



# 배치 정규화(Batch Normalization)

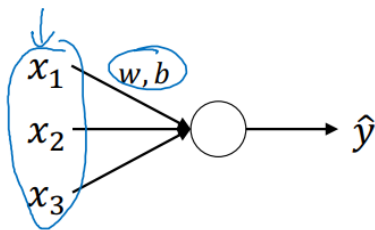
## 배치 정규화

하이퍼파라미터를 빨리 찾을 수 있게 하고, 신경망과 하이퍼파라미터의 상관관계를 줄여주어 더 넓은 범위의 하이퍼파라미터가 잘 동작하게 한다. 아주 깊은 신경망도 잘 작동하게 한다.

## 배치 정규화 개념

입력 변수를 정규화하면 학습이 빨라진다


Normalizing inputs to speed up learning



$$\mu = \frac{1}{m} \sum_i x^{(i)}$$

$$X = X - \mu$$

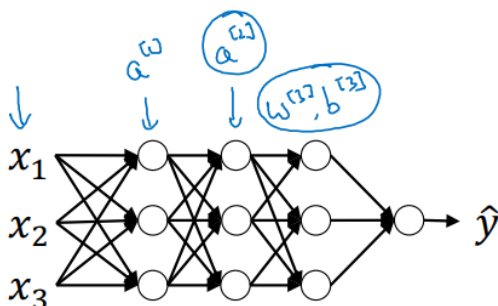
$$\sigma^2 = \frac{1}{m} \sum_i x^{(i)2} \quad \leftarrow \text{element-wise}$$

$$X = X / \sigma^2$$


로지스틱 회귀 등으로 모델을 학습시킬 때 입력 변수들을 정규화하면 학습이 빨라졌다. 평균을 계산할 때는 입력 변수의 평균을 뺐고, 분산을 계산할 때는 하나씩 제공해주었다.

→ 이 방법은 누워있는 학습 등고선을 경사하강법에 적합한 둥근 형태로 바꿔준다.

그렇다면 심층 신경망에서는 어떨까?



Can we normalize  $\frac{a^{[2]}}{w^{[2]}, b^{[2]}}$  so as to train faster

Normalize  $z^{[2]}$

↑

심층 신경망에서는 입력 변수  $x$ 뿐 아니라 **각 층의 활성화값  $a$** 가 있다.

입력층을 정규화하는 것보다  $w$ 와  $b$ 에게 직접적인 영향을 미치는 직전 층의  $a$ 를 정규화하는 것이 더 효율적일 것이다.

## 그런데 은닉층의 $a$ 값을 어떻게 정규화하는가? → Batch Normalization

활성 함수 이전의 값인  $z$ 를 정규화할 것인지, 활성 함수 이후의 값(최종  $y$ )인  $a$ 를 정규화할 것인지에는 논쟁이 있지만, **실제로는  $z$ 를 정규화하는 것이 더 자주 쓰인다.**

## 배치 정규화 구현하기

### 기본 구현 과정

### Implementing Batch Norm

Given some intermediate values in NN  $z^{(1)}, \dots, z^{(m)}$

$$\mu = \frac{1}{m} \sum_i z^{(i)}$$
$$\sigma^2 = \frac{1}{m} \sum_i (z^{(i)} - \mu)^2$$
$$z_{\text{norm}}^{(i)} = \frac{z^{(i)} - \mu}{\sqrt{\sigma^2 + \epsilon}}$$

신경망의 은닉 유닛 값은  $z^{(1)}$ 부터  $z^{(m)}$ 까지 있다.

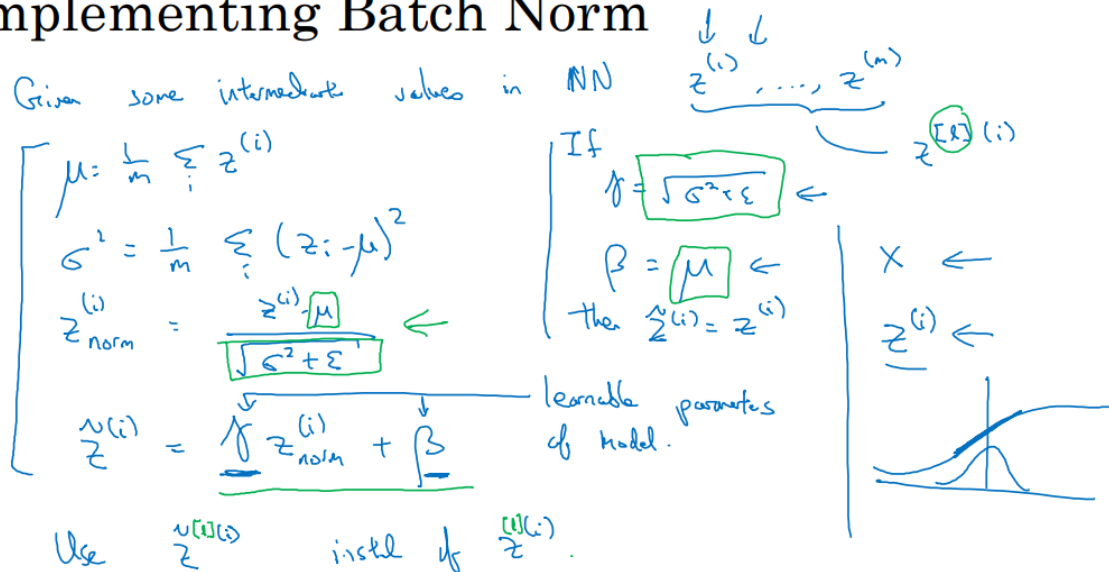
1. 은닉 유닛의 평균을 계산한다.
2. 은닉 유닛의 분산을 계산한다.
3. 각 은닉 유닛에 대해 정규화하여  $z_{\text{norm}}$ 을 얻는다.
  - a. 평균을 뺀 뒤에 표준편차로 나눈다.
  - b. 표준편차가 0인 경우를 대비해, 분모에  $\epsilon$ 을 추가한다.

→  $z$ 의 평균은 0, 표준편차는 1이 된다.

하지만 은닉 유닛은 다양한 분포를 가져야 하기 때문에 항상 평균 0, 표준편차 1을 갖는 것이 좋지만은 않다.

### $\tilde{z}$ ( $z_{\text{tilder}}$ )를 활용한 구현 과정

# Implementing Batch Norm



Andrew Ng

그래서 대신  $\tilde{z}$  ( $z_{\text{tilde}}$ )를 계산한다.  $\tilde{z} = \gamma * z^{(i)}_{\text{norm}} + \beta$  이다.

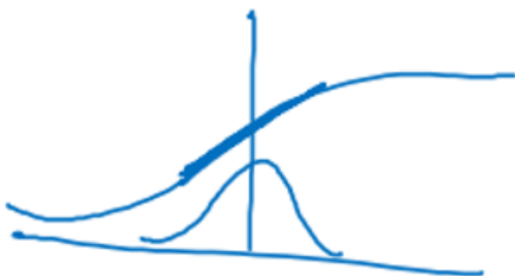
- $\gamma$ 와  $\beta$ 는 학습시킬 수 있는 변수이다. 모멘텀, RMSprop, Adam을 이용한 경사하강법 등 다양한 알고리즘을 이용해서  $\gamma$ 와  $\beta$ 를 학습시킬 수 있다.
- $\gamma$ 와  $\beta$ 를 이용하면  $\tilde{z}$ 의 평균을 원하는 대로 설정할 수 있다.

→ 다른  $\gamma$ 와  $\beta$  값을 정한다면 은닉 유닛의 값들이 서로 다른 평균이나 분산 값을 만들게 할 수 있다.

## 배치 정규화 개념 정리

배치 정규화는 입력층에만 정규화를 하는 것이 아니라 신경망 안 깊이 있는 은닉층의 값들까지도 정규화하는 것이다.

차이점은 은닉 유닛의 평균과 분산은 0, 1로 고정되는 게 좋지 않다는 것이다.

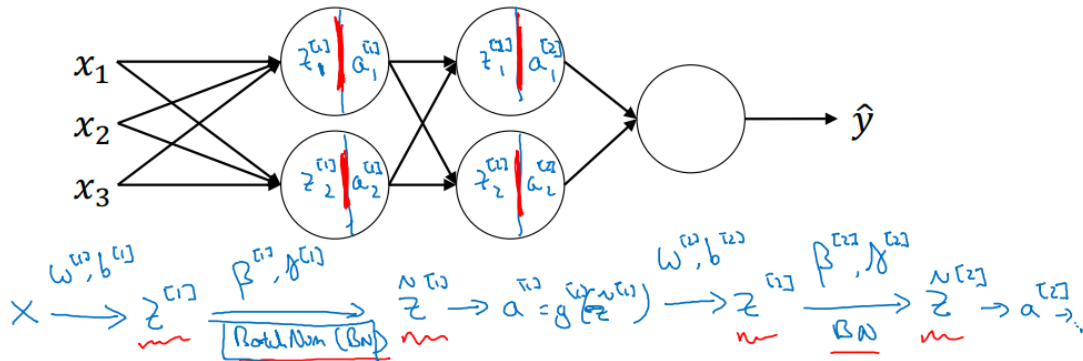


예를 들어, 왼쪽처럼 시그모이드 함수에서 은닉층 값이 저런 분포를 보이는 것은 좋지 않다.

시그모이드의 비선형성을 살릴 수 있도록 평균이 0이 아닌 다른 값을 갖는 것이 좋다.

## 심층 신경망 학습에 적용

### Adding Batch Norm to a network

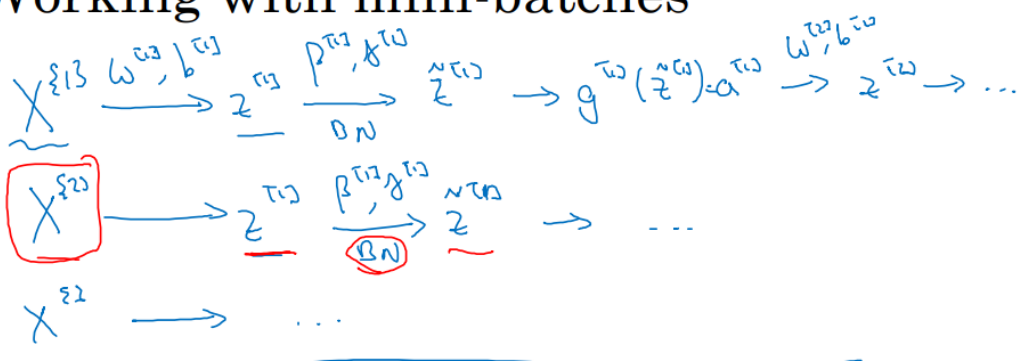


은닉 유닛의 계산은  $z$ 와  $a$  계산으로 나뉜다.

1.  $z$ 를 계산하고, 이를 배치 정규화한다.
  - a. Adam의 하이퍼파라미터  $\beta$ 와 배치 정규화의  $\beta$ 는 다르다!
  - b. 보통 딥러닝 프레임워크에서 배치 정규화는 함수로 구현할 수 있다. (tensorflow - `tf.nn.batch_normalization`)
2. 정규화 된 값들을 활성화 함수를 거쳐 활성화 값  $a$ 를 얻는다.

**배치 정규화는 훈련 세트의 미니배치에 적용된다.**

### Working with mini-batches



이렇게 첫 번째 미니 배치  $X^{[1]}$ 에 대해 경사하강법을 이용해서 과정을 마쳤으면 두 번째 미니 배치  $X^{[2]}$ 로 이동하고,  $z^{[1]}$ 을 계산한 다음 배치 정규화를 써서  $\tilde{z}^{[1]}$ 을 계산한다.

Parameters:  $w^{[l]}$ ,  ~~$b^{[l]}$~~ ,  $\beta^{[l]}$ ,  $\gamma^{[l]}$ .

- $z^{[l]} = w^{[l]} * a^{[l-1]} + b^{[l]} \rightarrow z^{[l]} = w^{[l]} * a^{[l-1]}$

배치 정규화에서  $z$ 의 평균을 계산한 뒤에 빼주기 때문에,  $b$ 의 값은 아무런 영향도 미치지 않는다. 따라서 제거한다.

- $z \sim \gamma^{[l]} * z^{[l]}_{\text{norm}} + \beta^{[l]}$

$z$ 의 평균을 결정하는  $\beta^{[l]}$ 은 써야 한다.  $b$  대신 편향 변수 역할을 한다.

$b$ 처럼  $\beta^{[l]}$ 과  $\gamma^{[l]}$ 의 차원도  $(n^{[l]}, 1)$ 이 된다.

- $n^{[l]}$ : 층  $l$ 에서의 은닉 유닛 수

## Gradient Descent 구현

### Implementing gradient descent

for  $t = 1 \dots \text{num Mini Batches}$   
 Compute forward pass on  $X^{t+1}$ .  
 In each hidden layer, use BN to replace  $z^{[l]}$  with  $\tilde{z}^{[l]}$ .  
 Use backprop to compute  $\frac{dw^{[l]}}{dz^{[l]}}$ ,  ~~$\frac{db^{[l]}}{dz^{[l]}}$~~ ,  $\frac{d\beta^{[l]}}{dz^{[l]}}$ ,  $\frac{d\gamma^{[l]}}{dz^{[l]}}$   
 Update params  $\left. \begin{aligned} w^{[l]} &:= w^{[l]} - \alpha \frac{dw^{[l]}}{dz^{[l]}} \\ \beta^{[l]} &:= \beta^{[l]} - \alpha \frac{d\beta^{[l]}}{dz^{[l]}} \\ \gamma^{[l]} &:= \dots \end{aligned} \right\} \leftarrow$   
 Works w/ momentum, RMSprop, Adam.

미니배치를 반복

1.  $X$ 의 forward propagation 계산
2. 모든 은닉 유닛에 대해 배치 정규화 적용
3. Backward propagation 계산  $\rightarrow dw, d\beta, d\gamma$  계산

## 배치 정규화의 원리

**입력 특성을 정규화하는 것은 학습 속도를 향상시킨다.**

**은닉층 값의 분포를 제한하여 값이 바뀌어서 발생하는 문제를 안정화시킨다.**

- 공변량 변화(Covariate Shift): X, Y 간의 대응을 학습시킬 때, X의 분포가 바뀌면 학습 알고리즘을 다시 학습해야 한다.
- 파라미터를 정규화한다. 일반화하는 것은 부수적인 효과이다.
- 은닉층에 곱셈잡음, 덧셈잡음을 추가하여 은닉층이 하나의 은닉 유닛에 너무 의존하지 않게 한다.

## 배치 정규화 테스트

테스트 시에는 배치가 하나이기 때문에 평균과 분산을 계산할 수 없습니다.

뮤와 시그마를 구할 수 없으므로, 독립적인 다른 식을 사용해야 한다.

따라서, 학습시에 사용된 미니배치들의 지수 가중 이동 평균을 추정치로 사용한다.