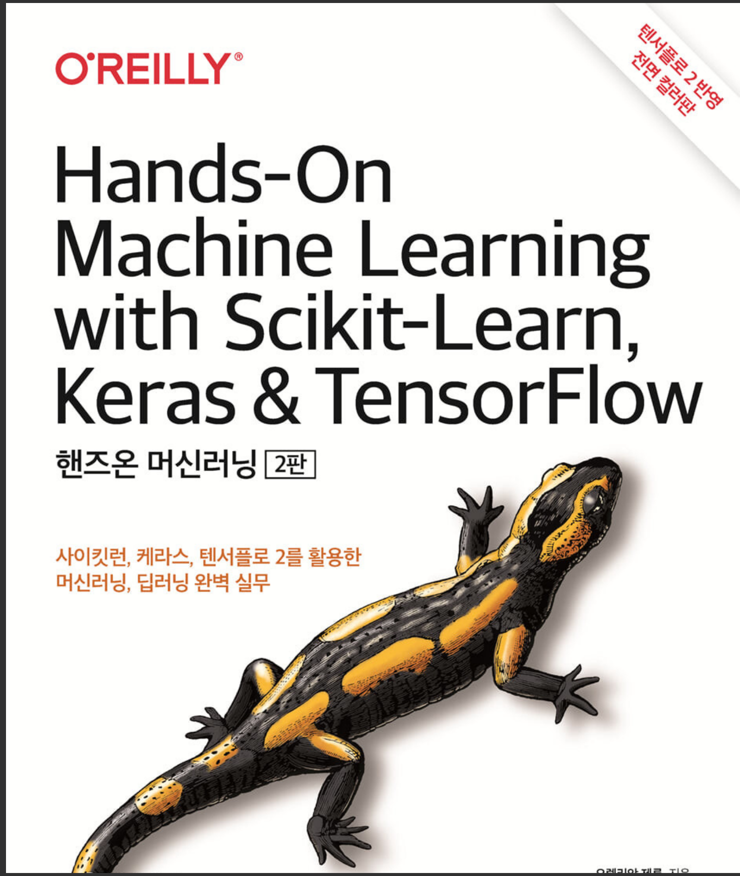
4장: 모델 훈련

4.2 경사하강법

4.3 다항회귀

4.6 로지스틱 회귀



4.2 경사하강법

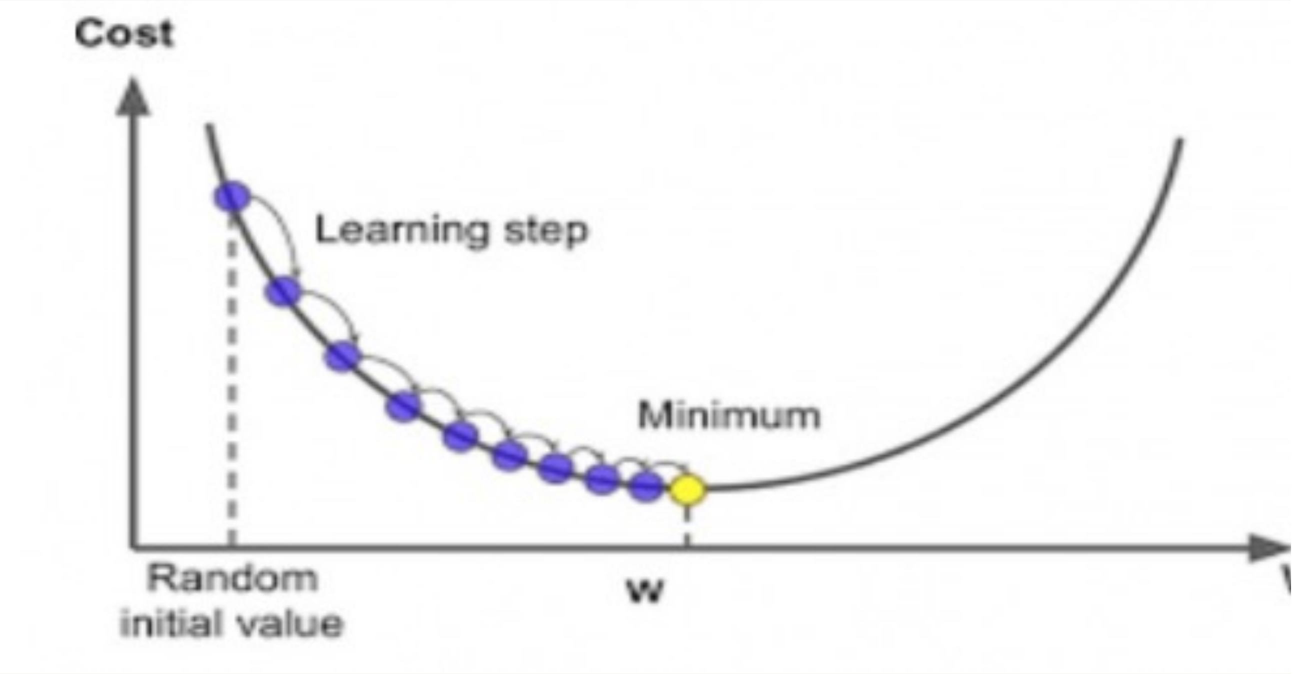
옵티마이저

손실함수의 크기를 줄이는 것이 목표

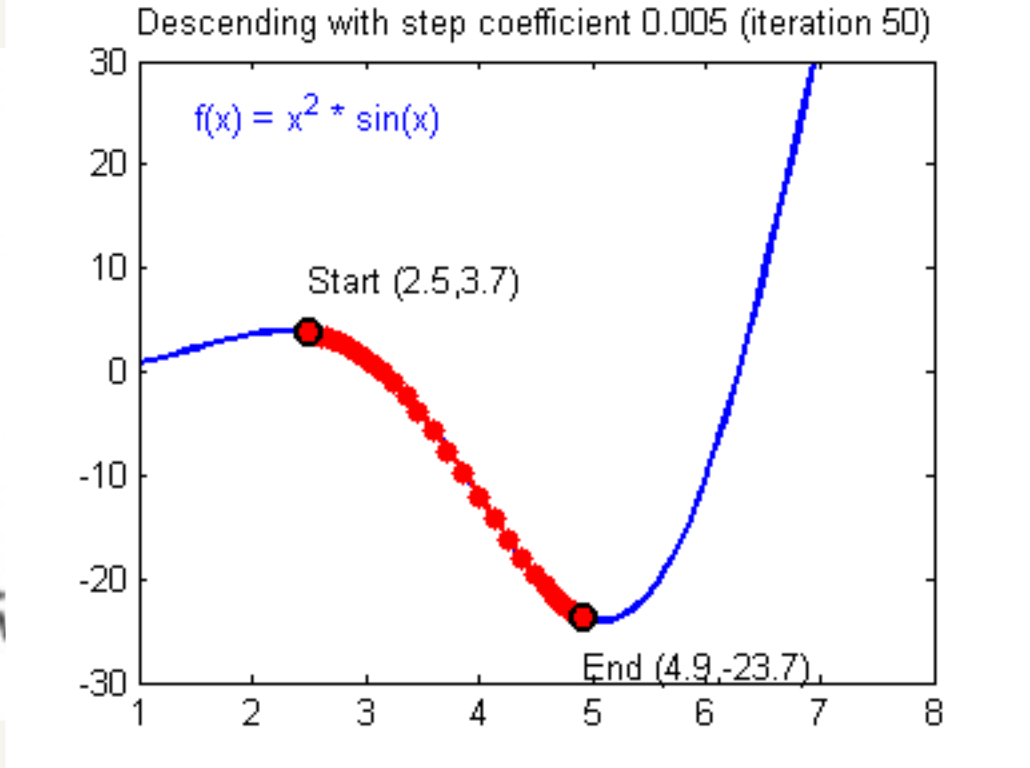
- 어떻게 줄일 것인가? -> “옵티마이저”로 줄인다.

경사하강법은 곧 손실함수의 크기를 줄이는 **옵티마이저**의 일종

경사(기울기)를 구한다.

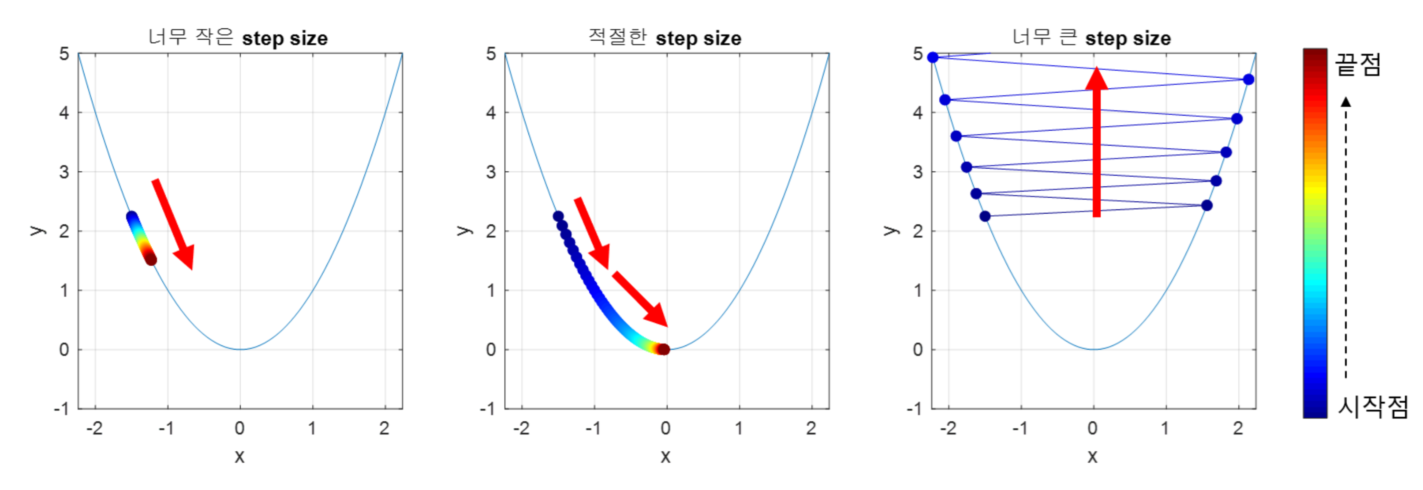


Cost
Learning step
Minimum
Random
initial value
w



Descending with step coefficient 0.005 (iteration 50)
f(x) = x2 * sin(x)
Start (2.5,3.7)
End (4.9,-23.7)
30
20
10
0
-10
-20
-30
1
2
3
4
5
6
7
8

학습률(스텝의 크기)



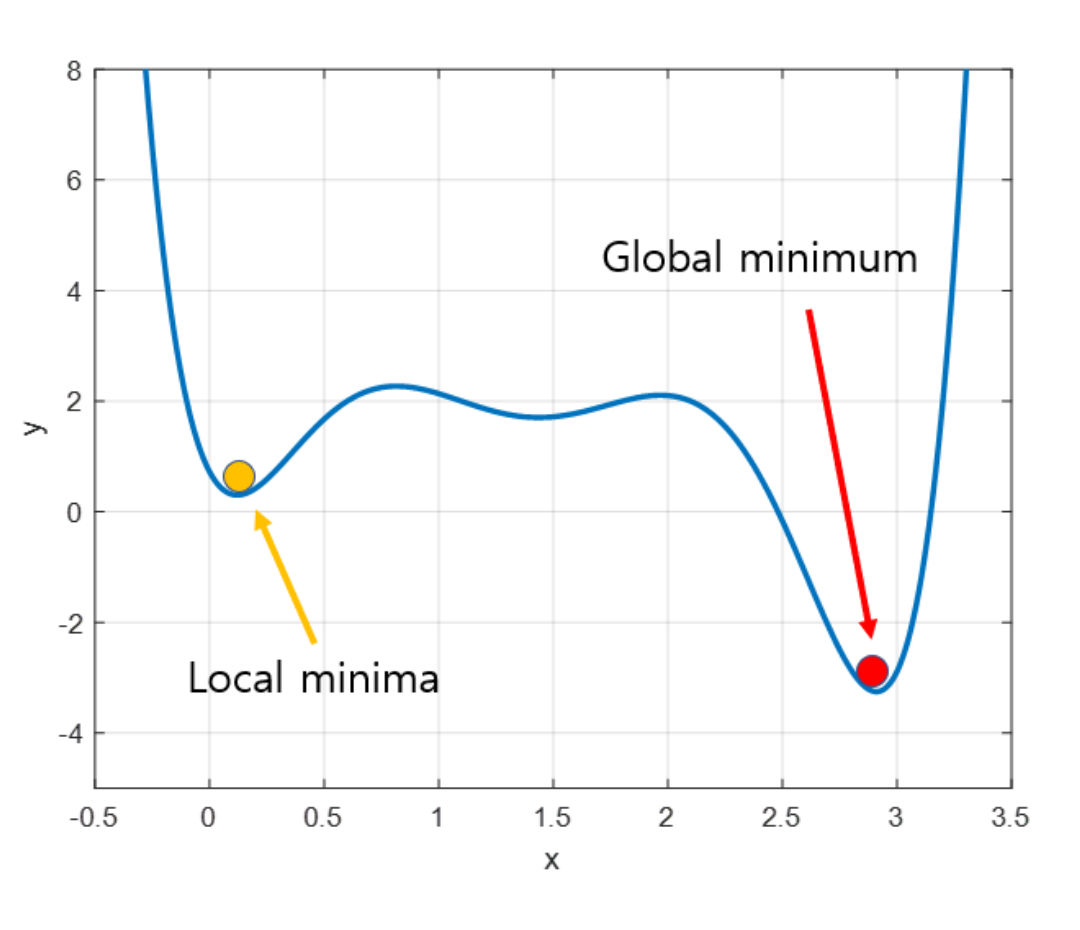
학습률이 너무 작다

-> 시간이 너무 오래 걸림

학습률이 너무 크다

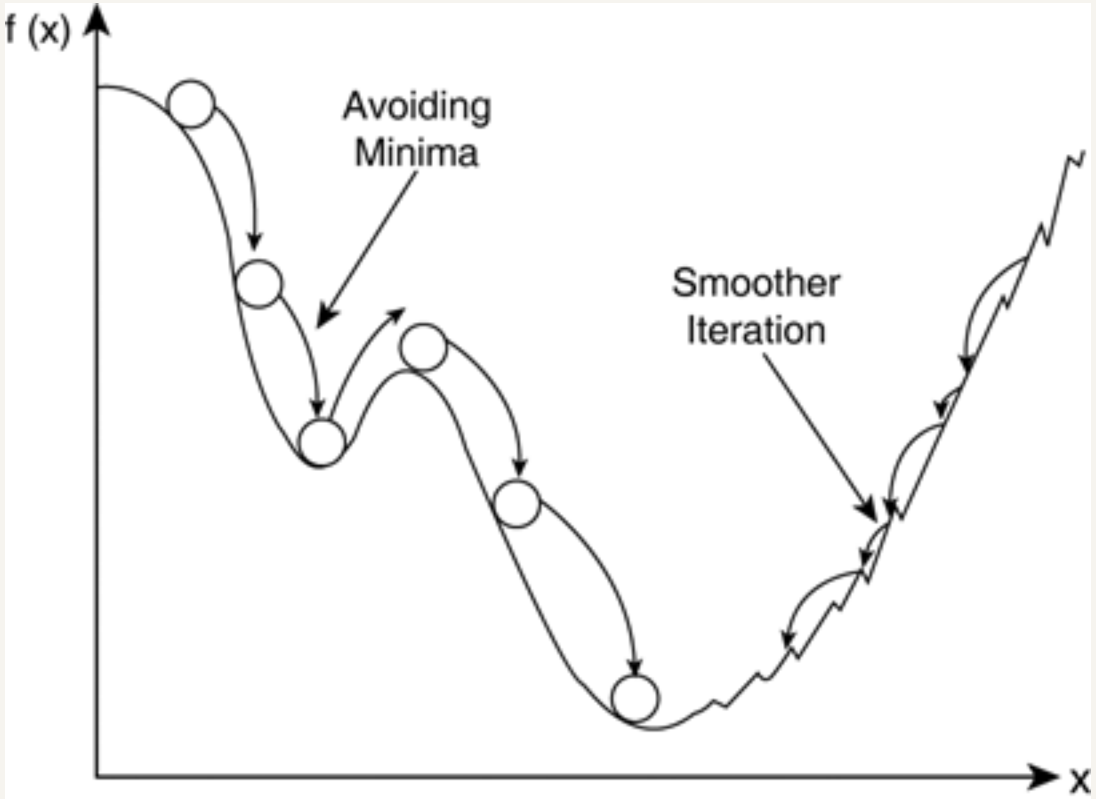
-> 이상한 곳으로 발산 가능

경사하강법의 문제점



“무작위 초기화”로 인하여
지역 최소값에 수렴할 수 있다.

모멘텀 (Momenetum)

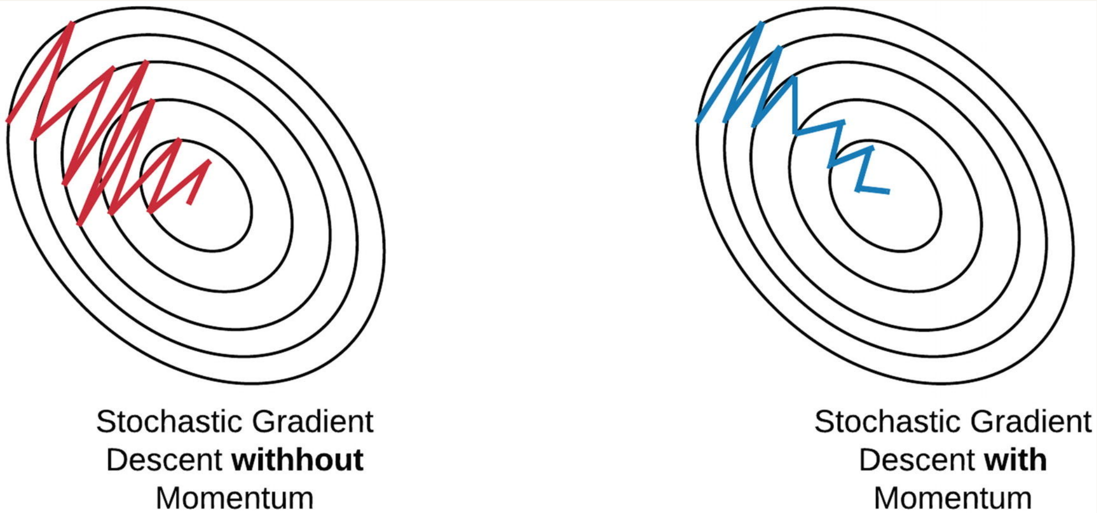


$$w_i^{k+1} = w_i^k + \eta\left(-\frac{d\epsilon}{dw_i^k}\right) + \alpha\left(\Delta w_i^{k-1}\right)$$

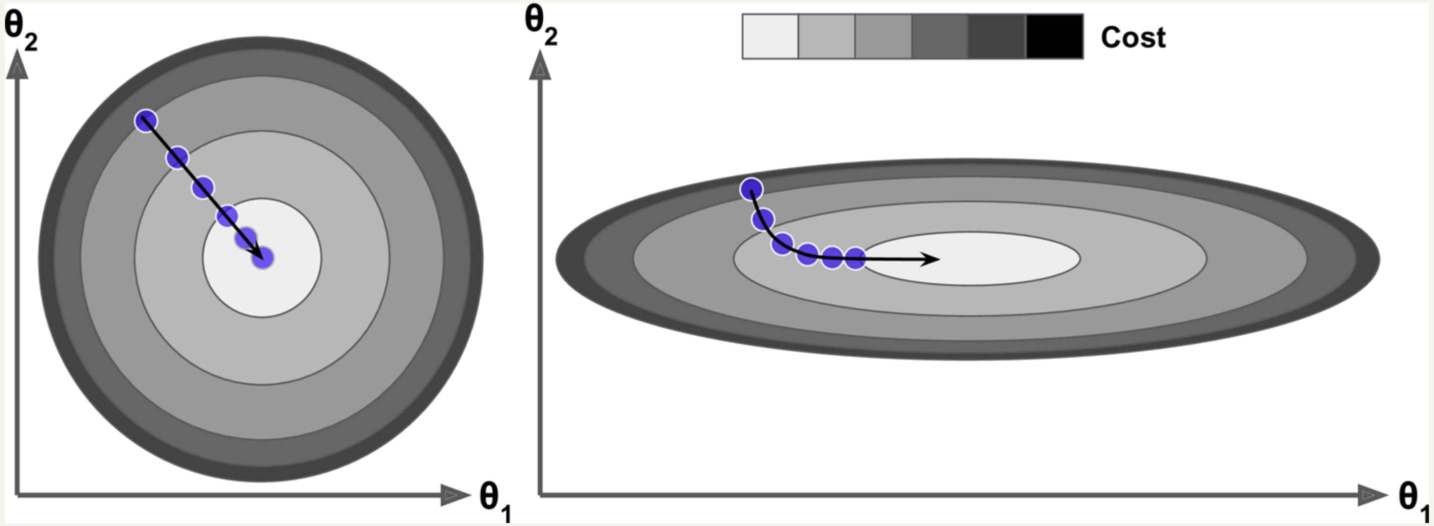
α : Momentum term

모멘텀 + SGD

지그재그 현상을 줄여준다.



경사하강법의 문제점



- 변수들의 스케일이 매우 다를 경우 발생한다.

사이킷 런의 StandardScaler 활용

4.2.1 배치 경사 하강법

편도함수

$$\frac{\partial}{\partial \theta_j} \mathrm{MSE}(\boldsymbol{\theta}) = \frac{2}{m} \sum_{i=1}^{m} \left(\boldsymbol{\theta}^T \mathbf{x}^{(i)} - y^{(i)} \right) x_j^{(i)}$$

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - t_i)^2$$

θ_j에 대해 편미분..

- θ_j가 변함에 따라 **비용함수가 변하는 정도 -> 편도함수**

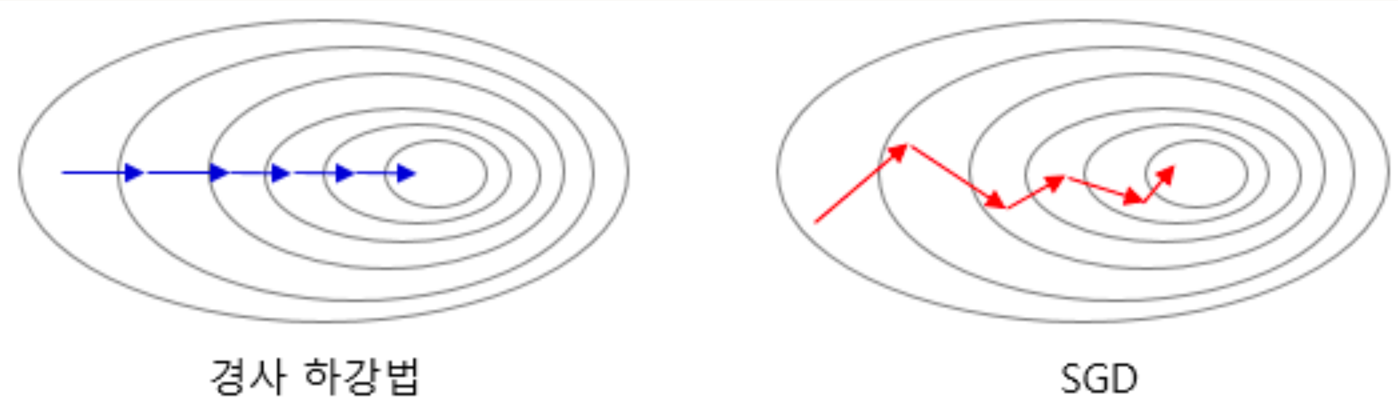
$$\boldsymbol{\theta}^{(\text{next step})} = \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} \mathrm{MSE}(\boldsymbol{\theta})$$

벡터화

$$\nabla_{\boldsymbol{\theta}} \mathrm{MSE}(\boldsymbol{\theta}) = \begin{pmatrix} \frac{\partial}{\partial \theta_0} \mathrm{MSE}(\boldsymbol{\theta}) \\ \frac{\partial}{\partial \theta_1} \mathrm{MSE}(\boldsymbol{\theta}) \\ \vdots \\ \frac{\partial}{\partial \theta_n} \mathrm{MSE}(\boldsymbol{\theta}) \end{pmatrix} = \frac{2}{m} \mathbf{X}^T (\mathbf{X}\boldsymbol{\theta} - \mathbf{y})$$

4.2.2 확률적 경사 하강법

- 매 스텝에서 <u>**한 개의 샘플을 무작위로 선택**</u>한다.

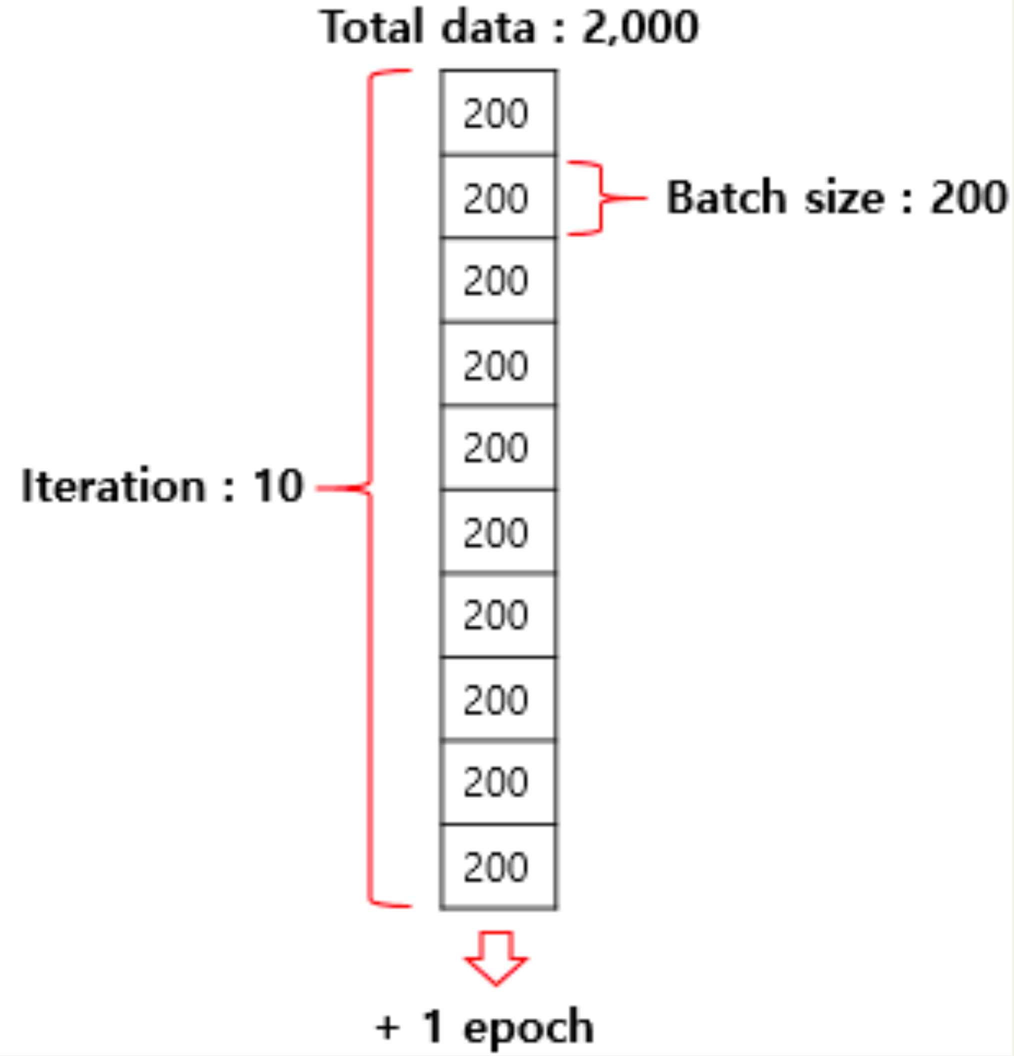


에포크와 배치 크기와 이터레이션
(Epochs and Batch size and Iteration)

에포크: 인공 신경망에서 전체 데이터에 대해서 순전파와 역전파가 끝난 상태

배치 크기: 몇 개의 데이터 단위로 매개변수를 업데이트 하는지

이터레이션(스텝): 한 번의 에포크를 끝내기 위해서 필요한 배치의 수



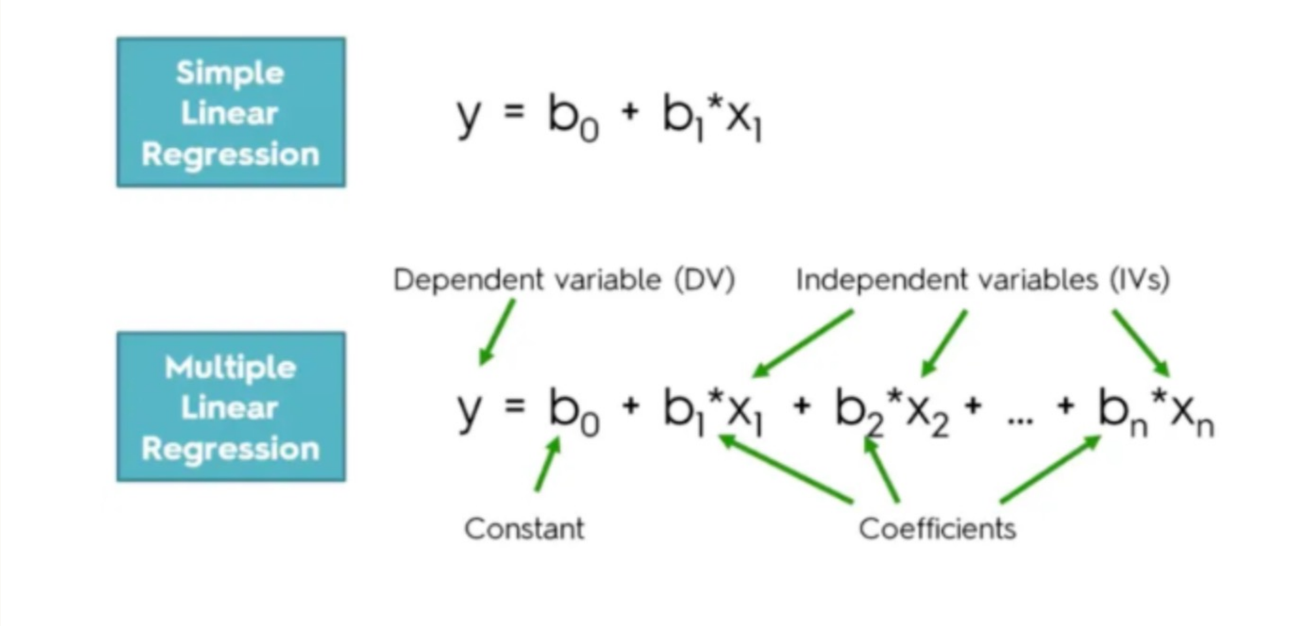
4.2.3 미니배치 경사 하강법

- **“미니배치”**에 대해 그레디언트를 계산한다.
- 미니배치: 임의의 작은 샘플 세트

4.3 다항회귀

VS 다중회귀

다항회귀



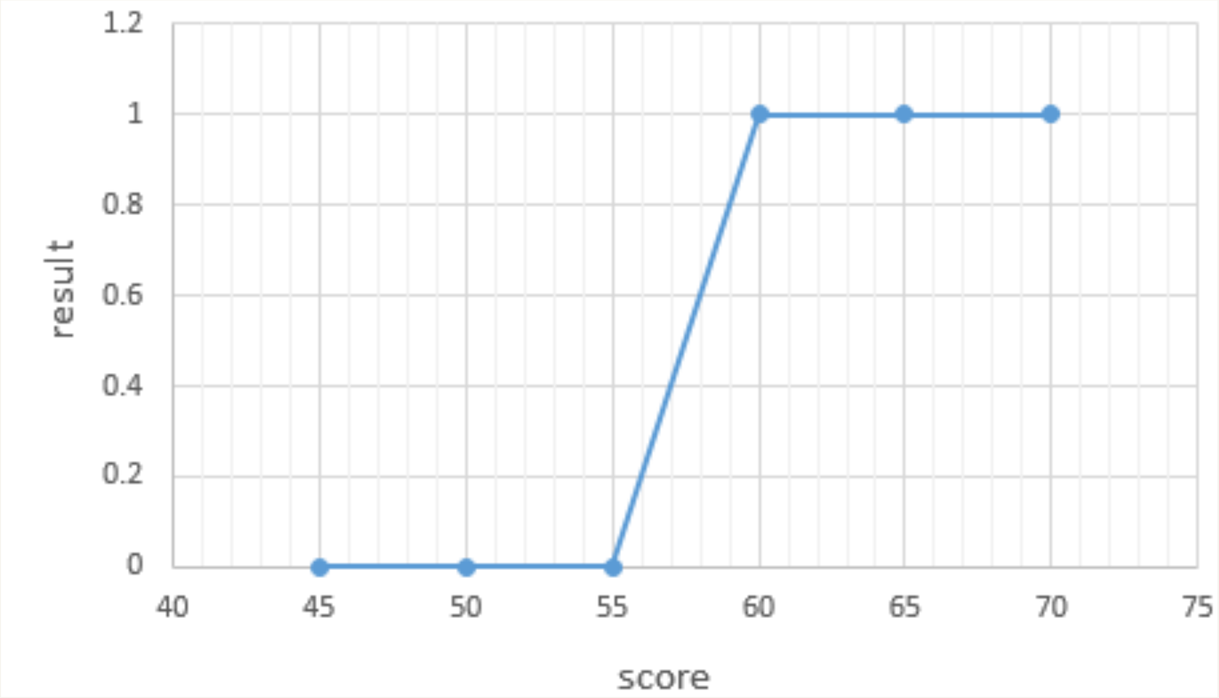
다중회귀

$$y = w_0 + w_1 x + w_2 x^2 + \cdots + w_d x^d$$

4.6 로지스틱 회귀

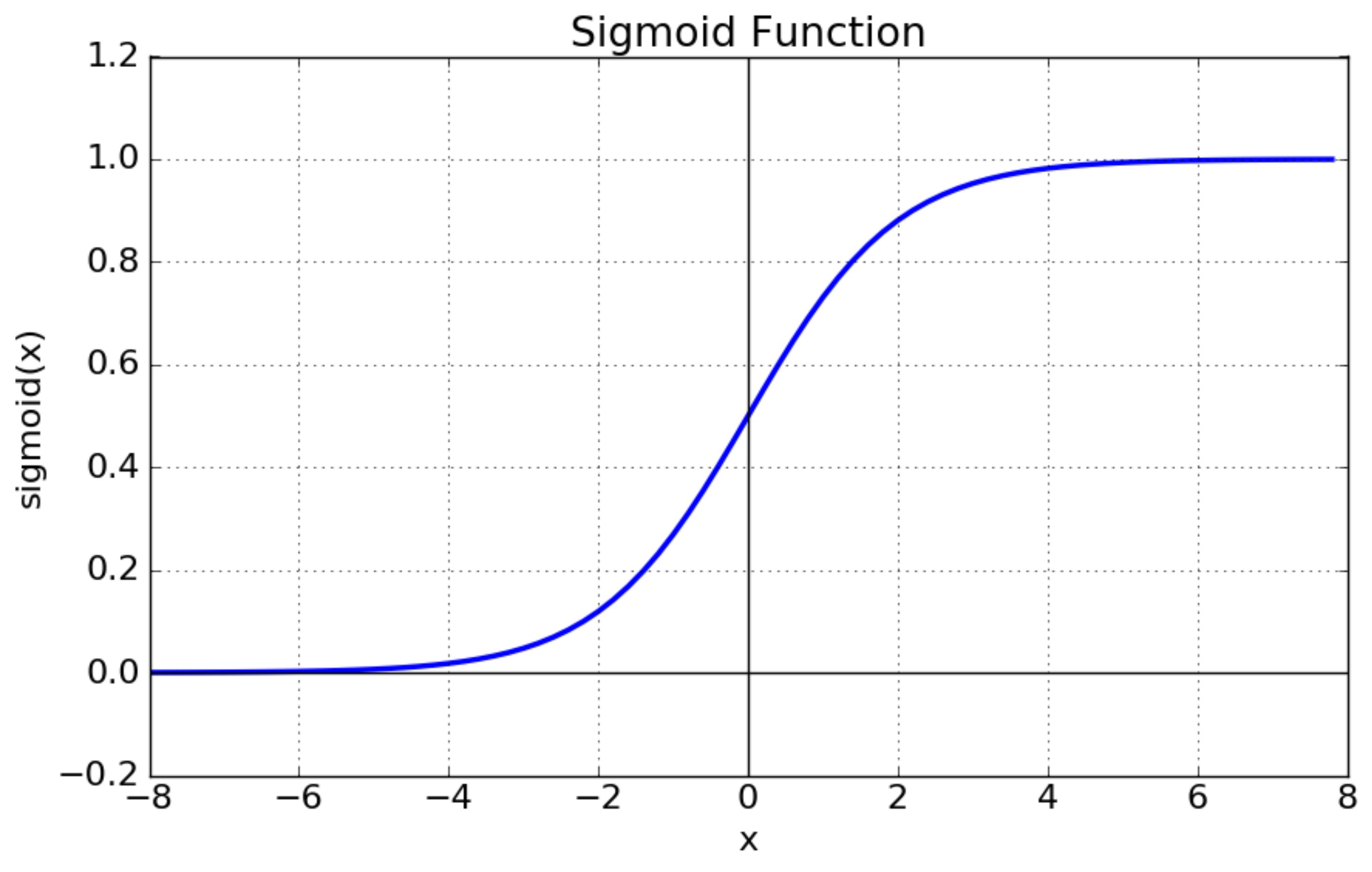
이진 분류

score(x)	result(y)
45	불합격
50	불합격
55	불합격
60	합격
65	합격
70	합격



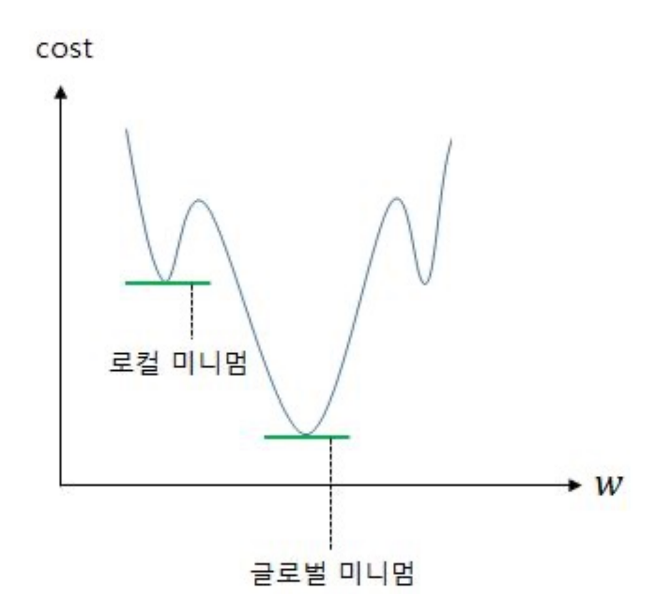
시그모이드 함수

$$H(x) = \frac{1}{1 + e^{-(wx+b)}} = sigmoid(wx + b) = \sigma(wx + b)$$



- 출력이 0과 1 사이의 값을 가지면서 S자 형태로 그려지는 함수
- w(weight), b(bias)

로지스틱 회귀의 비용함수



$$\text{if } y = 1 \rightarrow \text{cost}\ (H(x), y) = -\log(H(x))$$

$$\text{if } y = 0 \rightarrow \text{cost}\ (H(x), y) = -\log(1 - H(x))$$

$$\text{cost}\ (H(x), y) = -[ylogH(x) + (1 - y)log(1 - H(x))]$$

$$J(w) = -\frac{1}{n}\sum_{i=1}^{n}[y^{(i)}logH(x^{(i)}) + (1 - y^{(i)})log(1 - H(x^{(i)}))]$$

- MSE를 사용하지 않는다.
- 로그함수를 사용한다.

결론

- 로지스틱 회귀는 비용 함수로 크로스 엔트로피를 사용한다.
- 가중치를 찾기 위해서 크로스 엔트로피 함수의 평균을 취한 함수를 사용한다.

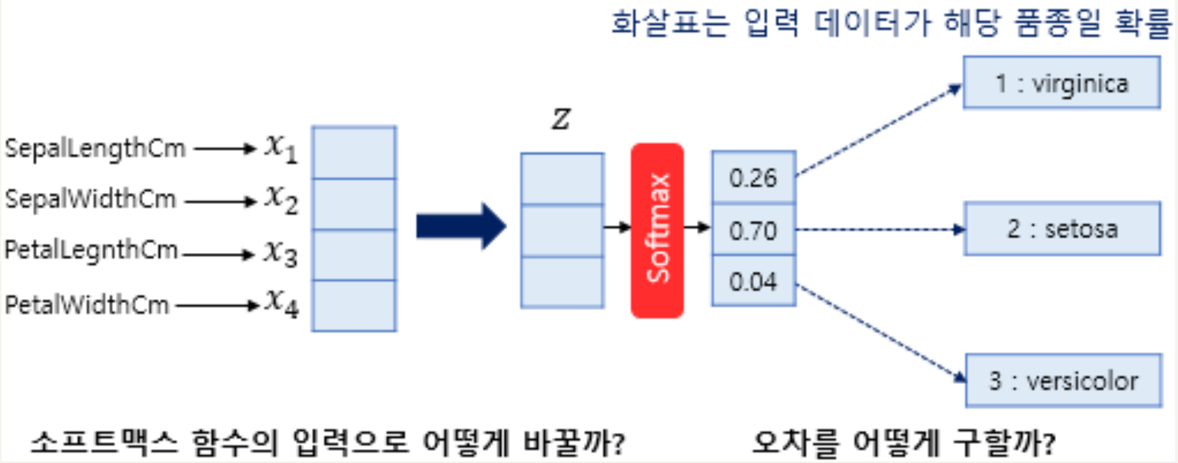
4.6.4
소프트맥스 회귀

다항 로지스틱 회귀라고도..

- 로지스틱 회귀 모델을 다중 클래스를 지원하도록 일반화

SepalLengthCm(x_1)	SepalWidthCm(x_2)	PetalLengthCm(x_3)	PetalWidthCm(x_4)	Species(y)
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
5.8	2.6	4.0	1.2	versicolor
6.7	3.0	5.2	2.3	virginica
5.6	2.8	4.9	2.0	virginica

소프트맥스 함수



$$p_i = \frac{e^{z_i}}{\sum_{j=1}^{k} e^{z_j}} \quad for\ i = 1, 2, \ldots k$$

$$softmax(z) = [\frac{e^{z_1}}{\sum_{j=1}^{3} e^{z_j}} \frac{e^{z_2}}{\sum_{j=1}^{3} e^{z_j}} \frac{e^{z_3}}{\sum_{j=1}^{3} e^{z_j}}] = [p_1, p_2, p_3] = \hat{y} = \text{예측값}$$

$$softmax(z) = [\frac{e^{z_1}}{\sum_{j=1}^{3} e^{z_j}} \frac{e^{z_2}}{\sum_{j=1}^{3} e^{z_j}} \frac{e^{z_3}}{\sum_{j=1}^{3} e^{z_j}}] = [p_1, p_2, p_3] = [p_{virginica}, p_{setosa}, p_{versicolor}]$$

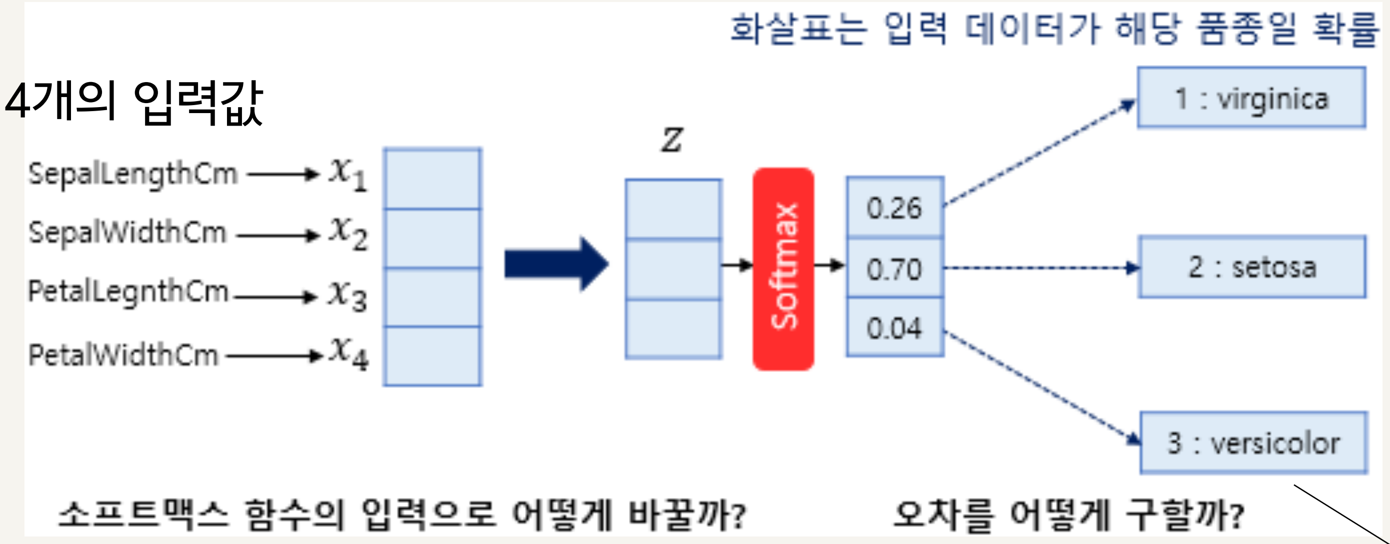
크로스 엔트로피

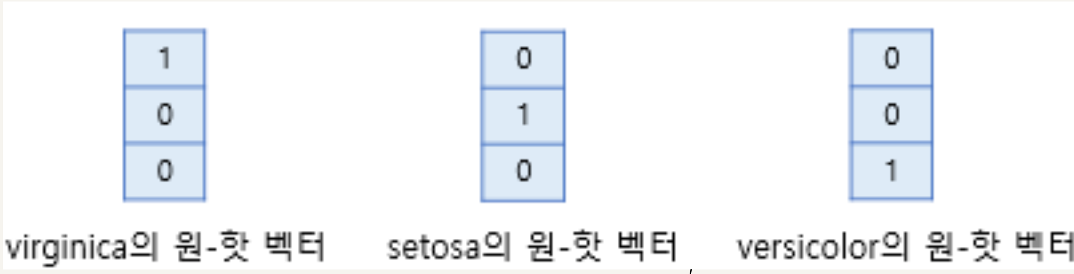
- 비용함수

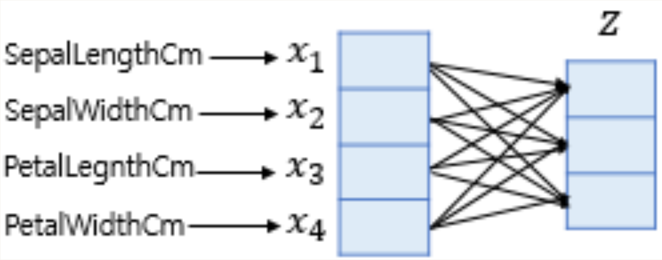
$$softmax(x) = \frac{x_i}{\sum_{j=0}^{k} e^{x_j}} \quad (i = 0, 1, \ldots k)$$

K = 2라면 로지스틱 회귀의 비용함수가 됩니다.

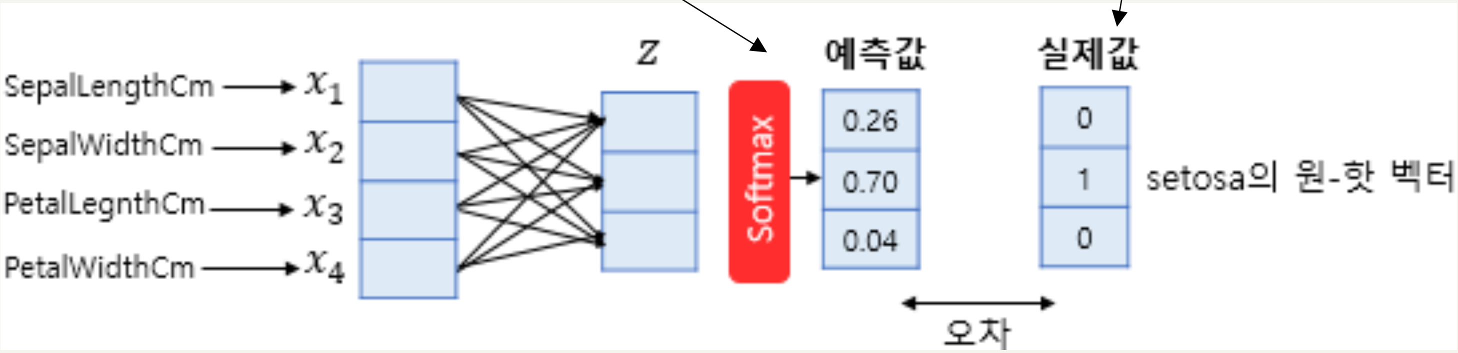
4개의 입력값

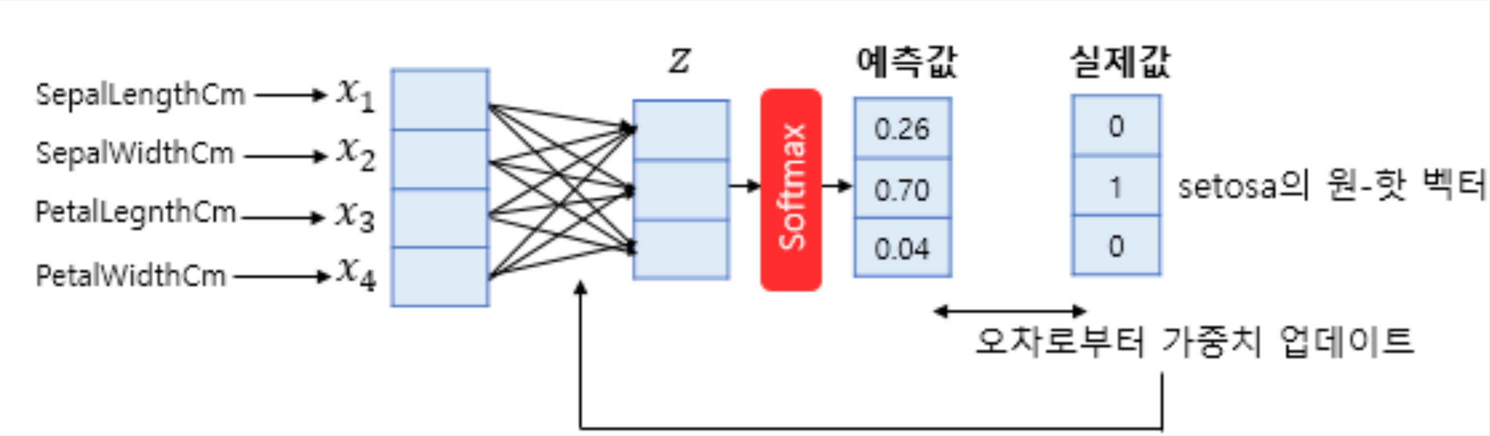


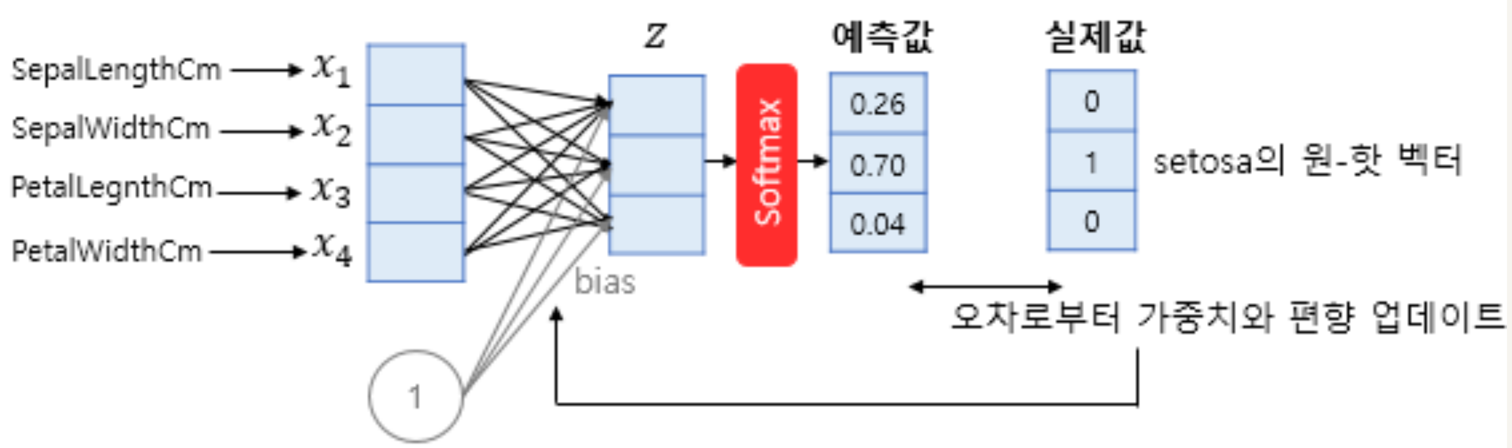


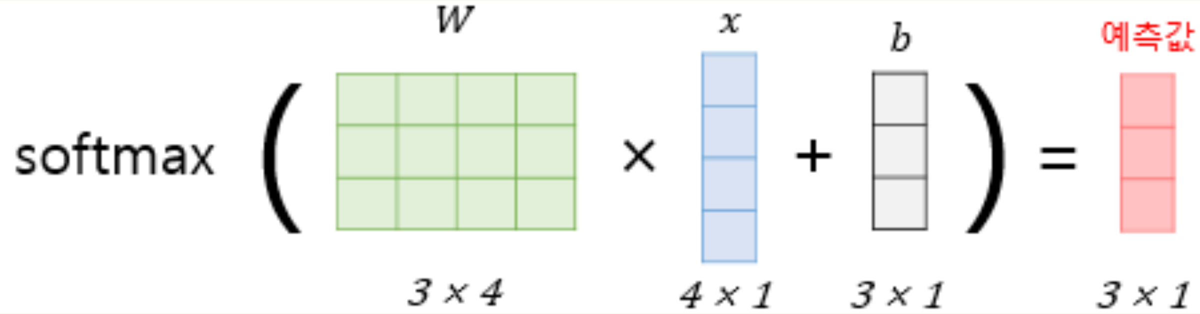


3차원 벡터로 변환




SepalLengthCm → x_1
SepalWidthCm → x_2
PetalLegnthCm → x_3
PetalWidthCm → x_4
Z
Softmax
예측값
0.26
0.70
0.04
실제값
0
1
0
setosa의 원-핫 벡터
오차로부터 가중치 업데이트


SepalLengthCm → x_1
SepalWidthCm → x_2
PetalLegnthCm → x_3
PetalWidthCm → x_4
1
bias
Z
Softmax
예측값
0.26
0.70
0.04
실제값
0
1
0
setosa의 원-핫 벡터
오차로부터 가중치와 편향 업데이트


softmax
W
3×4
x
4×1
b
3×1
예측값
3×1

감사합니다.