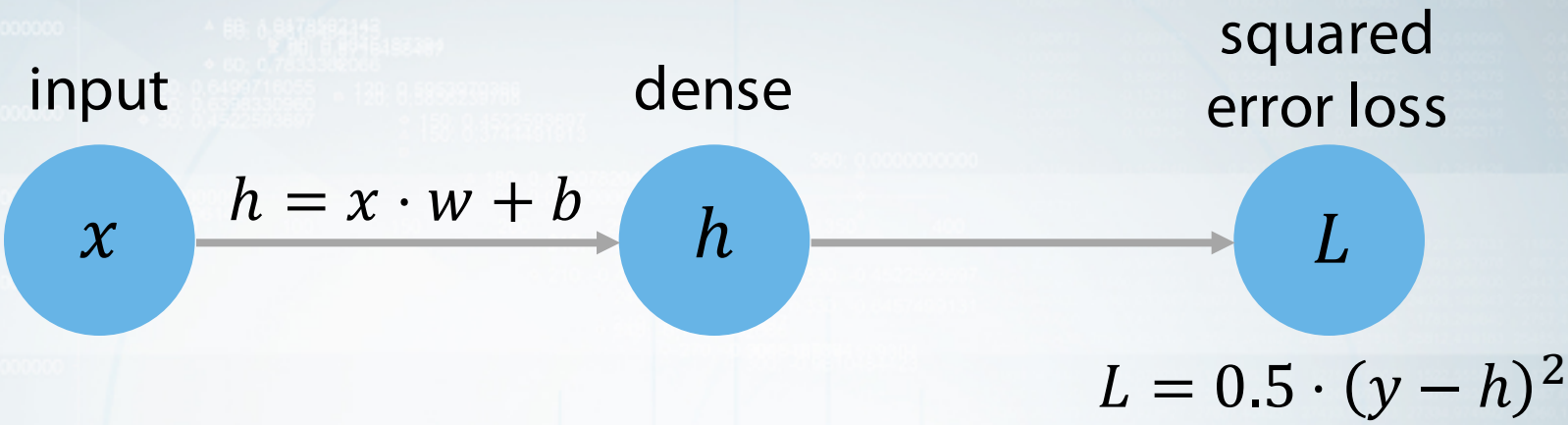
Backpropagation primer

In this video

- Step-by-step example of training a neural network via backprop
- You'll need the knowledge when using the advanced architectures

The simplest NN ever

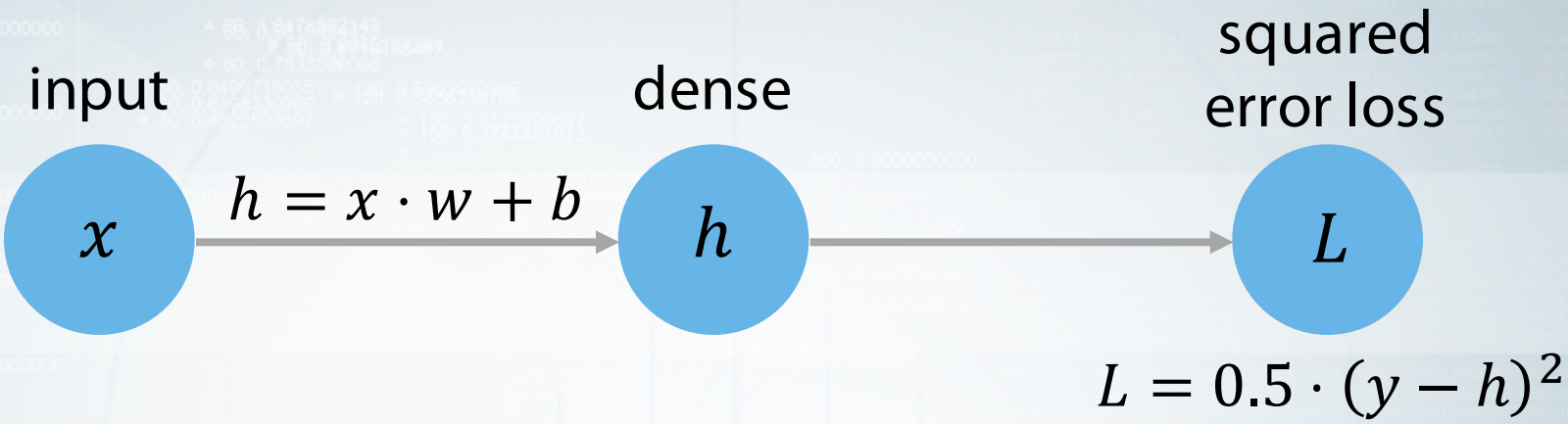


AKA
Least squares linear regression

Parameters:
Weight w, bias b

Input: x
Target: y

The simplest NN ever



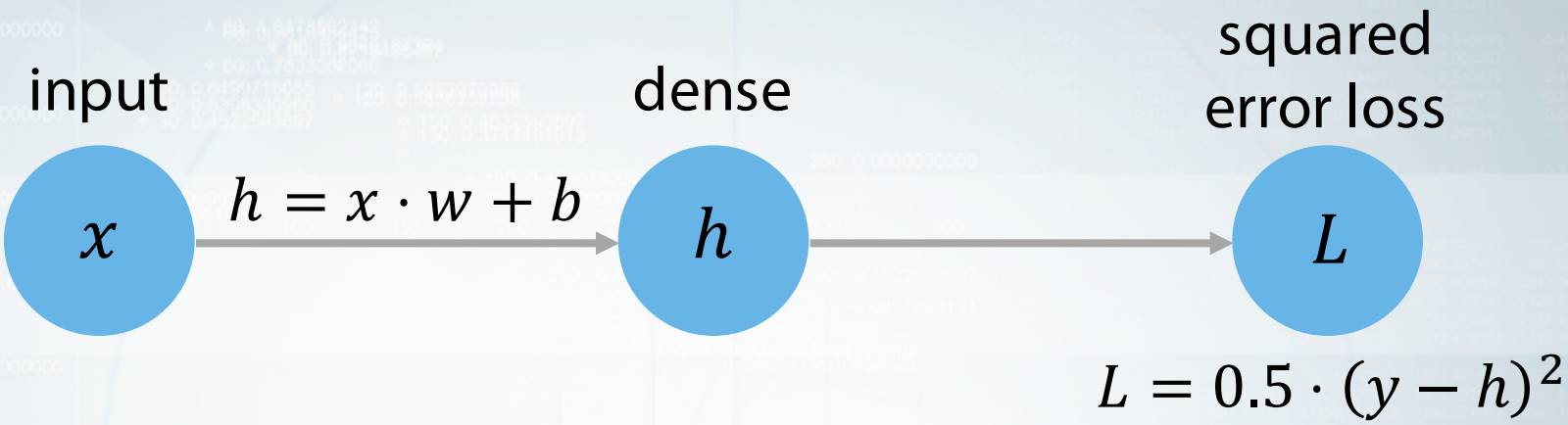
Parameters:
Weight w, bias b

Input: x
Target: y

L is just a function of parameters, features and target

$$L = f\big(y, g(x, w, b)\big)$$

The simplest NN ever

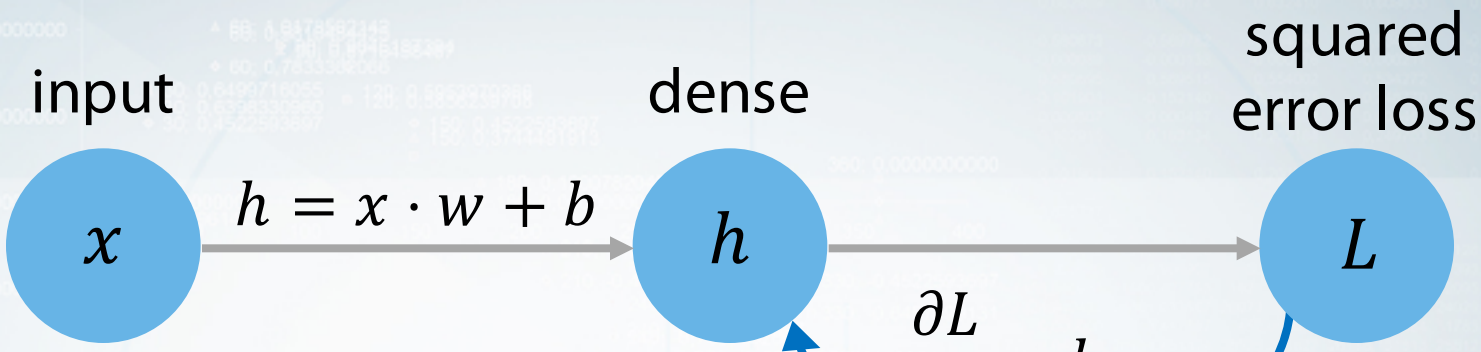


Gradient?

$$\frac{\partial L}{\partial w} = \frac{\partial L}{\partial h} \cdot \frac{\partial h}{\partial w}$$

$$\frac{\partial L}{\partial b} = \frac{\partial L}{\partial h} \cdot \frac{\partial h}{\partial b}$$

The simplest NN ever



$$\frac{\partial h}{\partial b} = 1$$

$$\frac{\partial h}{\partial w} = x$$

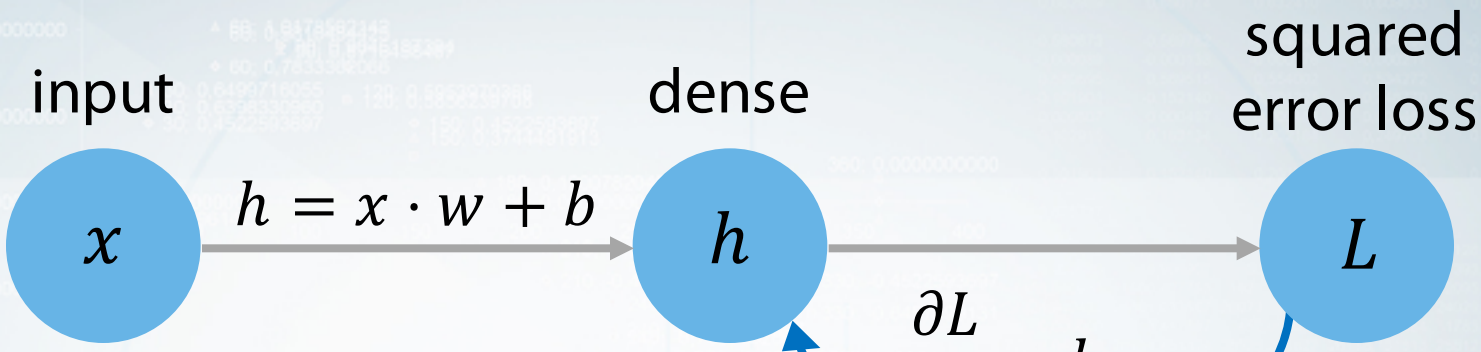
$$L = 0.5 \cdot (y - h)^2$$

Gradient?

$$\frac{\partial L}{\partial w} = \frac{\partial L}{\partial h} \cdot \frac{\partial h}{\partial w}$$

$$\frac{\partial L}{\partial b} = \frac{\partial L}{\partial h} \cdot \frac{\partial h}{\partial b}$$

The simplest NN ever



$$\frac{\partial h}{\partial b} = 1$$

$$\frac{\partial h}{\partial w} = x$$

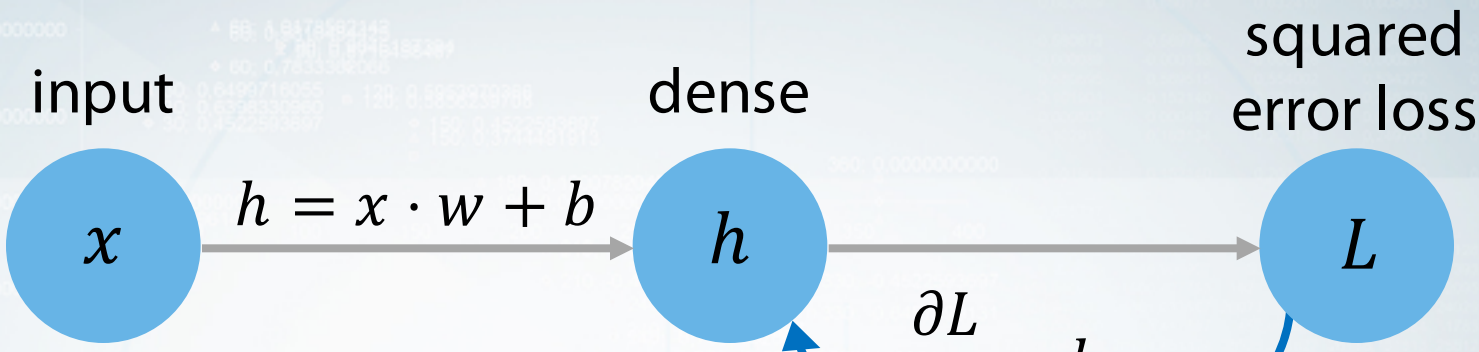
$$L = 0.5 \cdot (y - h)^2$$

Gradient?

$$\frac{\partial L}{\partial w} = \frac{\partial L}{\partial h} \cdot \frac{\partial h}{\partial w}$$

$$\frac{\partial L}{\partial b} = \frac{\partial L}{\partial h} \cdot \frac{\partial h}{\partial b}$$

The simplest NN ever



$$\frac{\partial h}{\partial b} = 1$$

$$\frac{\partial h}{\partial w} = x$$

$$L = 0.5 \cdot (y - h)^2$$

Let's fit

$y = 3$

$x = 1$

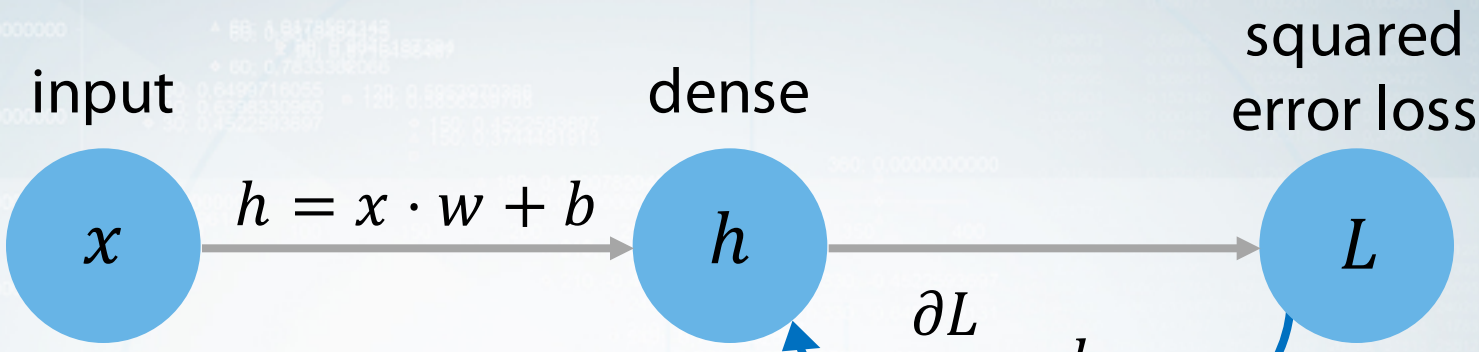
Initial

$w = 0.1$

$b = 1$

h	L	$\frac{\partial L}{\partial h}$	$\frac{\partial L}{\partial w}$	$\frac{\partial L}{\partial b}$	w	b
1.1	1.80					

The simplest NN ever



$$\frac{\partial h}{\partial b} = 1$$

$$\frac{\partial h}{\partial w} = x$$

$$L = 0.5 \cdot (y - h)^2$$

Let's fit

$y = 3$

$x = 1$

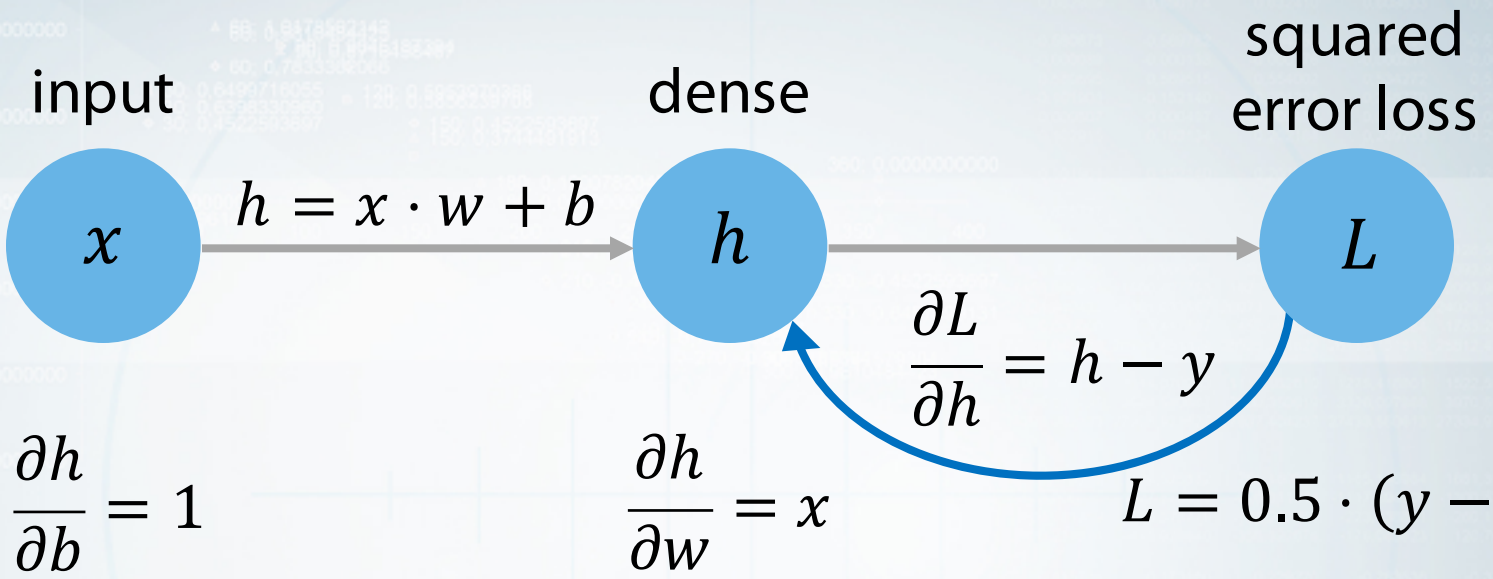
Initial

$w = 0.1$

$b = 1$

h	L	$\frac{\partial L}{\partial h}$	$\frac{\partial L}{\partial w}$	$\frac{\partial L}{\partial b}$	w	b
1.1	1.80	-1.9	-1.9	-1.9		

The simplest NN ever



Parameters update

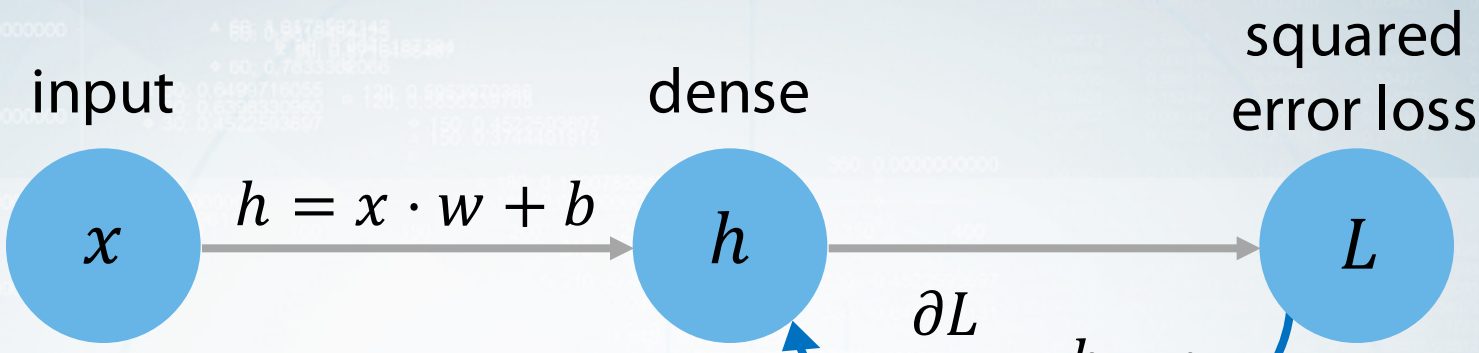
$$w -= \eta \frac{\partial L}{\partial w}$$

$$b -= \eta \frac{\partial L}{\partial b}$$

$$\eta = 0.2$$

h	L	$\frac{\partial L}{\partial h}$	$\frac{\partial L}{\partial w}$	$\frac{\partial L}{\partial b}$	w	b
1.1	1.80	-1.9	-1.9	-1.9	0.48	1.38

The simplest NN ever



Parameters update

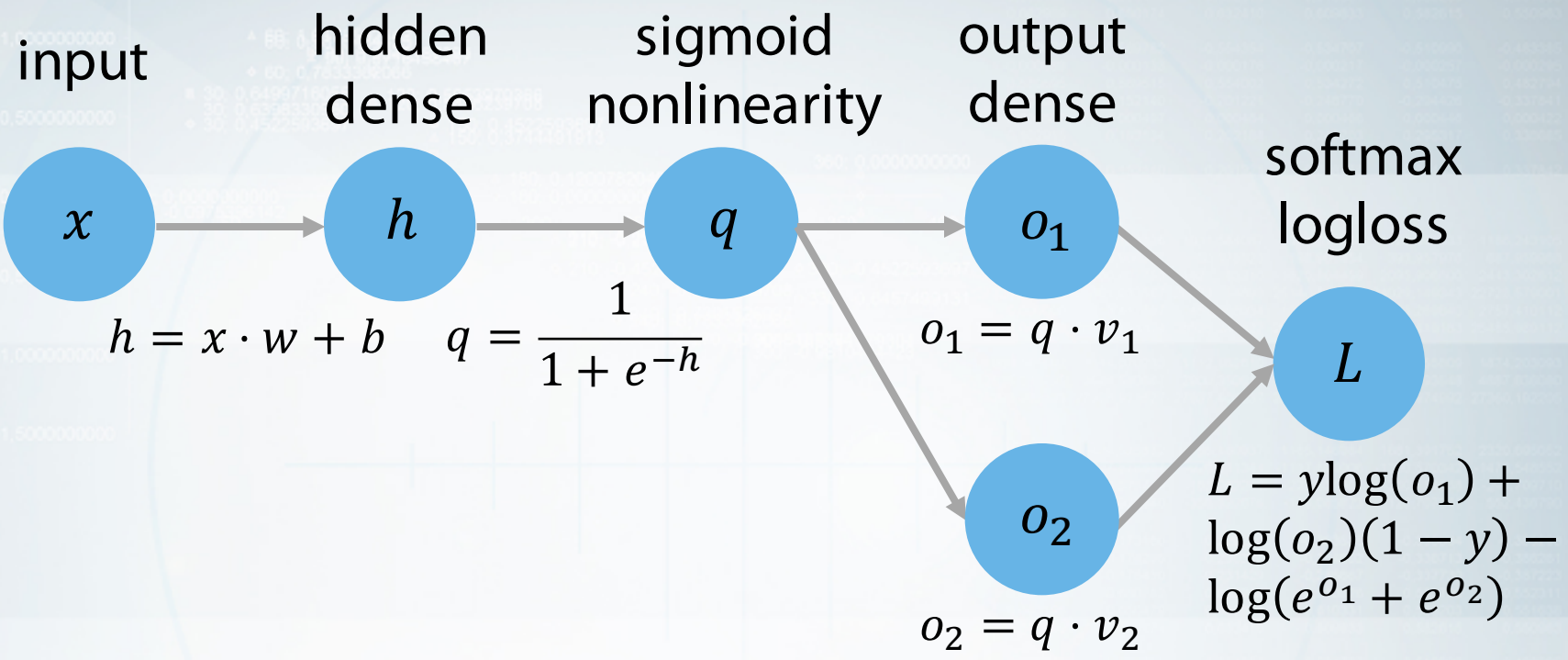
$$w -= \eta \frac{\partial L}{\partial w}$$

$$b -= \eta \frac{\partial L}{\partial b}$$

$$\eta = 0.2$$

h	L	$\frac{\partial L}{\partial h}$	$\frac{\partial L}{\partial w}$	$\frac{\partial L}{\partial b}$	w	b
1.1	1.80	-1.9	-1.9	-1.9	0.48	1.38
1.86	0.65	-1.14	-1.14	-1.14	0.71	1.61
2.32	0.23	-0.68	-0.68	-0.68	0.84	1.75

What if we go deeper?

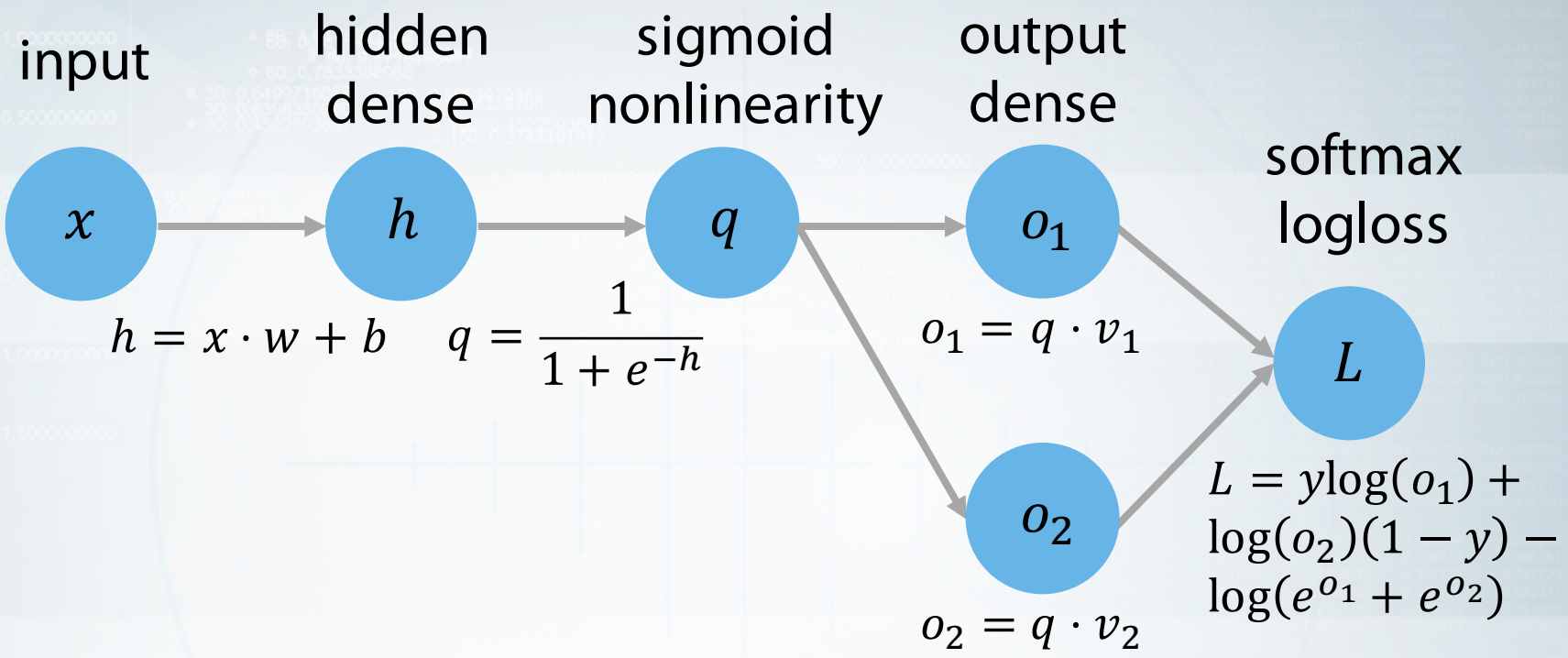


Parameters:

Weight w, bias b

Weight v_1, v_2

What if we go deeper?



Parameters:

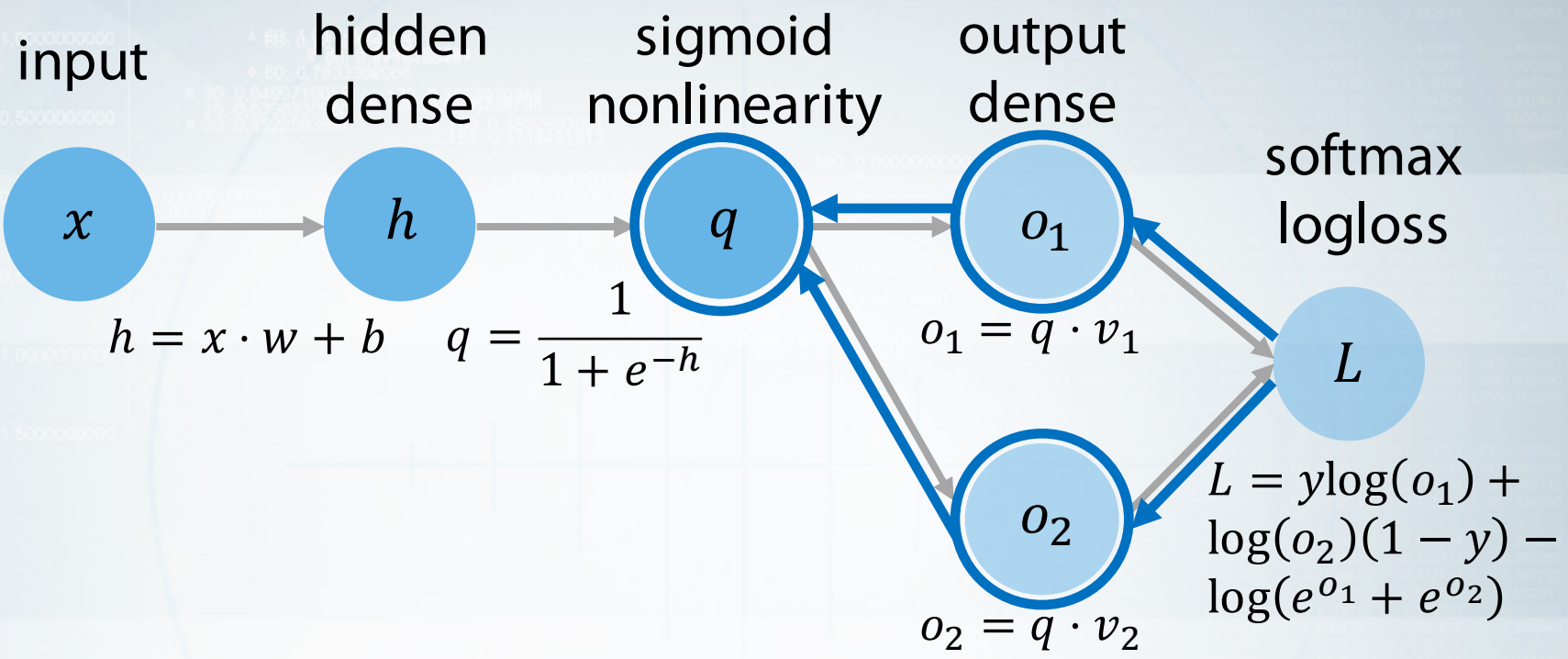
Weight w, bias b

Weight v_1, v_2

$$\frac{dL}{do_1} = \frac{y}{o_1} - \frac{e^{o_1}}{e^{o_2} + e^{o_1}}$$

$$\frac{dL}{do_2} = \frac{1 - y}{o_2} - \frac{e^{o_2}}{e^{o_2} + e^{o_1}}$$

What if we go deeper?



Parameters:

Weight w, bias b

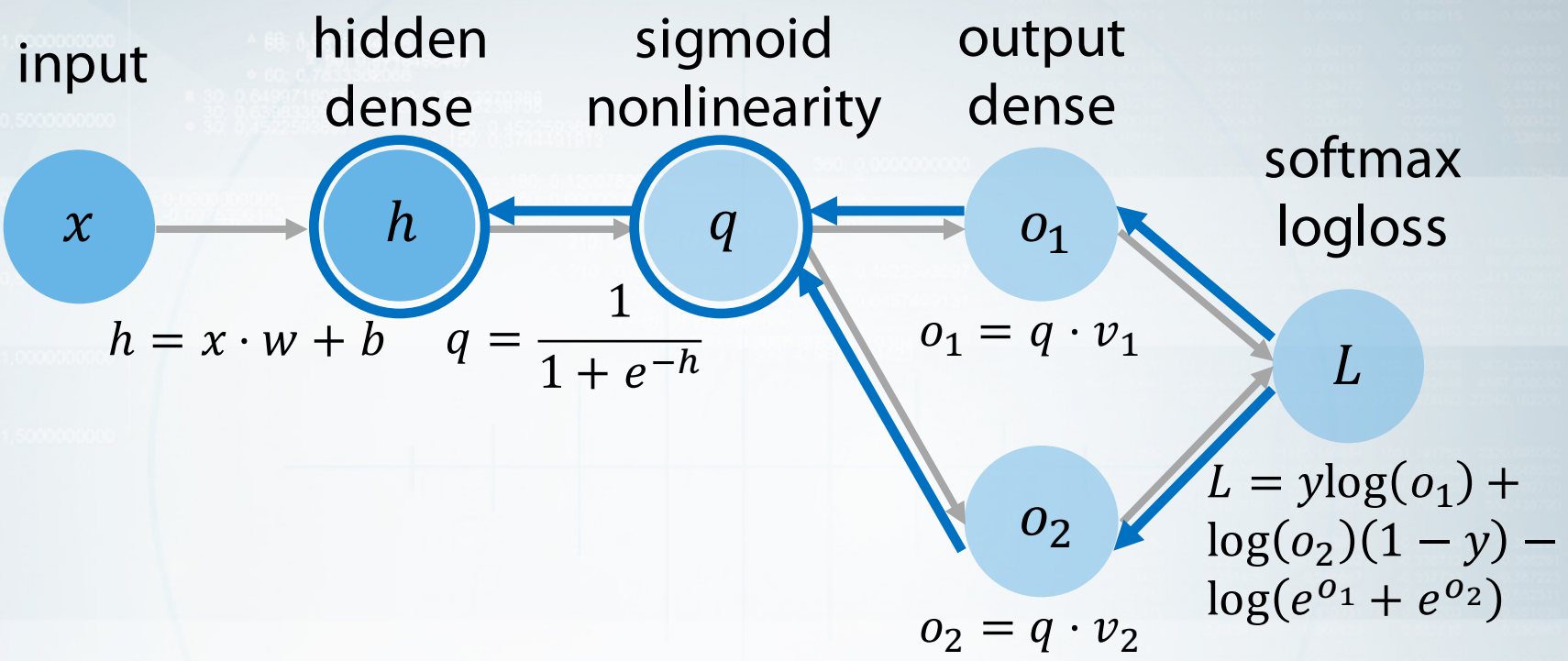
Weight v_1, v_2

$$\frac{\partial L}{\partial q} = v_1 \cdot \frac{\partial L}{\partial o_1} + v_2 \cdot \frac{\partial L}{\partial o_2}$$

Update v_1, v_2

$$\frac{\partial L}{\partial v_1} = \frac{\partial L}{\partial o_1} \cdot q \quad \frac{\partial L}{\partial v_2} = \frac{\partial L}{\partial o_2} \cdot q$$

What if we go deeper?



Parameters:

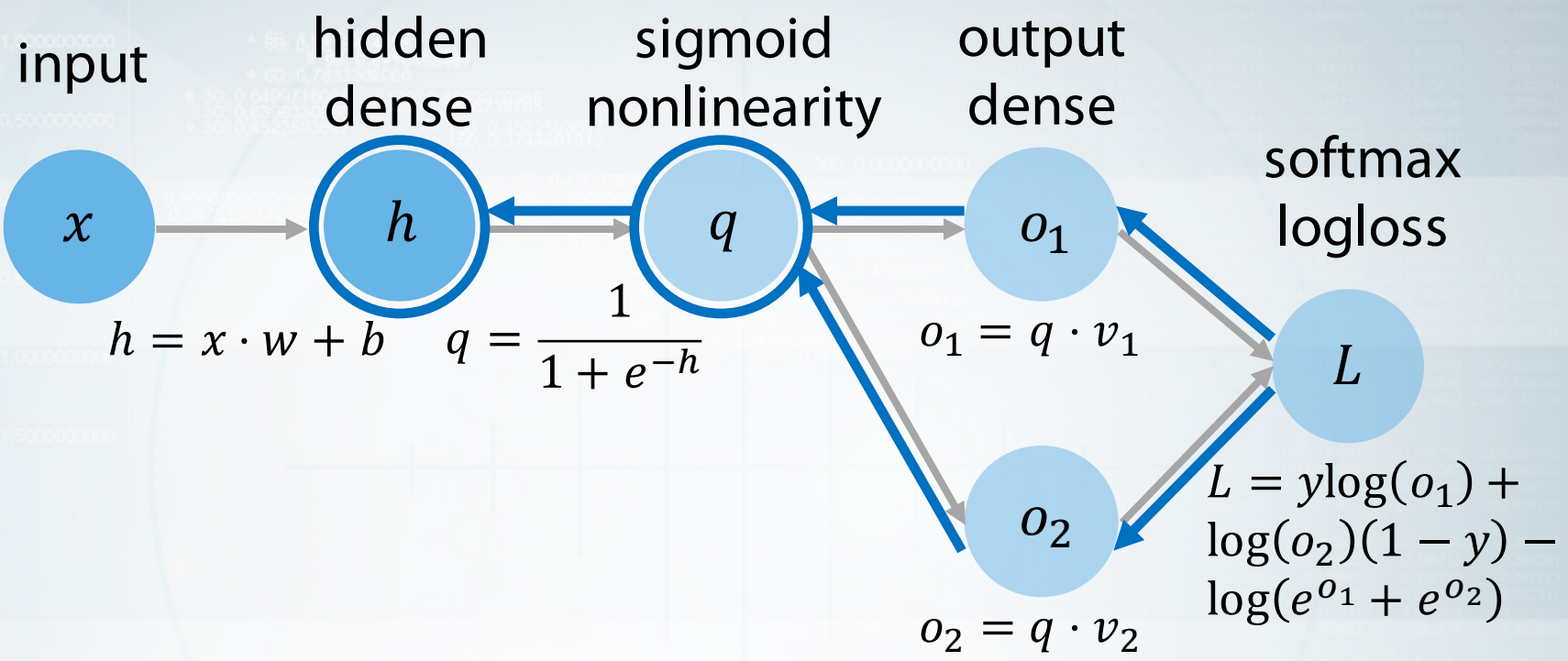
Weight w, bias b

Weight v_1, v_2

Question.

What will the derivative $\frac{\partial q}{\partial h}$ be?

What if we go deeper?



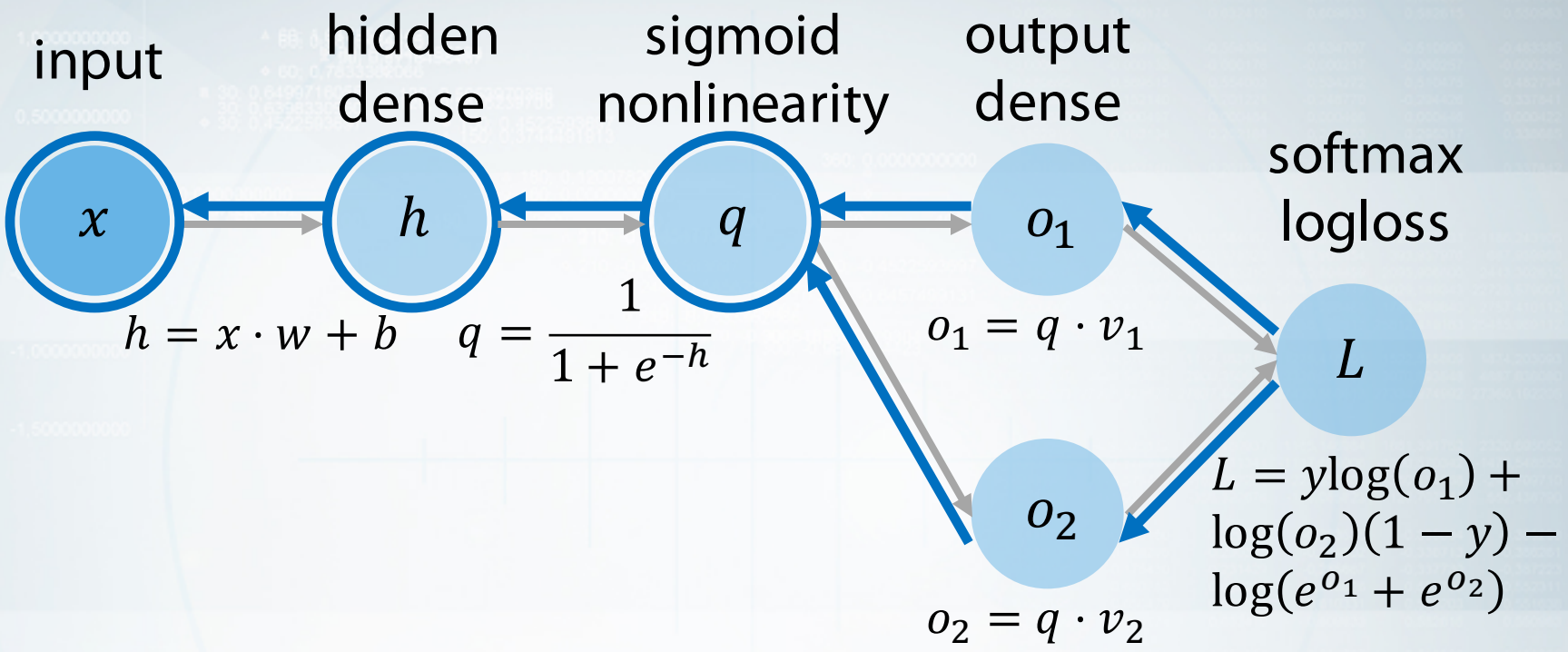
Parameters:

Weight w, bias b

Weight v_1, v_2

$$\frac{\partial L}{\partial h} = \frac{\partial L}{\partial q} \frac{e^{-q}}{(1 + e^{-q})^2}$$

What if we go deeper?



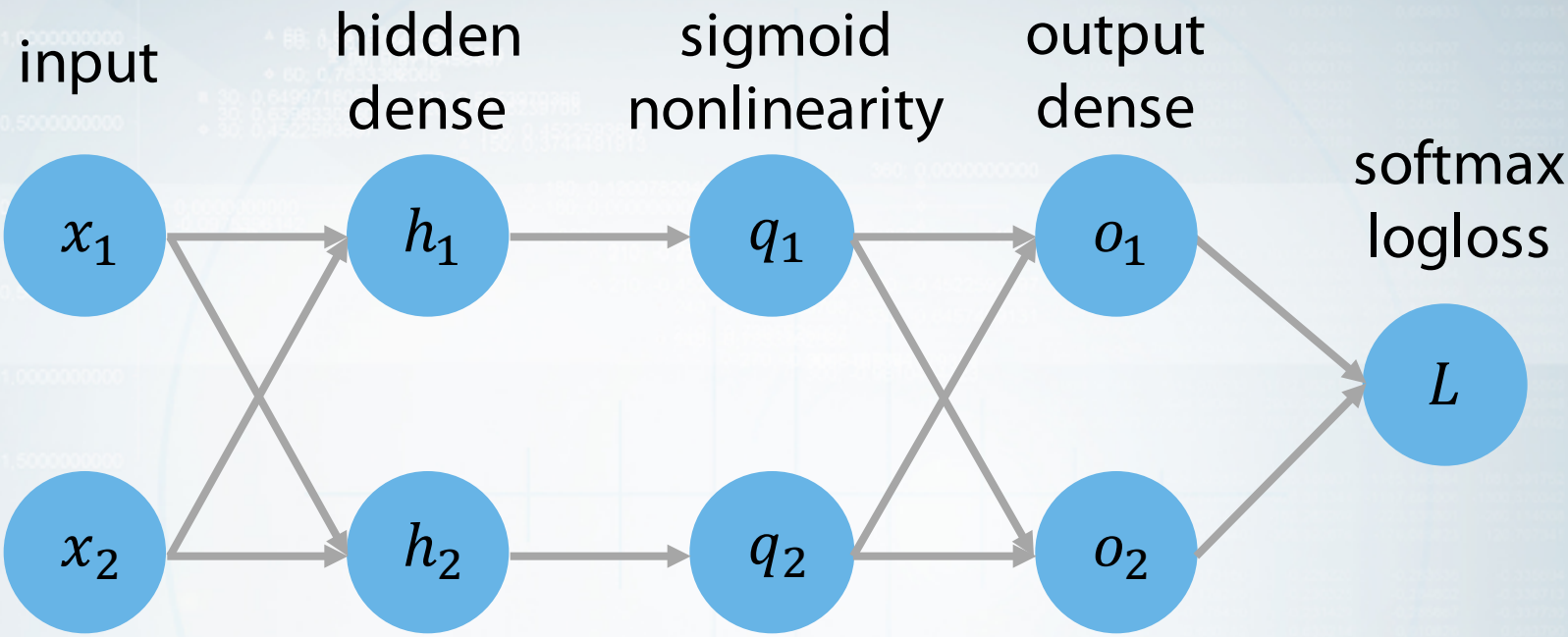
Parameters:

Weight w, bias b

Weight v_1, v_2

$$\frac{\partial L}{\partial w} = \frac{\partial L}{\partial h} \cdot x$$

What if we go wider?



$$\begin{bmatrix} h_1 \\ h_2 \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + b \qquad \begin{bmatrix} o_1 \\ o_2 \end{bmatrix} = \begin{bmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix} \cdot \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}$$

What if we go wider?

$$\frac{\partial(W \cdot \vec{x} + b)}{\partial \vec{x}} = ?$$

What if we go wider?

$$\frac{\partial(W \cdot \vec{x} + b)}{\partial \vec{x}} = \frac{\partial}{\partial \vec{x}} \left(\begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + b \right) =$$

$$= \frac{\partial}{\partial \vec{x}} \begin{bmatrix} w_{11}x_1 + w_{12}x_2 \\ w_{21}x_1 + w_{22}x_2 \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{bmatrix}$$

What if we go wider?

$$\frac{\partial(W \cdot \vec{x} + b)}{\partial \vec{x}} = \frac{\partial}{\partial \vec{x}} \left(\begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + b \right) =$$

$$= \frac{\partial}{\partial \vec{x}} \begin{bmatrix} w_{11}x_1 + w_{12}x_2 \\ w_{21}x_1 + w_{22}x_2 \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{bmatrix}$$

What if we go wider?

$$\frac{\partial(W \cdot \vec{x} + b)}{\partial W} = \frac{\partial}{\partial W}\begin{bmatrix} w_{11}x_1 + w_{12}x_2 \\ w_{21}x_1 + w_{22}x_2 \end{bmatrix}$$

$$= \begin{bmatrix} \begin{bmatrix} x_1 \\ 0 \end{bmatrix} \begin{bmatrix} x_2 \\ 0 \end{bmatrix} \\ \begin{bmatrix} 0 \\ x_1 \end{bmatrix} \begin{bmatrix} 0 \\ x_2 \end{bmatrix} \end{bmatrix}$$

What's in it for me?

You can have any crazy layer as long as you can compute its gradient

- No need to compute the gradients by hand
- There are frameworks for that