Computer Vision with Deep Learning

By

SUBRAHMANYAM MURALA
CVPR Lab
School of Computer Science and Statistics
Trinity College Dublin, Ireland

Source: Fei-Fei Li, CS231n: Convolutional Neural Networks for Visual Recognition, Lectures.
Goodfellow et al., Deep Learning, MIT Press

Classification

- K- Nearest Neighbor
- Neural Network
- Deep Learning (CNN)

K- Nearest Neighbor

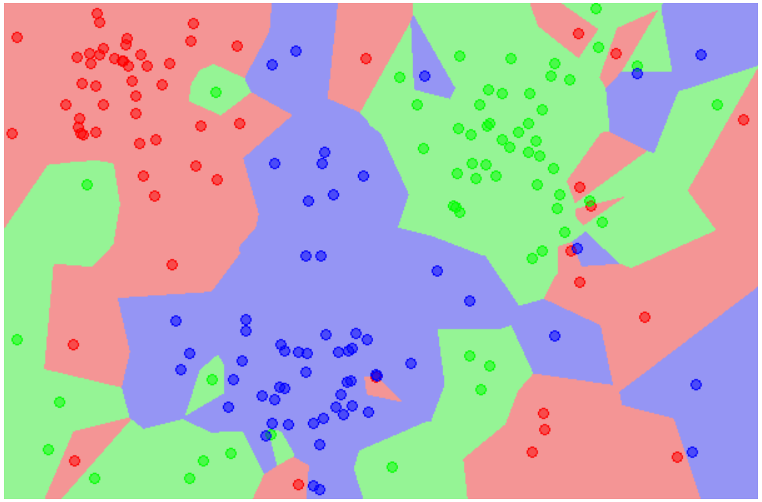
- **Most basic instance-based method**
- **Data represented in a vector space**
- **Supervised learning**

INSTANCE-BASED METHOD

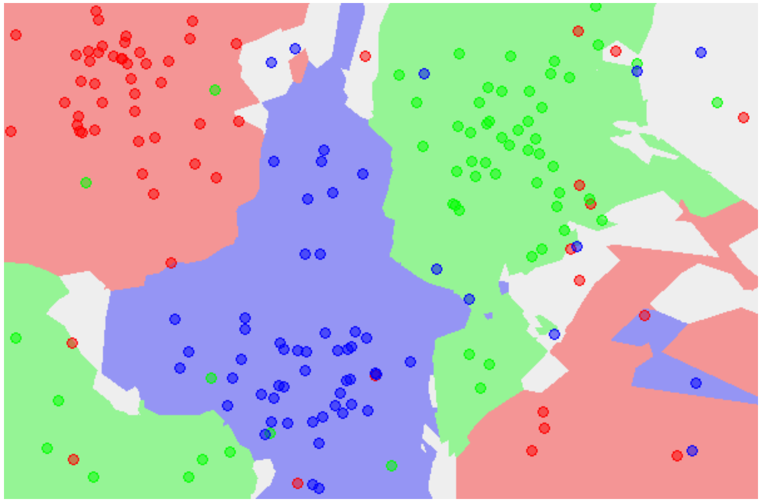
- Approximating real valued or discrete-valued target functions
- Learning in this algorithm consists of storing the presented training data
- When a new query instance is encountered, a set of similar related instances is retrieved from memory and used to classify the new query instance

K- Nearest Neighbor

- In *k-NN classification*, the output is a class membership.
- An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its *k* nearest neighbors (*k* is a positive integer, typically small).
- If *k* = 1, then the object is simply assigned to the class of that single nearest neighbor.



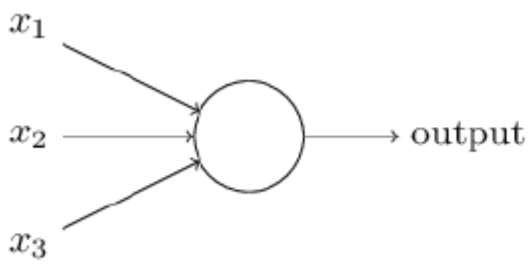
1-NN



5-NN

Neural Network

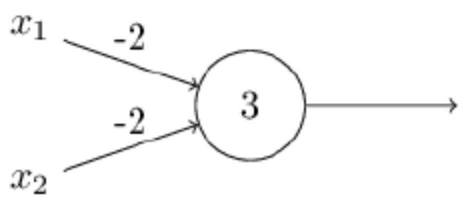
Perceptron: A perceptron takes several binary inputs, x1, x2,… and produces a single binary output.



$$\text{output} = \begin{cases} 0 & \text{if } \sum_j w_j x_j \leq \text{ threshold} \\ 1 & \text{if } \sum_j w_j x_j > \text{ threshold} \end{cases}$$

$$\text{output} = \begin{cases} 0 & \text{if } w \cdot x + b \leq 0 \\ 1 & \text{if } w \cdot x + b > 0 \end{cases}$$

Neural Network

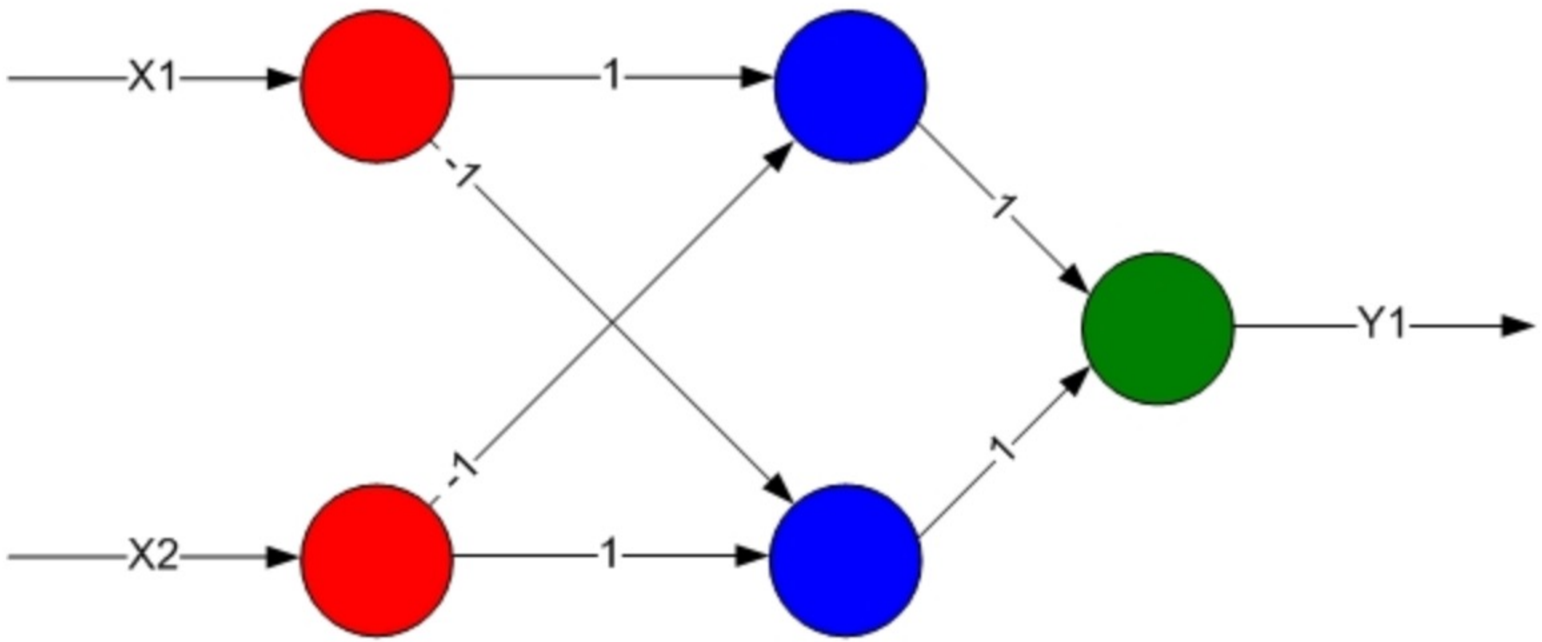


00	**1**
01	**1**
10	**1**
11	**0**

NAND

Neural Network

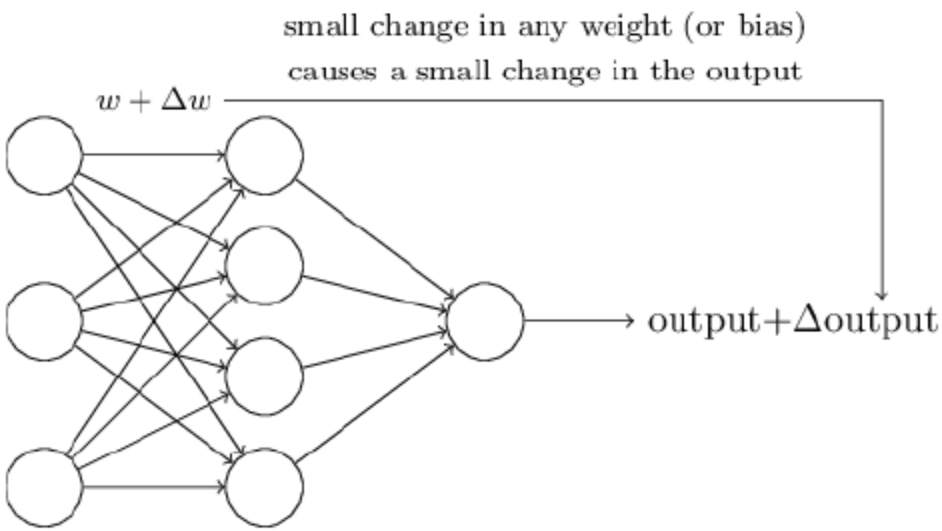
Y1 = XOR(X1, X2)



Neural Network

Sigmoid neurons

- Network need to learn weights and biases so that the output from the network correctly classifies the data.
- To see how learning might work, suppose we make a small change in some weight (or bias) in the network.
- What we'd like is for this small change in weight to cause only a small corresponding change in the output from the network.
- As we'll see in a moment, this property will make learning possible.



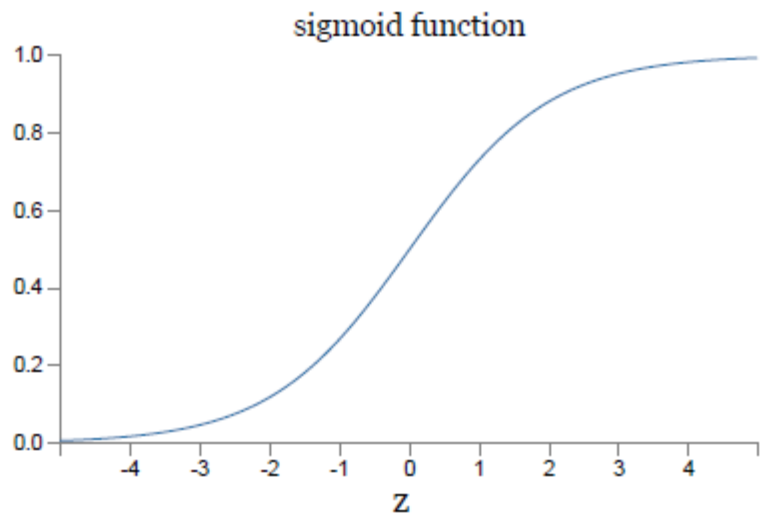
Neural Network

Sigmoid neurons

- When our network contains perceptrons, a small change in the weights or bias of any single perceptron in the network can sometimes cause the output of that perceptron to completely flip, say from 0 to 1.
- We can overcome this problem by introducing a new type of artificial neuron called a sigmoid neuron.

$$\sigma(z) \equiv \frac{1}{1+e^{-z}}.$$

$$\frac{1}{1+\exp(-\sum_j w_j x_j - b)}.$$



Neural Network

Tanh function

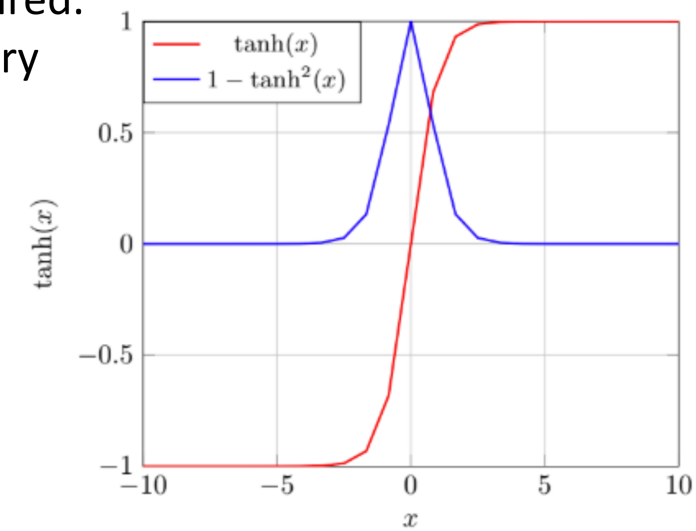
Tanh activation function take value between [-1, 1]. It's defined as follows.

$$T(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

Its derivative is

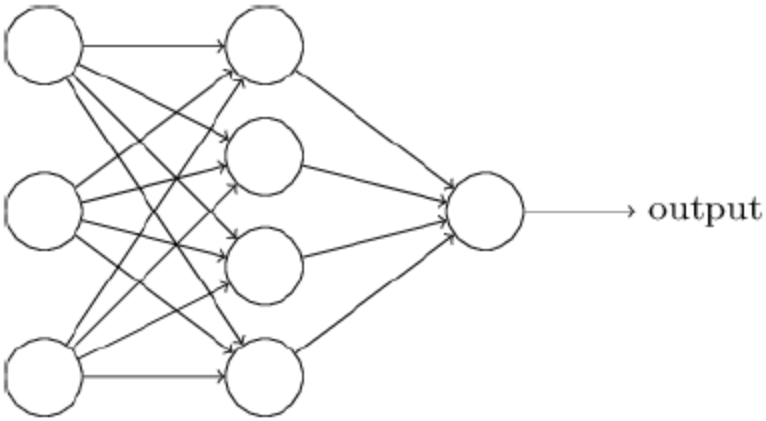
$$\tanh'(x) = 1 - \tanh^2(x)$$

- It's zero centred function.
- It's saturated function so rescaling of data is required.
- Because of exponential terms computationally very expensive.

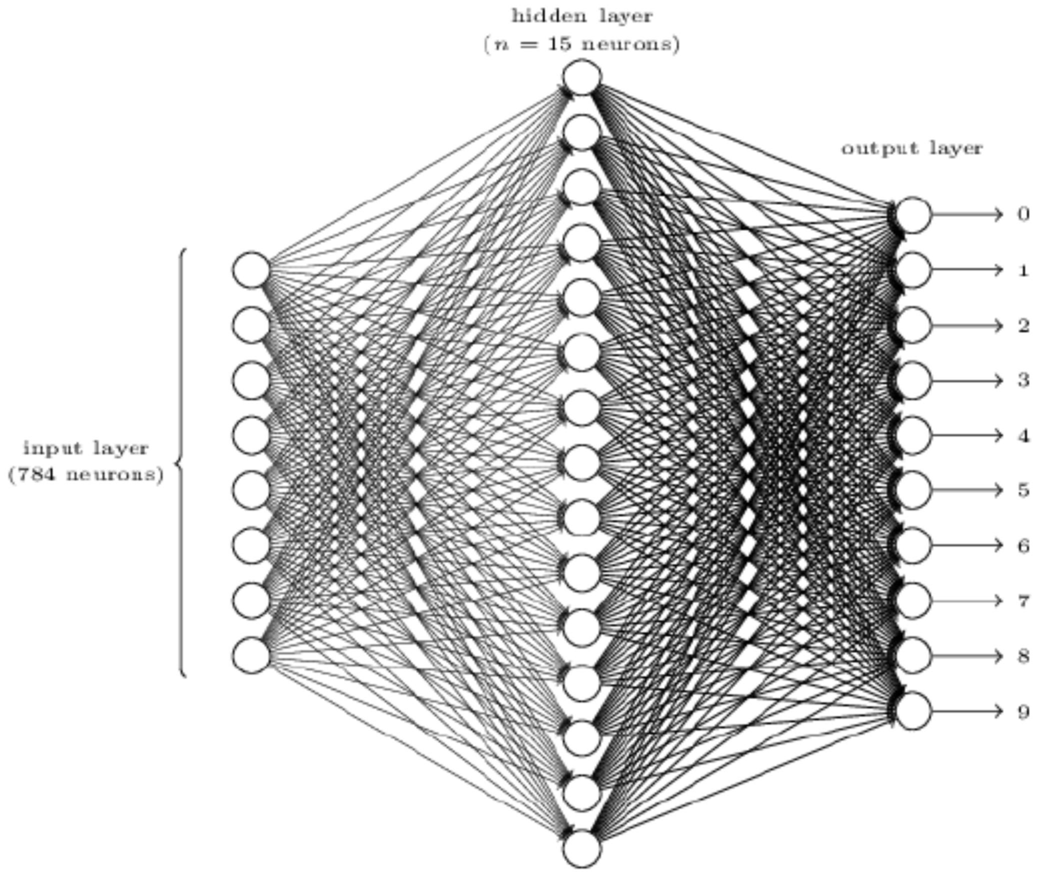


Neural Network

$$\Delta\text{output} \approx \sum_j \frac{\partial\,\text{output}}{\partial w_j} \Delta w_j + \frac{\partial\,\text{output}}{\partial b} \Delta b,$$



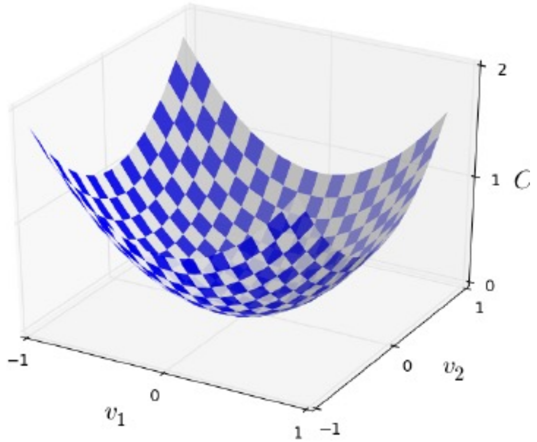
Neural Network



Neural Network

Learning with gradient descent

$$C(w,b) \equiv \frac{1}{2n} \sum_x \|y(x) - a\|^2.$$



$$\Delta C \approx \frac{\partial C}{\partial v_1} \Delta v_1 + \frac{\partial C}{\partial v_2} \Delta v_2.$$

$$\nabla C \equiv \left(\frac{\partial C}{\partial v_1}, \frac{\partial C}{\partial v_2} \right)^T.$$

$$\Delta v = (\Delta v_1, \ldots, \Delta v_m)^T$$

$$\Delta C \approx \nabla C \cdot \Delta v.$$

$$\Delta v = -\eta \nabla C,$$

$$\Delta C \approx -\eta \nabla C \cdot \nabla C = -\eta \|\nabla C\|^2$$

$$v \rightarrow v' = v - \eta \nabla C.$$

Neural Network

Learning with gradient descent

$$w_k \rightarrow w'_k = w_k - \eta \frac{\partial C}{\partial w_k}$$
$$b_l \rightarrow b'_l = b_l - \eta \frac{\partial C}{\partial b_l}.$$

$$C = \frac{1}{n} \sum_x C_x \qquad C_x \equiv \frac{\|y(x) - a\|^2}{2}$$

$$\nabla C = \frac{1}{n} \sum_x \nabla C_x.$$

Neural Network

Learning with stochastic gradient descent (SGD)

$$\frac{\sum_{j=1}^{m} \nabla C_{X_j}}{m} \approx \frac{\sum_x \nabla C_x}{n} = \nabla C,$$

$$\nabla C \approx \frac{1}{m} \sum_{j=1}^{m} \nabla C_{X_j},$$

$$w_k \rightarrow w'_k = w_k - \frac{\eta}{m} \sum_j \frac{\partial C_{X_j}}{\partial w_k}$$

$$b_l \rightarrow b'_l = b_l - \frac{\eta}{m} \sum_j \frac{\partial C_{X_j}}{\partial b_l},$$

Neural Network

The backpropagation algorithm

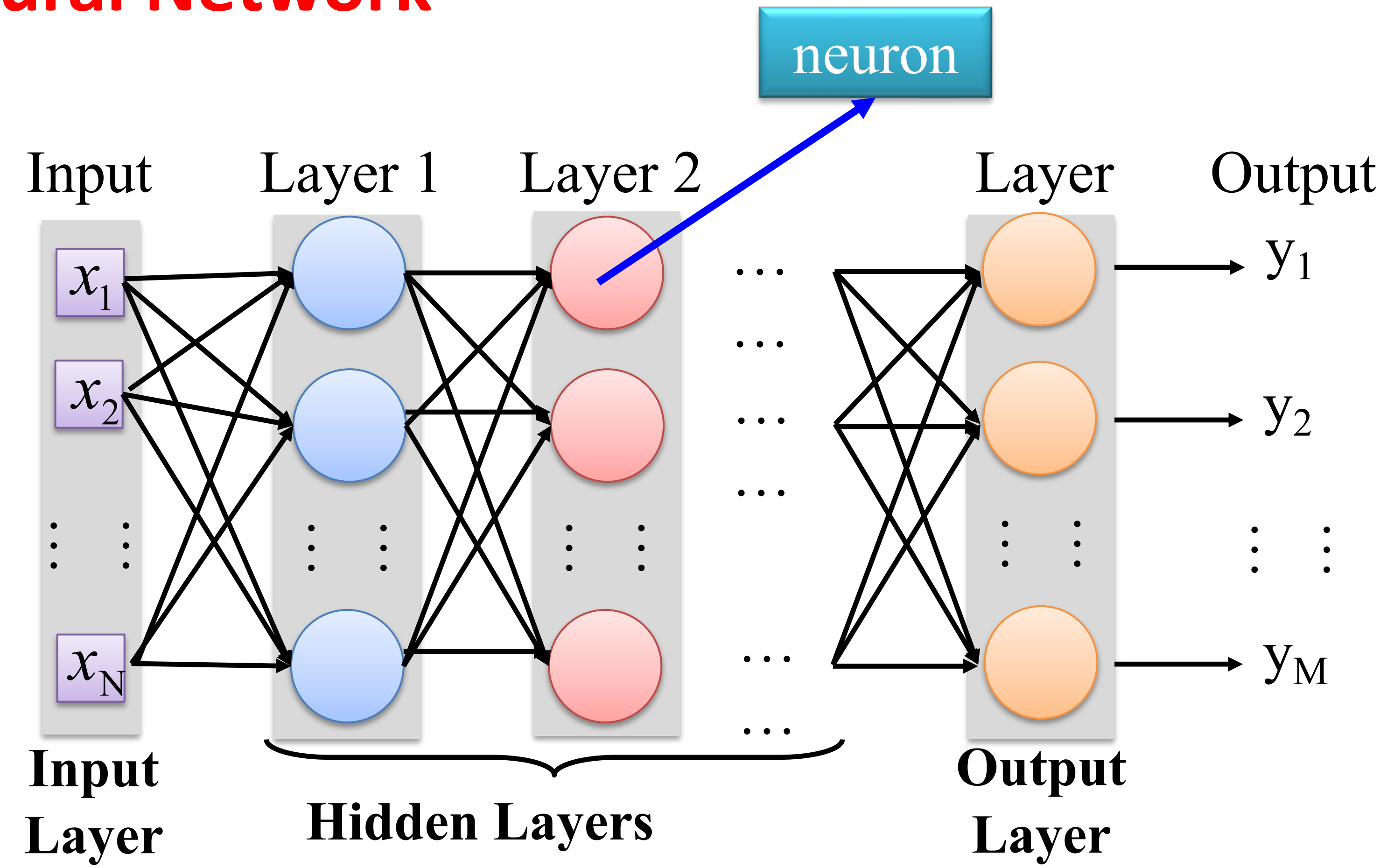
$$\frac{\partial C}{\partial w_j} \approx \frac{C(w + \epsilon e_j) - C(w)}{\epsilon},$$

$$C = \frac{(y-a)^2}{2},$$

$$\frac{\partial C}{\partial w} = (a-y)\sigma'(z)x$$

$$\frac{\partial C}{\partial b} = (a-y)\sigma'(z)$$

Neural Network



Deep means many hidden layers

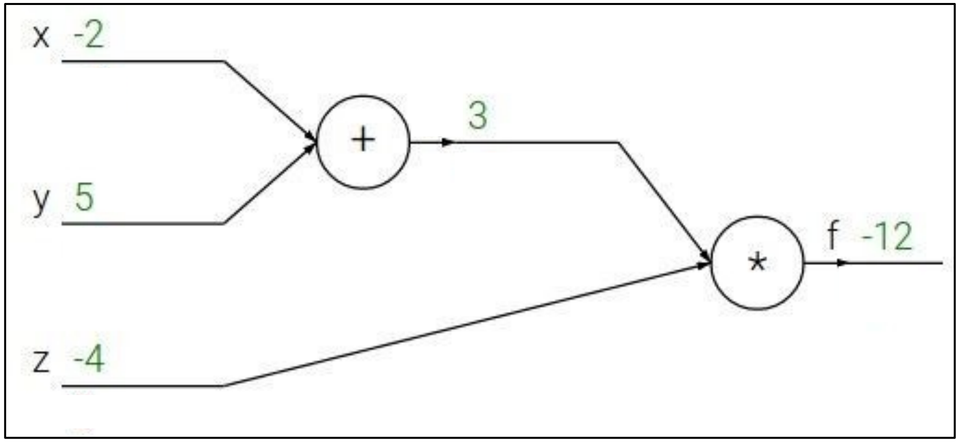
Slide credit: L. Lazebnik

Neural Network

Backpropagation: a simple example

$$f(x, y, z) = (x + y)z$$

e.g. x = -2, y = 5, z = -4

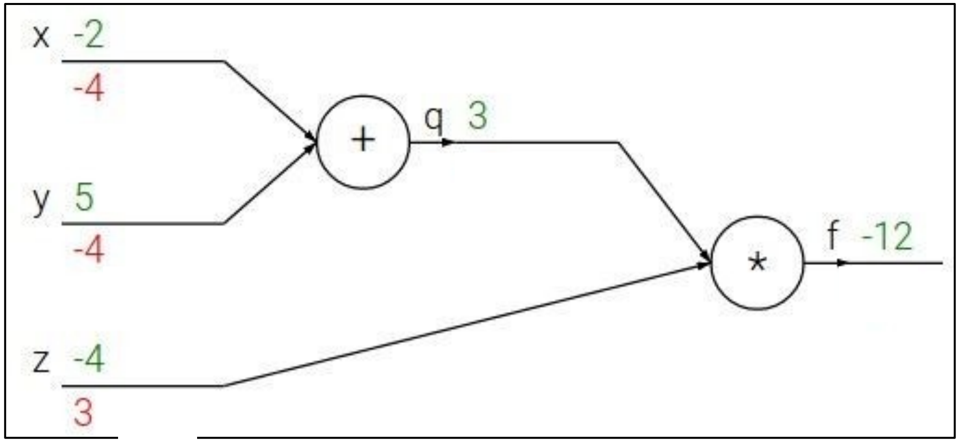


Slide credit: Fei-Fei Li

Classification

Neural Network

Backpropagation: a simple example



$$f(x, y, z) = (x + y)z$$

e.g. x = -2, y = 5, z = -4

$$q = x + y \qquad \frac{\partial q}{\partial x} = 1, \frac{\partial q}{\partial y} = 1$$

$$f = qz \qquad \frac{\partial f}{\partial q} = z, \frac{\partial f}{\partial z} = q$$

Want: $\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}$

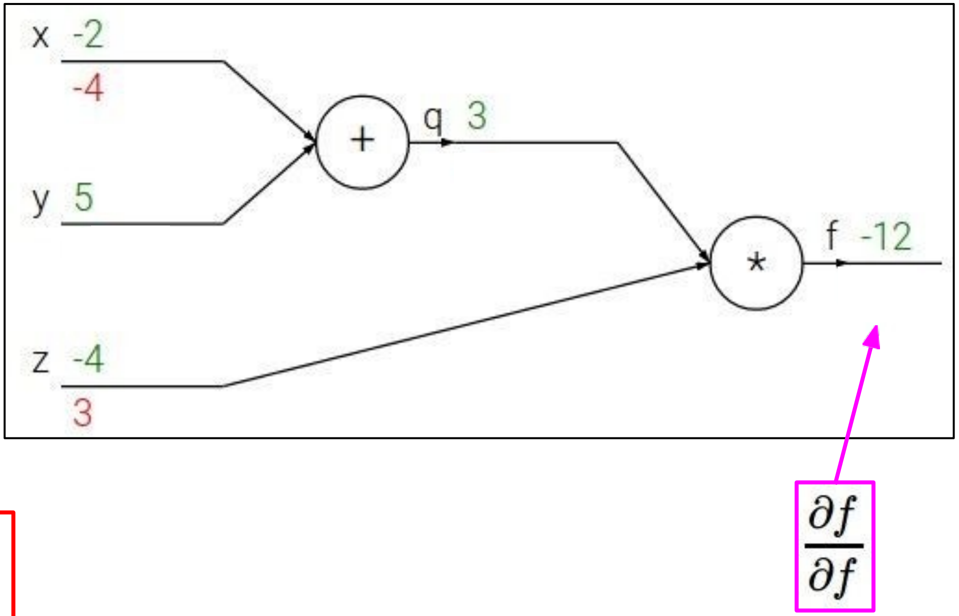
Classification

Neural Network

Backpropagation: a simple example

$$f(x, y, z) = (x + y)z$$

e.g. x = -2, y = 5, z = -4



$$q = x + y \qquad \frac{\partial q}{\partial x} = 1, \frac{\partial q}{\partial y} = 1$$

$$f = qz \qquad \frac{\partial f}{\partial q} = z, \frac{\partial f}{\partial z} = q$$

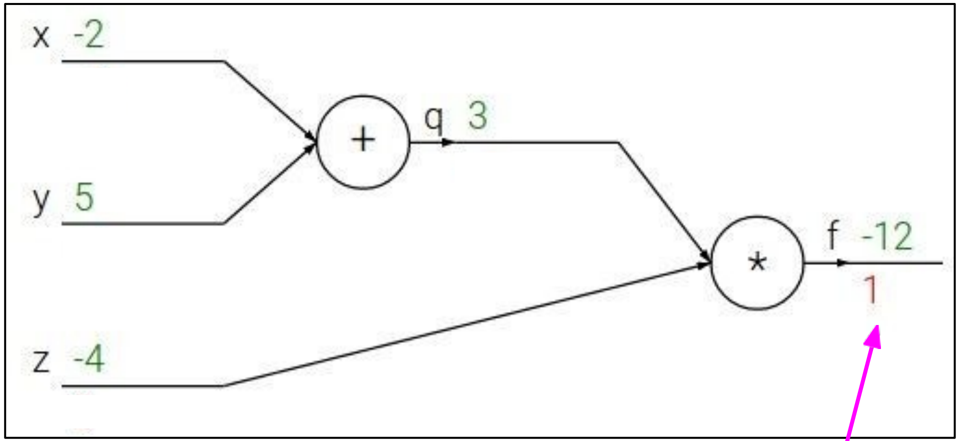
Want: $\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}$

Neural Network

Backpropagation: a simple example

$$f(x, y, z) = (x + y)z$$

e.g. x = -2, y = 5, z = -4



$$\frac{\partial f}{\partial f}$$

$$q = x + y \qquad \frac{\partial q}{\partial x} = 1, \frac{\partial q}{\partial y} = 1$$

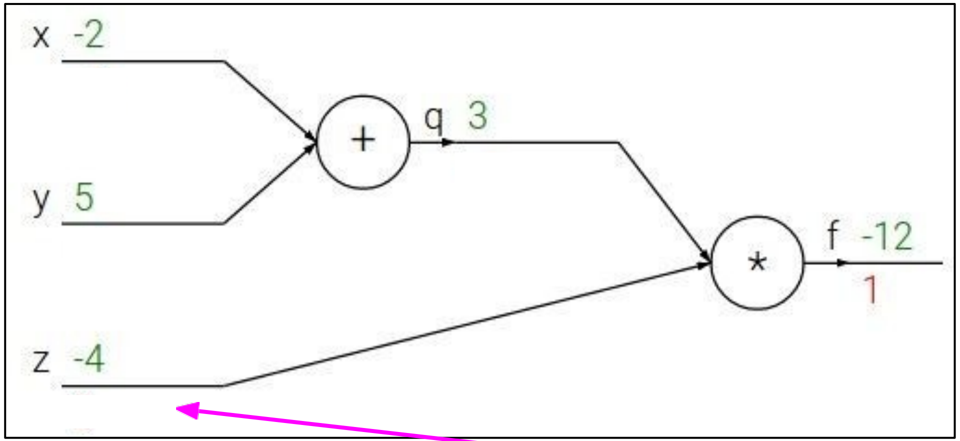
$$f = qz \qquad \frac{\partial f}{\partial q} = z, \frac{\partial f}{\partial z} = q$$

Want: $\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}$

Slide credit: Fei-Fei Li

Classification

Neural Network



$$\frac{\partial f}{\partial z}$$

Backpropagation: a simple example

$$f(x, y, z) = (x + y)z$$

e.g. x = -2, y = 5, z = -4

$$q = x + y \qquad \frac{\partial q}{\partial x} = 1, \frac{\partial q}{\partial y} = 1$$

$$f = qz \qquad \frac{\partial f}{\partial q} = z, \frac{\partial f}{\partial z} = q$$

Want: $\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}$

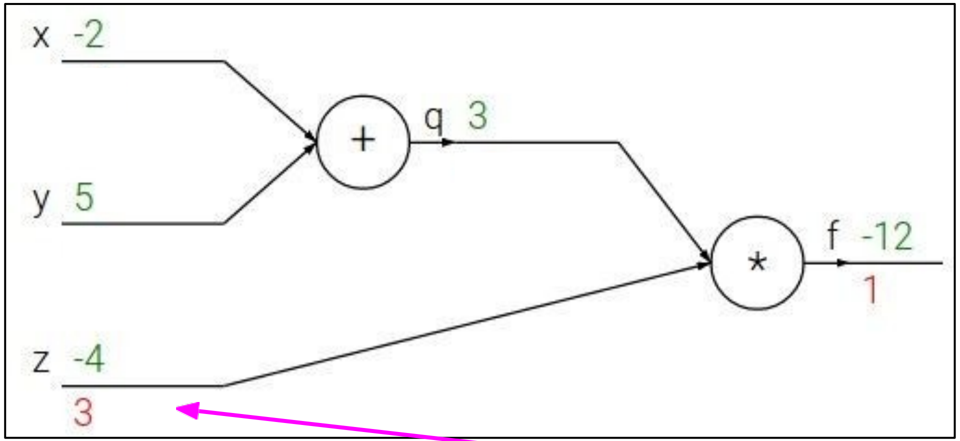
Slide credit: Fei-Fei Li

Neural Network

Backpropagation: a simple example

$$f(x, y, z) = (x + y)z$$

e.g. x = -2, y = 5, z = -4



$$\frac{\partial f}{\partial z}$$

$$q = x + y \qquad \frac{\partial q}{\partial x} = 1, \frac{\partial q}{\partial y} = 1$$

$$f = qz \qquad \frac{\partial f}{\partial q} = z, \frac{\partial f}{\partial z} = q$$

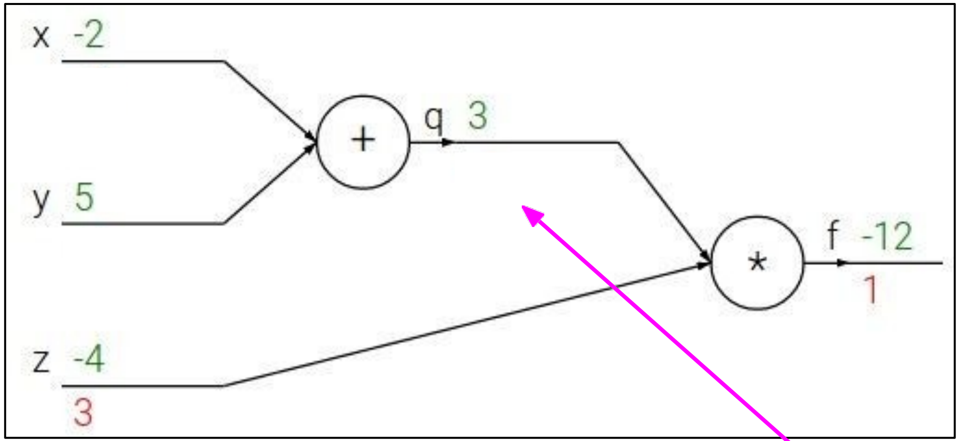
Want: $\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}$

Neural Network

Backpropagation: a simple example

$$f(x, y, z) = (x + y)z$$

e.g. x = -2, y = 5, z = -4



$$q = x + y \qquad \frac{\partial q}{\partial x} = 1, \frac{\partial q}{\partial y} = 1$$

$$f = qz \qquad \frac{\partial f}{\partial q} = z, \frac{\partial f}{\partial z} = q$$

$$\frac{\partial f}{\partial q}$$

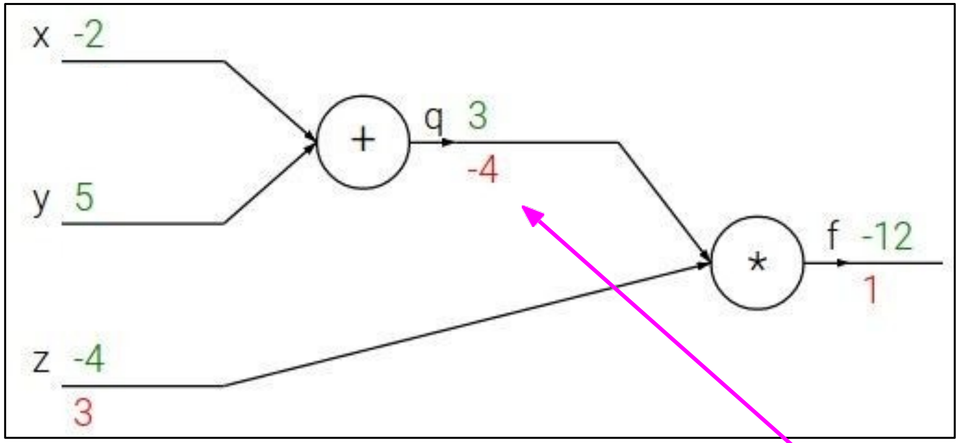
Want: $\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}$

Neural Network

Backpropagation: a simple example

$$f(x, y, z) = (x + y)z$$

e.g. x = -2, y = 5, z = -4



$$q = x + y \qquad \frac{\partial q}{\partial x} = 1, \frac{\partial q}{\partial y} = 1$$

$$f = qz \qquad \frac{\partial f}{\partial q} = z, \frac{\partial f}{\partial z} = q$$

$$\frac{\partial f}{\partial q}$$

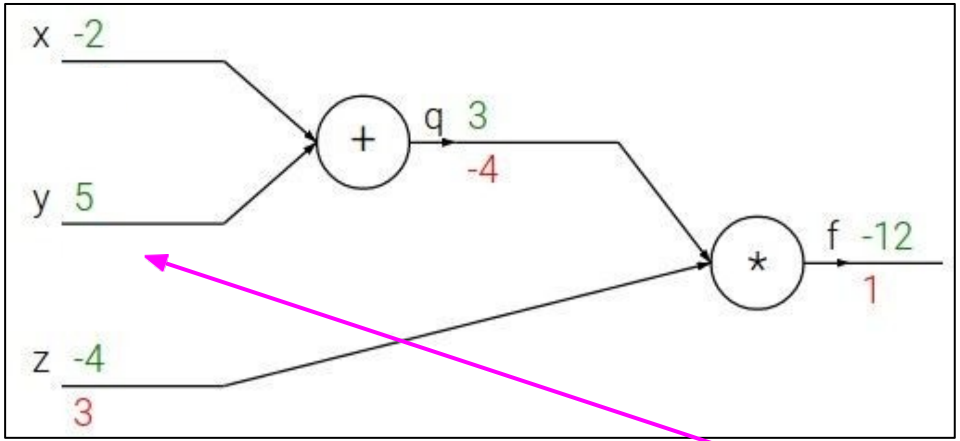
Want: $\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}$

Neural Network

Backpropagation: a simple example

$$f(x, y, z) = (x + y)z$$

e.g. x = -2, y = 5, z = -4



$$\frac{\partial f}{\partial y}$$

$$q = x + y \qquad \frac{\partial q}{\partial x} = 1, \frac{\partial q}{\partial y} = 1$$

$$f = qz \qquad \frac{\partial f}{\partial q} = z, \frac{\partial f}{\partial z} = q$$

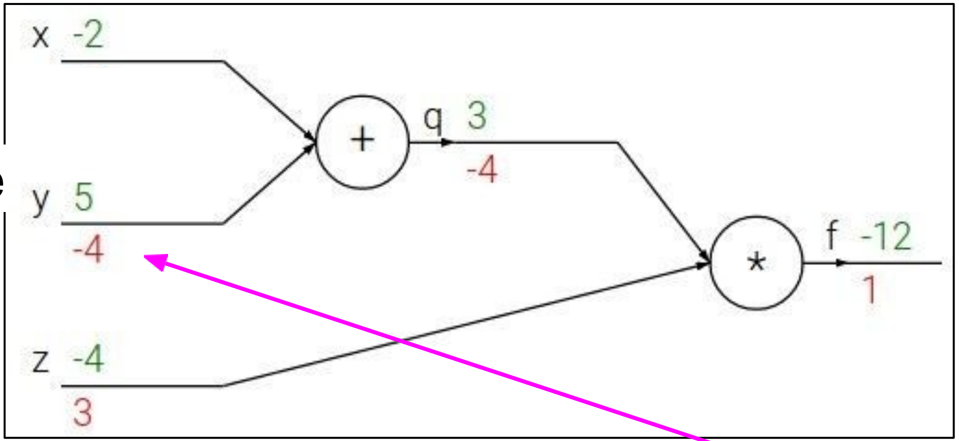
Want: $\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}$

Slide credit: Fei-Fei Li

Classification

Neural Network

Backpropagation: a simple example



$$f(x,y,z) = (x+y)z$$

e.g. x = -2, y = 5, z = -4

$$q = x + y \qquad \frac{\partial q}{\partial x} = 1, \frac{\partial q}{\partial y} = 1$$

$$f = qz \qquad \frac{\partial f}{\partial q} = z, \frac{\partial f}{\partial z} = q$$

Want: $\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}$

$$\frac{\partial f}{\partial y}$$

Chain rule:

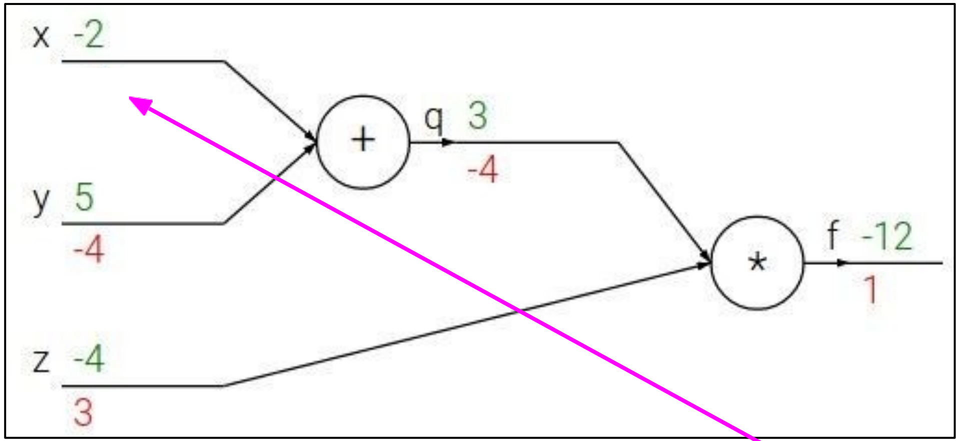
$$\frac{\partial f}{\partial y} = \frac{\partial f}{\partial q}\frac{\partial q}{\partial y}$$

Neural Network

Backpropagation: a simple example

$$f(x, y, z) = (x + y)z$$

e.g. x = -2, y = 5, z = -4



$$\frac{\partial f}{\partial x}$$

$$q = x + y \qquad \frac{\partial q}{\partial x} = 1, \frac{\partial q}{\partial y} = 1$$

$$f = qz \qquad \frac{\partial f}{\partial q} = z, \frac{\partial f}{\partial z} = q$$

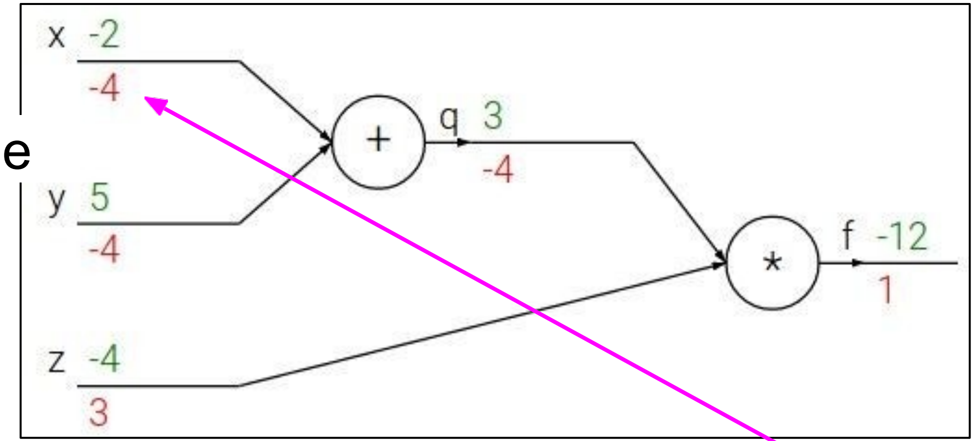
Want: $\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}$

Neural Network

Backpropagation: a simple example

$$f(x, y, z) = (x + y)z$$

e.g. x = -2, y = 5, z = -4



$$q = x + y \qquad \frac{\partial q}{\partial x} = 1, \frac{\partial q}{\partial y} = 1$$

$$f = qz \qquad \frac{\partial f}{\partial q} = z, \frac{\partial f}{\partial z} = q$$

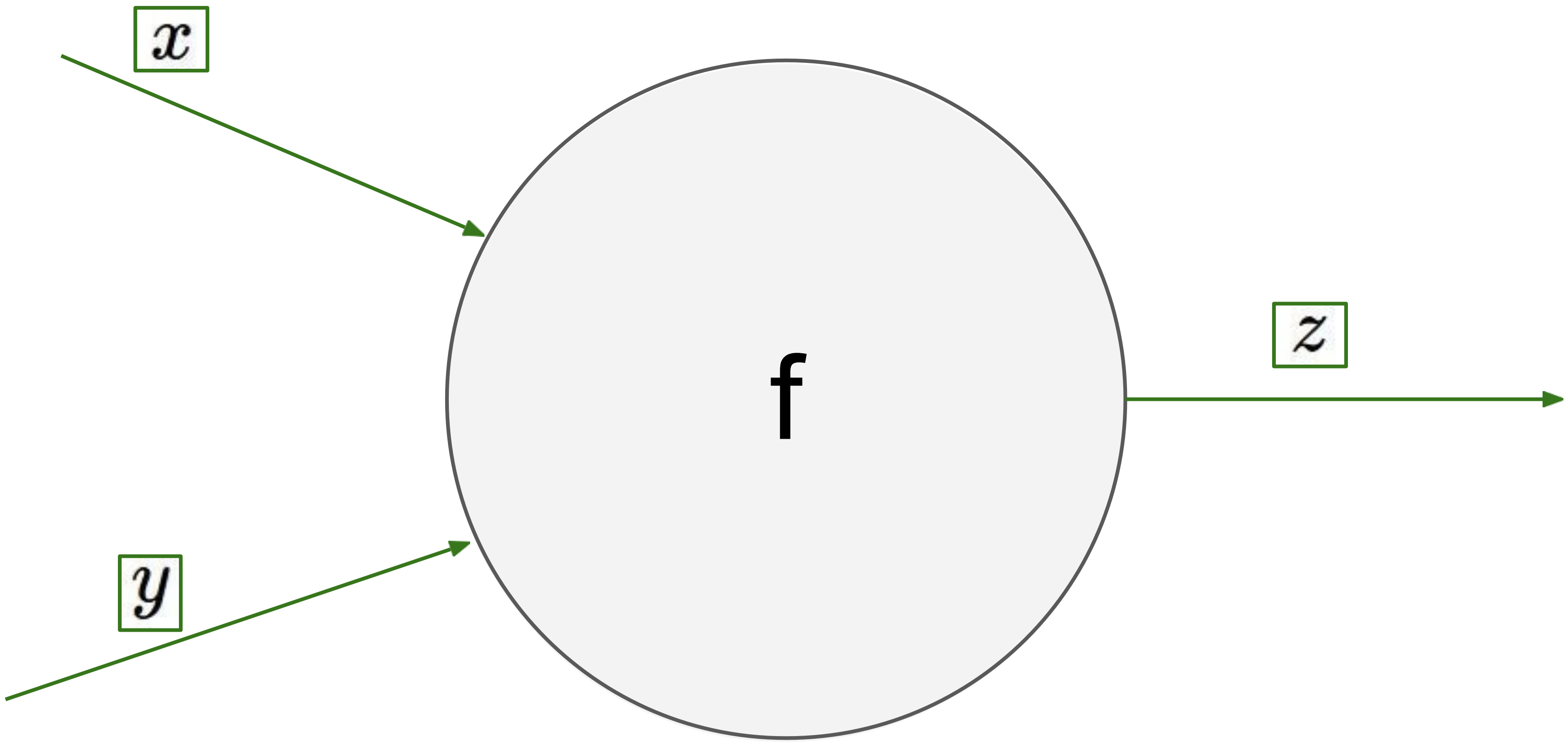
Want: $\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}$

$\frac{\partial f}{\partial x}$

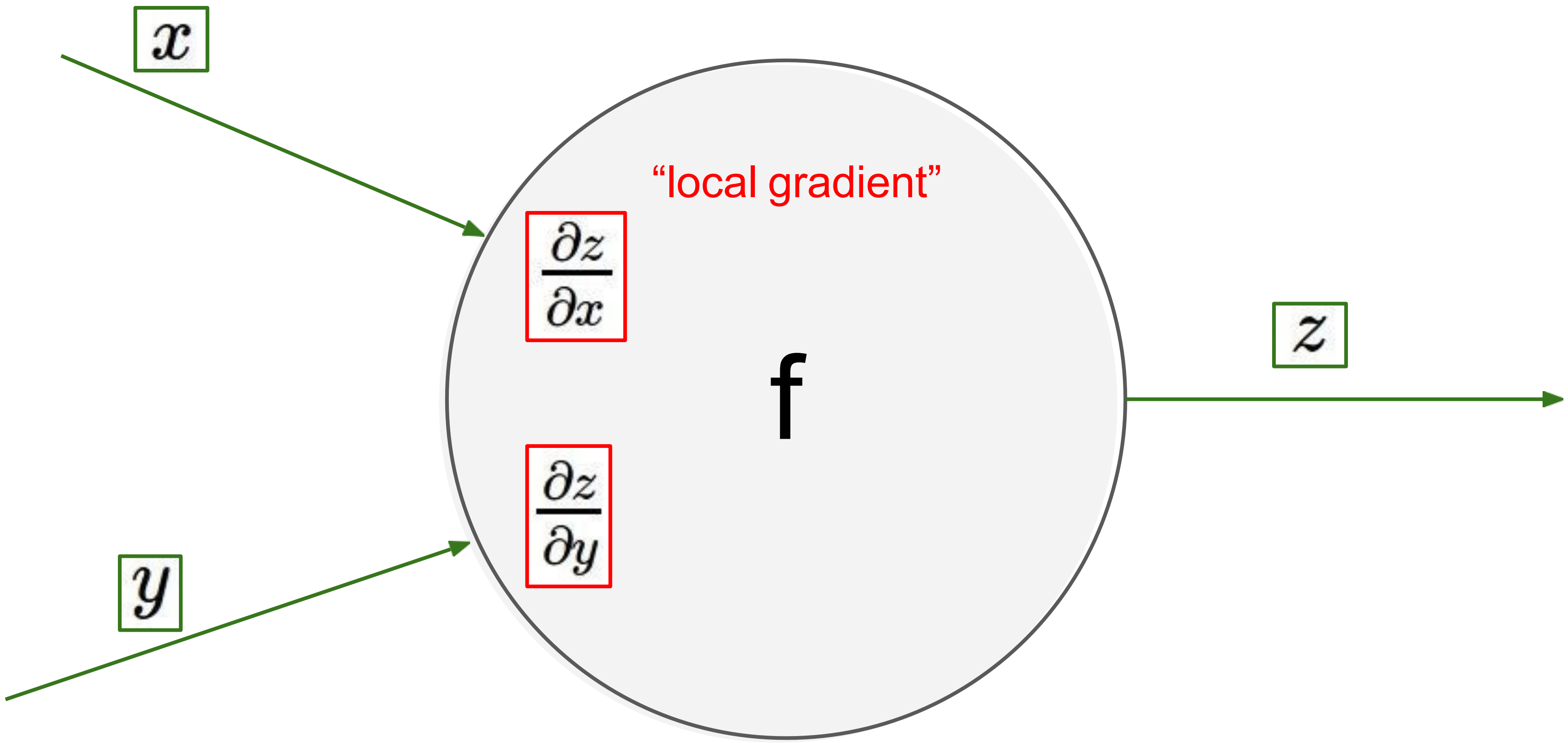
Chain rule:

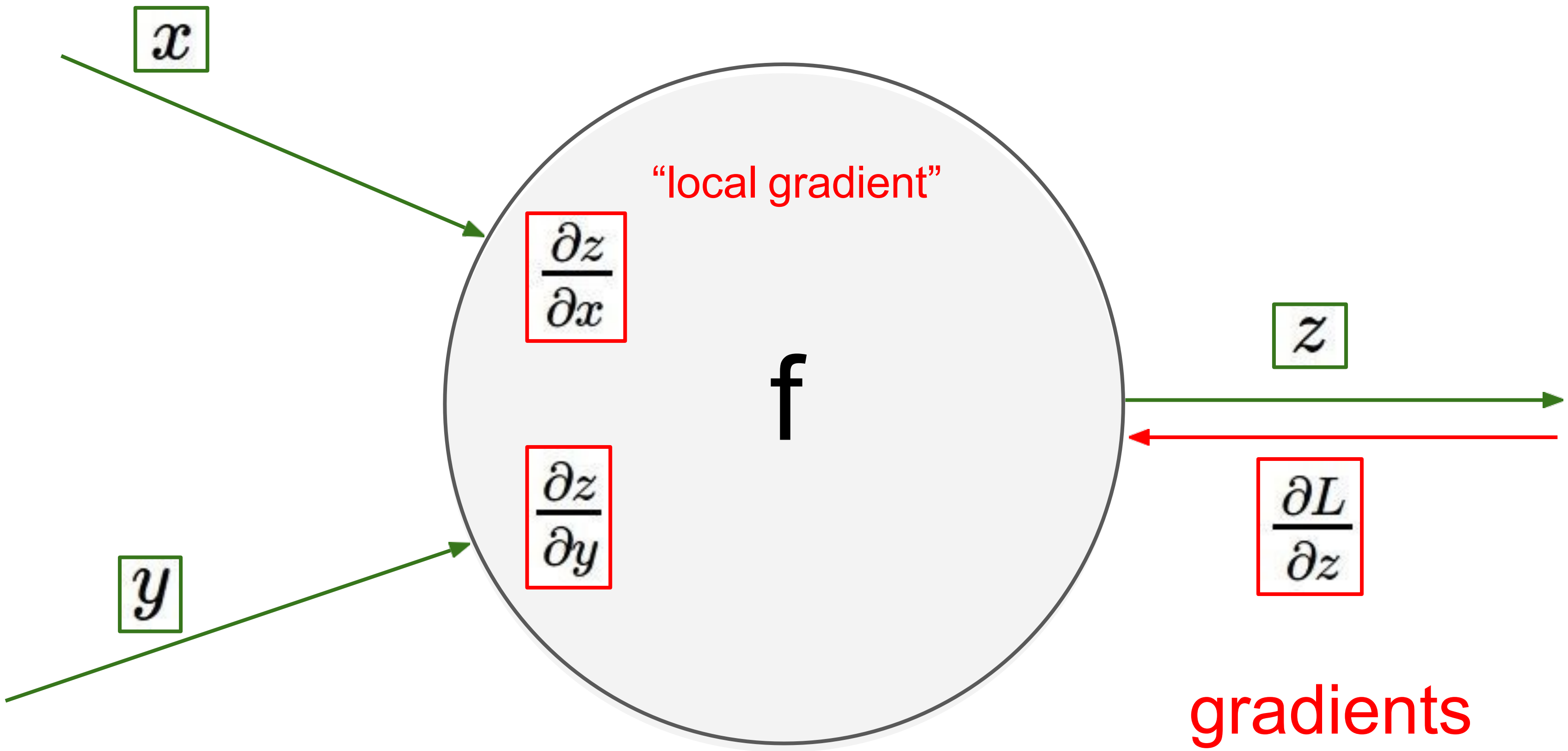
$$\frac{\partial f}{\partial x} = \frac{\partial f}{\partial q}\frac{\partial q}{\partial x}$$

Slide credit: Fei-Fei Li

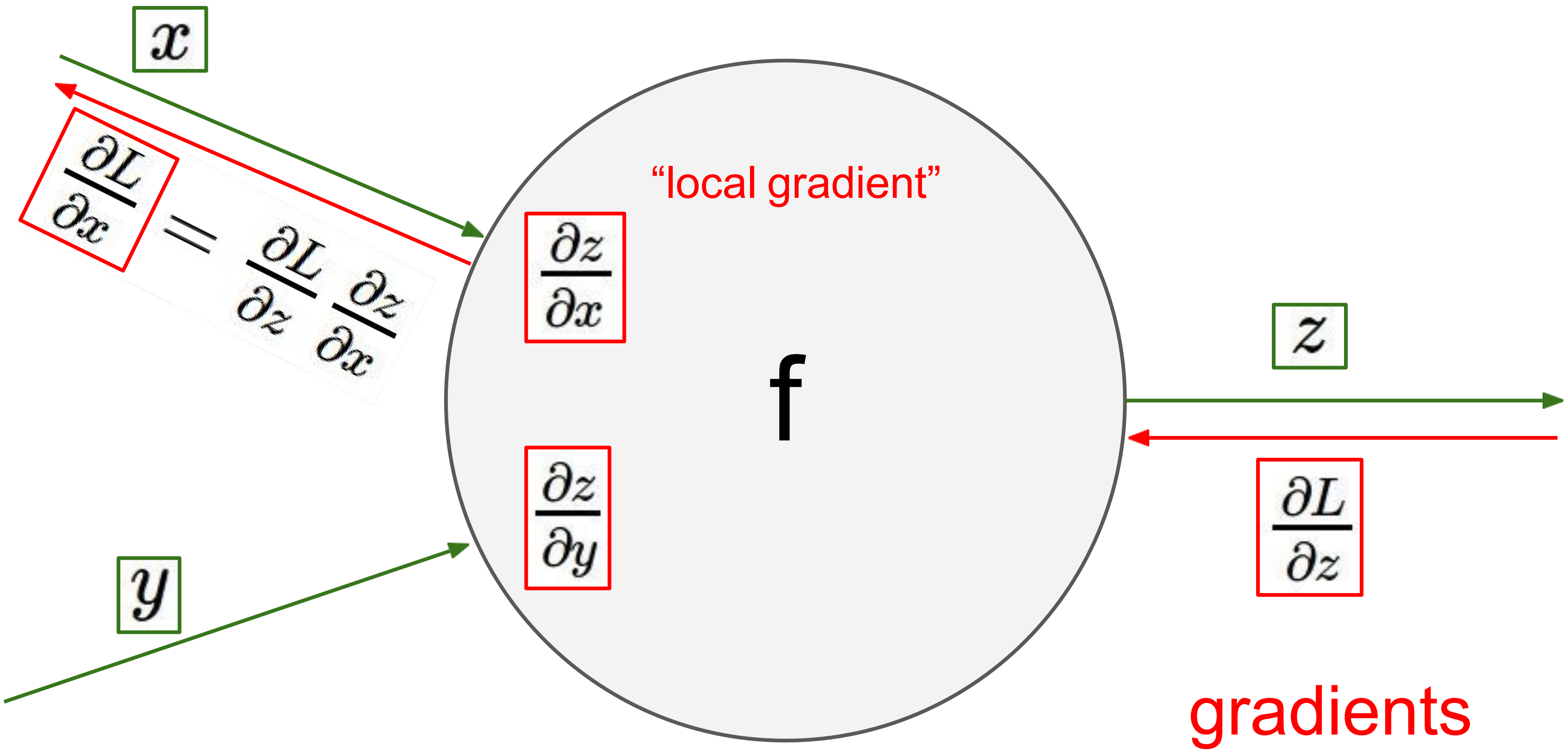


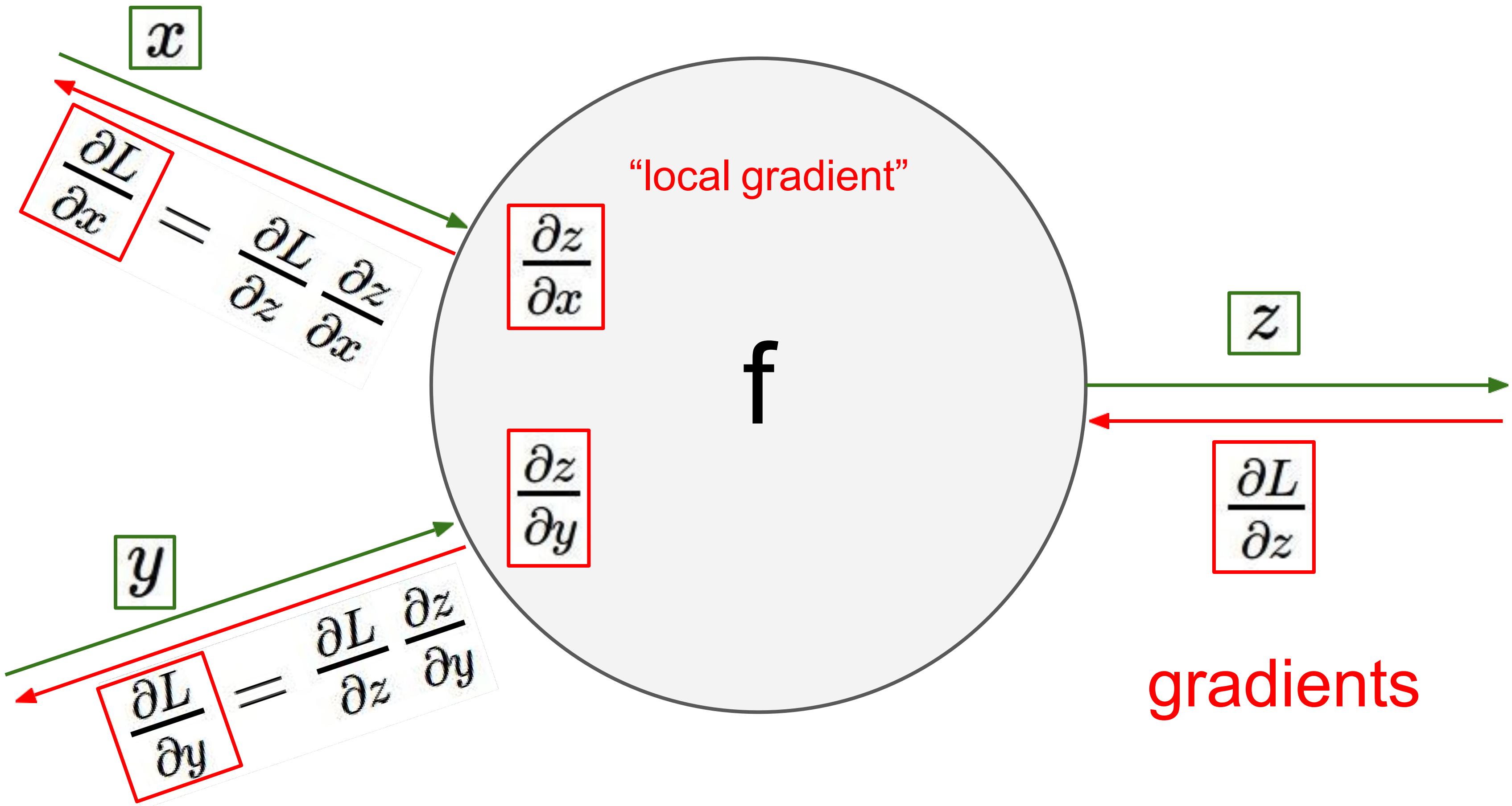
Slide credit: Fei-Fei Li

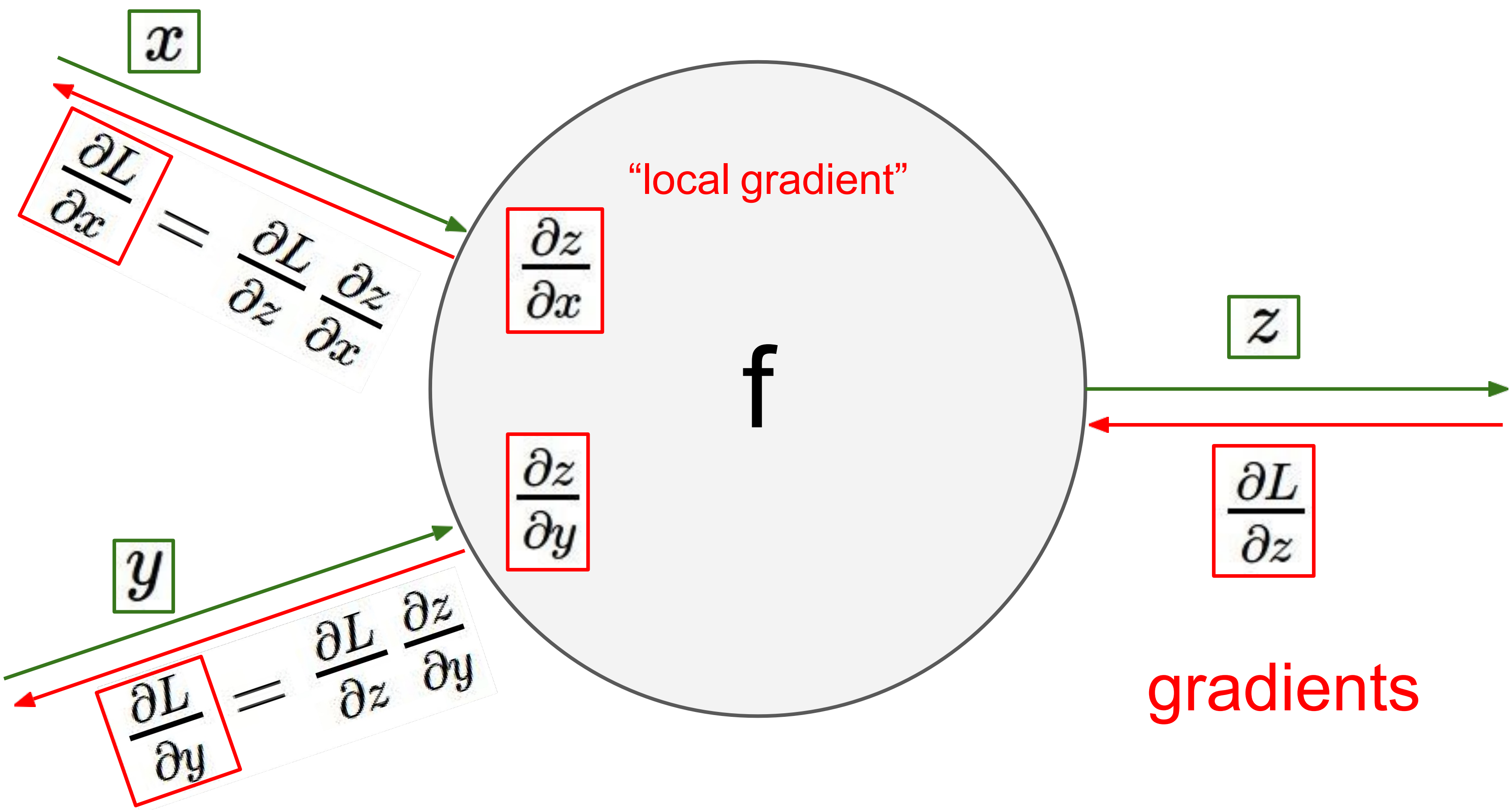

x
"local gradient"
$\frac{\partial z}{\partial x}$
f
z
$\frac{\partial z}{\partial y}$
y



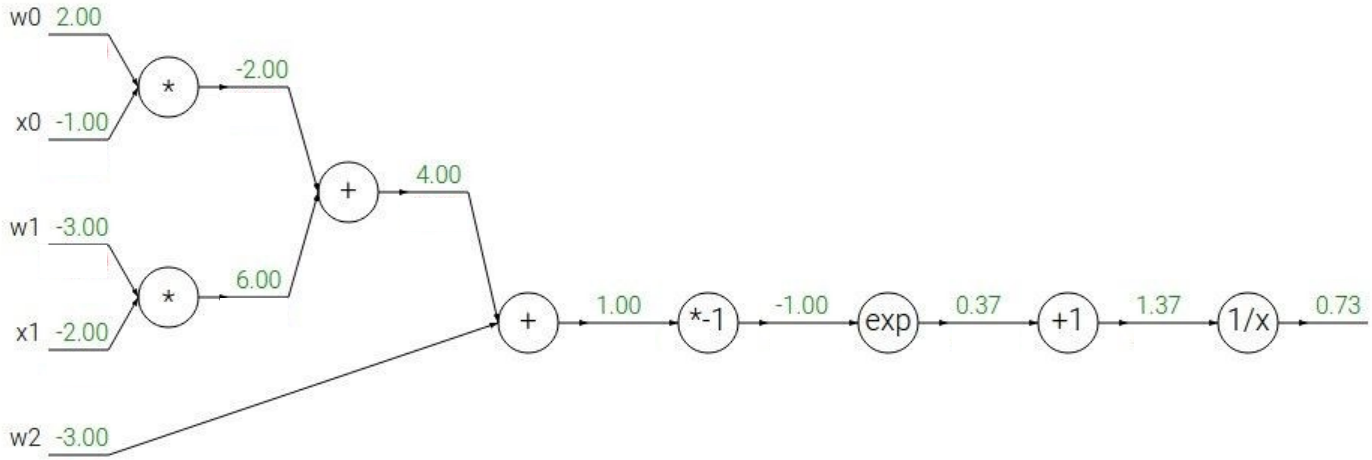
Slide credit: Fei-Fei Li


x
$\frac{\partial L}{\partial x} = \frac{\partial L}{\partial z}\frac{\partial z}{\partial x}$
"local gradient"
$\frac{\partial z}{\partial x}$
f
$\frac{\partial z}{\partial y}$
z
$\frac{\partial L}{\partial z}$
y
gradients


x
$\frac{\partial L}{\partial x} = \frac{\partial L}{\partial z}\frac{\partial z}{\partial x}$
"local gradient"
$\frac{\partial z}{\partial x}$
f
$\frac{\partial z}{\partial y}$
z
$\frac{\partial L}{\partial z}$
y
$\frac{\partial L}{\partial y} = \frac{\partial L}{\partial z}\frac{\partial z}{\partial y}$
gradients

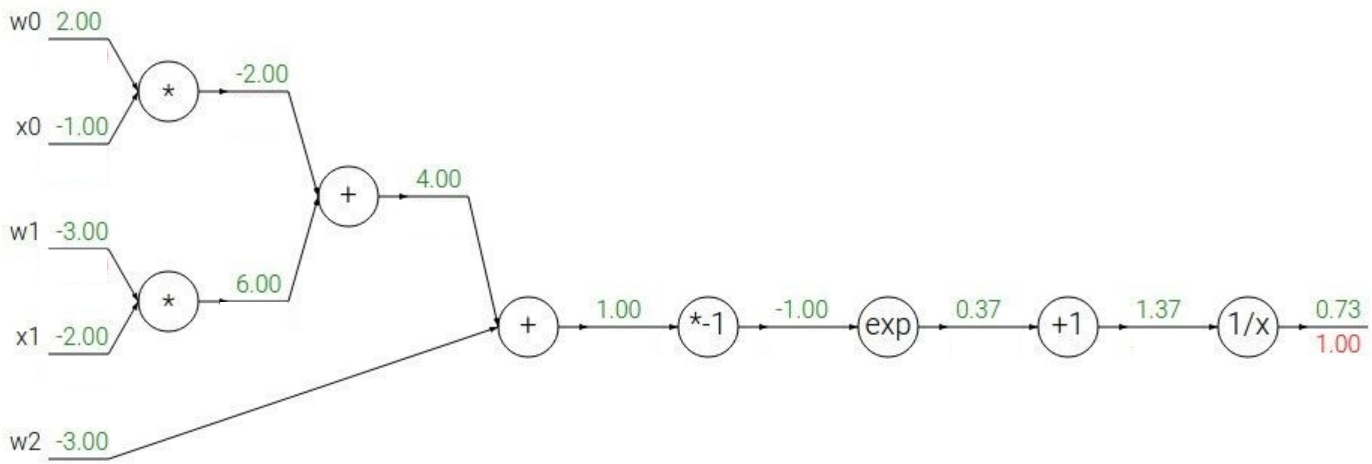

x
∂L/∂x = ∂L/∂z ∂z/∂x
"local gradient"
∂z/∂x
f
z
∂z/∂y
∂L/∂z
y
∂L/∂y = ∂L/∂z ∂z/∂y
gradients

Another example: $$f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$$



Slide credit: Fei-Fei Li

Another example: $$f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$$



$$f(x) = e^x \quad \rightarrow \quad \frac{df}{dx} = e^x$$

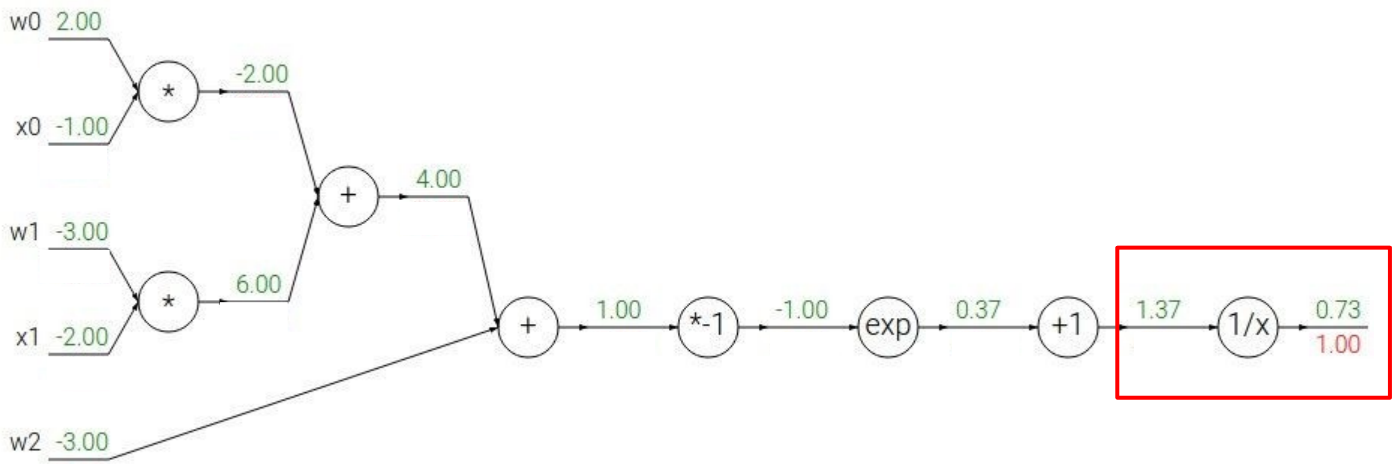
$$f_a(x) = ax \quad \rightarrow \quad \frac{df}{dx} = a$$

$$f(x) = \frac{1}{x} \quad \rightarrow \quad \frac{df}{dx} = -1/x^2$$

$$f_c(x) = c + x \quad \rightarrow \quad \frac{df}{dx} = 1$$

Slide credit: Fei-Fei Li

Another example: $$f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$$



$$f(x) = e^x \quad \rightarrow \quad \frac{df}{dx} = e^x$$

$$f_a(x) = ax \quad \rightarrow \quad \frac{df}{dx} = a$$

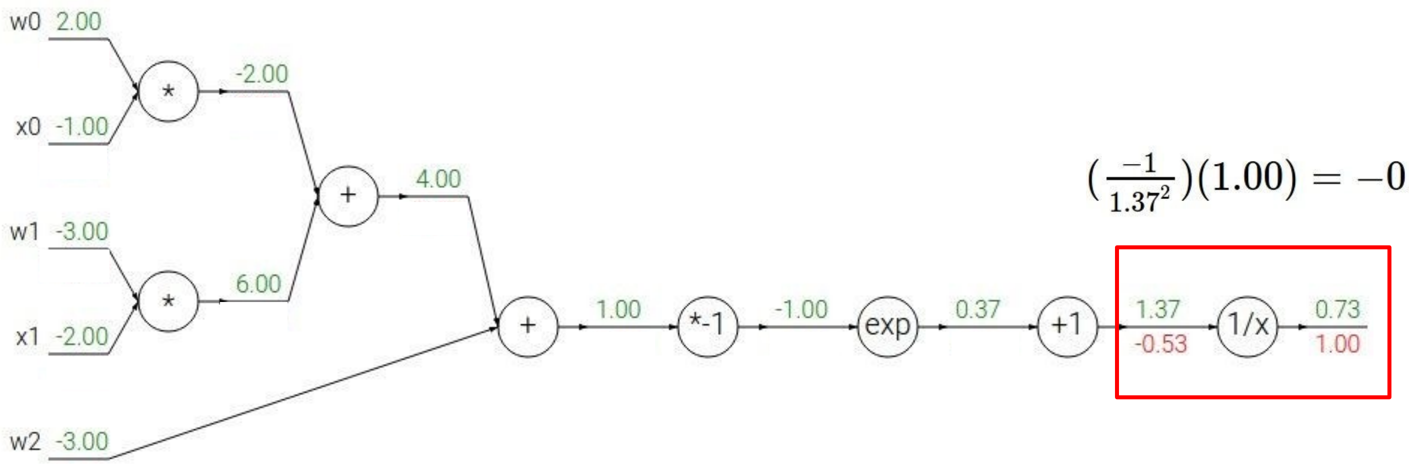
$$\boxed{f(x) = \frac{1}{x} \quad \rightarrow \quad \frac{df}{dx} = -1/x^2}$$

$$f_c(x) = c + x \quad \rightarrow \quad \frac{df}{dx} = 1$$

Slide credit: Fei-Fei Li

Another example: $f(w,x) = \dfrac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$

$\left(\frac{-1}{1.37^2}\right)(1.00) = -0.53$



$f(x) = e^x \quad \rightarrow \quad \dfrac{df}{dx} = e^x$

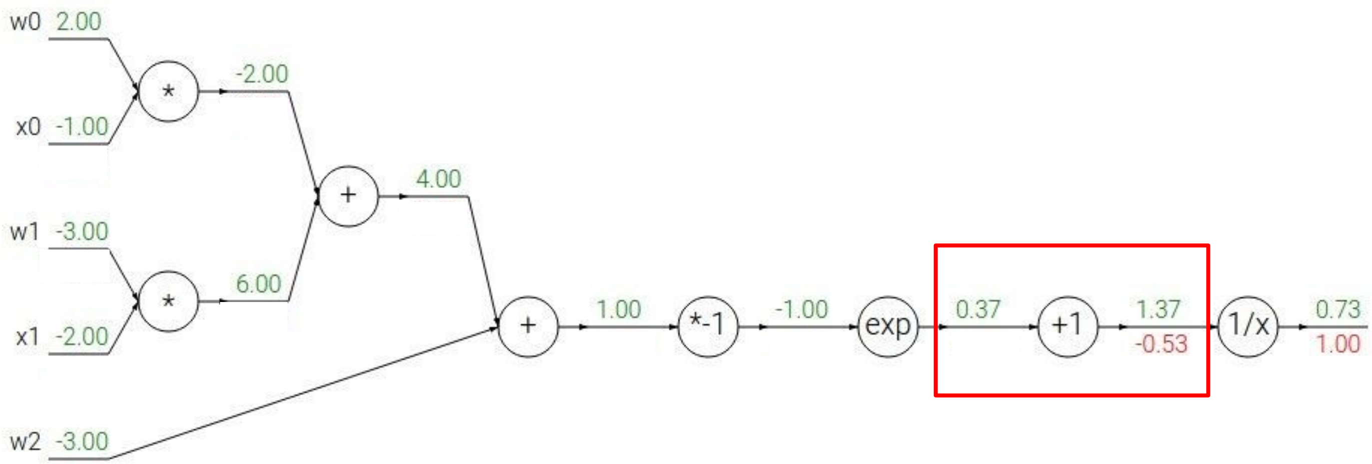
$f_a(x) = ax \quad \rightarrow \quad \dfrac{df}{dx} = a$

$\boxed{f(x) = \dfrac{1}{x} \quad \rightarrow \quad \dfrac{df}{dx} = -1/x^2}$

$f_c(x) = c + x \quad \rightarrow \quad \dfrac{df}{dx} = 1$

Slide credit: Fei-Fei Li

Another example: $$f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$$



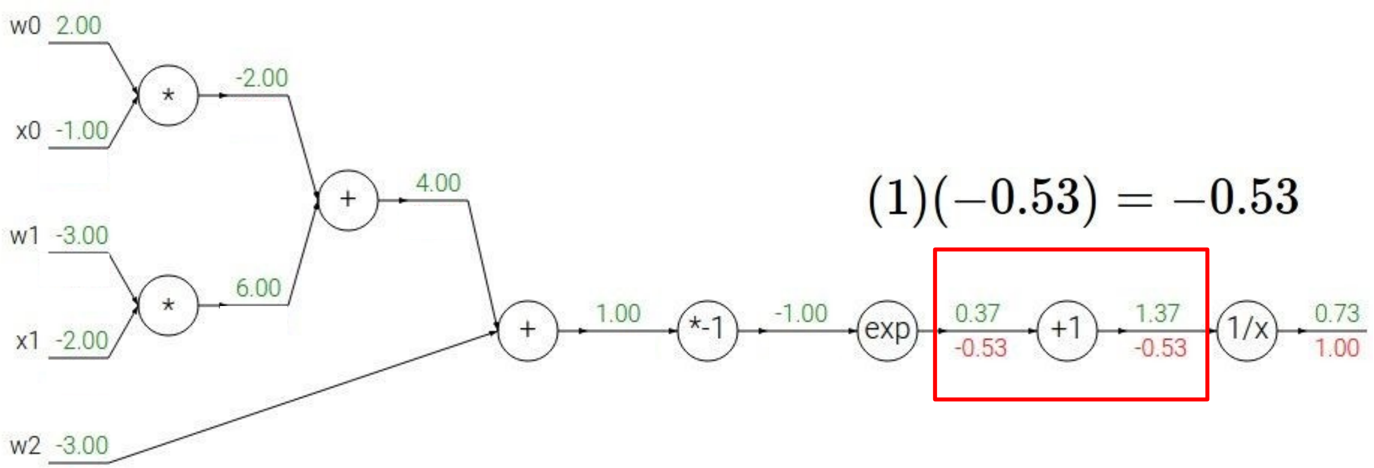
$$f(x) = e^x \quad \to \quad \frac{df}{dx} = e^x$$

$$f_a(x) = ax \quad \to \quad \frac{df}{dx} = a$$

$$f(x) = \frac{1}{x} \quad \to \quad \frac{df}{dx} = -1/x^2$$

$$\boxed{f_c(x) = c + x \quad \to \quad \frac{df}{dx} = 1}$$

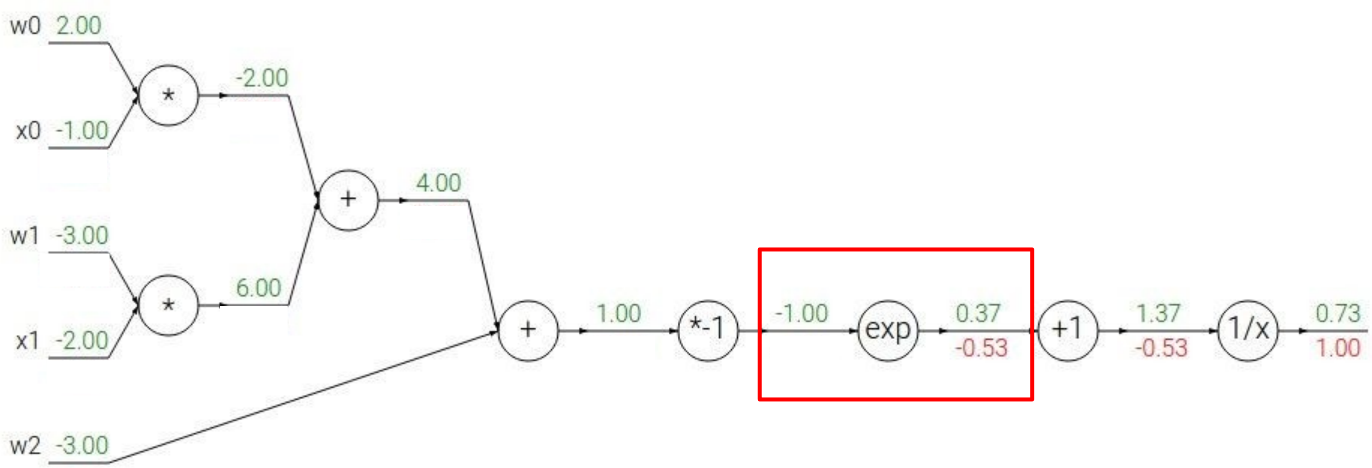
Another example: $f(w,x) = \dfrac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$



$$(1)(-0.53) = -0.53$$

$f(x) = e^x$	$\rightarrow$	$\dfrac{df}{dx} = e^x$	$f(x) = \dfrac{1}{x}$	$\rightarrow$	$\dfrac{df}{dx} = -1/x^2$
$f_a(x) = ax$	$\rightarrow$	$\dfrac{df}{dx} = a$	$\boxed{f_c(x) = c + x \quad \rightarrow \quad \dfrac{df}{dx} = 1}$		

Another example: $$f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$$



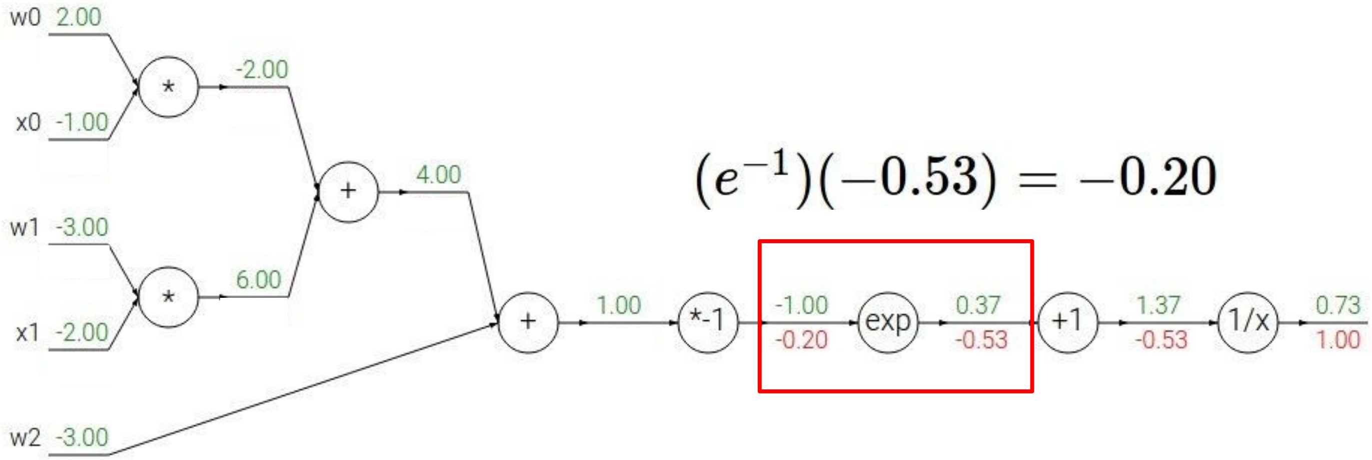
$$\boxed{f(x) = e^x \quad \rightarrow \quad \frac{df}{dx} = e^x}$$

$$f_a(x) = ax \quad \rightarrow \quad \frac{df}{dx} = a$$

$$f(x) = \frac{1}{x} \quad \rightarrow \quad \frac{df}{dx} = -1/x^2$$

$$f_c(x) = c + x \quad \rightarrow \quad \frac{df}{dx} = 1$$

Another example: $$f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$$



$$(e^{-1})(-0.53) = -0.20$$

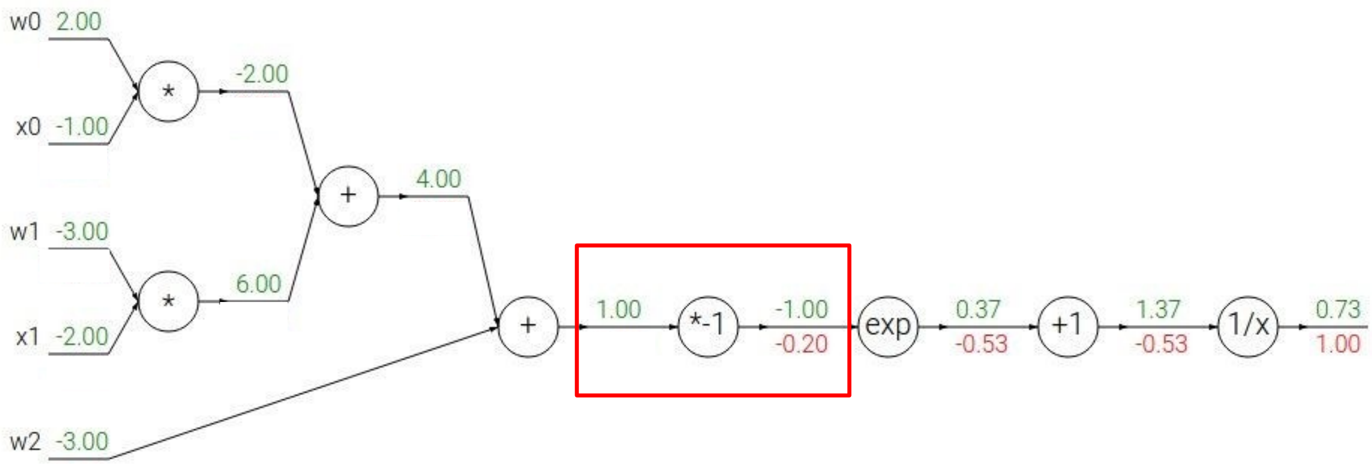
$$\boxed{f(x) = e^x \quad \rightarrow \quad \frac{df}{dx} = e^x}$$

$$f_a(x) = ax \quad \rightarrow \quad \frac{df}{dx} = a$$

$$f(x) = \frac{1}{x} \quad \rightarrow \quad \frac{df}{dx} = -1/x^2$$

$$f_c(x) = c + x \quad \rightarrow \quad \frac{df}{dx} = 1$$

Another example: $$f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$$



$$f(x) = e^x \quad \rightarrow \quad \frac{df}{dx} = e^x$$

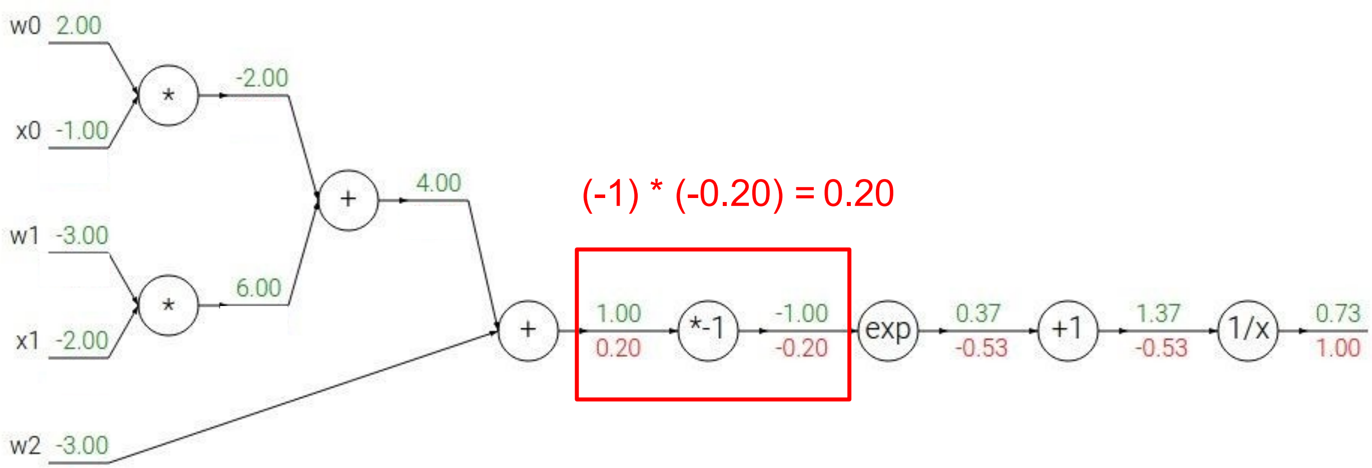
$$f_a(x) = ax \quad \rightarrow \quad \frac{df}{dx} = a$$

$$f(x) = \frac{1}{x} \quad \rightarrow \quad \frac{df}{dx} = -1/x^2$$

$$f_c(x) = c + x \quad \rightarrow \quad \frac{df}{dx} = 1$$

Slide credit: Fei-Fei Li

Another example: $$f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$$



$$f(x) = e^x \quad \rightarrow \quad \frac{df}{dx} = e^x$$

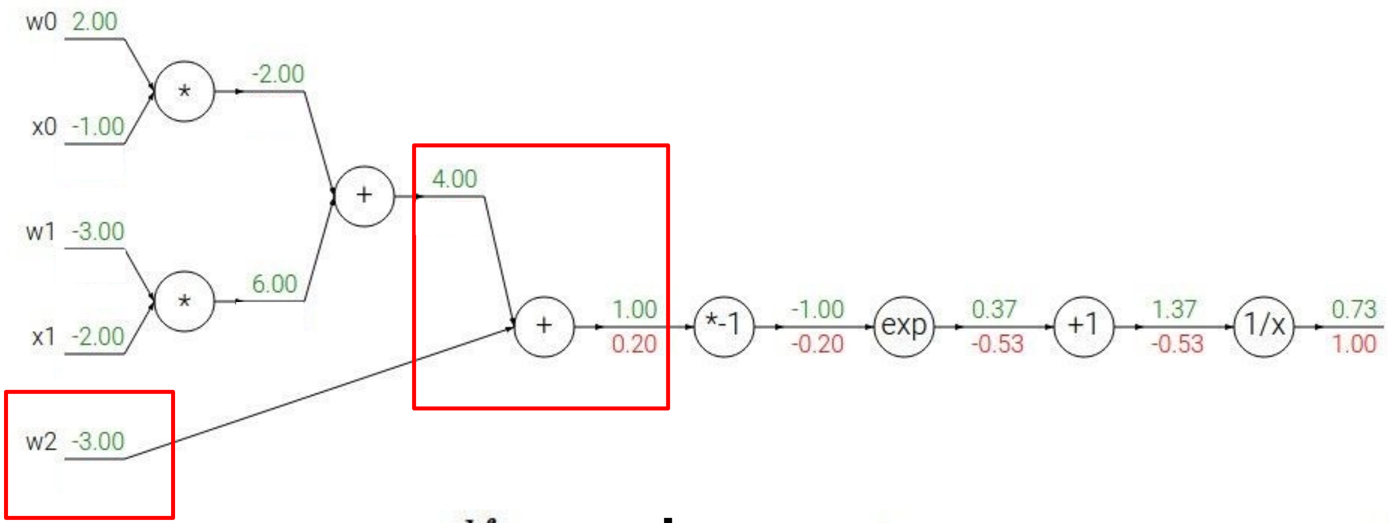
$$f(x) = \frac{1}{x} \quad \rightarrow \quad \frac{df}{dx} = -1/x^2$$

$$\boxed{f_a(x) = ax \quad \rightarrow \quad \frac{df}{dx} = a}$$

$$f_c(x) = c + x \quad \rightarrow \quad \frac{df}{dx} = 1$$

Slide credit: Fei-Fei Li

Another example: $$f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$$



$$f(x) = e^x \quad \rightarrow \quad \frac{df}{dx} = e^x$$

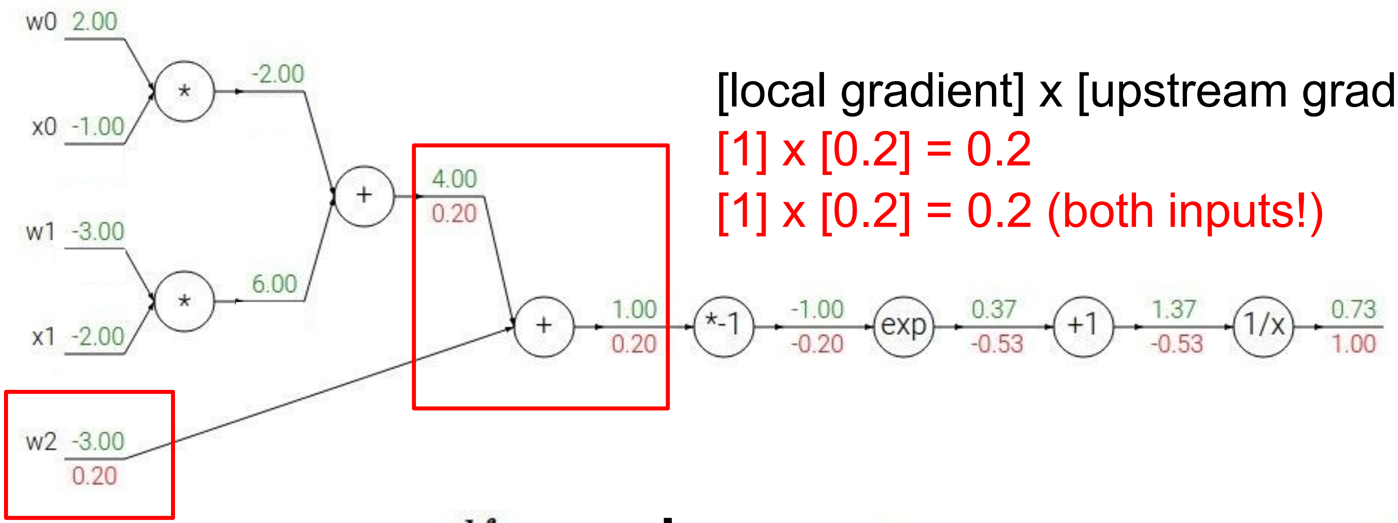
$$f_a(x) = ax \quad \rightarrow \quad \frac{df}{dx} = a$$

$$f(x) = \frac{1}{x} \quad \rightarrow \quad \frac{df}{dx} = -1/x^2$$

$$f_c(x) = c + x \quad \rightarrow \quad \frac{df}{dx} = 1$$

Slide credit: Fei-Fei Li

Another example: $f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$



[local gradient] x [upstream gradient]

[1] x [0.2] = 0.2

[1] x [0.2] = 0.2 (both inputs!)

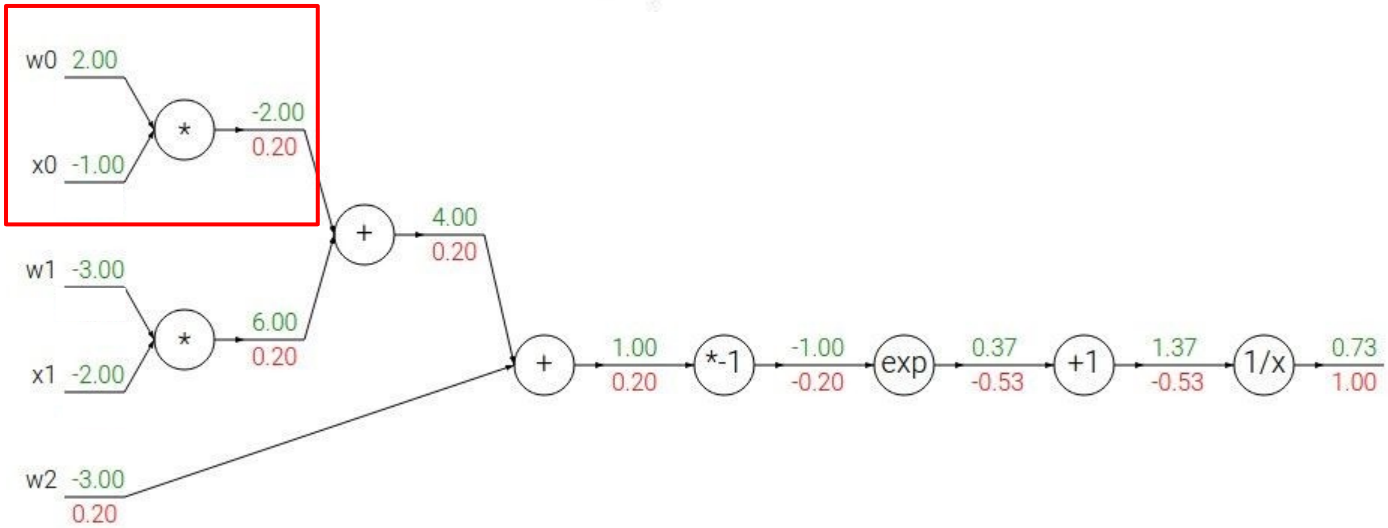
$f(x) = e^x \quad \rightarrow \quad \frac{df}{dx} = e^x$

$f_a(x) = ax \quad \rightarrow \quad \frac{df}{dx} = a$

$f(x) = \frac{1}{x} \quad \rightarrow \quad \frac{df}{dx} = -1/x^2$

$f_c(x) = c + x \quad \rightarrow \quad \frac{df}{dx} = 1$

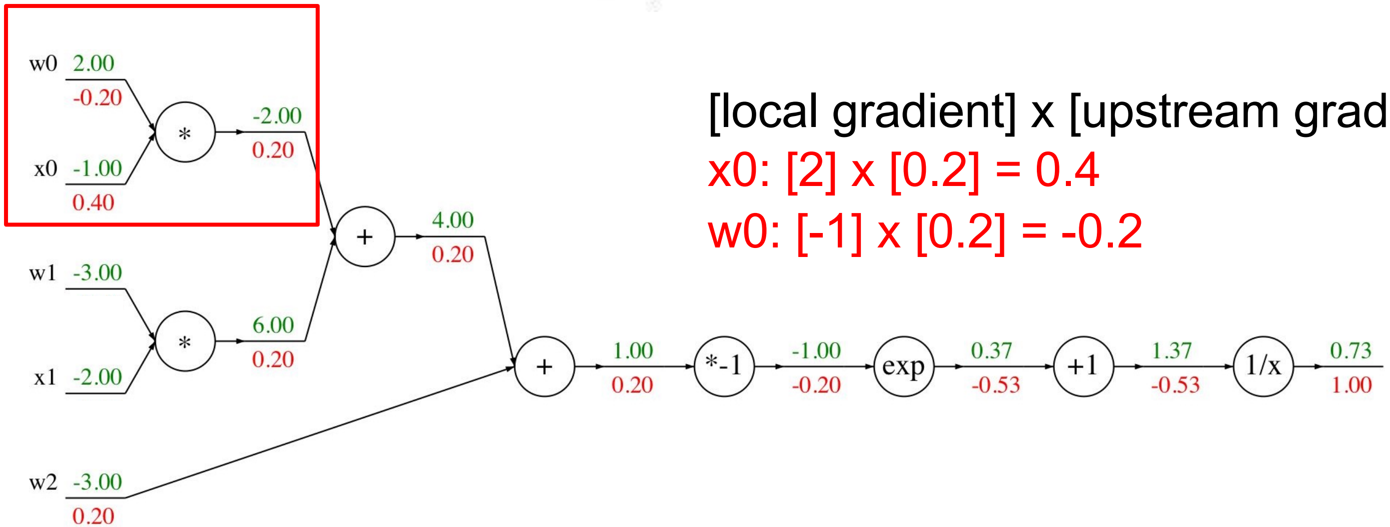
Another example: $$f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$$



$$f(x) = e^x \quad \rightarrow \quad \frac{df}{dx} = e^x \qquad\qquad f(x) = \frac{1}{x} \quad \rightarrow \quad \frac{df}{dx} = -1/x^2$$

$$f_a(x) = ax \quad \rightarrow \quad \frac{df}{dx} = a \qquad\qquad f_c(x) = c + x \quad \rightarrow \quad \frac{df}{dx} = 1$$

Another example: $f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$



[local gradient] x [upstream gradient]
x0: [2] x [0.2] = 0.4
w0: [-1] x [0.2] = -0.2

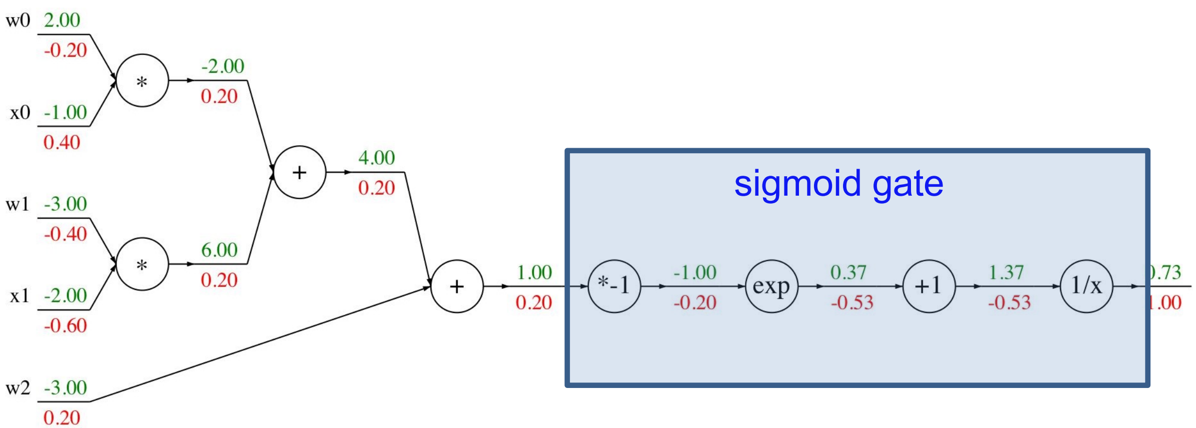
$$f(x) = e^x \quad \rightarrow \quad \frac{df}{dx} = e^x \qquad\qquad f(x) = \frac{1}{x} \quad \rightarrow \quad \frac{df}{dx} = -1/x^2$$

$$f_a(x) = ax \quad \rightarrow \quad \frac{df}{dx} = a \qquad\qquad f_c(x) = c + x \quad \rightarrow \quad \frac{df}{dx} = 1$$

$$f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$$

$$\boxed{\sigma(x) = \frac{1}{1+e^{-x}}}$$ sigmoid function

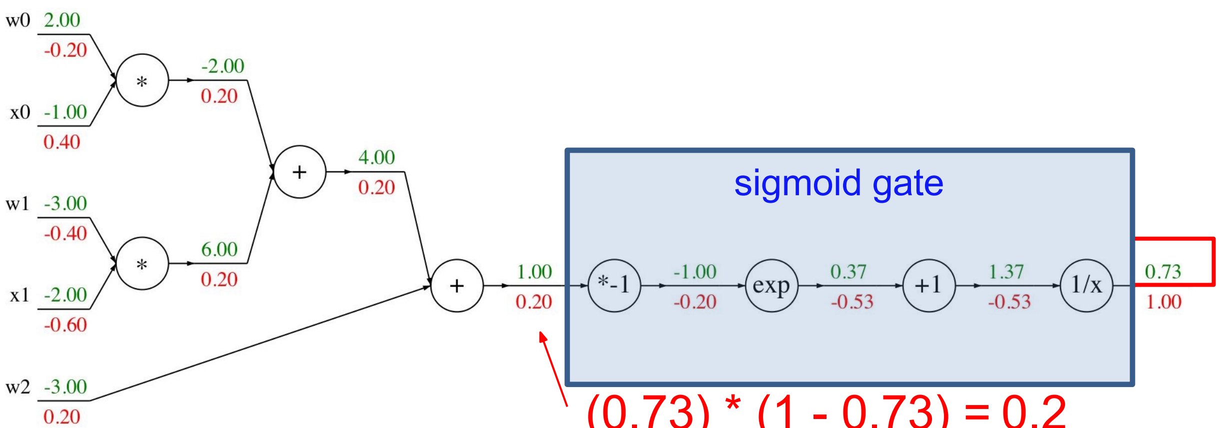
$$\frac{d\sigma(x)}{dx} = \frac{e^{-x}}{(1+e^{-x})^2} = \left(\frac{1+e^{-x}-1}{1+e^{-x}}\right)\left(\frac{1}{1+e^{-x}}\right) = (1-\sigma(x))\,\sigma(x)$$



$$f(w,x) = \frac{1}{1+e^{-(w_0x_0+w_1x_1+w_2)}}$$

$$\boxed{\sigma(x) = \frac{1}{1+e^{-x}}}$$ sigmoid function

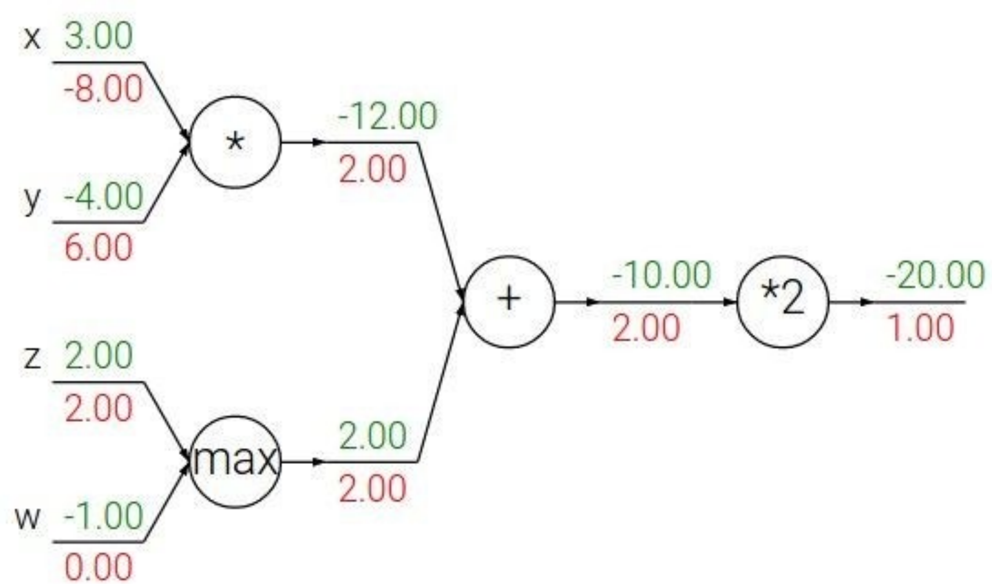
$$\frac{d\sigma(x)}{dx} = \frac{e^{-x}}{(1+e^{-x})^2} = \left(\frac{1+e^{-x}-1}{1+e^{-x}}\right)\left(\frac{1}{1+e^{-x}}\right) = (1-\sigma(x))\,\sigma(x)$$



(0.73) * (1 - 0.73) = 0.2

Patterns in backward flow

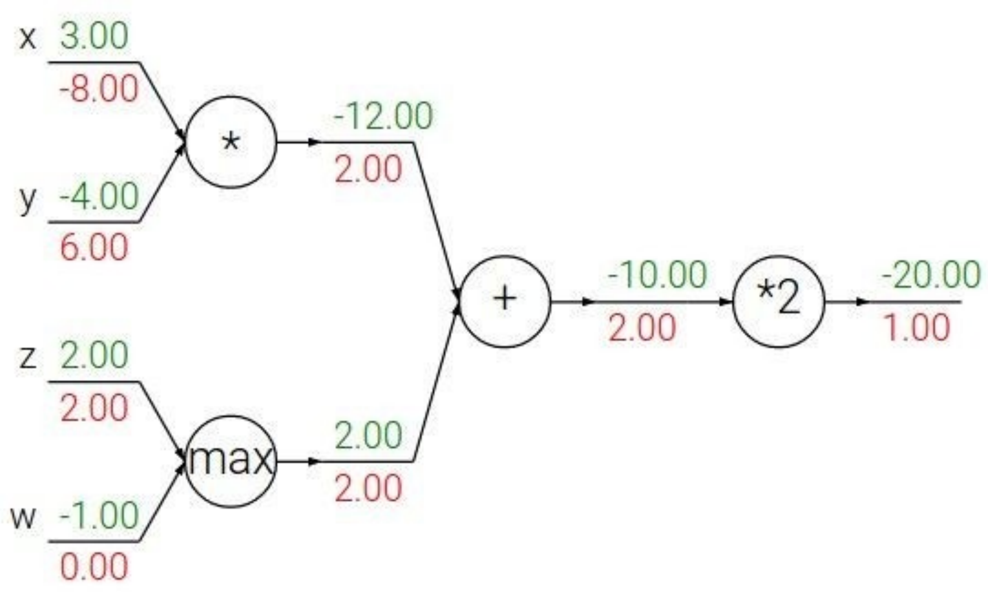
add gate: gradient distributor



Patterns in backward flow

add gate: gradient distributor

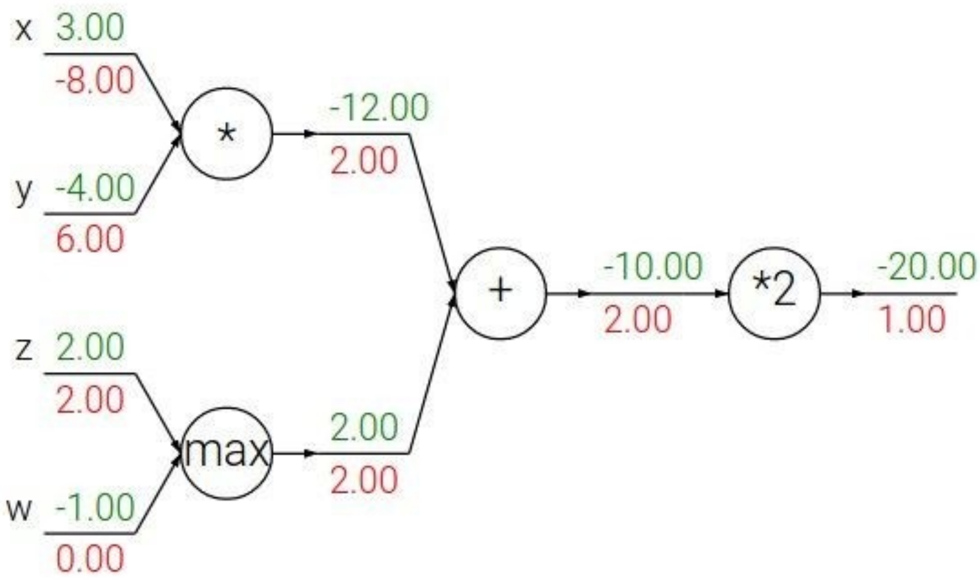
Q: What is a **max** gate?



Patterns in backward flow

add gate: gradient distributor

max gate: gradient router

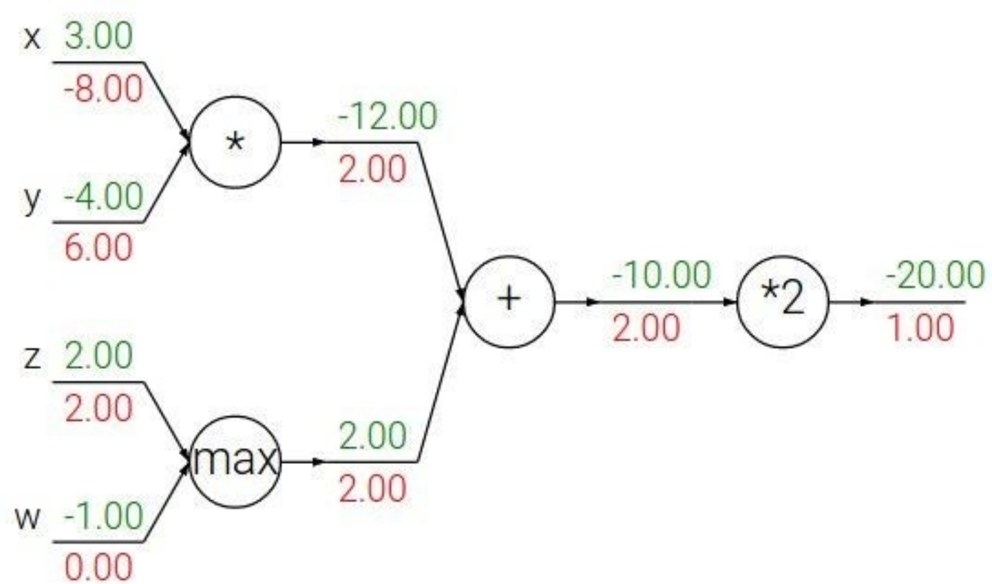


Patterns in backward flow

add gate: gradient distributor

max gate: gradient router

Q: What is a **mul** gate?

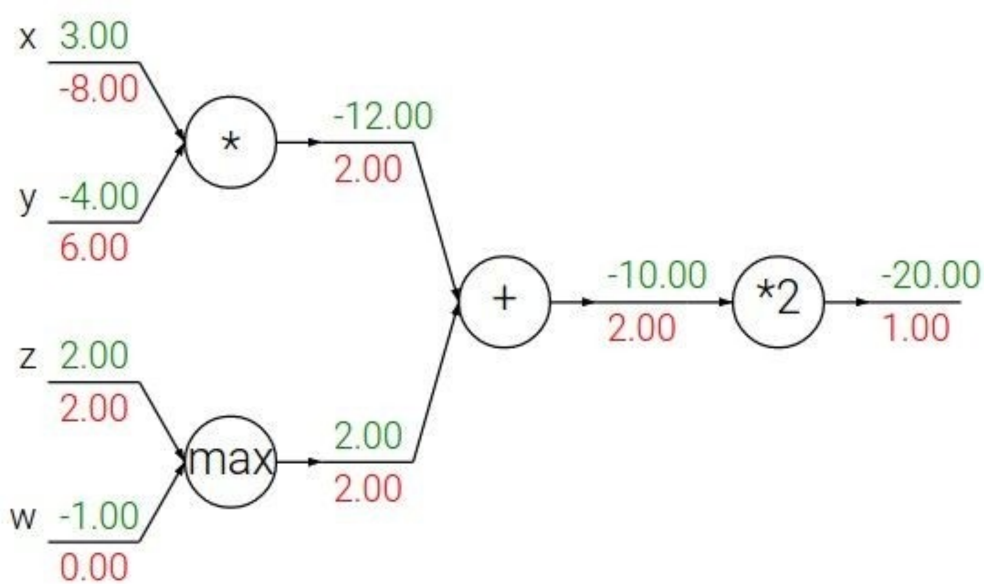


Patterns in backward flow

add gate: gradient distributor

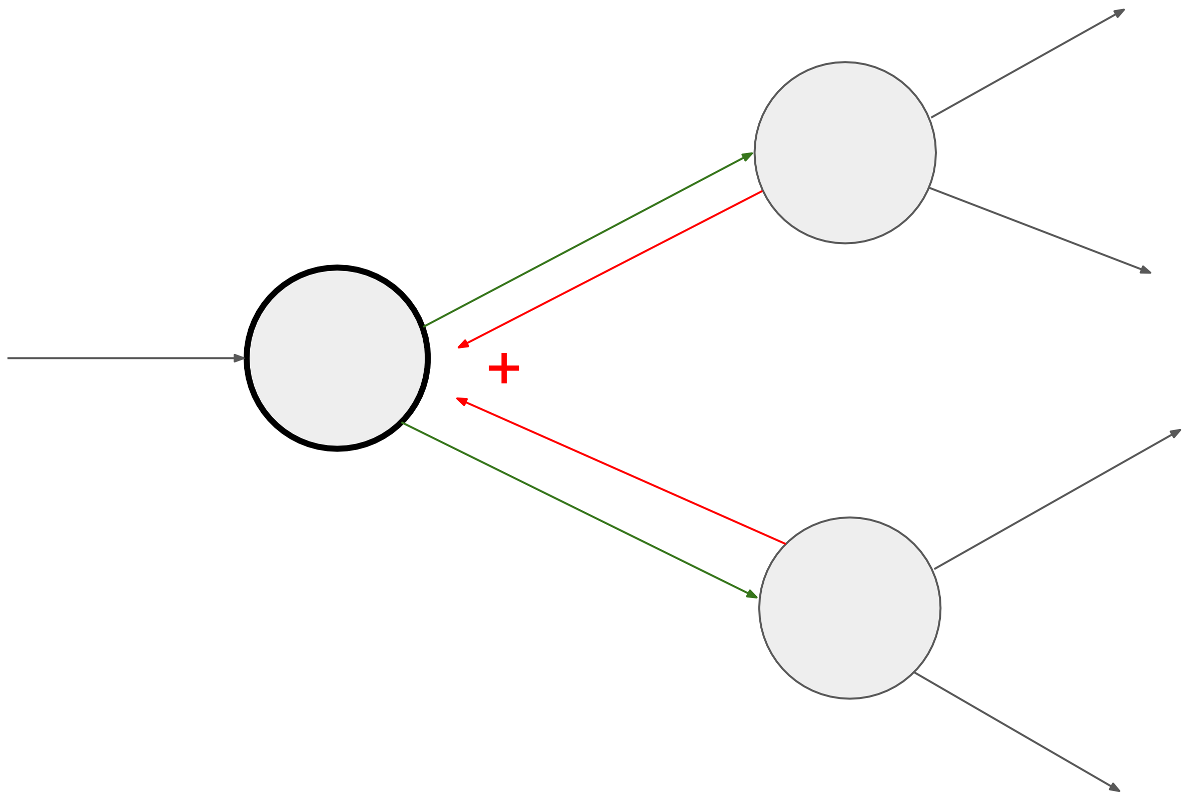
max gate: gradient router

mul gate: gradient switcher



Slide credit: Fei-Fei Li

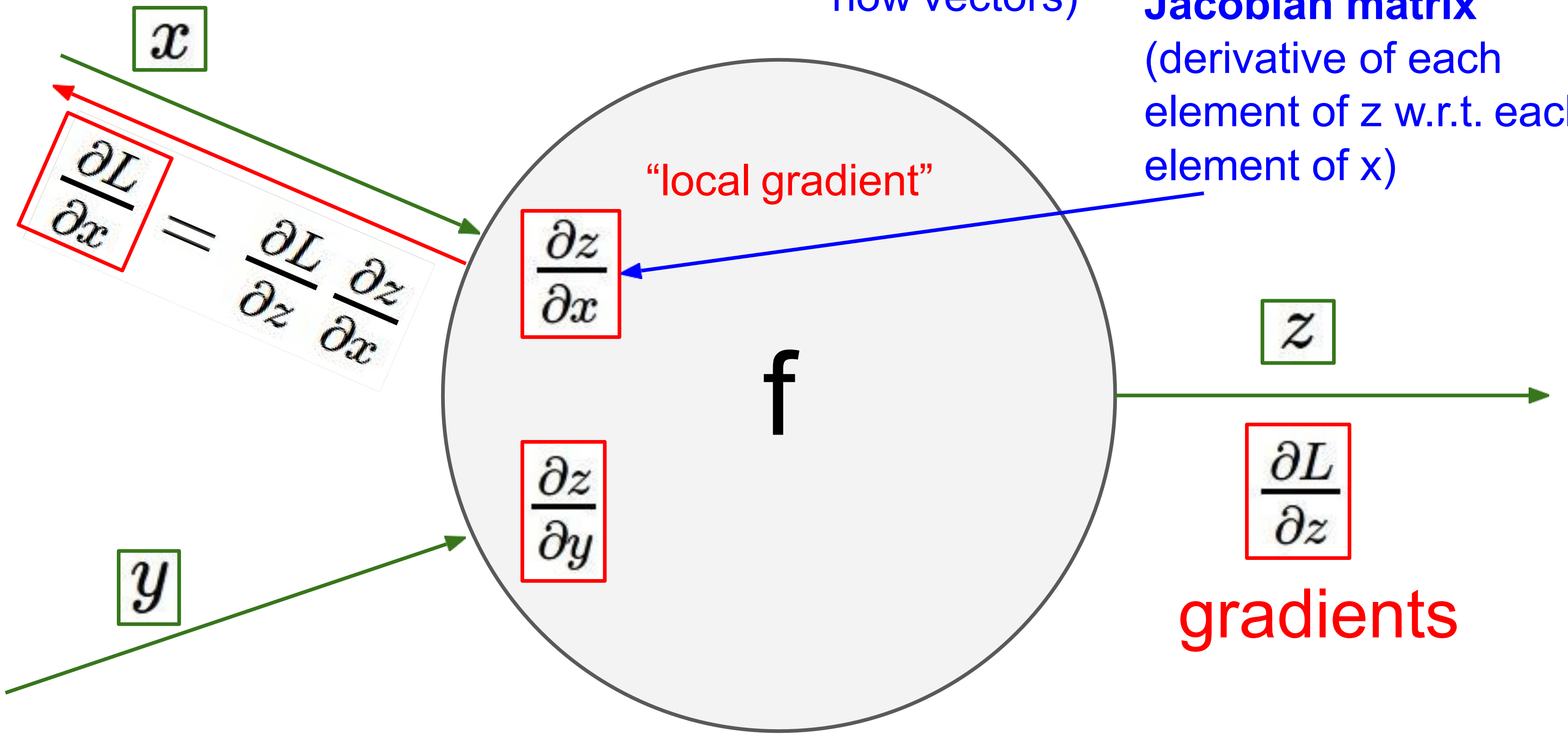
Gradients add at branches

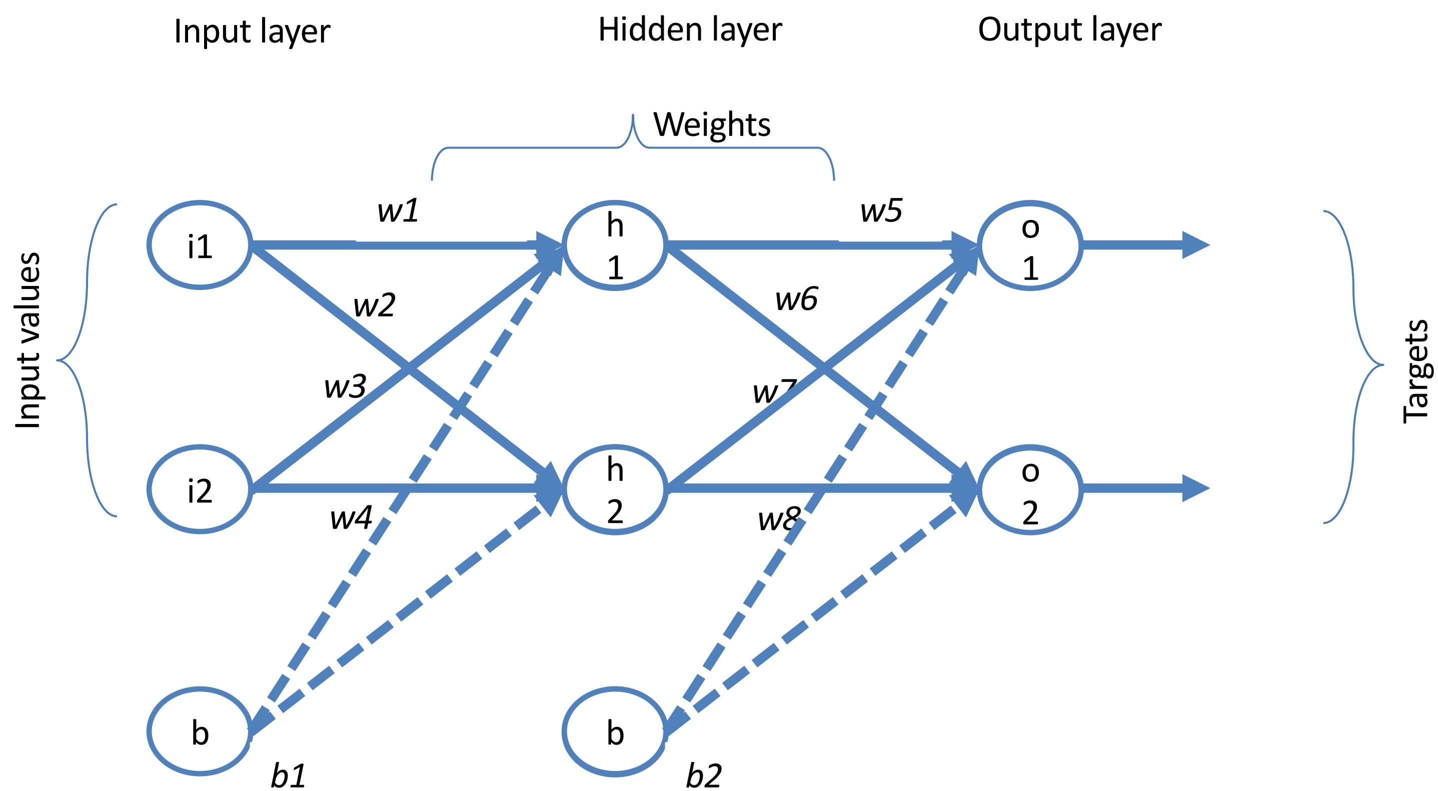


Gradients for vectorized code

(x,y,z are now vectors)

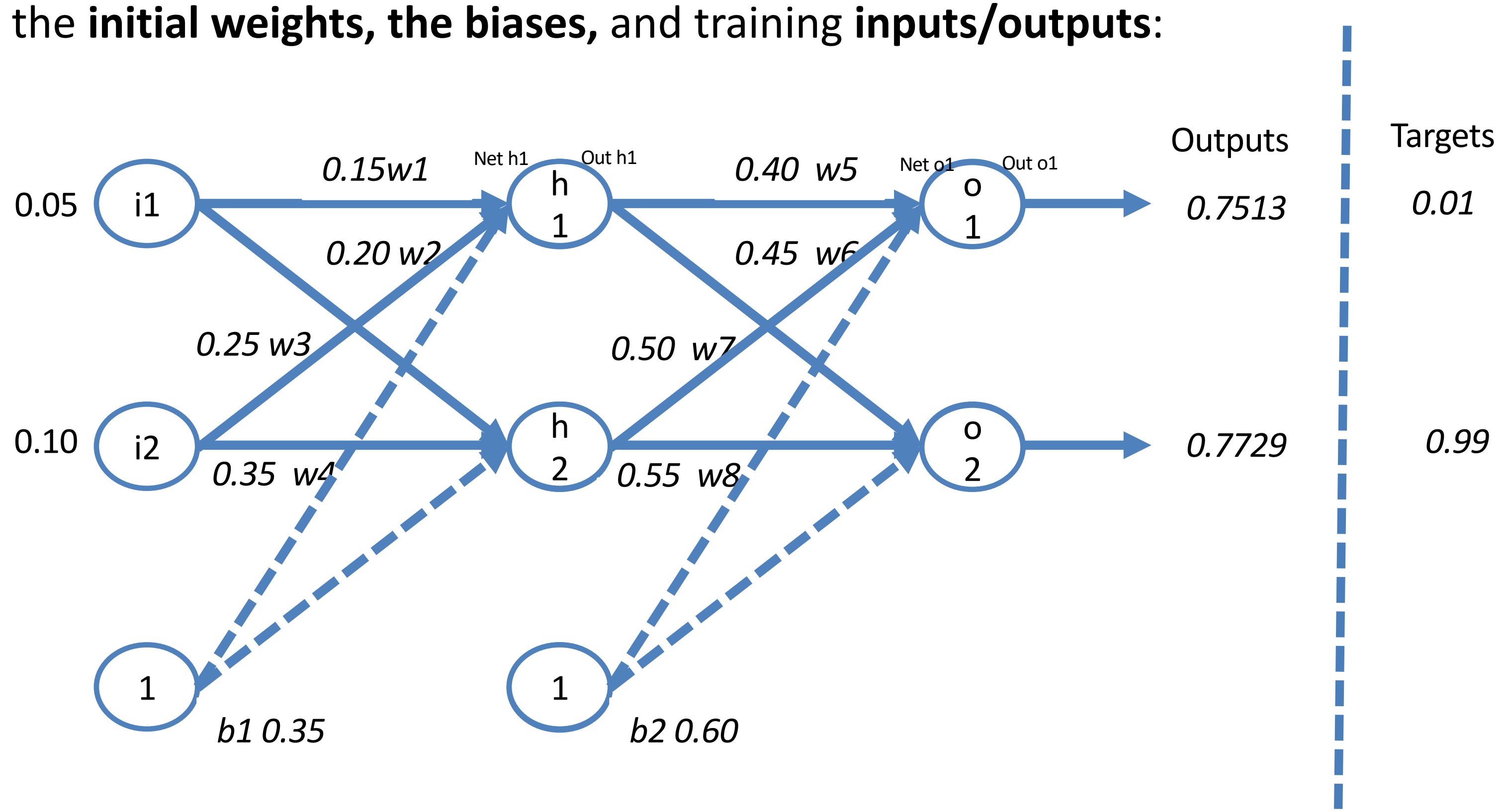
This is now the **Jacobian matrix** (derivative of each element of z w.r.t. each element of x)





Basic Structure of NN

Here are the **initial weights, the biases,** and training **inputs/outputs**:



Example of NN

Forward Pass

Lets see what the neural network currently predicts given the weights and biases above and inputs of 0.05 and 0.10.

=> Output for **hidden layer** with **sigmoid activation function:**

$$net_{h1} = w_1 * i_1 + w2 * i_2 + b_1 * 1$$

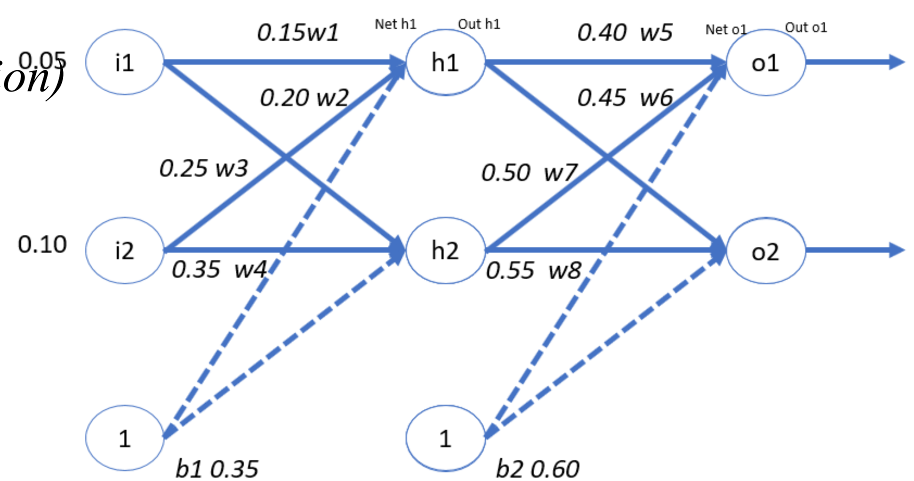
$$net_{h1} = 0.05 * 0.15 + 0.2 * 0.1 + 0.35 * 1 \rightarrow \mathbf{0.3775}$$

$$out\ h1 = \frac{1}{1+e^{-net\ h1}}$$ (**sigmoid** *activation function*)

$$out\ h1 = \frac{1}{1+e^{-0.3775}} \rightarrow \mathbf{0.5932699}$$

similarly,

$$out\ h2 = \mathbf{0.5968843}$$



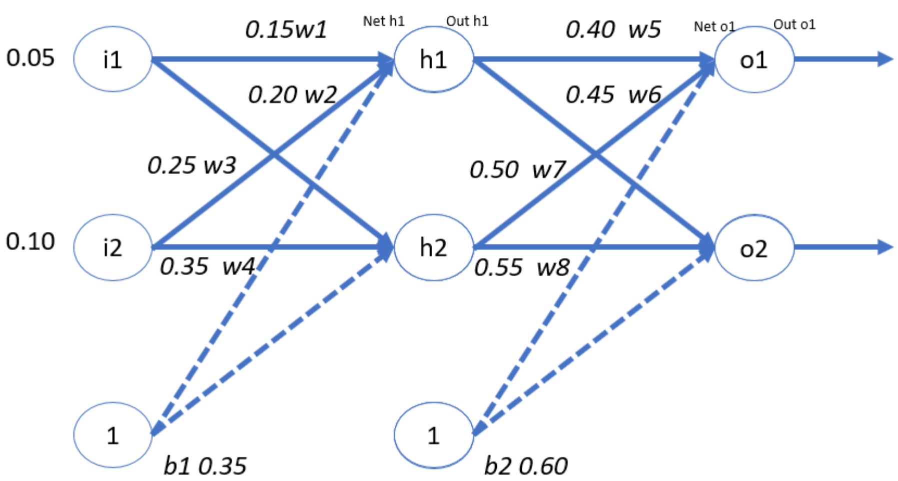
Repeat above process for the output layer neurons, using the output from the hidden layer neurons as inputs.

net $_{o1}$ ***= w*** $_5$ **** out*** $_{h1}$ ***+ w*** $_5$ **** out*** $_{h1}$ ***+ b*** $_2$ **** 1***

net $_{o1}$ *= 0.4 x 0.5932699 + 0.45 x 0.5968843 + 0.6 x 1* ➔ *1.105905967*

out $_{o1} = \frac{1}{1+e^{-1.105905}}$ ➔ *0.75136507* *(Out1 but target is 0.01)*

out $_{o2}$ *=0.772928* *(Out2 but target is 0.99)*



Total Error

We can now calculate the error for each output neuron using the **squared error function** and sum them to get the total error:

$$\boldsymbol{E_{total} = \sum \frac{1}{2}(target - output)^2}$$

$$E_{total} = Eo1 + Eo2$$

$$E_{o1} = \frac{1}{2}(0.01 - 0.75136507)^2 \rightarrow \boldsymbol{0.274811083}$$

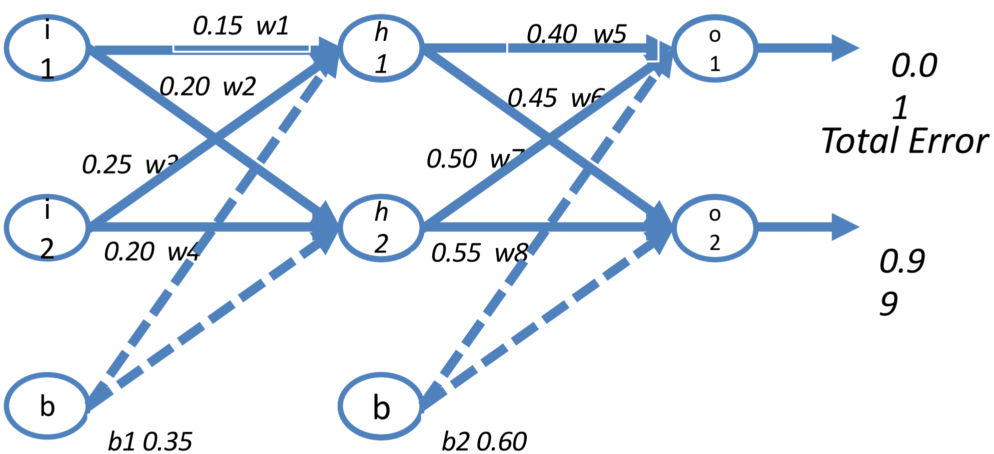
$$E_{o2} = \boldsymbol{0.023560026}$$

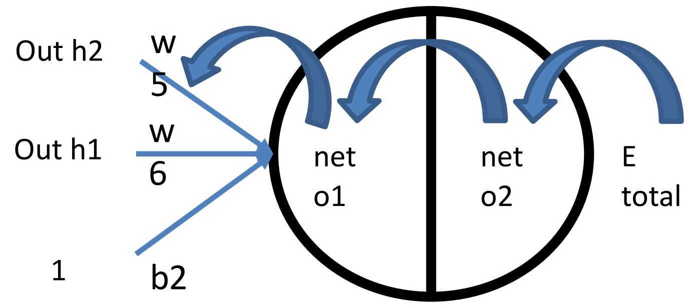
$$E\ total = Eo1 + Eo2 = \boldsymbol{0.298371109}$$

Backward Propagation

For output layer :

$$\frac{\partial Etotal}{\partial w5} = \frac{\partial Etotal}{\partial outo1} * \frac{\partial outo1}{\partial neto1} * \frac{\partial neto1}{\partial w5}$$





E total = $\frac{1}{2}(\textcolor{red}{target\ o1} - Out\ o1)^2 + \frac{1}{2}(\textcolor{red}{target\ o2} - Out\ o2)^2$

Derivative w.r.t Out o1

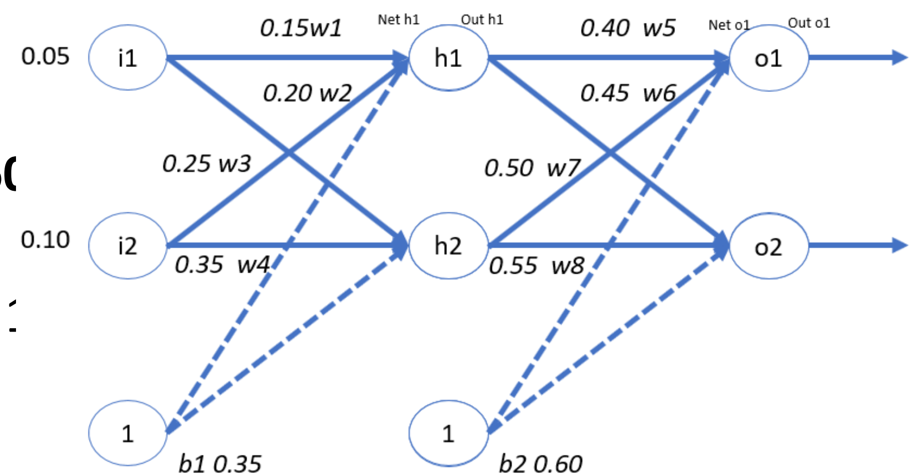
$$\frac{\partial Etotal}{\partial outo1} = -\ (\textcolor{red}{target\ o1} - Out\ o1) + 0 = \mathbf{0.74136507}$$

$$\text{Out o1} = \frac{1}{1 + e^{-net\ o1}}$$

$$\frac{\partial outo1}{\partial neto1} = \text{Out o1 (1 – Out o1)} = \mathbf{0.1868156(}$$

net o1 = w5 x out h1+ w6 x out h2 + b2 x 1

$$\frac{\partial neto1}{\partial w5} = \text{Out h1} = \mathbf{0.5932699}$$

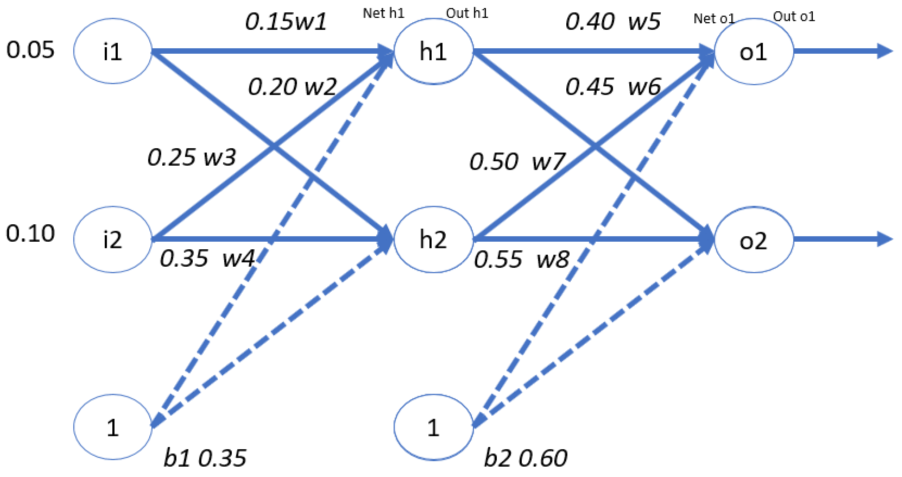


Constant are in RED color

Backward Propagation

$$\frac{\partial Etotal}{\partial w5} = \frac{\partial Etotal}{\partial outo1} * \frac{\partial outo1}{\partial neto1} * \frac{\partial neto1}{\partial w5}$$

$$\frac{\partial Etotal}{\partial w5} = \mathbf{0.082167041}$$



Updation of ***weight w5 :***

w5_new = w5 – η x $\frac{\partial Etotal}{\partial w5}$

W5_new = 0.40 – 0.5 x 0.082167 = .358916

η is learning rate here 0.5

w5 is now updated to *w5_new*
In next Forward pass w5_new is used

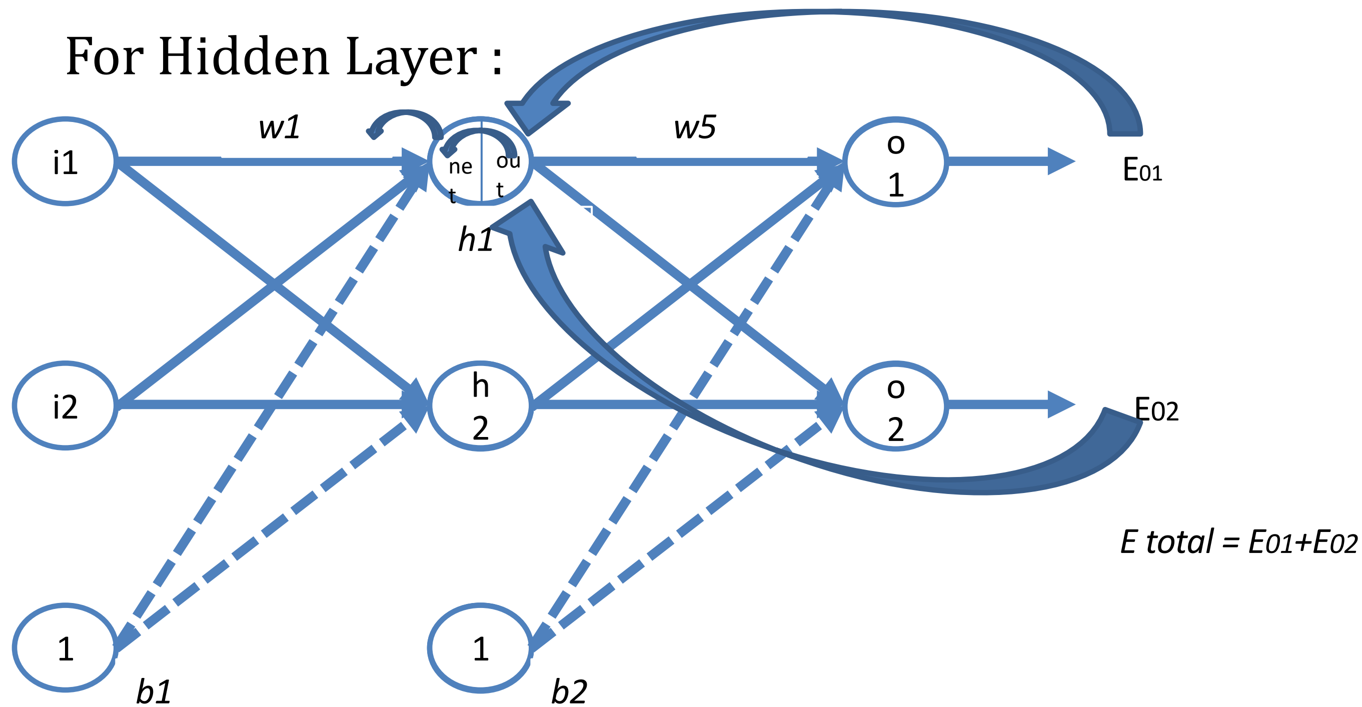
Find out updated values of weights w6, w7, w8 and bias b2 with the same procedure.

w6_new = 0.408666186
w7_new = 0.511301270
w8_new = 0.561370121

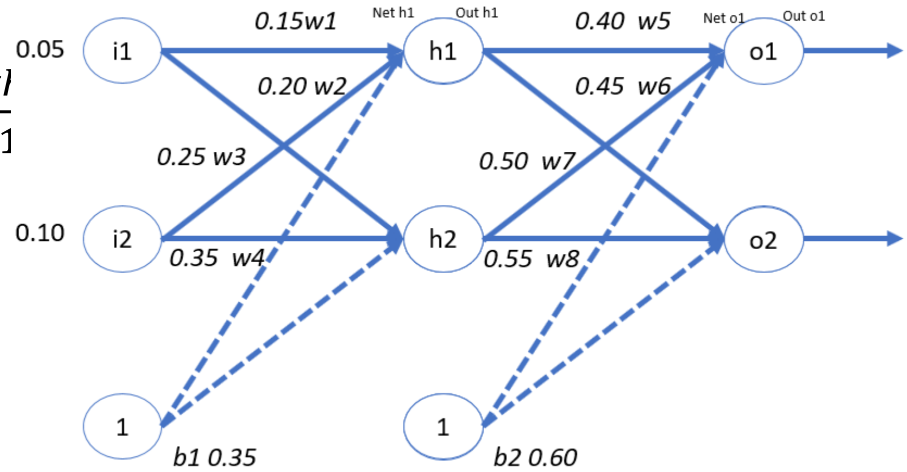
***Remember new values only considered in next Forward pass after complete updation of weights.**

Next, we'll continue the backwards pass by calculating new values for w1

For Hidden Layer :



$$\frac{\partial Etotal}{\partial w1} = \frac{\partial Etotal}{\partial outh1} * \frac{\partial outh1}{\partial neth1} * \frac{\partial neth1}{\partial w1}$$



Etotal = Eo1 + Eo2

$$Eo1 = \frac{1}{2}(targeto1 - Outo1)^2$$
$$Eo2 = \frac{1}{2}(targeto2 - Outo2)^2$$

(Eo1 and Eo2 not directly depend on outh1)

$$\frac{\partial Etotal}{\partial outh1} = \frac{\partial Eo1}{\partial outh1} + \frac{\partial Eo2}{\partial outh1}$$

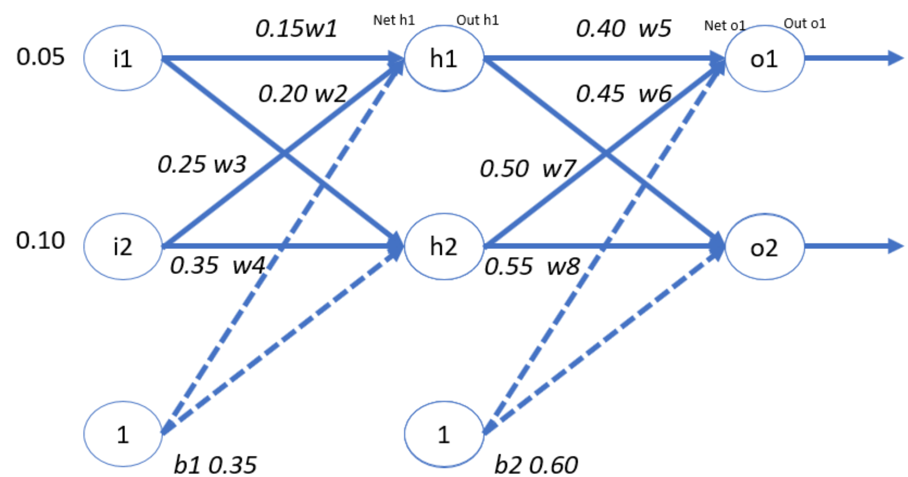
we will take both separately

$$\frac{\partial Eo1}{\partial outh1} = \frac{\partial Eo1}{\partial neto1} * \frac{\partial neto1}{\partial outh1}$$

$$\frac{\partial Eo1}{\partial neto1} = \left.\frac{\partial Eo1}{\partial outo1} * \frac{\partial outo1}{\partial neto1}\right\}$$ Both already calculated

$$\frac{\partial Eo1}{\partial neto1} = 0.74136507 \text{ x } 0.18681560 = 0.1384985$$

$$\frac{\partial neto1}{\partial outh1} = ?$$

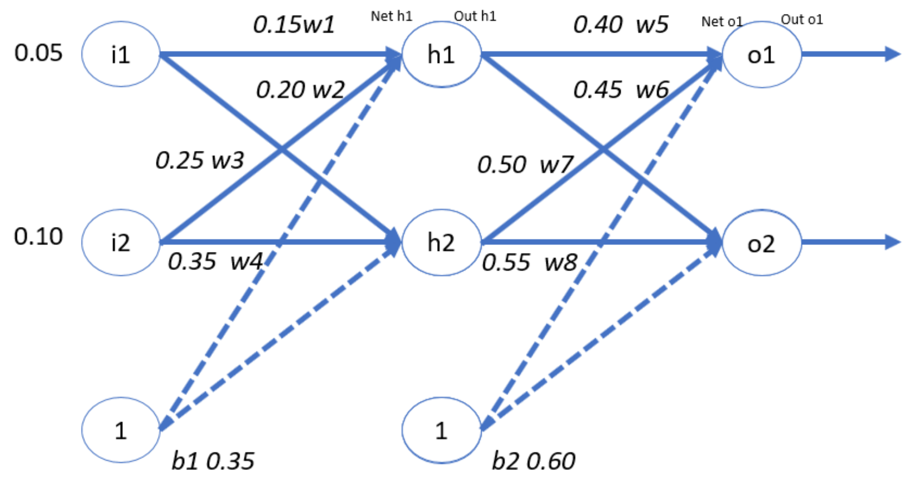


neto1 = w5 $* outh1 +$ w6 $* outh2 + b2 * 1$

$\frac{\partial neto1}{\partial outh1}$ = w5 = 0.40

$\frac{\partial Eo1}{\partial outh1} = \frac{\partial Eo1}{\partial neto1} * \frac{\partial neto1}{\partial outh1}$ = 0.1384985 x 0.40 = 0.0553994

Classification



$$\frac{\partial Eo2}{\partial outh1} = \frac{\partial Eo2}{\partial neto2} * \frac{\partial neto2}{\partial outh1}$$

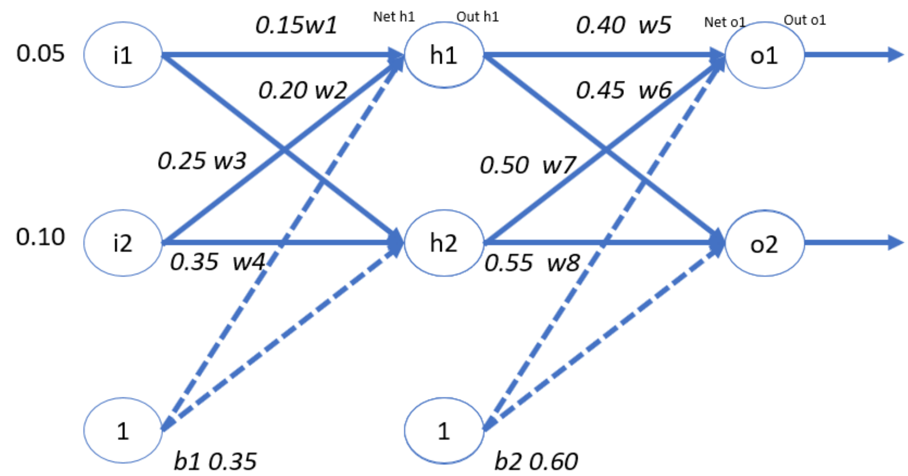
$$\frac{\partial Eo2}{\partial neto2} = \frac{\partial Eo2}{\partial outo2} * \frac{\partial outo2}{\partial neto2}$$

This time both not calculated

$$\text{Eo2} = \frac{1}{2}(\text{target o2} - \text{out o2})^2$$

$$\frac{\partial Eo2}{\partial outo2} = -(\text{target o2} - \text{out o2}) = -(0.99 - 0.772928)$$

$$\frac{\partial Eo2}{\partial outo2} = -0.217072$$



$$\frac{\partial Eo2}{\partial neto2} = \frac{\partial Eo2}{\partial outo2} * \frac{\partial outo2}{\partial neto2}$$

$$\text{Out o2} = \frac{1}{1+ e^{-neto2}}$$

$$\frac{\partial outo2}{\partial neto2} = \text{Out o2 } (1 - \text{Out o2})$$

$$= (0.7729284)(1 - 0.7729284) = 0.1755100$$

$$\frac{\partial Eo2}{\partial neto2} = (-0.217072) * (0.1755100) = -\,0.0380983$$

$$\frac{\partial Eo2}{\partial outh1} = \frac{\partial Eo2}{\partial neto2} * \frac{\partial neto2}{\partial outh1}$$

$$\frac{\partial Eo2}{\partial neto2} =- 0.0380983$$

neto2 = w7 $* outh1 +$ w8 $* outh2 + b2 * 1$

$$\frac{\partial neto2}{\partial outh1} = \text{w7} = 0.50$$

$$\frac{\partial Eo2}{\partial outh1} = \frac{\partial Eo2}{\partial neto2} * \frac{\partial neto2}{\partial outh1}$$

$$\frac{\partial Eo2}{\partial outh1} = -0.0380983 * 0.50 = -0.0190491$$

$$\frac{\partial Etotal}{\partial w1} = \frac{\partial Etotal}{\partial outh1} * \frac{\partial outh1}{\partial neth1} * \frac{\partial neth1}{\partial w1}$$

$$\frac{\partial Etotal}{\partial outh1} = \frac{\partial Eo1}{\partial outh1} + \frac{\partial Eo2}{\partial outh1}$$

$$\frac{\partial Etotal}{\partial outh1} = 0.0553994 + -0.0190491 = 0.0363503$$

$$\frac{\partial Etotal}{\partial w1} = \frac{\partial Etotal}{\partial outh1} * \frac{\partial outh1}{\partial neth1} * \frac{\partial neth1}{\partial w1}$$

outh1 = $\frac{1}{1+ e^{-net\,h1}}$

$\frac{\partial outh1}{\partial neth1}$ = outh1 x (1 - outh1) = 0.5932699 X (1- 0.5932699)

$\frac{\partial outh1}{\partial neth1}$ = 0.2413007

$$\frac{\partial Etotal}{\partial w1} = \frac{\partial Etotal}{\partial outh1} * \frac{\partial outh1}{\partial neth1} * \frac{\partial neth1}{\partial w1}$$

neth1 = i1 $*$ w1 + i2 $*$ w2 + b1 $*$ 1

$$\frac{\partial neth1}{\partial w1} = \text{i1} = 0.05$$

$$\frac{\partial Etotal}{\partial w1} = 0.0363503 * 0.2413007 * 0.05$$

$$\frac{\partial Etotal}{\partial w1} = \mathbf{0.00043856}$$

We can write this in short as :

$$\frac{\partial Etotal}{\partial w1} = \left(\sum_o \frac{\partial Etotal}{\partial outo} * \frac{\partial outo}{\partial neto} * \frac{\partial neto}{\partial outh1}\right) * \frac{\partial outh1}{\partial neth1} * \frac{\partial neth1}{\partial w1}$$

$$\frac{\partial Etotal}{\partial w1} = \underbrace{\left(\sum_o \boldsymbol{\delta}_o * w_{ho}\right) * (\mathrm{outh1})\ (1\text{-}\mathrm{outh1})} * i1$$

$$\frac{\partial Etotal}{\partial w1} = \boldsymbol{\delta}_{h1} * i1$$

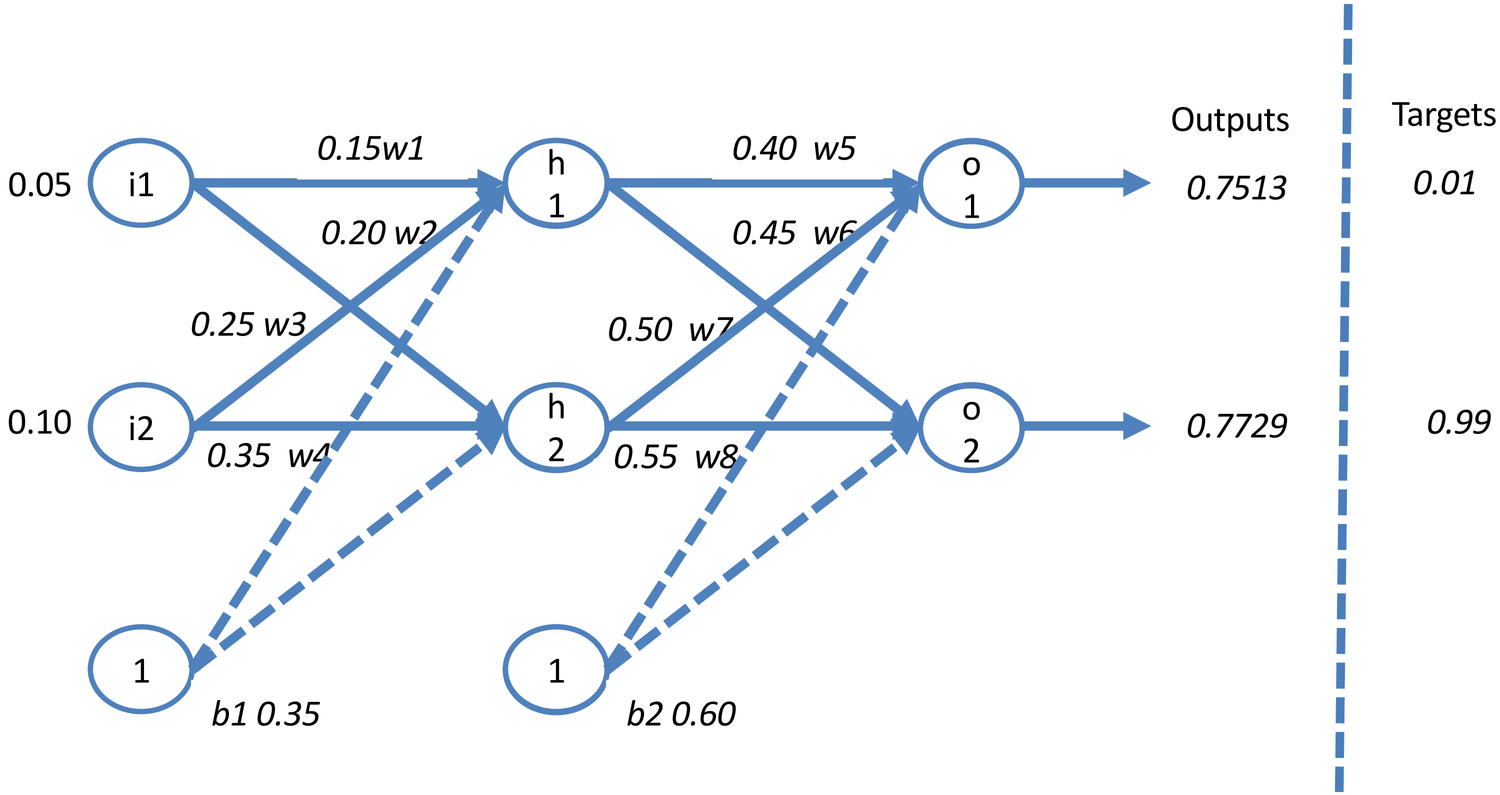
Updation of ***weight w1 :***

$$w1\_new = w1 - \eta \times \frac{\partial Etotal}{\partial w1}$$

$$w1\_new = 0.15 - 0.5 * 0.00043856 = 0.149780$$

With the same procedure weights **w2 w3 w4** and bias **b1** will be computed.

w1_new = 0.19956143
w2_new = 0.24975114
w3_new = 0.29950229



Example of NN

- Finally, we've updated all of our weights! When we fed forward the 0.05 and 0.1 inputs originally, the error on the network was 0.298371109.
- After this first round of backpropagation, the total error is now down to 0.291027924.
- It might not seem like much, but after repeating this process 10,000 times, for example, the error plummets to 0.0000351085.
- At this point, when we feed forward 0.05 and 0.1, the two outputs neurons generate 0.015912196 (vs 0.01 target) and 0.984065734 (vs 0.99 target).

Neural Network

Learning

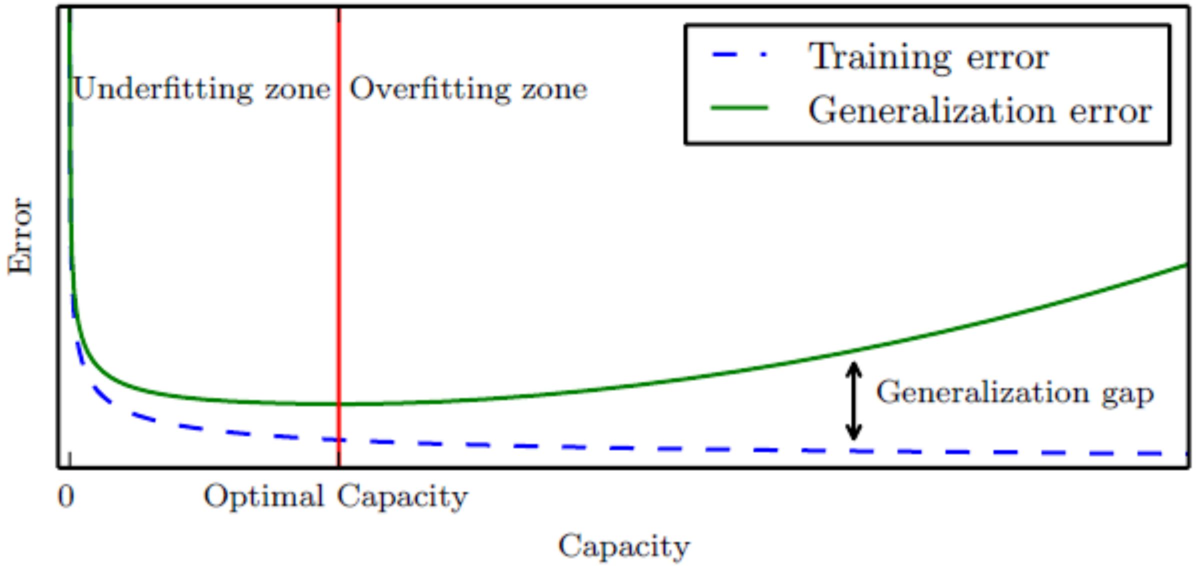


Figure 5.3: Typical relationship between capacity and error. Training and test error behave differently. At the left end of the graph, training error and generalization error are both high. This is the **underfitting regime**. As we increase capacity, training error decreases, but the gap between training and generalization error increases. Eventually, the size of this gap outweighs the decrease in training error, and we enter the **overfitting regime**, where capacity is too large, above the **optimal capacity**.