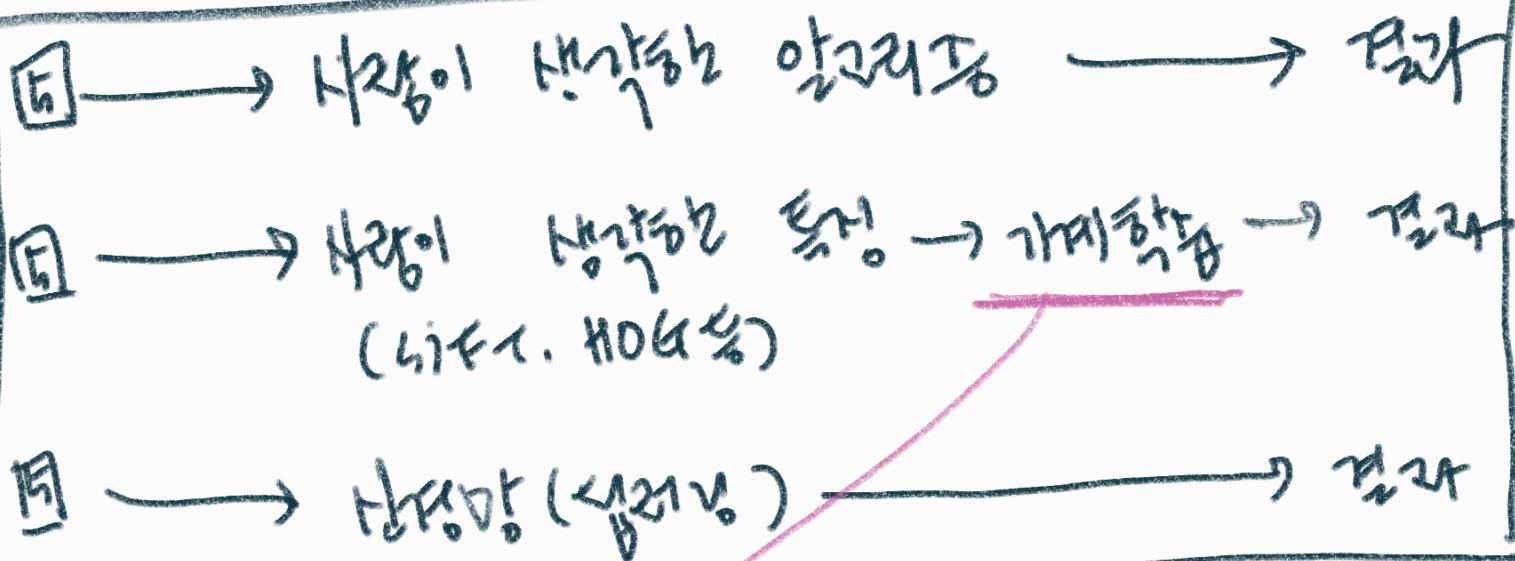


230331(금)

w, b 파라미터를 자동으로 학습하는
방법.

chapter 4. 신경망 학습

< 두 가지 학습 방법 예시 >



* 딥러닝

; end-to-end machine learning

; 사이킷 라이브러리 ~ 결과 출력 No, 사람 X

• 훈련데이터와 시험 데이터 나누는 이유

; 성능 평가 위해

; 오버피팅 피하기

$$E_{\text{full}} = \text{Const}$$

• 손실함수 (loss function)

→ 학습률의 높은 차이를 줄이기 위한 하위의 기준

w^2 로 찾기 위한 풍태, 오류 확률 $\downarrow \rightarrow$ 비용

→ 광장-평균제곱오차, 고차에 대한 오차 등

① 평균제곱오차 (Mean Square error; MSE)

(예시)

$$X = [1, 2, 3]$$

$$Y = [3, 5, 7] \quad) \text{이 데이터를 학습해보면, } \\ X=4 \text{ 일 때, } Y \text{ 값 구할 수 있어야} \\ (\text{예: } f(X) = 2X + 1)$$

$$H(w_1, b) = w_1x + b \quad (\text{목표: } w=2, b=1)$$

가중치(H) 추적

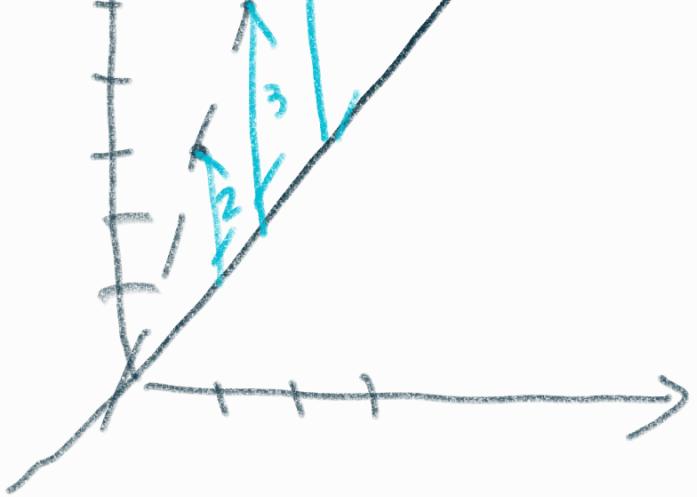
$$w=1$$

$$(b=0) \Rightarrow \text{점수} \rightarrow 3+5+7=15 \quad (3/0)$$

최대제곱을 통해 얼마나 잘못되었는지

목표

$$H(w, b) = \frac{2^2 + 3^2 + 4^2}{3} = \frac{29}{3}$$



가설이 틀렸어
모델이 틀렸어.

$$\text{Cost}(w, b) = \frac{1}{m} \times \sum_{i=1}^m \frac{(w_i x_i + b - y_i)^2}{\text{가설}} = \frac{29}{3}$$

※ 절대값 사용하지 않고, 제곱하는 이유
- 오류값 정복 위해서

- 절대값은 계산하기도, 고민을 활용하기도
쉽지 않아 늘어놓는다.

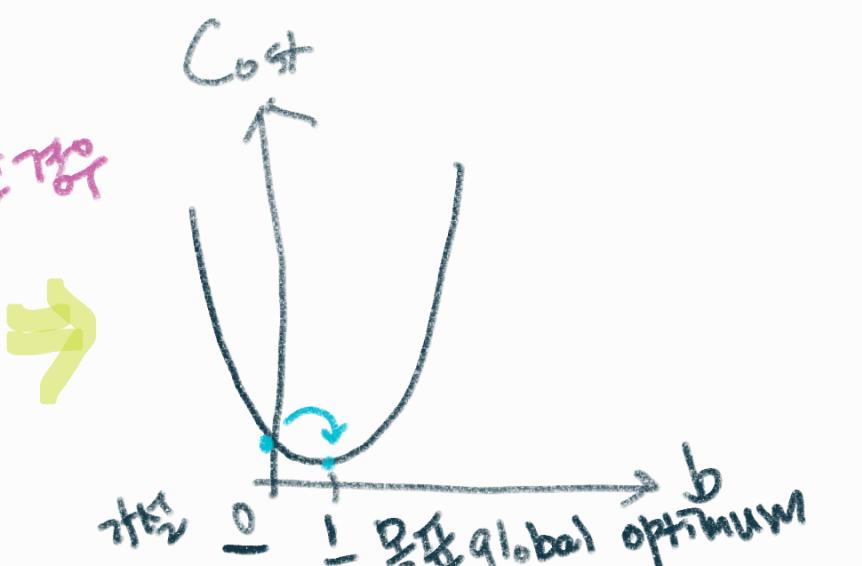
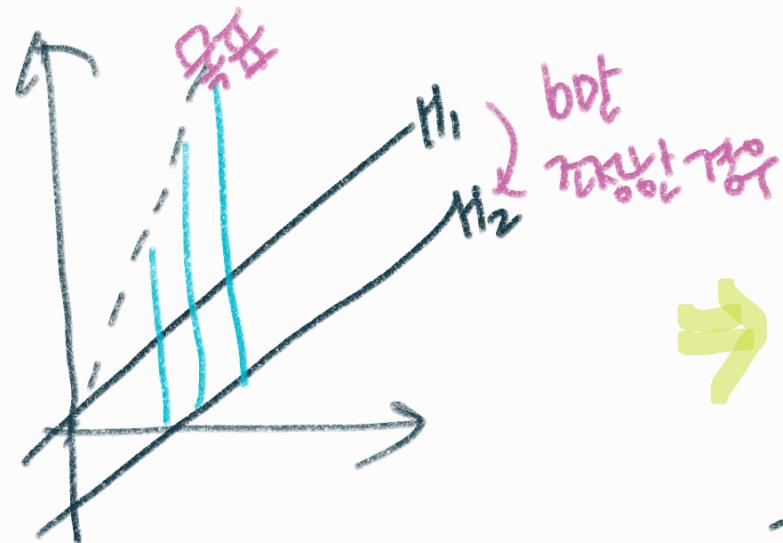
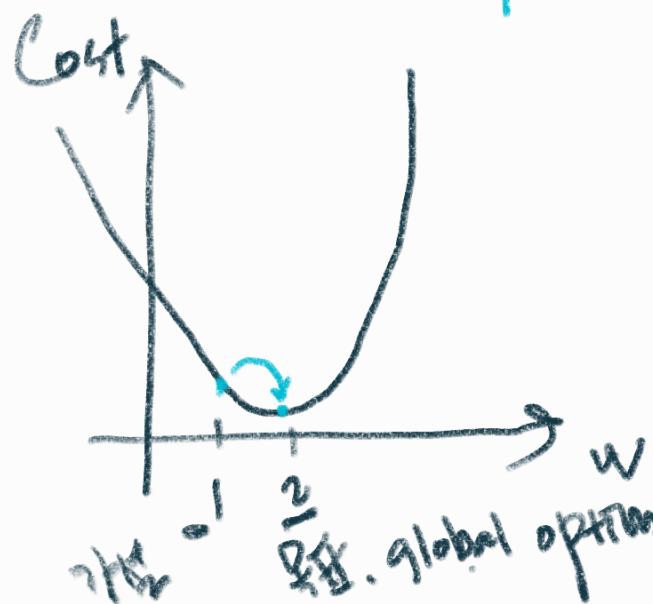
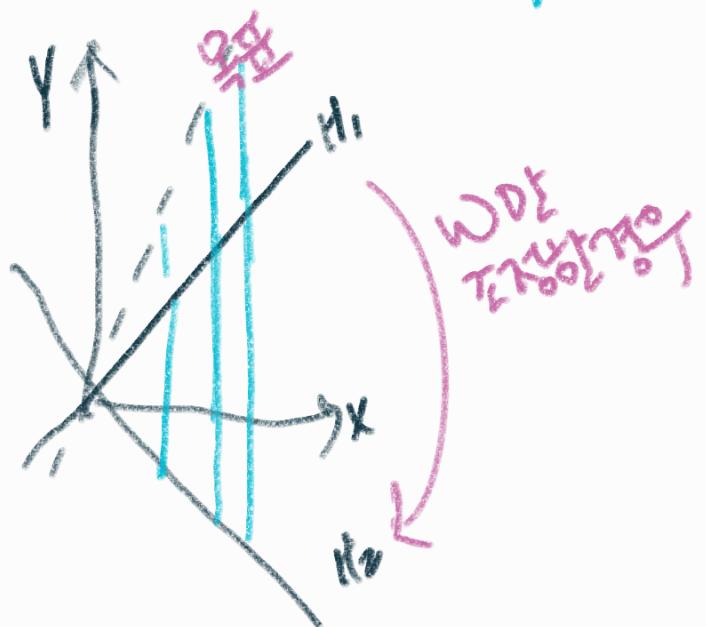
제곱한 후 더한 후 평균을 할 때까지,
 $\min_w \min_b \sum_{i=1}^m \frac{(w_i x_i + b - y_i)^2}{\text{가설}} = \frac{\text{목표}}{\text{목표}}$

$$\text{Cost}(w, b) = \frac{1}{m} \times \sum_{i=1}^m (w_i x_i + b - y_i)^2$$

()
가설 목표

W, b를 최소화해야 하므로,
데이터 개수 n을 크게 잇이 X

W와 b가 같은 경우 Cost는 기하급수적으로 증가



하지만 W와 b 둘 다 같아야!!

Cost는 어떤지 알지 못함? 두 경우가?

- ⇒ W와 b 둘 다 기울기가 0인 지점 같아야!!!
- ⇒ 기울기가 양수이면 오른쪽에 (+)
(양수) 왼 (-)

⇒ 경사하강법

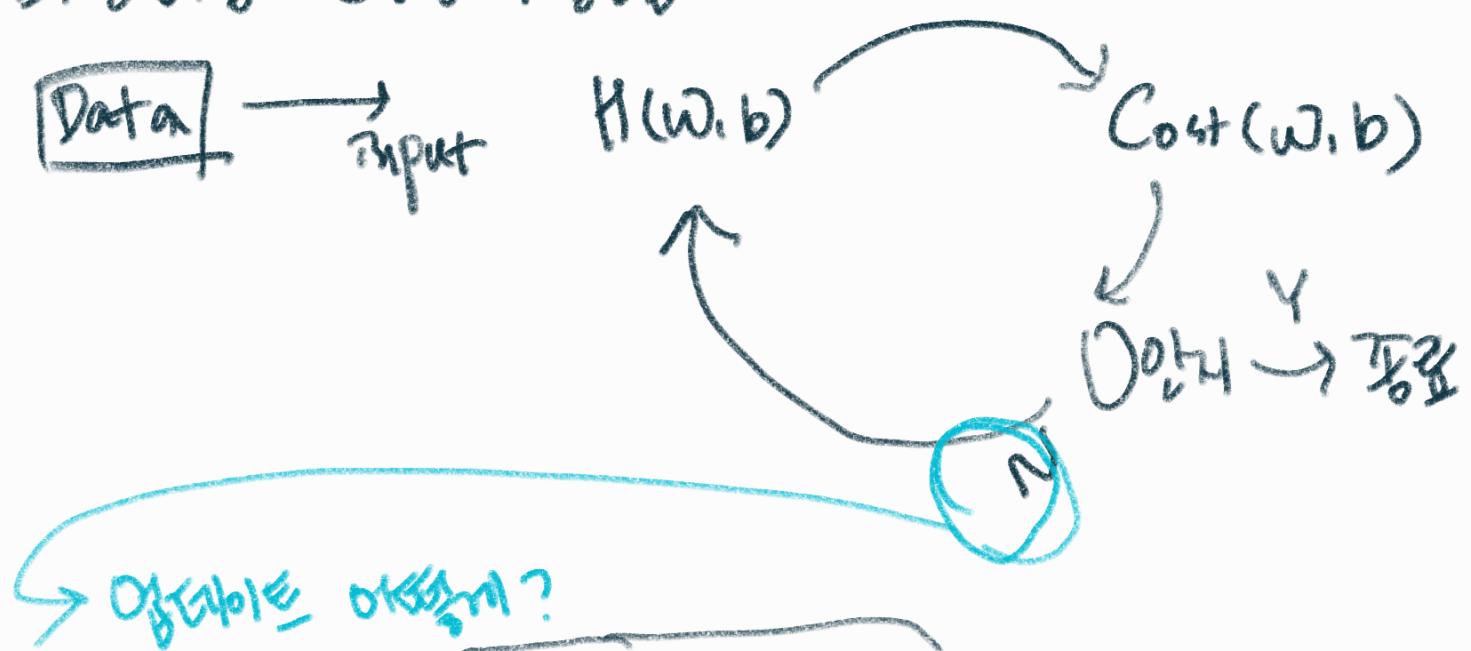
; Cost를 줄이기 위해서 반복적으로 기울기를

계산해 변수의 값을 업데이트해 가는 과정

기울기를 어떻게 구하나?

; w와 b에 대해 각각 평균을 수행

머신러닝 정상화 방법



$$w = w - \frac{\partial}{\partial w} \text{Cost}(w, b)$$

$$b = b - \frac{\partial}{\partial b} \text{Cost}(w, b)$$

W와 b에 각각을 평균화한 값은 바로 그

제대로 수렴하기 훨씬 빠르게 0.0001 까지

값을 넣어놓고 학습이 잘
이려면 업데이트로 1000번 이상 반복
global optimization을 찾을 때 까지

이해가 안되는 장표... 왜 3bit와 3.3bit가 나오는 거지??

026 Entropy

www.cs.rochester.edu/u/james/CSC248/Lec6.pdf

"Classification의 loss function으로는 cross entropy를 주로 사용합니다."
"두 분포 차이를 줄이기 위해 KL-divergence를 최소화 시킵니다."

- **Entropy** : 정보를 최적으로 인코딩하기 위해 필요한 bit의 수

ex) 오늘이 무슨 요일인지 bit로 전송한다. → 3bit 필요
월: 000, 화: 001, 수: 010, 목: 011, 금: 100, 토: 101, 일: 110 $\log_2 N$

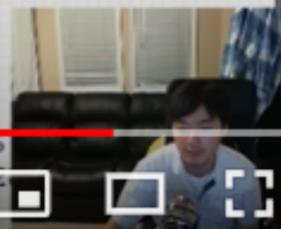
- 만약 각각의 발생 확률이 다르다면?

ex) 40개의 문자 (A,B,C,D,...,Z,1,2,3,...14)를 bit로 전송한다. $\log_2 40 = 5.3$

그런데 A,B,C,D가 전체의 22.5%씩 전체 90% 확률로 발생한다

1st bit : A,B,C,D 인지 아닌지 YES → 추가로 2bit 더 필요 (ABCD 구분)
NO → 추가로 6bit 더 필요 ($\log_2 36$)

필요한 bit 수 = $0.9 \times 3 + 0.1 \times 6 = 3.3$ bit!



- Entropy는 각 label들의 확률분포의 함수!

$$H(y) = -\sum y_i \log \frac{1}{8/16} = 9.48$$

$$\begin{aligned} H &= -4 * (0.225 \\ &\quad - 36 * (0.00 \\ &= 2.2 \text{ bit} \end{aligned}$$

② 교차 엔트로피 오류 (CEE,
cross entropy error)

• 엔트로피: 무질서도, 다양성에 대한 측정

• CEE

$$E = - \sum_i t_i \cdot \log(y_i)$$

정답

예측값

$$t = [0, 0, 1, 0, 0, 0, 0, 0]$$

$$y_{01} = [1, 0.05, 0.1, 0, 0.05, 0.1, 0.0, 0.0]$$

$$y_{04} = [1, 0.05, 0.1, 0, 0.05, 0.0, 0.0, 0.0]$$

$\Rightarrow t$ 가 0인 경우, y_{01} 은 모든 값이 0이다

$\Rightarrow t$ 가 1인 경우에도 결과 예측값에 영향을 미치지

$\therefore y_{01}$ 과 y_{04} 의 결과 예측값은 같음

$\Rightarrow 0 \times y_{01}$ 은 계산하지 않고 바로 계산량 ↓ 시간 ↓