

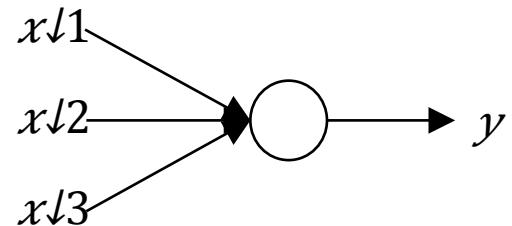


deeplearning.ai

Batch Normalization

Why does Batch Norm work?

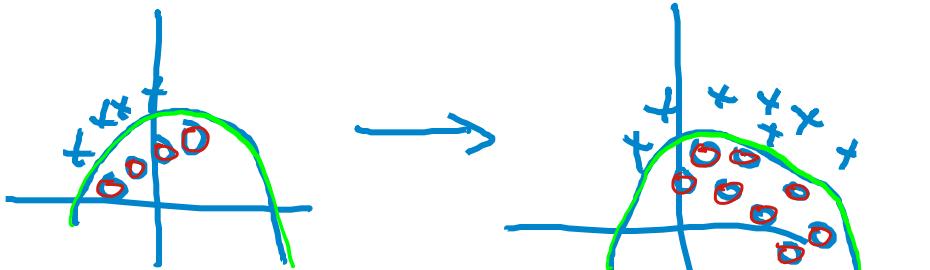
Learning on shifting input distribution



Cat
 $y = 1$



Non-Cat
 $y = 0$



$y = 1$



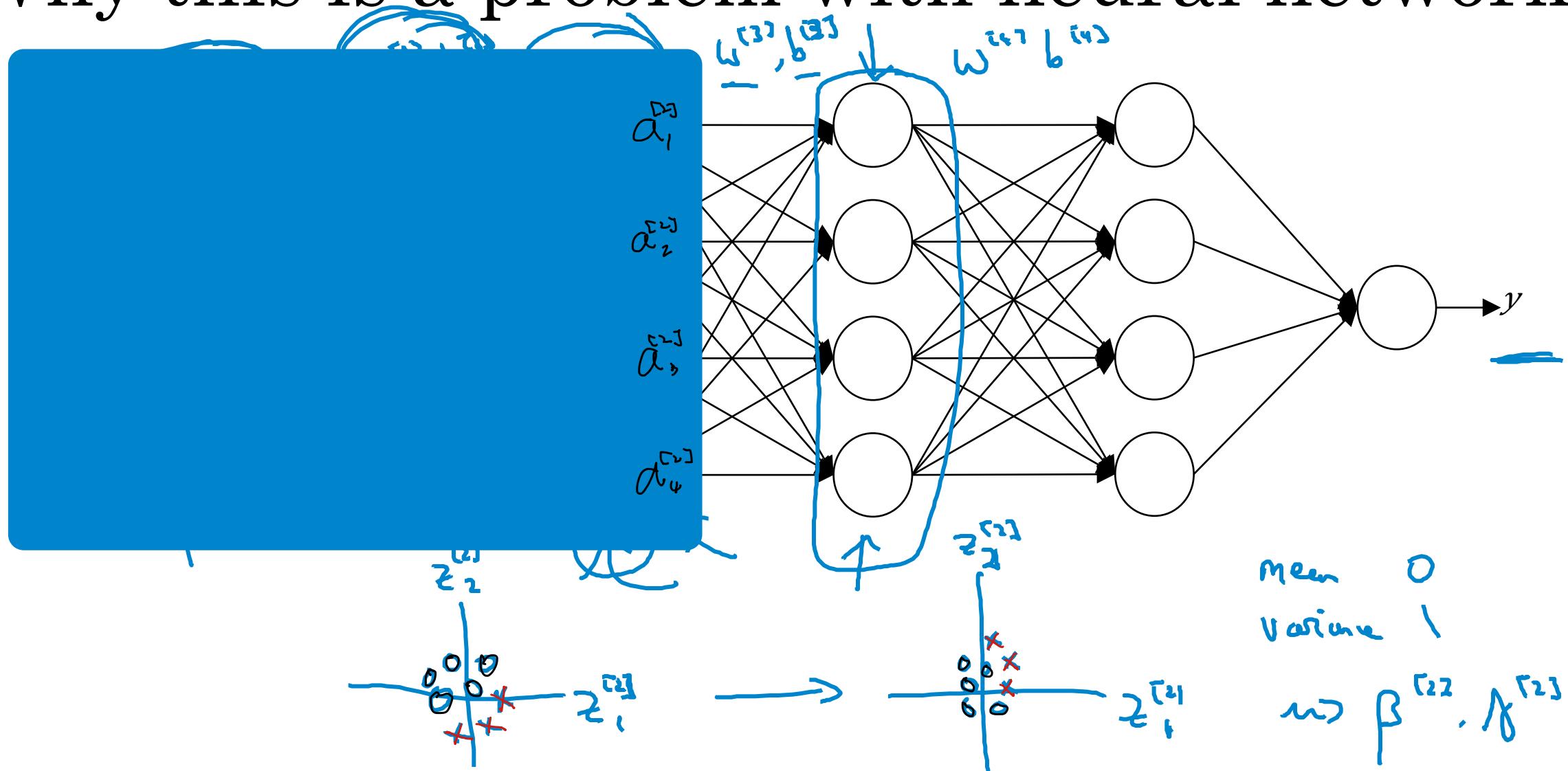
$y = 0$



"Covariate shift"
covariate shift

$\text{X} \rightarrow \text{Y}$

Why this is a problem with neural networks?



앞을 막아놓고 생각하면 3번째 히든 레이어의 job은 $a[2]$ 를 받아서 그들을 $y_{\hat{}}$ 으로 매핑하는 방법을 찾는 것이다. 근데 uncover하면 $a[2]$ 도 결국은 W 와 b 들에 의해서 결정되는 값이므로 W , b 가 바뀌면 a 도 바뀐다. 즉, covariate shift가 발생하게 된다. BN은 hidden unit value의 distribution이 shift around하는 양을 줄여준다. z_1 , z_2 그림으로 놓고 보면 z_1 , z_2 의 값 자체는 바뀌더라도 적어도 z_1 , z_2 의 mean, variance는 stay same 할 것이다. 이전 레이어의 변화에 영향을 덜 받게 된다는 것은 이전 레이어의 파라미터가 할 일과 뒷 레이어의 파라미터가 할 일의 커플링을 약화시킨다는 것을 의미한다.

Batch Norm as regularization

X

- Each mini-batch is scaled by the mean/variance computed on just that mini-batch.
 $\hat{z}^{(i)}$ $\underline{64}, \underline{128}$ $\underline{z}^{(i)}$
- This adds some noise to the values $z^{(i)}[l]$ within that minibatch. So similar to dropout, it adds some noise to each hidden layer's activations.
 μ, σ^2
- This has a slight regularization effect.

mini-batch : 64 \longrightarrow 512