 초기 조건 설정

- ✓ 초기 가중치 → 작은 범위의 실수값으로 랜덤하게 설정
- ✓ 학습률 → 1보다 작은 값에서 시작하여 진행 상황에 따라 조정

MLP의 학습 전략

○ 모델 설정

□ 활성화 함수

✓ 은닉 노드 → 시그모이드 함수, 하이퍼탄젠트 함수, ReLU 함수

✓ 출력 노드

- 목표 출력값이 임의의 실수값(회귀 문제) → 선형함수

- 목표 출력값이 클래스 레이블(분류 문제)

→ 시그모이드 함수, 소프트맥스 softmax 함수

MLP의 학습 전략

○ 오차함수 손실함수 loss function

□ 제곱 오차함수 squared error function

$$E_{sqe}(X, \theta) = \sum_{p=1}^N \|t_p - f(x_p, \theta)\|^2 = \sum_{p=1}^N \sum_{k=1}^M (t_{pk} - f_k(x_i, \theta))^2$$

✓ 목표 출력값이 연속한 실수값을 갖는 회귀 문제에 적합

□ 교차엔트로피 오차함수 cross entropy error function

$$E_{crs}(X, \theta) = \sum_{p=1}^N \sum_{k=1}^M t_{pk} \ln f_k(x_i, \theta)$$

$$y_k = f_k(x_i, \theta) = \frac{\exp(u_k^o)}{\sum_{i=1}^M \exp(u_i^o)}$$

✓ 목표 출력값이 1 또는 0의 값을 갖는 분류 문제에 적합

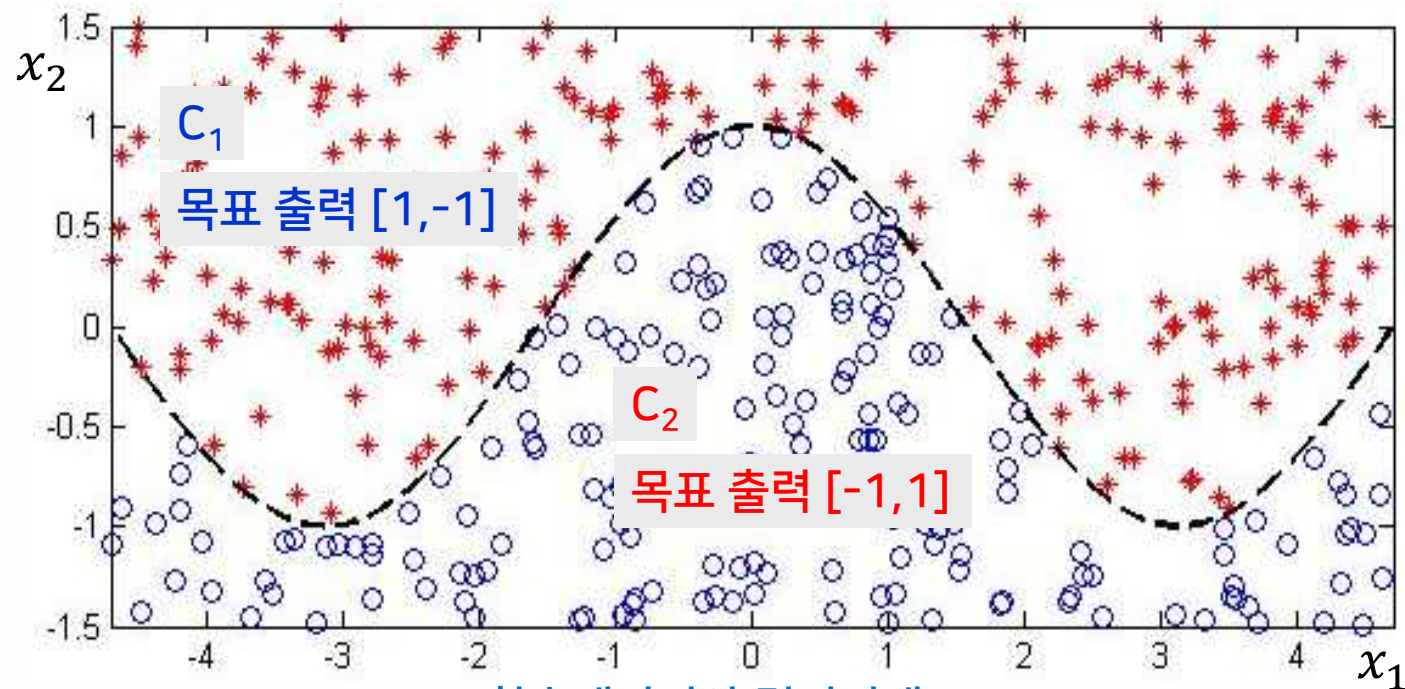
✓ 출력 노드의 활성화 함수로는 소프트맥스 함수를 주로 사용

2

응용: 숫자인식

MLP를 이용한 간단한 패턴 분류 실험

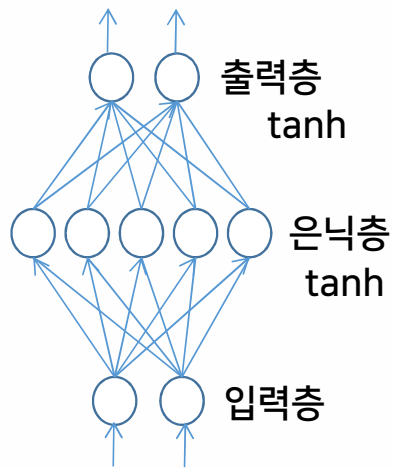
$\{(x_1, x_2) | x_1 \in [-1.5\pi, 1.5\pi], x_2 \in [-1.5, 1.5]\} \rightarrow 400\text{개}$



학습데이터와 결정경계

MLP를 이용한 간단한 패턴 분류 실험

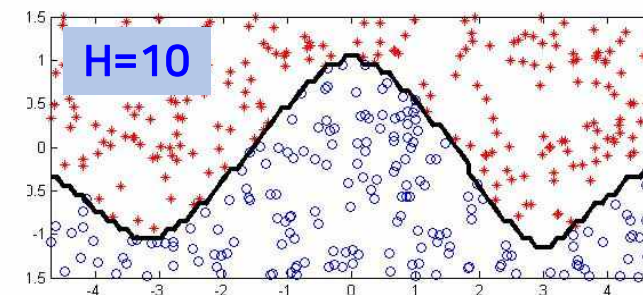
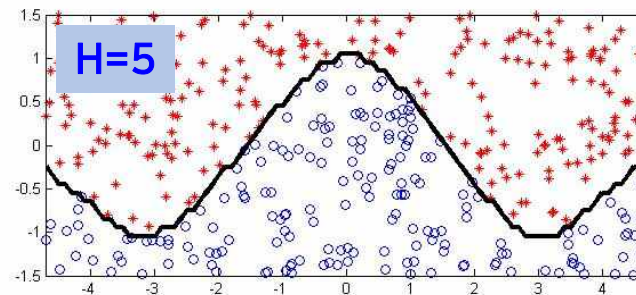
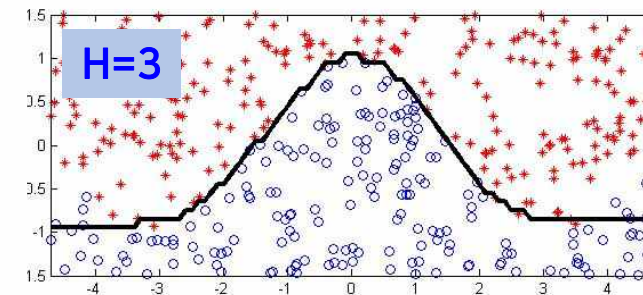
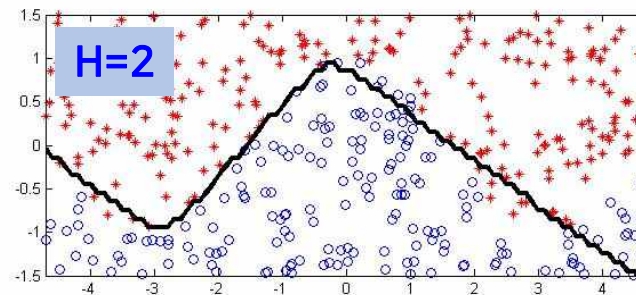
MLP 구조



은닉 뉴런의 개수에 따른 분류오차

은닉 뉴런의 수	2	3	5	10
오분류율	6%	5%	1.5%	1%

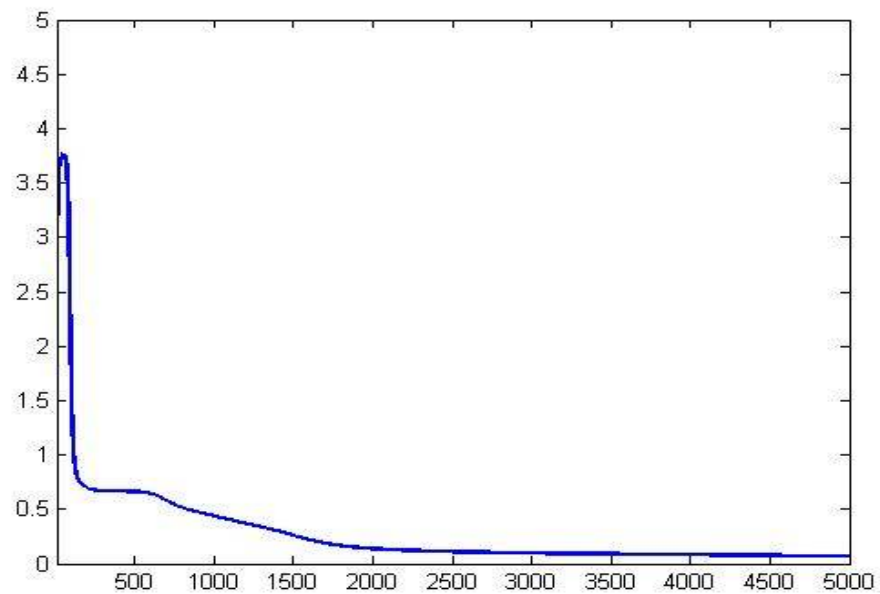
은닉 뉴런의 개수에 따른 결정경계의 변화



MLP를 이용한 간단한 패턴 분류 실험

학습횟수에 따른 학습오차의 변화

학습오차



학습 에포크 수

숫자 인식 digit recognition

3 6 8 1 7 9 6 6 9 1
 6 7 5 7 8 6 3 4 8 5
 2 1 7 9 7 1 2 8 4 6
 4 8 1 9 0 1 8 8 9 4
 7 6 1 8 6 4 1 5 6 0
 7 5 9 2 6 5 8 1 9 7
 2 2 2 2 2 3 4 4 8 0
 0 2 3 8 0 7 3 8 5 7
 0 1 4 6 4 6 0 2 4 3
 7 1 2 8 1 6 9 8 6 1



입력 이미지



2

목표 출력값
(클래스 레이블)

0
0
1
0
0
0
0
0
0
0
0

데이터 준비

필기 숫자인식

- ☐ 패턴 분류의 대표적인 문제
- ☐ 손으로 쓴 숫자 영상을 10개의 클래스로 분류하는 문제

벤치마크 데이터

- ☐ MNIST 데이터 <http://yann.lecun.com/exdb/mnist>
- ☐ 데이터 개수 → 7만 (학습용 6만, 테스트용 1만)
- ☐ 28×28 크기의 흑백 영상
 - ✓ 각 픽셀은 0~255의 명도값을 가짐
- ☐ 각 데이터에 대해 클래스 레이블(0~9)이 함께 주어짐



데이터 셋팅

○ 입력값 전처리: 정규화 normalization

- ☐ 데이터가 가지는 값이 일정 범위 안에 있도록 조정

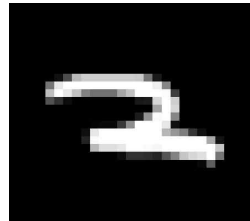
- ## □ 정규화의 필요성

- ✓ 신경세포의 입력값이 크면
세포화가 가능성이 높아짐

- ## ✓ 학습의 어려움

- 0~255 범위의 값을 0~1 범위 값으로 조정

- ✓ $\tilde{x} = (x - 0)/(255 - 0)$

[illegible]

데이터 셋팅

○ 목표 출력값 설정

- ☐ 출력 뉴런의 수 = 클래스 레이블의 수로 설정
- ☐ i 번째 클래스에 속하는 데이터의 출력값
 - ✓ i 번째 출력 뉴런은 1, 나머지는 0의 값으로 설정

목표 출력값

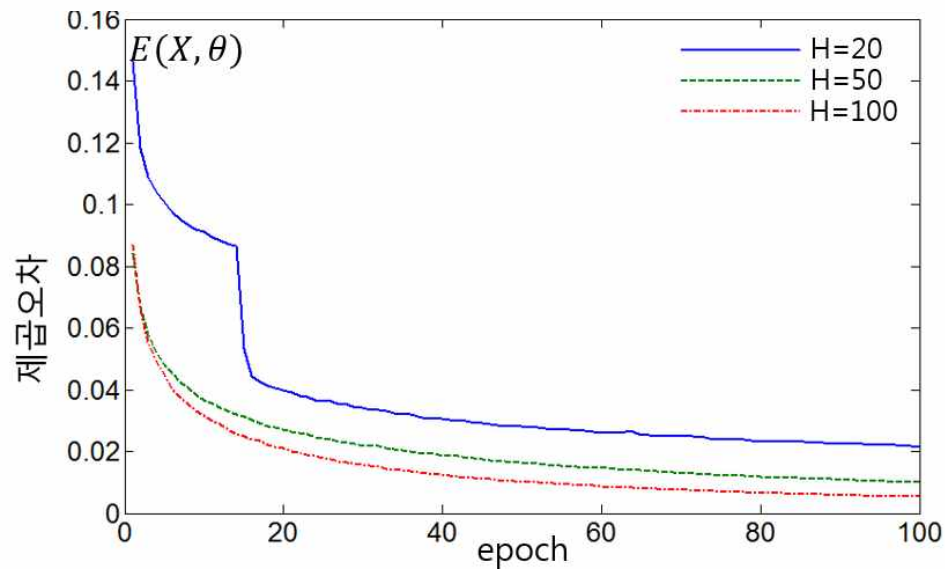
0
0
1
0
0
0
0
0
0
0

○ 데이터 집합의 구성

- ☐ 학습 데이터 → 60,000개
- ☐ 테스트 데이터 → 10,000개

학습 곡선

- 학습 곡선 learning curve을 이용한 학습 상황 관찰
 - 한 에포크 epoch가 끝날 때마다 학습 오차를 계산하여 그 변화를 살펴보는 그래프
- 은닉 노드 수에 따른 학습 성능 비교



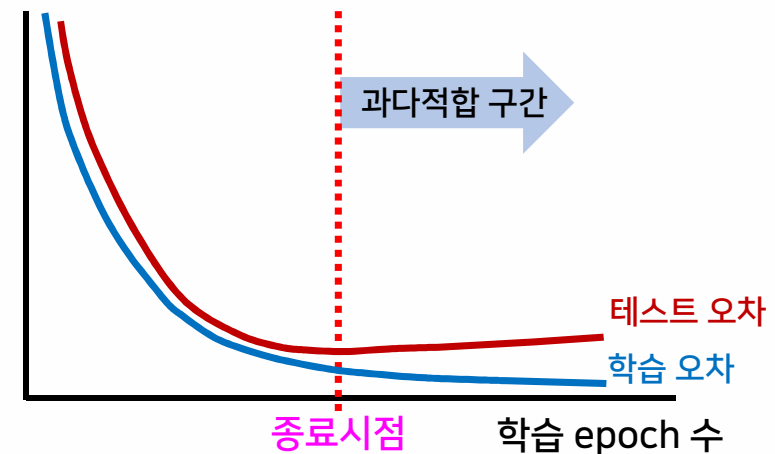
성능 평가

○ 일반화 성능

- 학습에 사용되지 않은 새로운 데이터에 대한 신경망 출력의 정확도
 - ✓ 평가 방법 → 테스트 데이터 집합을 별도로 수집하여 오차 계산

○ 학습 오차와 테스트 오차의 관계

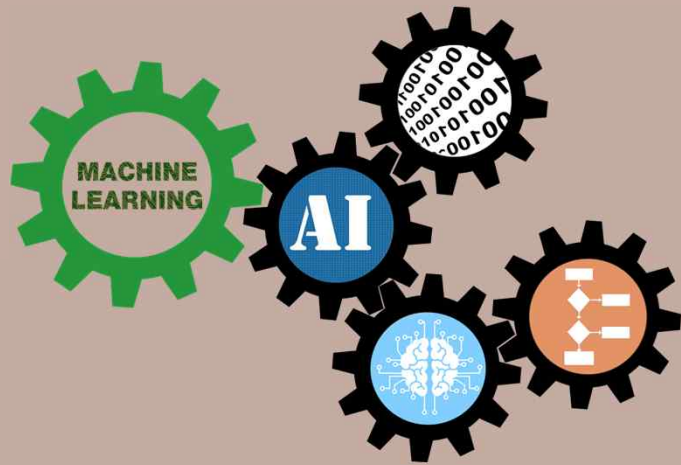
- 학습이 진행되면 학습 오차는 계속 감소,
테스트 오차는 일정 시점에서 다시 증가
→ 과다적합



성능 평가

- 오차함수와 은닉 노드 수에 따른 성능(오분류율) 변화

활성화 함수	오차함수	H=20	H=50	H=100
시그모이드	제곱오차	4.51	3.17	2.49
소프트맥스	교차엔트로피	4.69	3.07	2.35



다음시간안내

제11강

딥러닝 (1)