

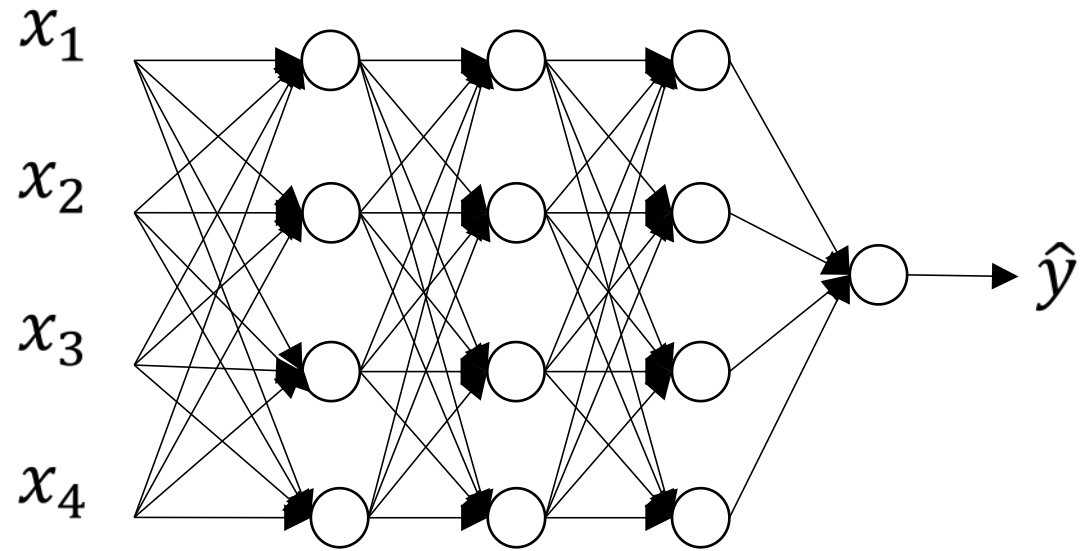


deeplearning.ai

Regularizing your neural network

Dropout regularization

Dropout regularization



↑
0.5 ↑
0.5 ↑
0.5

Implementing dropout ("Inverted dropout")

Illustrate with layer $l=3$. keep-prob = 0.8 0.2

→ $d3 = \text{np.random.rand}(a3.\text{shape}[0], a3.\text{shape}[1]) < \text{keep-prob}$

$a3$ = np.multiply($a3$, $d3$)

$a3 \neq d3$.

→ $a3 /= \text{keep-prob}$ ←

50 units. \leadsto 10 units shut off

$$z^{[4]} = w^{[4]} \cdot \underbrace{a^{[3]}}_{\text{reduced by 20\%}} + b^{[4]}$$

↑

reduced by 20%

$$/= \underline{0.8}$$

Test

Making predictions at test time

$$a^{[0]} = X$$

No drop out.

$$z^{[1]} = W^{[1]} \frac{a^{[0]}}{\quad} + b^{[1]}$$

$$a^{[1]} = g^{[1]}(z^{[1]})$$

$$z^{[2]} = W^{[2]} \frac{a^{[1]}}{\quad} + b^{[2]}$$

$$a^{[2]} = \dots$$

↓
↑
 \hat{y}

\neq keep-prob



deeplearning.ai

Regularizing your neural network

Understanding dropout

Why does drop-out work?

Intuition: Can't rely on any one feature, so have to spread out weights. \rightarrow Shrink weights. b_2

