Machine Learning

Neural Networks

And Deep learning

Session 4

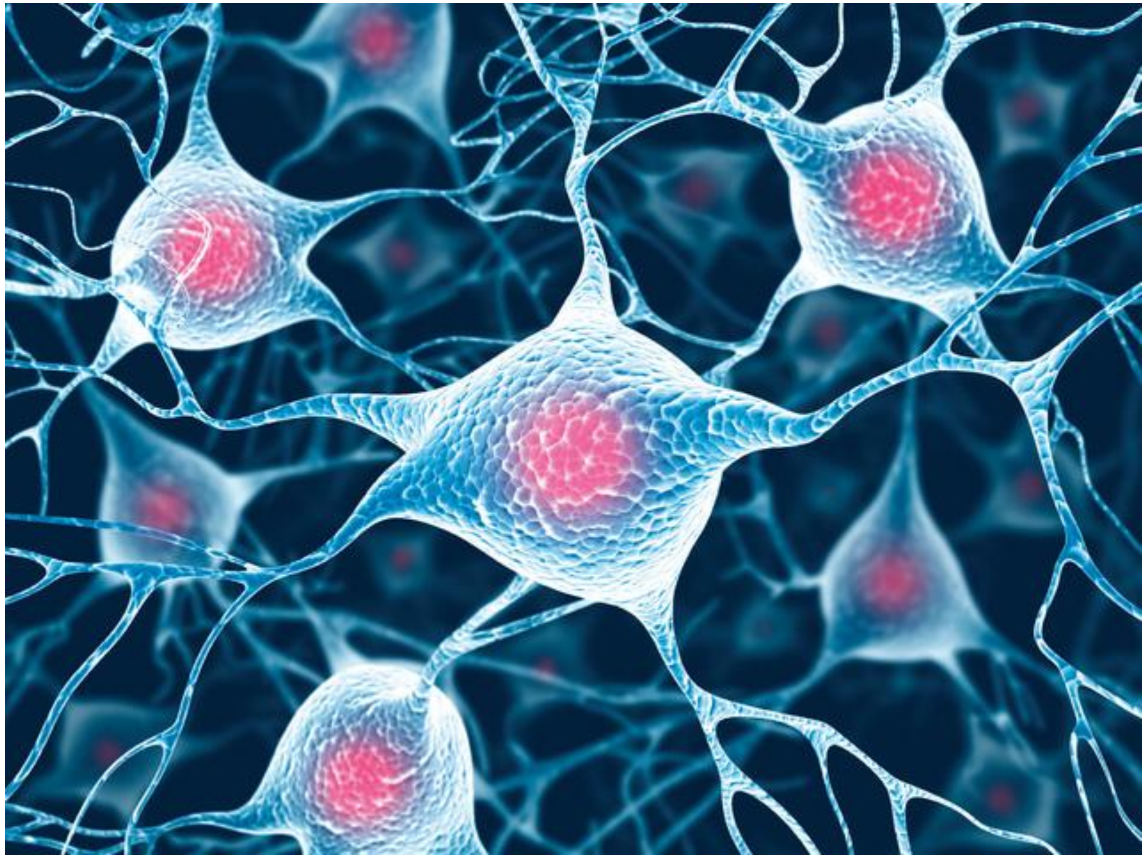
Session 4

Content

- What is neural network
- what can neural network do
- how does neural network works
- one neuron
- one layer
- multi layers

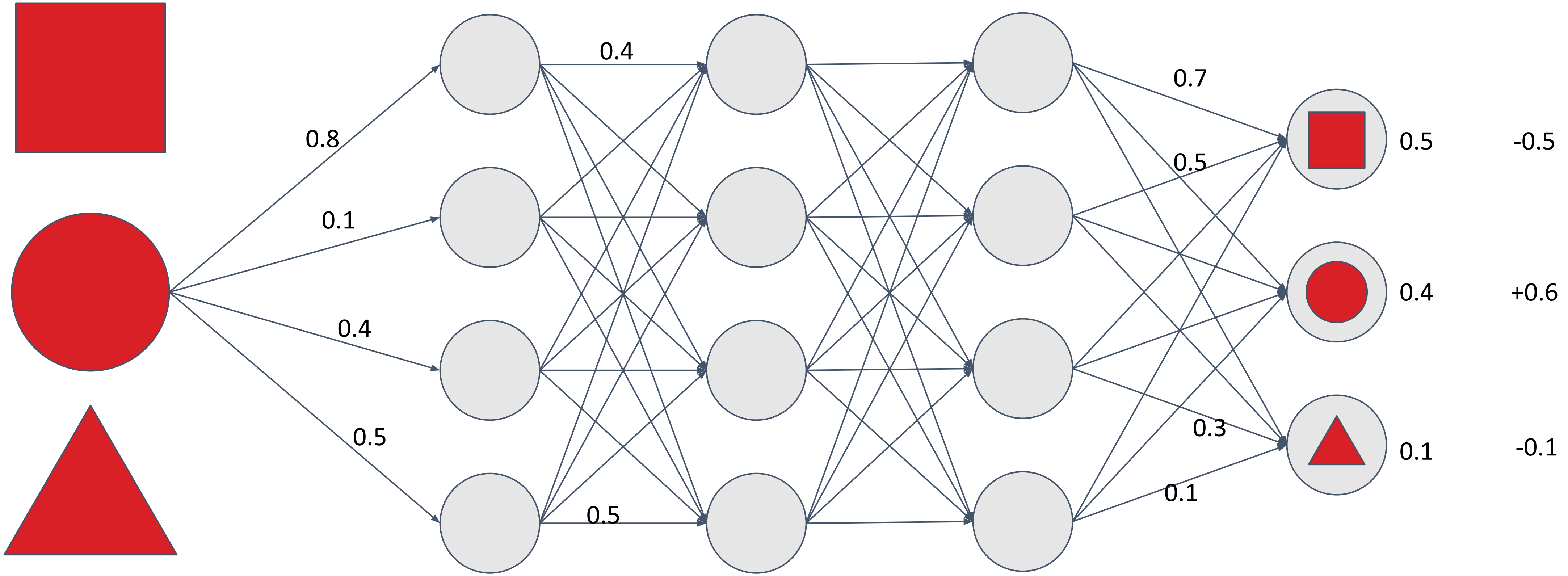
Session 4

What is neural network



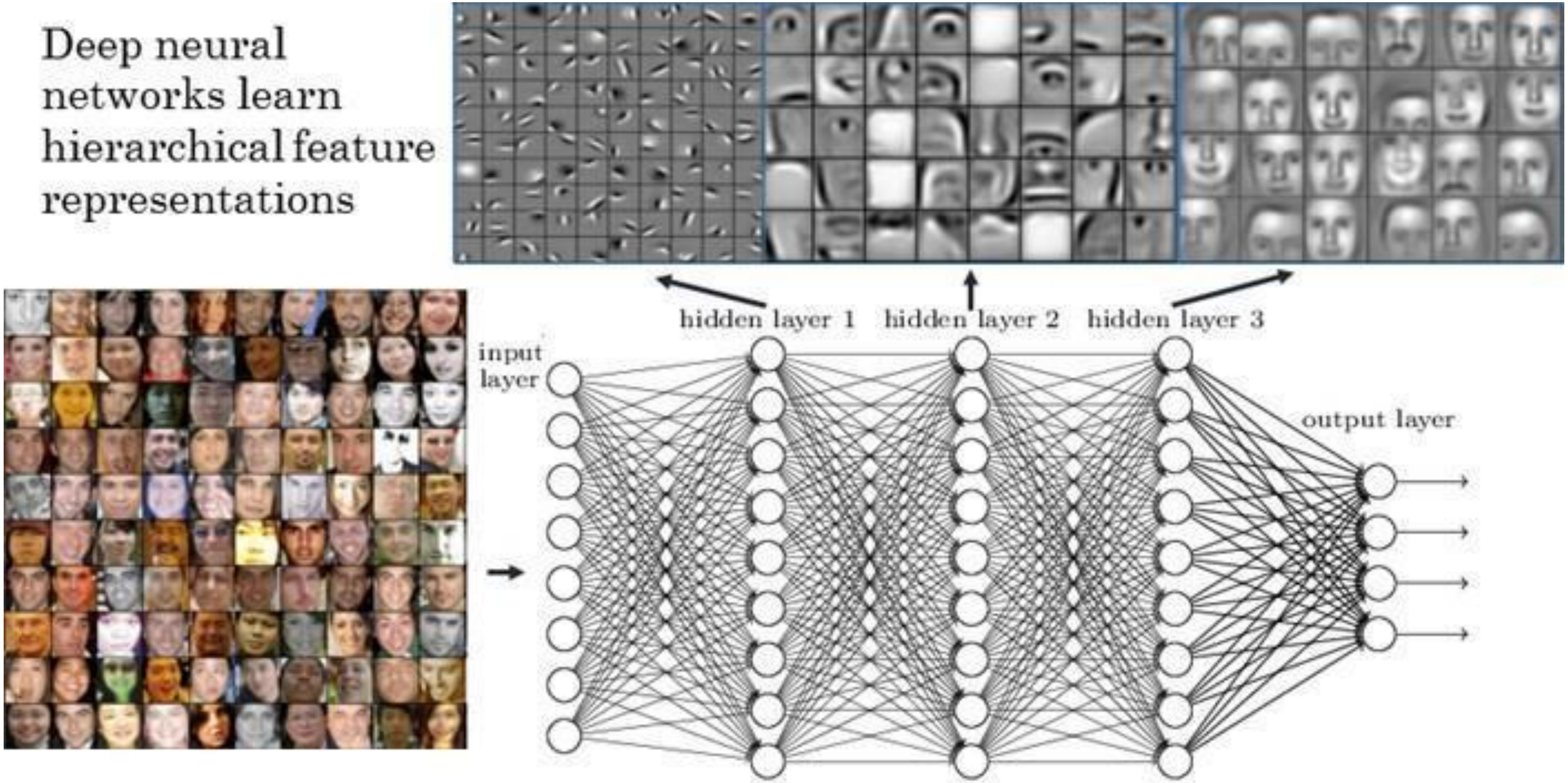
Session 4

What is neural network



What can neural network do

Deep neural networks learn hierarchical feature representations



Session 4

What is neural network

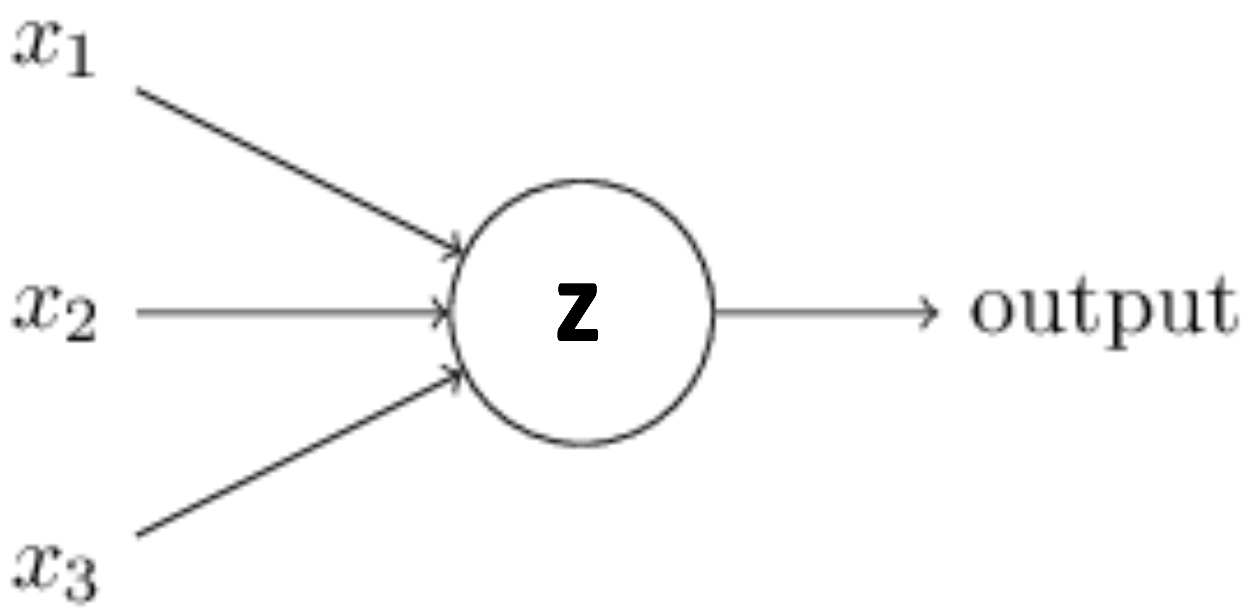
$$h_{\Theta}(x) = \Theta_0 + \Theta_1 x_1 + \Theta_2 x_2$$

$$z = b + w_1 x_1 + w_2 x_2$$

Session 4

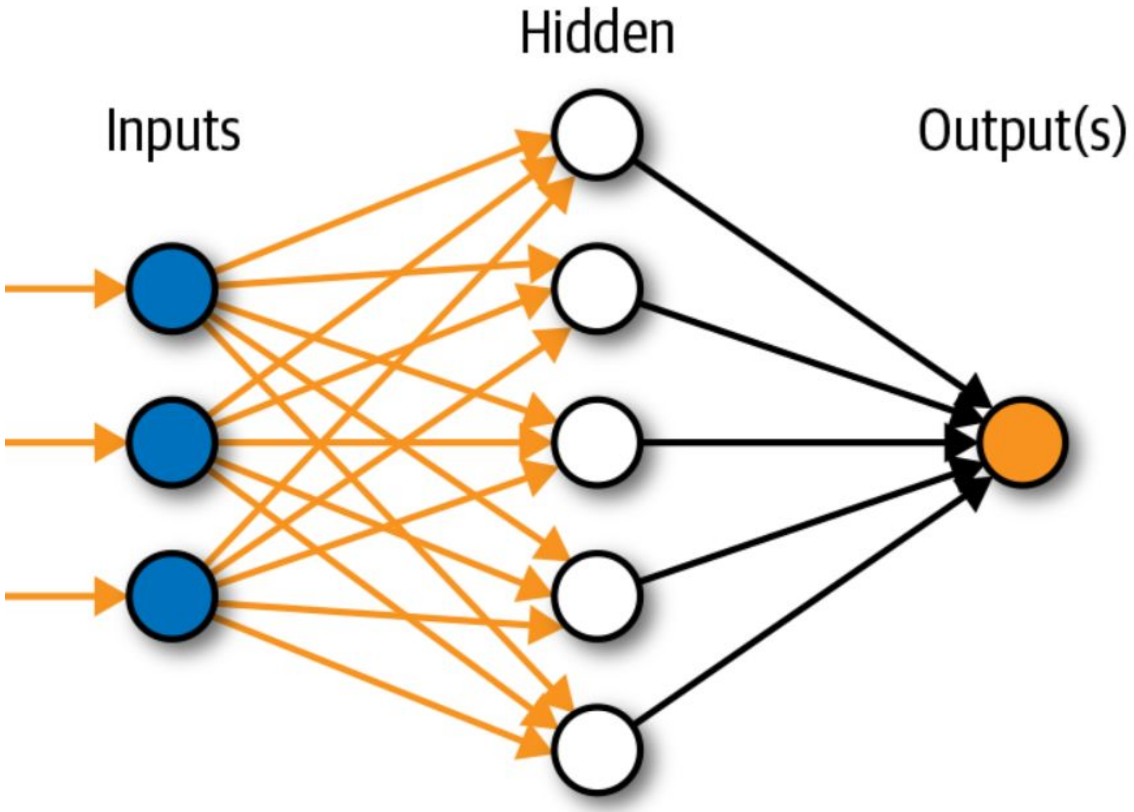
What is neural network

$$z = b + w_1 x_1 + w_2 x_2 + w_3 x_3$$



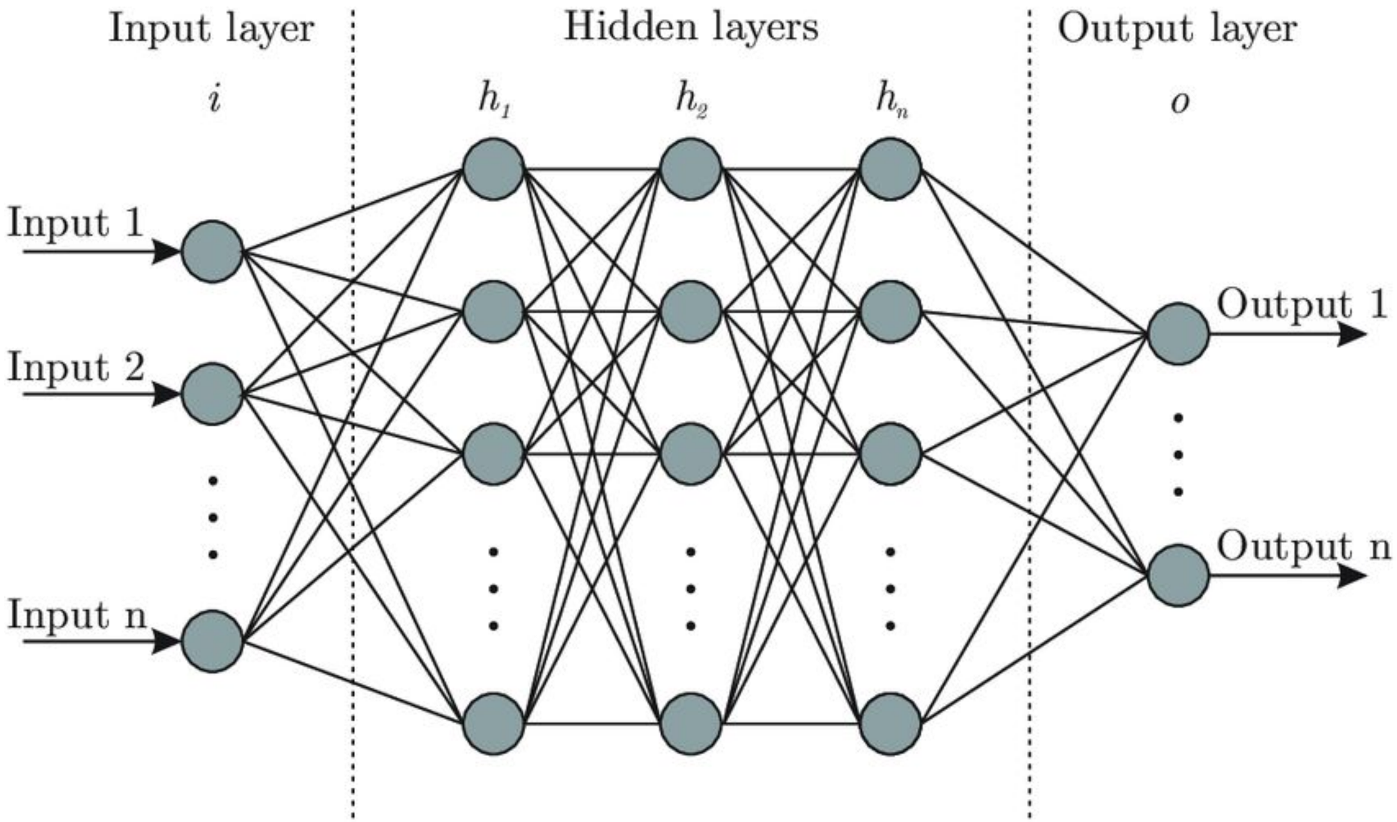
Session 4

What is neural network

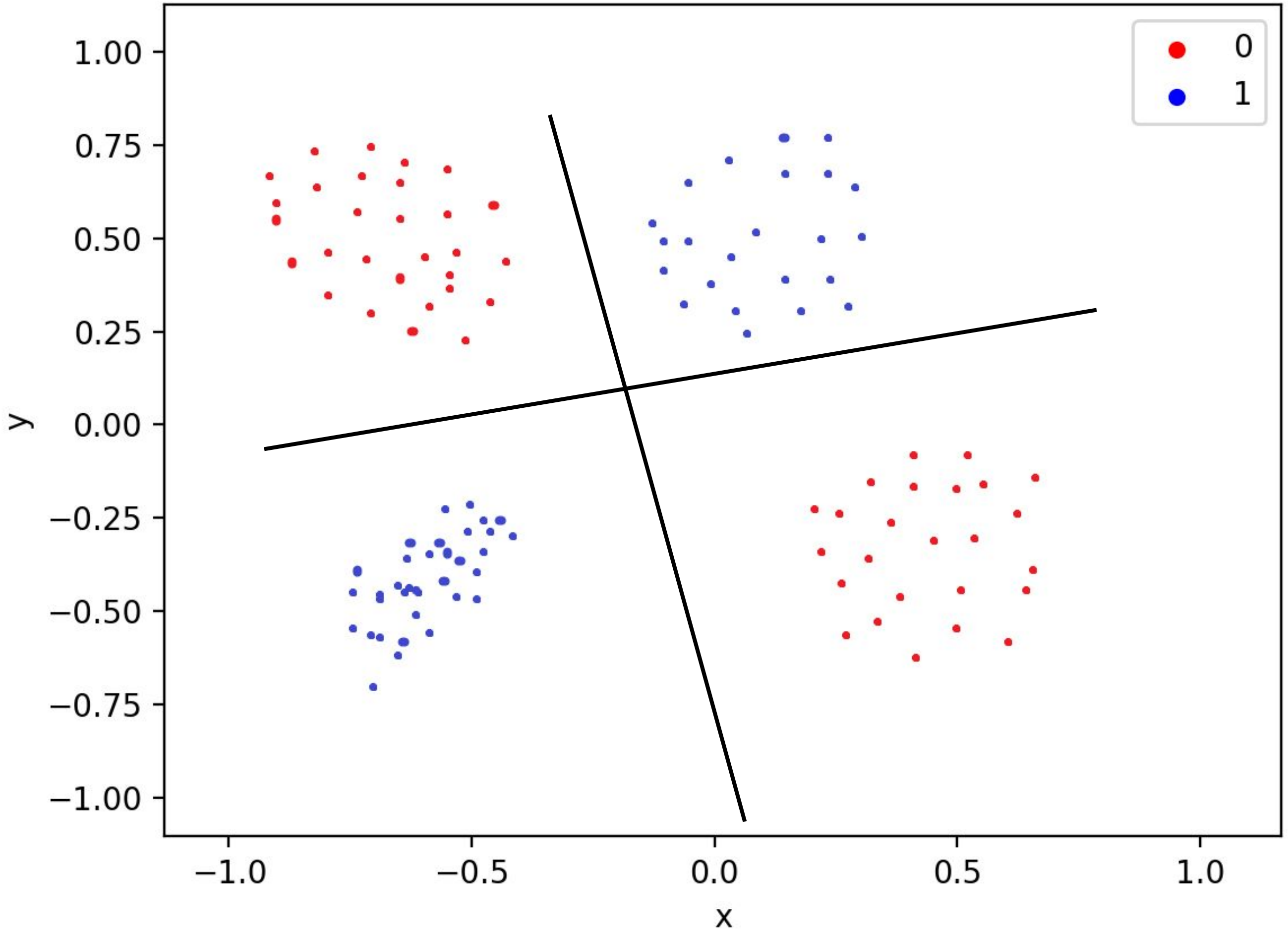


Session 4

What is neural network



What can neural network do

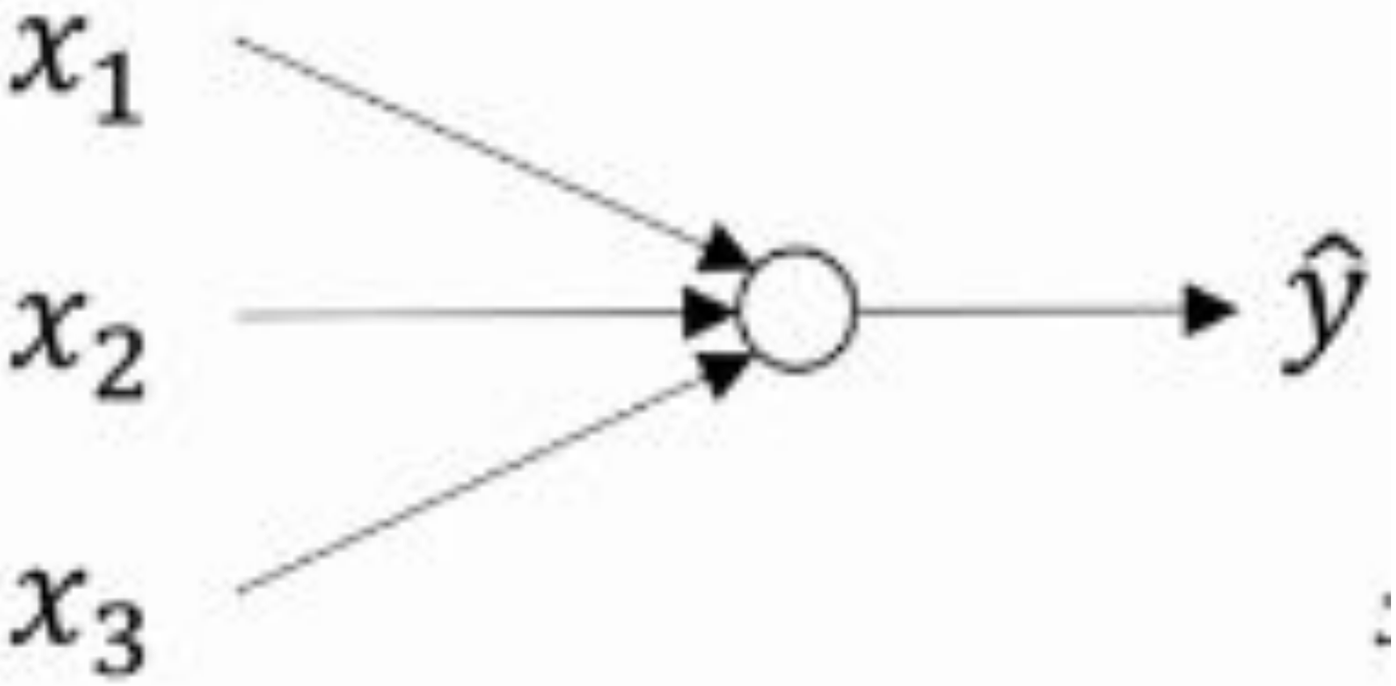


Review

1. hypothesis
2. cost
3. gradient descent

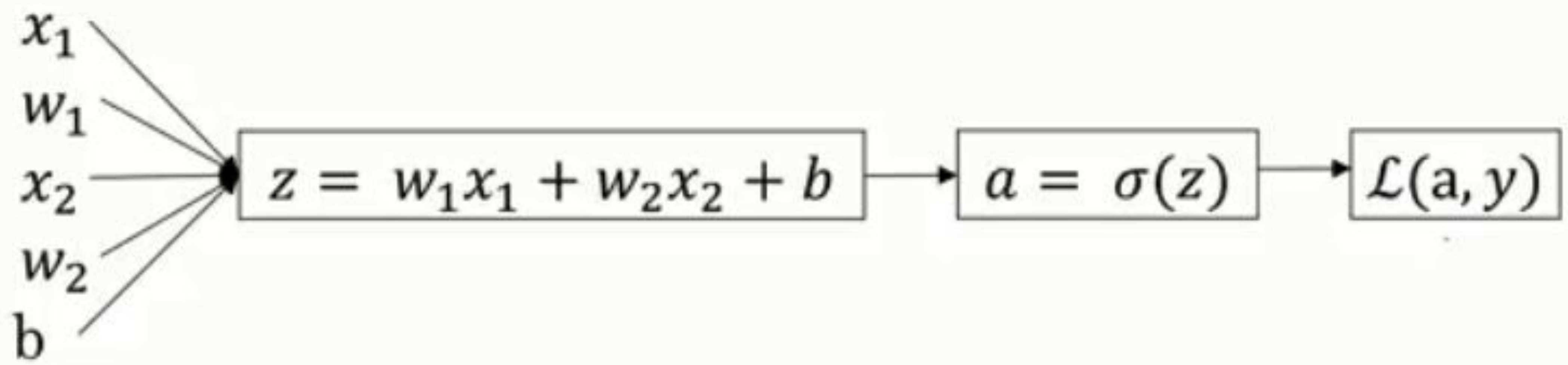
How does neural network work

One neuron



How does neural network work

One neuron



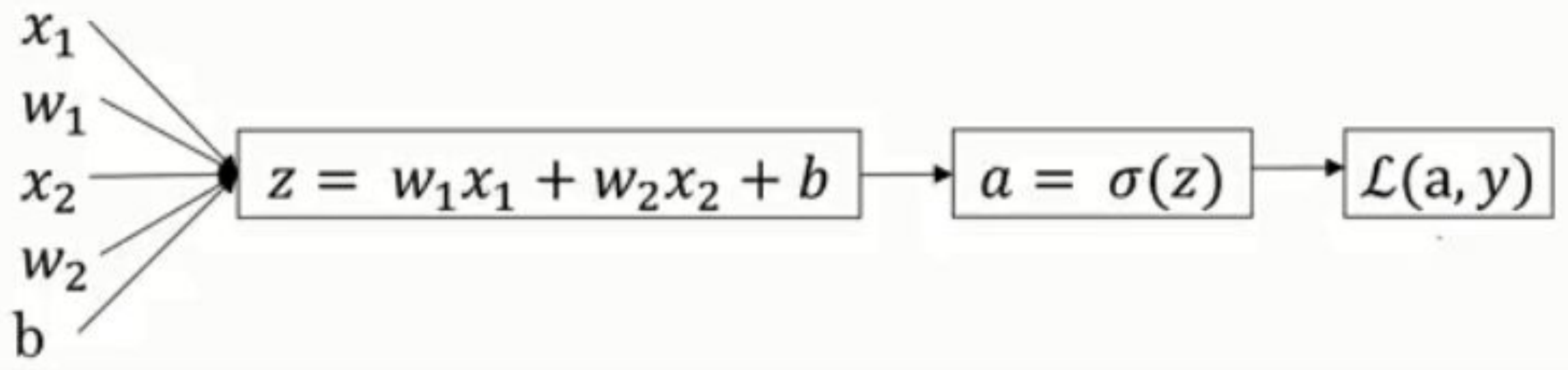
$$\hat{y} = a = \sigma(z)$$

$$\mathcal{L}(a, y) = -(y \log(a) + (1 - y) \log(1 - a))$$

$$J(w, b) = \frac{1}{m}\sum_{i=1}^{m}\mathcal{L}(\hat{y}^{(i)}, y^{(i)}) = -\frac{1}{m}\sum_{i=1}^{m} y^{(i)} \log \hat{y}^{(i)} + (1 - y^{(i)}) \log(1 - \hat{y}^{(i)})$$

How does neural network work

One neuron



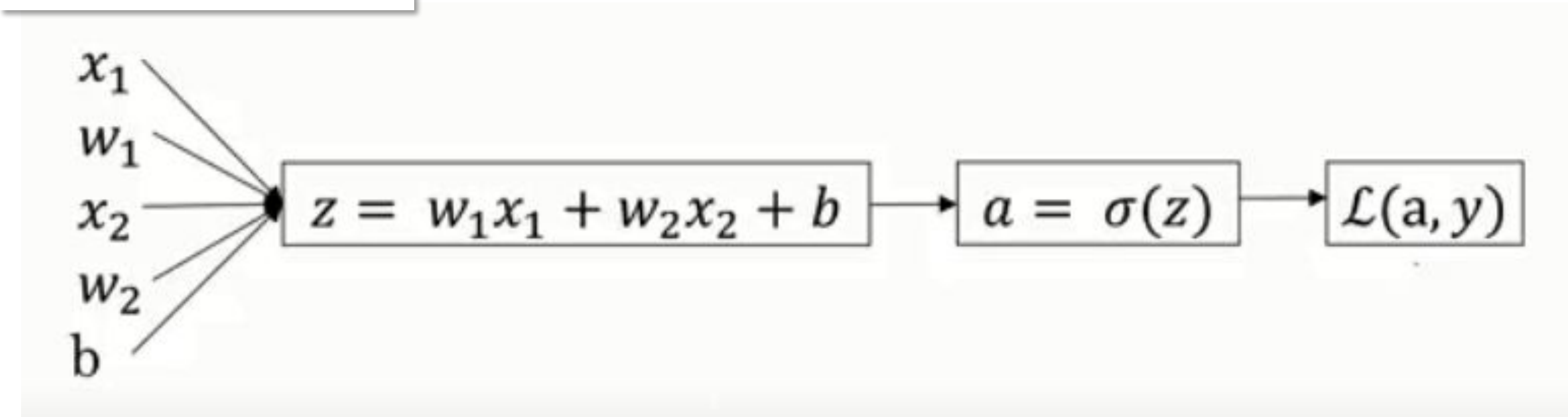
$$w_1 = w_1 - \alpha \, dw_1$$

$$w_2 = w_2 - \alpha \, dw_2$$

$$b = b - \alpha \, db$$

How does neural network work

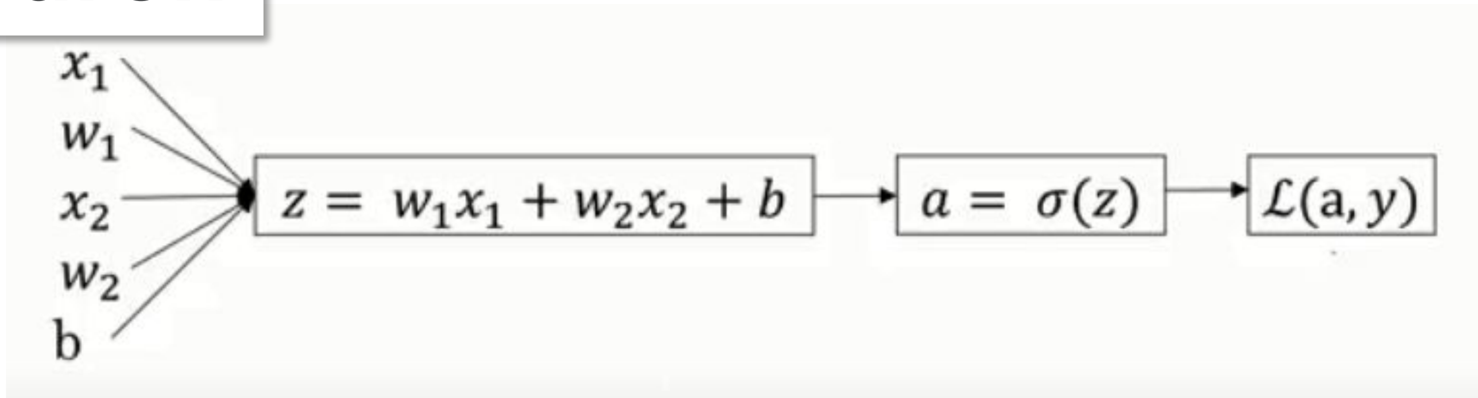
One neuron



$$\mathrm{d}w_1 = \frac{\partial J(w,b)}{\partial w_1} \quad \mathrm{d}w_2 = \frac{\partial J(w,b)}{\partial w_2} \quad \mathrm{d}b = \frac{\partial J(w,b)}{\partial b}$$

How does neural network work

One neuron



$$\boxed{a = \sigma(z) = \frac{1}{1 + e^{-z}}}$$

$$\text{"da"} = \frac{\partial\, l(a,y)}{\partial a} = -\frac{y}{a} + \frac{1-y}{1-a}$$

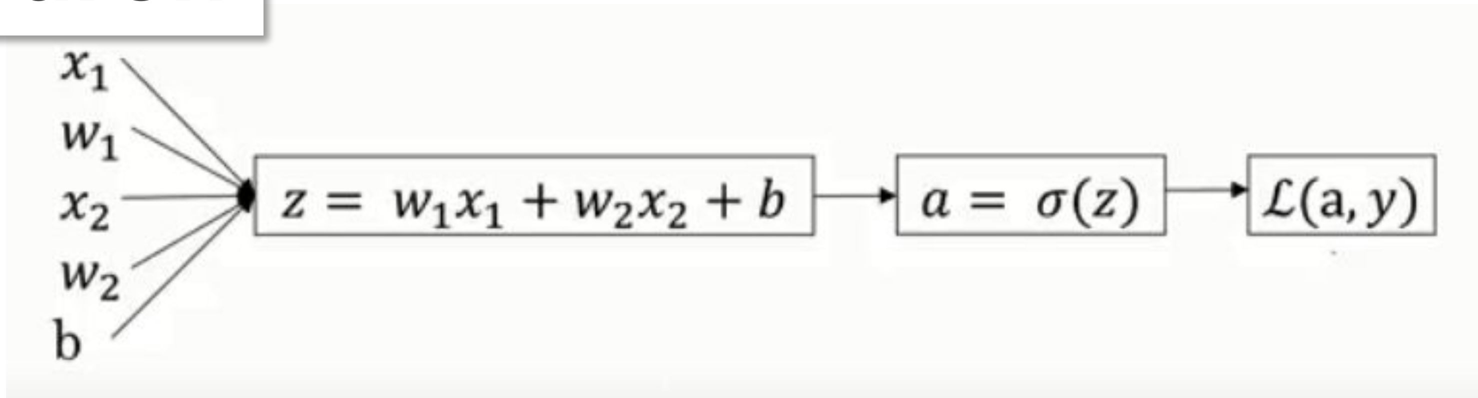
$$dz = \frac{\partial\, \ell(a,y)}{\partial z} = \frac{\partial\, \ell(a,y)}{\partial a} \times \frac{da}{\partial z}$$

$$dz = da * \frac{\partial a}{\partial z} = \left(-\frac{y}{a} + \frac{1-y}{1-a}\right) * \mathrm{a}(\mathrm{a}-1)$$

$$= \mathbf{a - y}$$

How does neural network work

One neuron



$$dz = a - y$$

$$dw_1 = x_1\, dz$$
$$dw_2 = x_2\, dz$$
$$db = dz$$

How does neural network work

One neuron

- ❑ Forward propagation
- ❑ Backward propagation
- ❑ Update

How does neural network work

$$Z = w^T X + b = np.dot(w.T, X) + b$$

$$A = \sigma(Z)$$

J = - [Y * log(A) + (1 – Y) *log (1 – A)]

dZ = A-Y

dw= $\frac{1}{m}$ X dZ^T

db = $\frac{1}{m}$ np.sum(dZ)

recap

Forward propagation

$$Z = w^T X + b$$
$$A = \sigma(Z)$$

$$Z = [z^{(1)} \quad z^{(2)} \ldots \ldots \quad z^{(m)}] \qquad (1,m)$$

$$A = [a^{(1)} \quad a^{(2)} \ldots \ldots \quad a^{(m)}] \qquad (1,m)$$

$$w^T = [w_1 \quad w_2 \ldots \ldots \quad w_n] \qquad (1,n)$$

$$X = \begin{bmatrix} | & | & & | \\ x^{(1)} & x^{(2)} & \ldots\ldots & x^{(m)} \\ | & | & & | \end{bmatrix} \qquad (n,m)$$

Backward propagation

dZ = A-Y

dw= $\frac{1}{m}$ X dZ^T

db = $\frac{1}{m}$ np.sum(dZ)

$$dZ = [dz^{(1)} \quad dz^{(2)} \ldots \ldots \quad dz^{(m)}] \qquad (1,m)$$

$$Y = [y^{(1)} \quad y^{(2)} \ldots \ldots \quad y^{(m)}] \qquad (1,m)$$

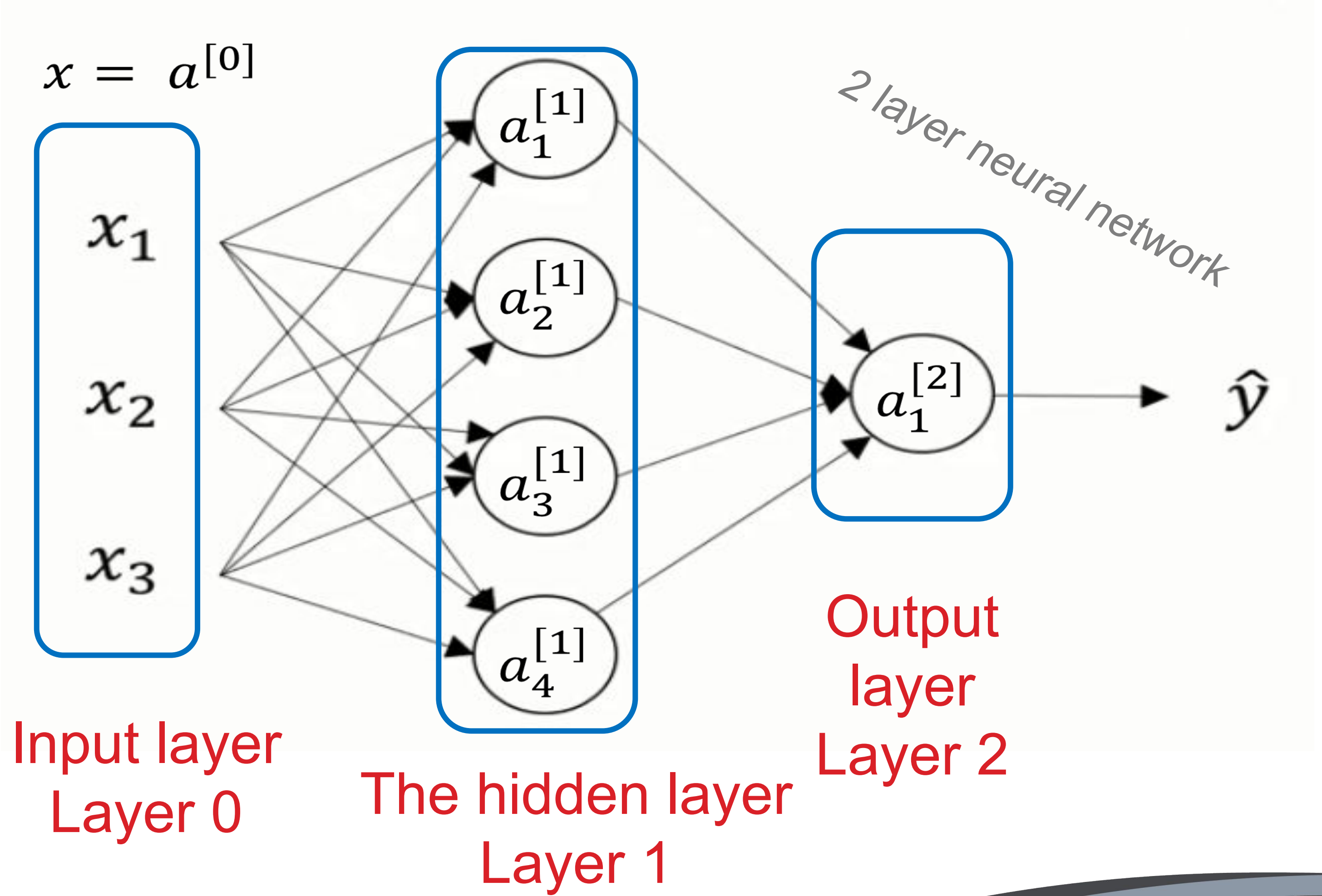
$$dw^T = [dw_1 \quad dw_2 \ldots \ldots \quad dw_n] \qquad (1,n)$$

db *is* 1 *by* 1 *matrix*

$$w = w - \alpha\, dw \qquad b = b - \alpha\, db$$

Grow Through

One hidden layer



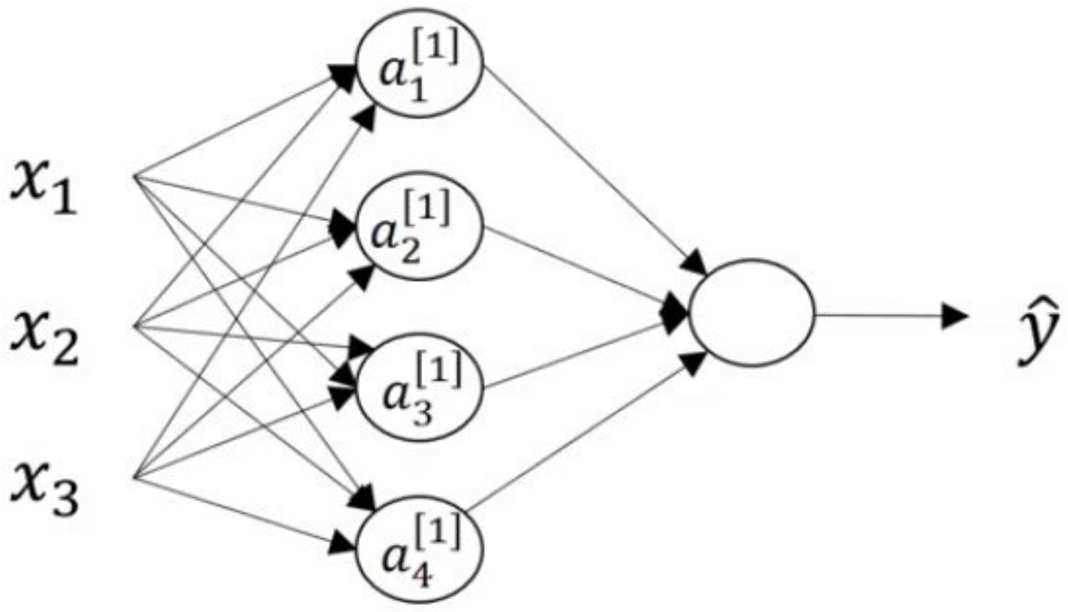
One hidden layer

$$z_1^{[1]} = w_1^{[1]T} x + b_1^{[1]}, \ a_1^{[1]} = \sigma(z_1^{[1]})$$

$$z_2^{[1]} = w_2^{[1]T} x + b_2^{[1]}, \ a_2^{[1]} = \sigma(z_2^{[1]})$$

$$z_3^{[1]} = w_3^{[1]T} x + b_3^{[1]}, \ a_3^{[1]} = \sigma(z_3^{[1]})$$

$$z_4^{[1]} = w_4^{[1]T} x + b_4^{[1]}, \ a_4^{[1]} = \sigma(z_4^{[1]})$$



One hidden layer

$$z_1^{[1]} = w_1^{[1]T} x + b_1^{[1]}$$

$$\underbrace{\begin{bmatrix} z_1^{[1]} \\ z_2^{[1]} \\ z_3^{[1]} \\ z_4^{[1]} \end{bmatrix}}_{z^{[1]}} = \underbrace{\begin{bmatrix} \cdots & w_1^{[1]T} & \cdots \\ \cdots & w_2^{[1]T} & \cdots \\ \cdots & w_3^{[1]T} & \cdots \\ \cdots & w_4^{[1]T} & \cdots \end{bmatrix}}_{W^{[1]}} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}_{x} + \underbrace{\begin{bmatrix} b_1^{[1]} \\ b_2^{[1]} \\ b_3^{[1]} \\ b_4^{[1]} \end{bmatrix}}_{b^{[1]}}$$

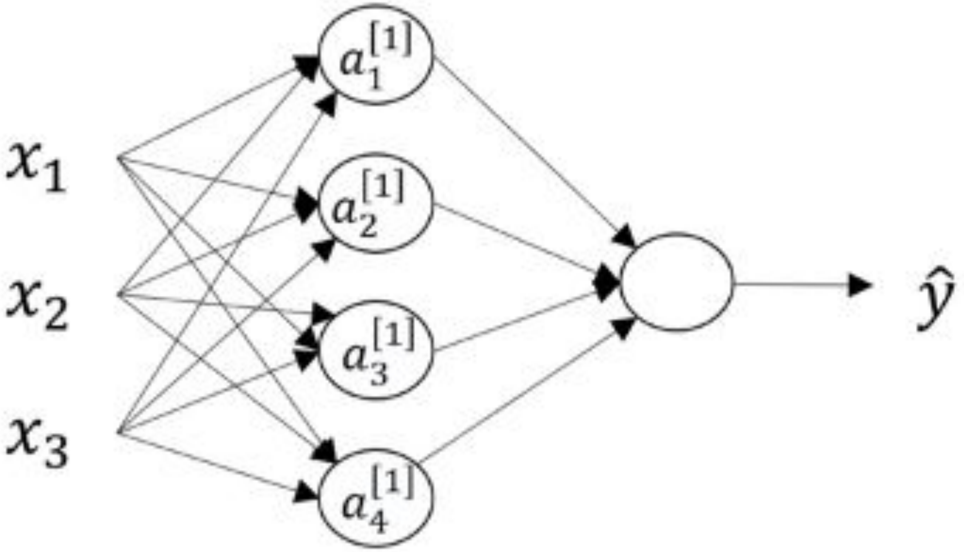
One hidden layer

$$a_1^{[1]} = \sigma(z_1^{[1]})$$

$$\begin{bmatrix} a_1^{[1]} \\ a_2^{[1]} \\ a_3^{[1]} \\ a_4^{[1]} \end{bmatrix} = \sigma\left(\begin{bmatrix} z_1^{[1]} \\ z_2^{[1]} \\ z_3^{[1]} \\ z_4^{[1]} \end{bmatrix} \right)$$

$$a^{[1]} \qquad\qquad\qquad z^{[1]}$$

One hidden layer



Given input x:

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

(4,1) (4,3) (3,1) (4,1)

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

(4,1) (4,1)

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

(1,1) (1,4) (4,1) (1,1)

$$a^{[2]} = \sigma(z^{[2]})$$

(1,1) (1,1)

One hidden layer

```
for i = 1 to m
```

$$z^{[1](i)} = W^{[1]}x^{(i)} + b^{[1]}$$

$$a^{[1](i)} = \sigma(z^{[1](i)})$$

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

$$a^{[2](i)} = \sigma(z^{[2](i)})$$

One hidden layer

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

$$\begin{bmatrix} | & | & & | \\ z^{[1](1)} & z^{[1](2)} & \cdots & z^{[1](m)} \\ | & | & & | \end{bmatrix} = \begin{bmatrix} \cdots & w_1^{[1]T} & \cdots \\ \cdots & w_2^{[1]T} & \cdots \\ \cdots & w_3^{[1]T} & \cdots \\ \cdots & w_4^{[1]T} & \cdots \end{bmatrix} \begin{bmatrix} | & | & & | \\ x^{(1)} & x^{(2)} & \ldots\ldots & x^{(m)} \\ | & | & & | \end{bmatrix} + \begin{bmatrix} b_1^{[1]} \\ b_2^{[1]} \\ b_3^{[1]} \\ b_4^{[1]} \end{bmatrix}$$

$$Z^{[1]} \quad = \quad W^{[1]} \quad X \quad + \quad b^{[1]}$$

(4,m) (4,n) (n,m) (4,1)

One hidden layer

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

$$\begin{bmatrix} | & | & & | \\ a^{[1](1)} & a^{[1](2)} & \cdots & a^{[1](m)} \\ | & | & & | \end{bmatrix} = \sigma\left(\begin{bmatrix} | & | & & | \\ z^{[1](1)} & z^{[1](2)} & \cdots & z^{[1](m)} \\ | & | & & | \end{bmatrix} \right)$$

$$A^{[1]} = \sigma(\ Z^{[1]}\ )$$

(4,m) (4,m)

One hidden layer

forward propagation

$$Z^{[1]} = W^{[1]}X + b^{[1]}$$
$$A^{[1]} = \sigma(Z^{[1]})$$
$$Z^{[2]} = W^{[2]}A^{[1]} + b^{[2]}$$
$$A^{[2]} = \sigma(Z^{[2]})$$

One hidden layer

back propagation

$$dw^{[1]} = \frac{\partial\, J(w^{[1]}, b^{[1]}, w^{[2]}, b^{[2]})}{\partial w_1} = \frac{1}{m}\sum_{i=1}^{m}\frac{\partial\, l(a^{[2]}, y)}{\partial w_1}$$

$$db^{[1]}$$

$$dw^{[2]}$$

$$db^{[2]}$$

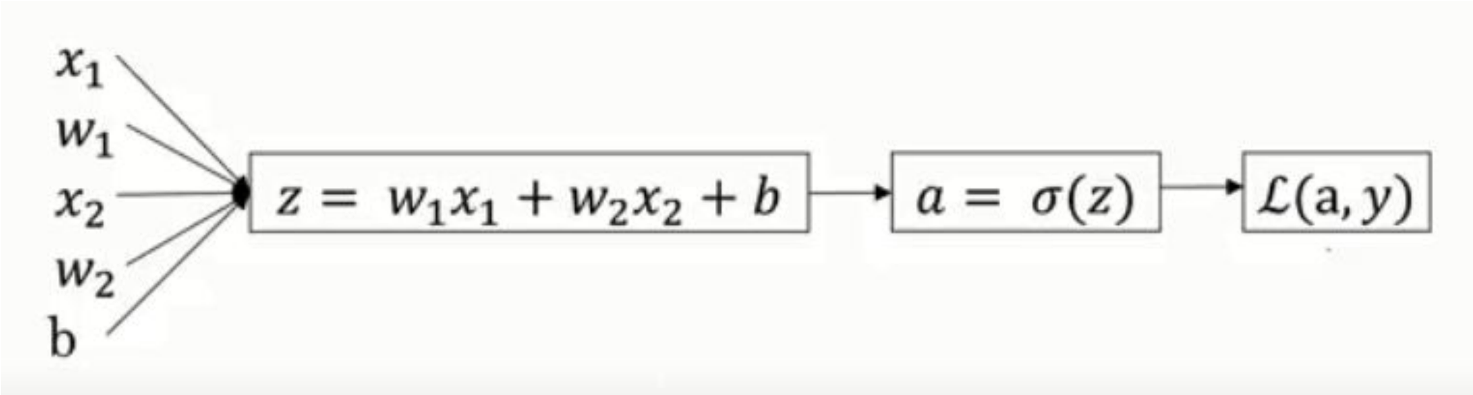
$$w^{[1]} = w^{[1]} - \alpha\, dw^{[1]}$$

$$b^{[1]} = b^{[1]} - \alpha\, db^{[1]}$$

$$w^{[2]} = w^{[2]} - \alpha\, dw^{[2]}$$

$$b^{[2]} = b^{[2]} - \alpha\, db^{[2]}$$

One hidden layer



$$\boxed{a = \sigma(z) = \frac{1}{1 + e^{-z}}}$$

$$\text{"da"} = \frac{\partial\, l(a,y)}{\partial a} = -\frac{y}{a} + \frac{1-y}{1-a}$$

$$dz = \frac{\partial\, \ell(a,y)}{\partial z} = \frac{\partial\, \ell(a,y)}{\partial a} \times \frac{da}{\partial z}$$

$$dz = da * \frac{\partial a}{\partial z} = \left(-\frac{y}{a} + \frac{1-y}{1-a}\right) * \mathrm{a(a-1)}$$

$$= \mathbf{a - y}$$

One hidden layer

$$"da" = \frac{\partial\, l(a,y)}{\partial a} = -\frac{y}{a} + \frac{1-y}{1-a}$$

$$x,\ W^{[1]},\ b^{[1]} \rightarrow \boxed{z^{[1]} = W^{[1]}x + b^{[1]}} \rightarrow \boxed{a^{[1]} = \sigma(z^{[1]})} \rightarrow \boxed{z^{[2]} = W^{[2]}a^{[1]} + b^{[2]}} \rightarrow \boxed{a^{[2]} = \sigma(z^{[2]})} \rightarrow \boxed{\mathcal{L}(a^{[2]},y)}$$

with $W^{[2]}$, $b^{[2]}$ entering the third box.

$$dz^{[1]} = \frac{\partial\, l(a^{[2]},y)}{\partial z^{[1]}}$$

$$da^{[1]} = \ \ldots$$

$$dz^{[2]} = a^{[2]} - y$$

$$dw^{[2]} = dz^{[2]} * a^{[1]T}$$

$$db^{[2]} = dz^{[2]}$$

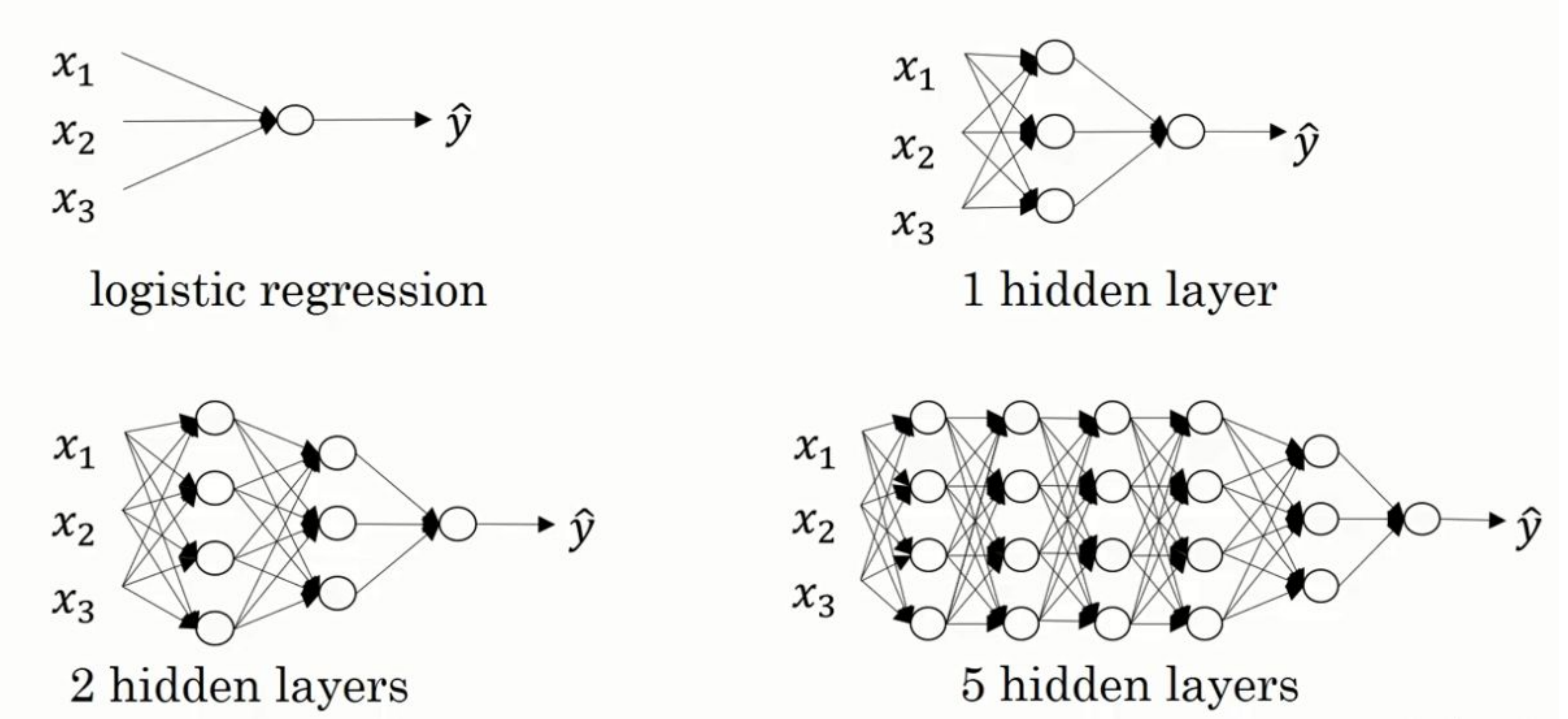
$$dz^{[1]} = \frac{\partial\, l(a^{[2]},y)}{\partial z^{[2]}} * \frac{\partial z^{[2]}}{\partial a^{[1]}} * \frac{\partial a^{[1]}}{\partial z^{[1]}}$$

$$\underset{(n^{[1]},1)}{\mathbf{dz}^{[1]}} = \underset{(n^{[1]},n^{[2]})}{\mathbf{w}^{[2]T}} * \underset{(n^{[2]},1)}{\mathbf{dz}^{[2]}} * \underset{(n^{[1]},1)}{\mathbf{g}^{[1]\prime}(\mathbf{z}^{[1]})} \Longrightarrow$$

$$\mathbf{dw}^{[1]} = \mathbf{dz}^{[1]} * \mathbf{x}^{T}$$

$$\mathbf{db}^{[1]} = \mathbf{dz}^{[1]}$$

Multi hidden layers



Multi hidden layers

Forward propagation

$$Z^{[L]} = W^{[L]} A^{[L-1]} + b^{[L]}$$

$(\boldsymbol{n}^{[\mathrm{l}]},\mathrm{m})$ $(\boldsymbol{n}^{[\mathrm{L}]}, \boldsymbol{n}^{[\mathrm{l}-1]})$ $(\boldsymbol{n}^{[\mathrm{L}-1]},\mathrm{m})$ $(\boldsymbol{n}^{[L]},1)$

broadcasted

$$A^{[L]} = g(Z^{[L]})$$

$(\boldsymbol{n}^{[\mathrm{l}]},\mathrm{m})$ $(\boldsymbol{n}^{[\mathrm{l}]},\mathrm{m})$

Multi hidden layers

Back propagation

Element wise multiplication

$$dZ^{[L]} = dA^{[L]} * g^{[L]\prime}(z^{[L]}) = dW^{[L+1]T} \ dZ^{[L+1]} * g^{[L]\prime}(z^{[L]})$$

$(\boldsymbol{n}^{[l]},m)$ $(\boldsymbol{n}^{[l]},m)$ $(\boldsymbol{n}^{[l]},m)$ $(\boldsymbol{n}^{[L]}, \boldsymbol{n}^{[l+1]})$ $(\boldsymbol{n}^{[L+1]}, \boldsymbol{n}^{[l]})$ $(\boldsymbol{n}^{[l]},m)$

$$dW^{[L]} = \frac{1}{m} dZ^{[L]} A^{[L-1]T}$$

$(\boldsymbol{n}^{[l]}, \boldsymbol{n}^{[l-1]})$ $(\boldsymbol{n}^{[l]},m)$ $(m, \boldsymbol{n}^{[l-1]})$

Vertical axis Keep it 2D matrix

$$db^{[L]} = \frac{1}{m} np.sum(dZ^{[L]}, axis = 1, keepdims = True)$$

$(\boldsymbol{n}^{[l]},1)$

$$dA^{[L-1]} = dW^{[L]T} \ dZ^{[L]}$$

$(\boldsymbol{n}^{[l-1]},m)$ $(\boldsymbol{n}^{[l-1]}, \boldsymbol{n}^{[l]})$ $(\boldsymbol{n}^{[l]},m)$

Goodby
e