

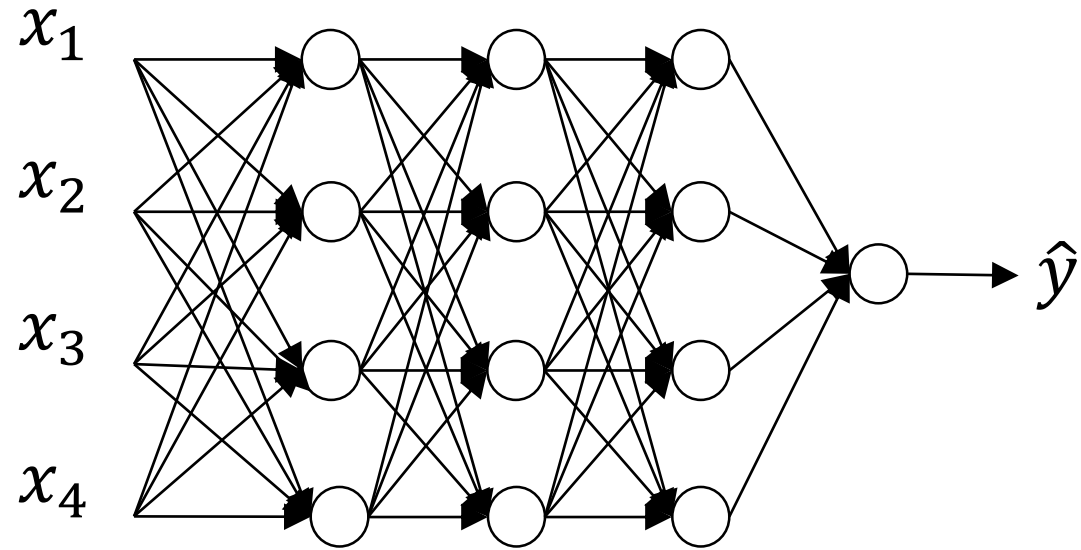


deeplearning.ai

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
 x

0.2

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

$a3 = \text{np.multiply}(a3, d3)$

$a3 \neq d3$.

$\rightarrow a3 /= \text{keep-prob}$

50 units. \leadsto 10 units shut off

$$z^{[4]} = w^{[4]} \cdot a^{[3]} + b^{[4]}$$

\nwarrow reduced by 20%.

$= 0.8$

Hence, about 20% of the nodes will be removed.

If keep_prob = 0.8, then approximately 80% of the nodes will be kept in the neural network.

Making predictions at test time

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

No dropout.

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

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

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

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

↓
↑
y

Don't use dropout during the test time, otherwise it will affect the distribution.

$\neq \text{keep-prob}$



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. \leadsto Shrink weights. \hookrightarrow hence, it has

Shrink weights. b_2

hence, it has the
effect of
regularization

