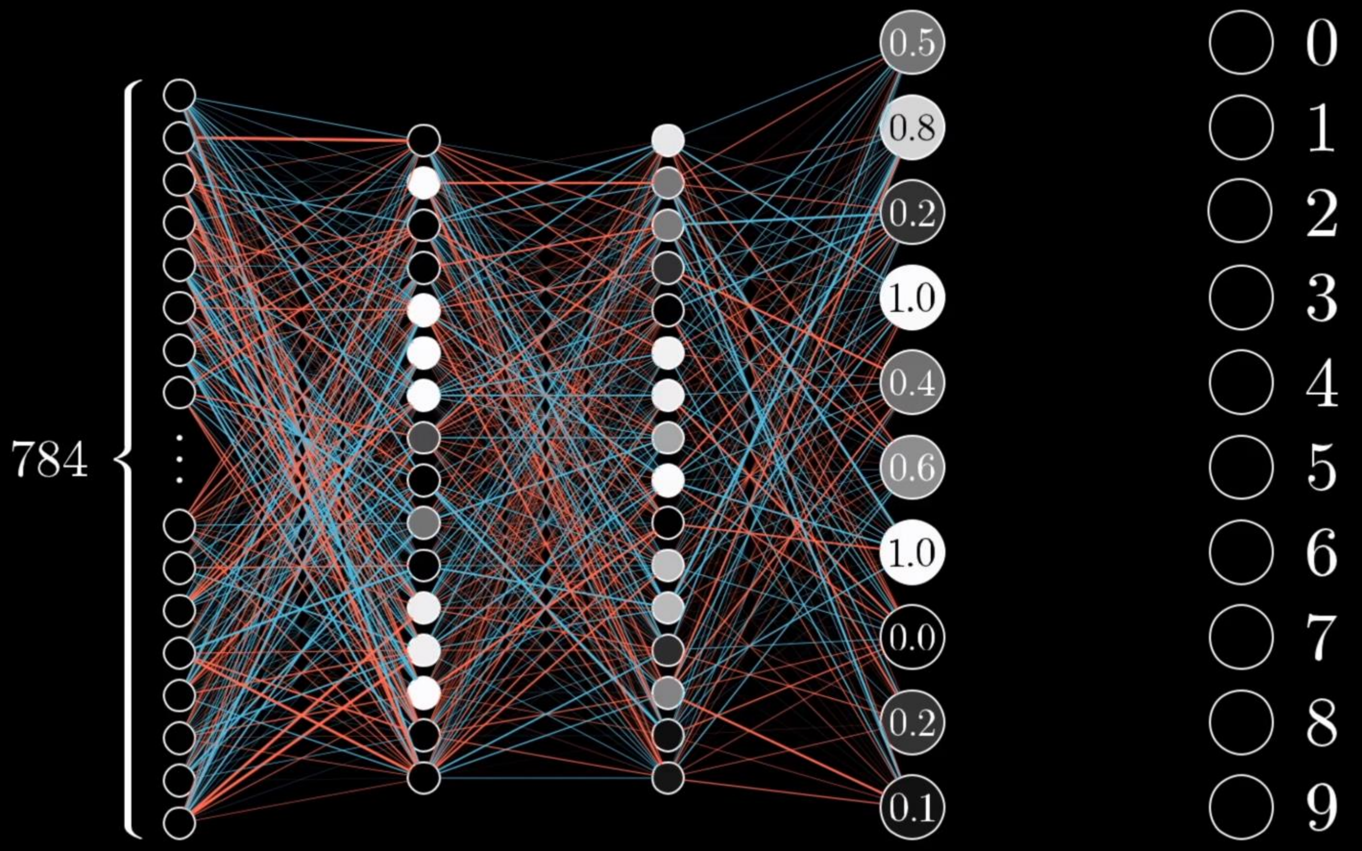
Back Propagation and Activation Functions

Mass Street University
PER EDUCATIONEM PROGRESSUS

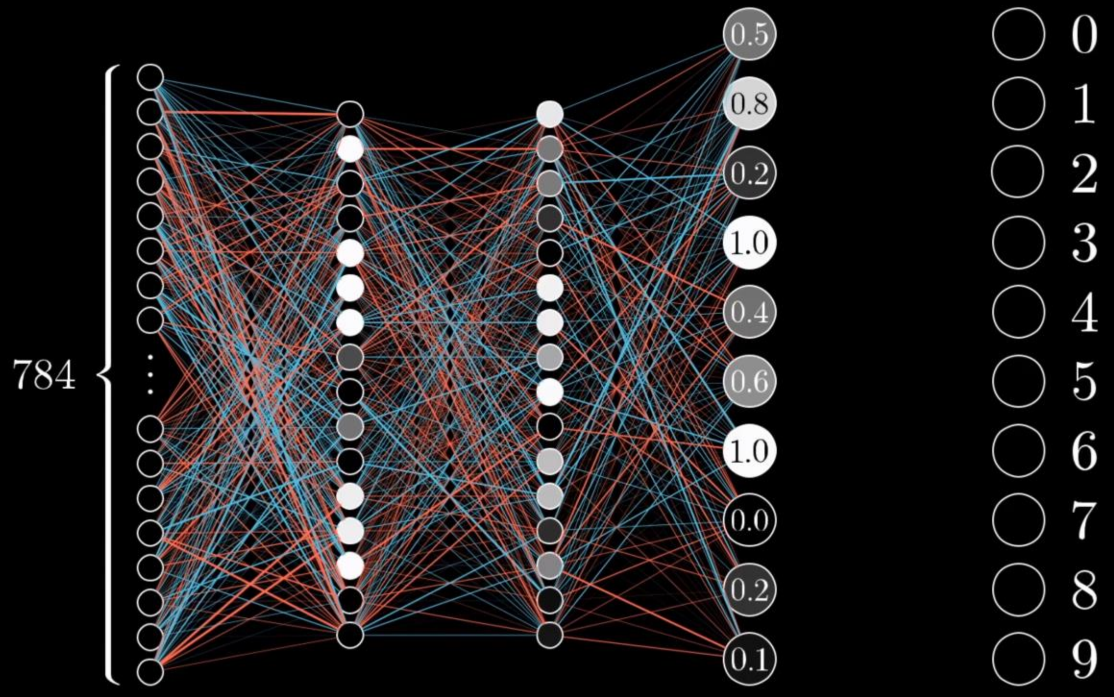
An example Network

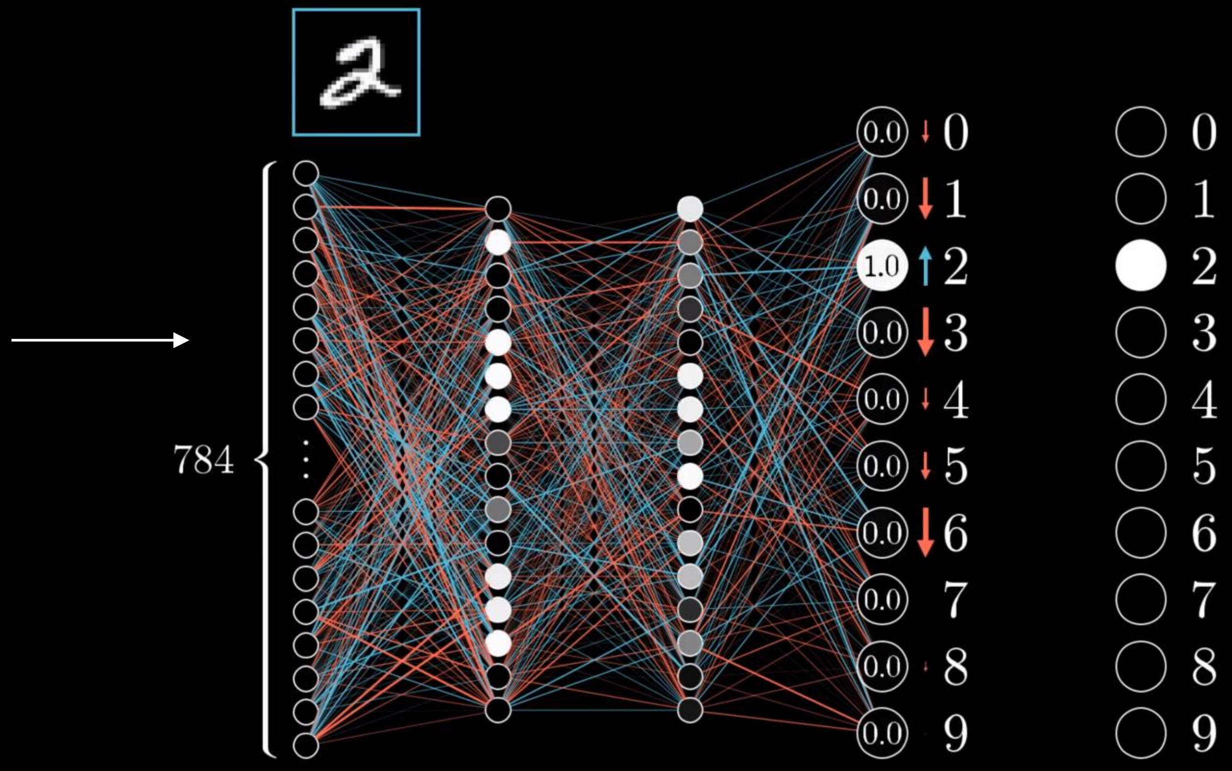
- 28 x 28 pixel images as input.
- Input vector of 784.
- 10 target variables or ‘classes’
- It has 16 neurons in two hidden layers



Mass Street University
PER EDUCATIONEM PROGRESSUS

Example Cont.





Mass Street University
PER EDUCATIONEM PROGRESSUS

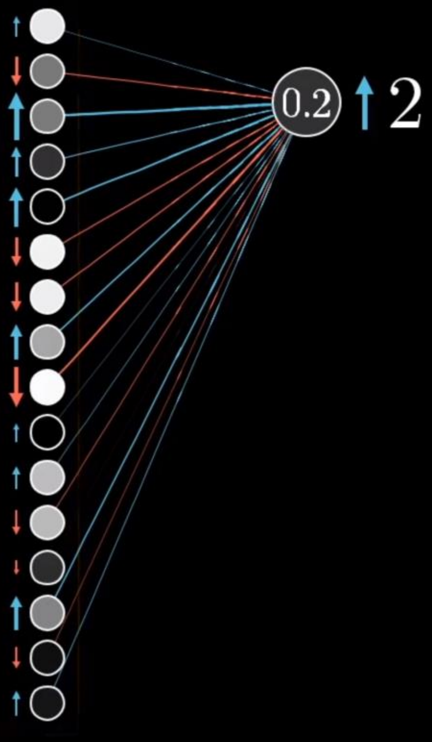
Updating a Neuron through back prop.

$$0.2 = \sigma(w_0 a_0 + w_1 a_1 + \cdots + w_{n-1} a_{n-1} + b)$$

Increase b

Increase w_i
in proportion to a_i

Change a_i
in proportion to w_i

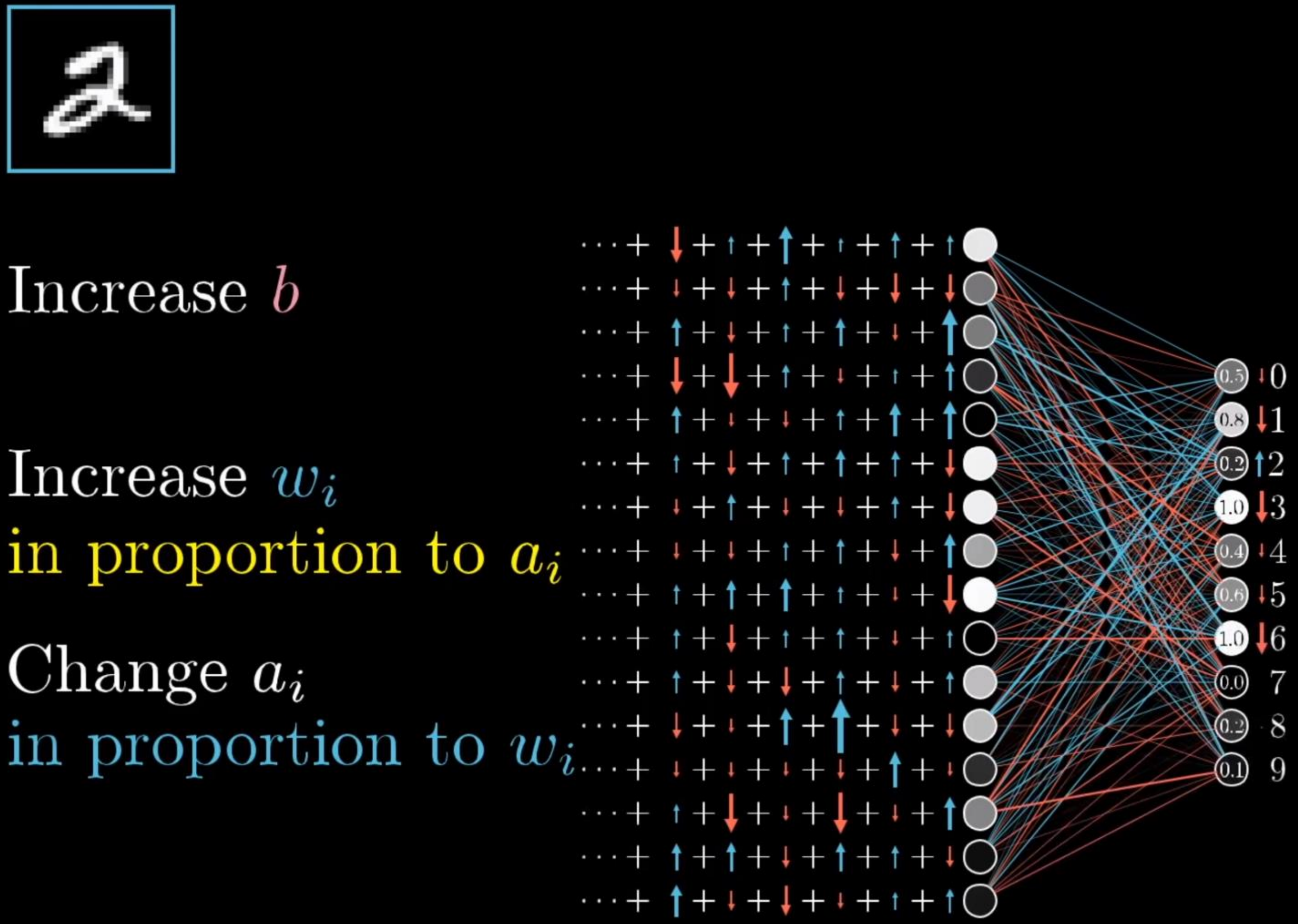


Weight Matrix Update

Average over all training data

	2	5	0	4	1	9	$\cdots$	
w_0	-0.08	$+0.02$	-0.02	$+0.11$	-0.05	-0.14	$\cdots$	$\rightarrow -0.08$
w_1	-0.11	$+0.11$	$+0.07$	$+0.02$	$+0.09$	$+0.05$	$\cdots$	$\rightarrow +0.12$
w_2	-0.07	-0.04	-0.01	$+0.02$	$+0.13$	-0.15	$\cdots$	$\rightarrow -0.06$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
$w_{13,001}$	$+0.13$	$+0.08$	-0.06	-0.09	-0.02	$+0.04$	$\cdots$	$\rightarrow +0.04$

Updating whole network through back prop



General Rules

- The more data you use to update the longer it will take to train.
- Batches speed up the learning process.
- We update weights after each batch is trained on.

Cost Functions in Deep Learning

- Two primary functions are binary and categorical cross entropy.
- Binary cross entropy takes in a single binary value of either 1 or 0.
- Categorical takes in n length binary chain of 0’s and 1’s where 1’s are positive classes.
- Categorical can have multiple correct classes.

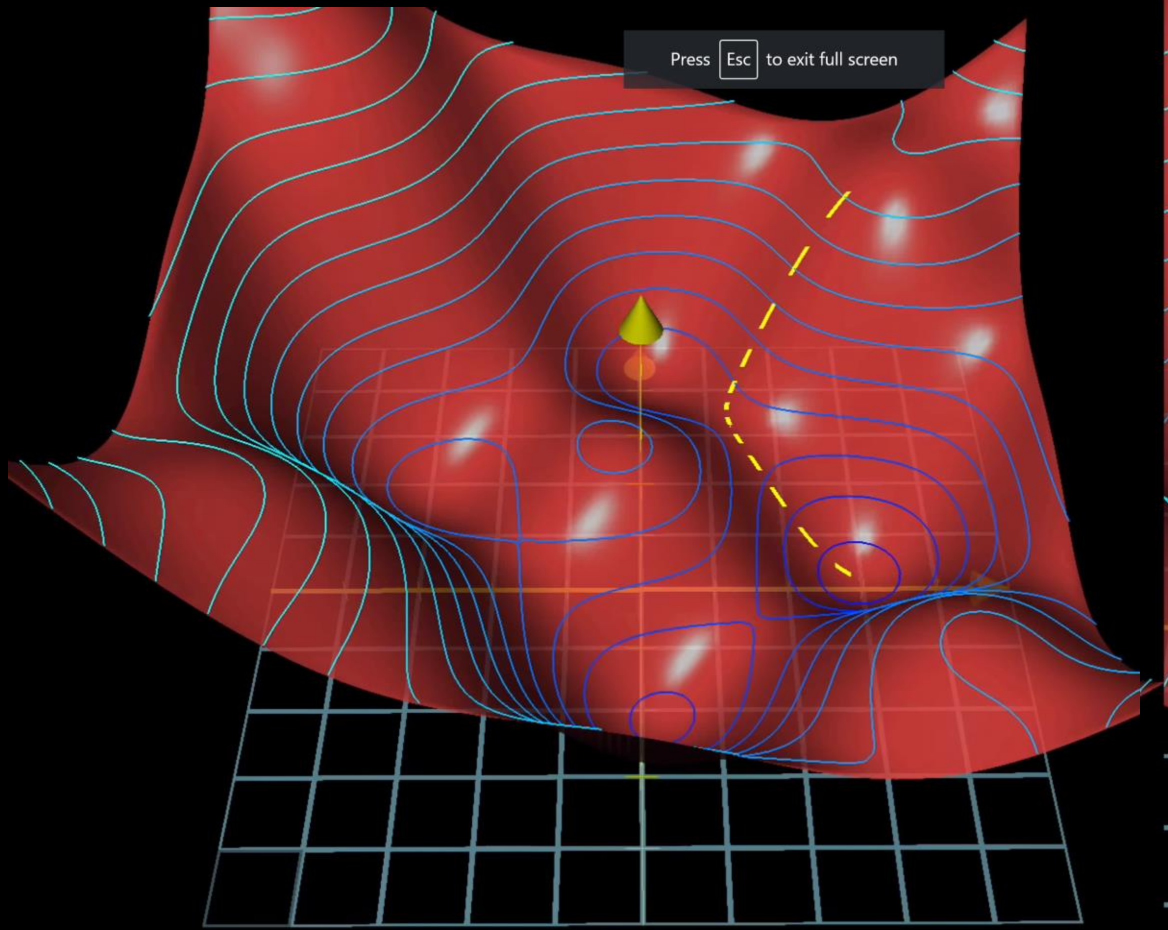
Or 0, 0, 0, 1, 0

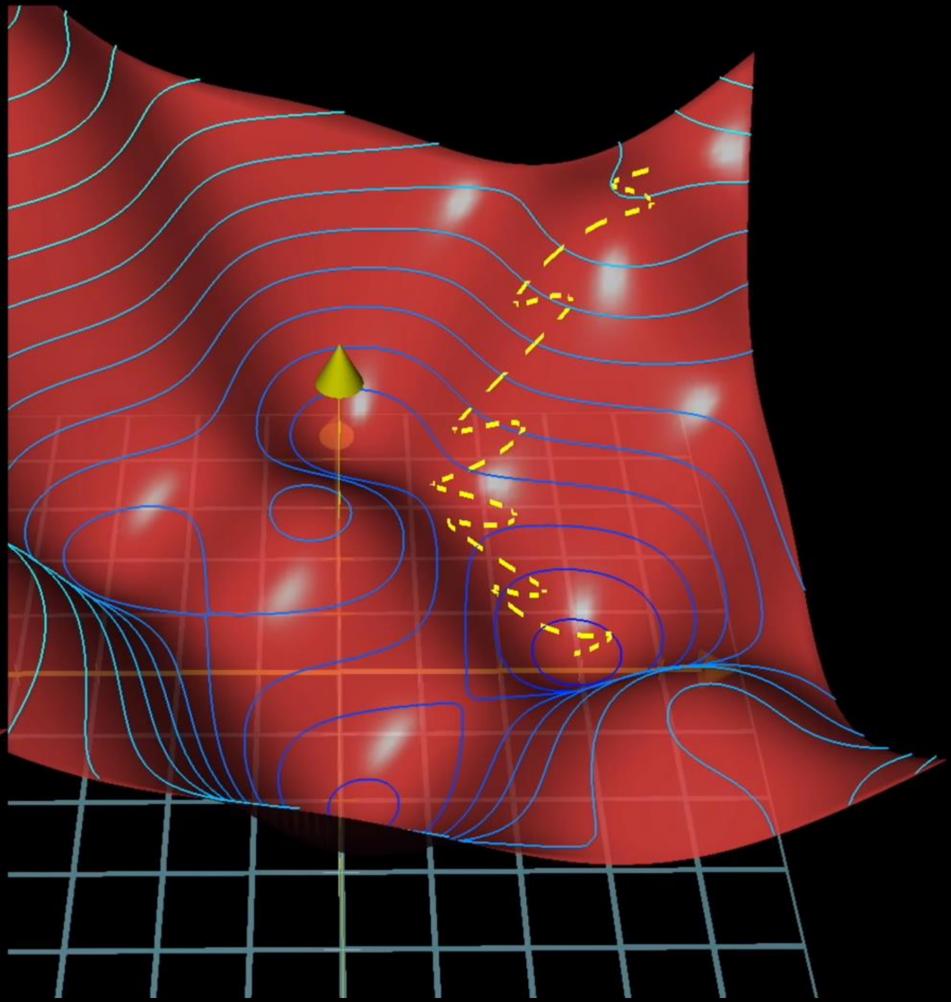
Mass Street University
PER EDUCATIONEM PROGRESSUS

Stochastic Gradient Descent

- Is used to find the local and global minima for any function.
- Works like Gradient Descent except it is batched and it has random starting points.
- It works like blind hill climbing.

GD vs SGD





Mass Street University

PER EDUCATIONEM PROGRESSUS

Learning Rate

- Low learning rate converges slower but is more stable.
- High learning rate converges quickly but may ‘step off’ the minima and fall off a cliff. This leads to the model not being able to converge at all.
- Start high and lower it each training epoch.

SGD update

$$\theta_1 = \theta_0 - \alpha \Delta Q \theta$$

- θ_1 is next position
- θ_0 is current position
- α is learning rate
- $\Delta Q \theta$ is the cost function
- When $\theta_1 = \theta_0$ then the network can no longer learn new things.

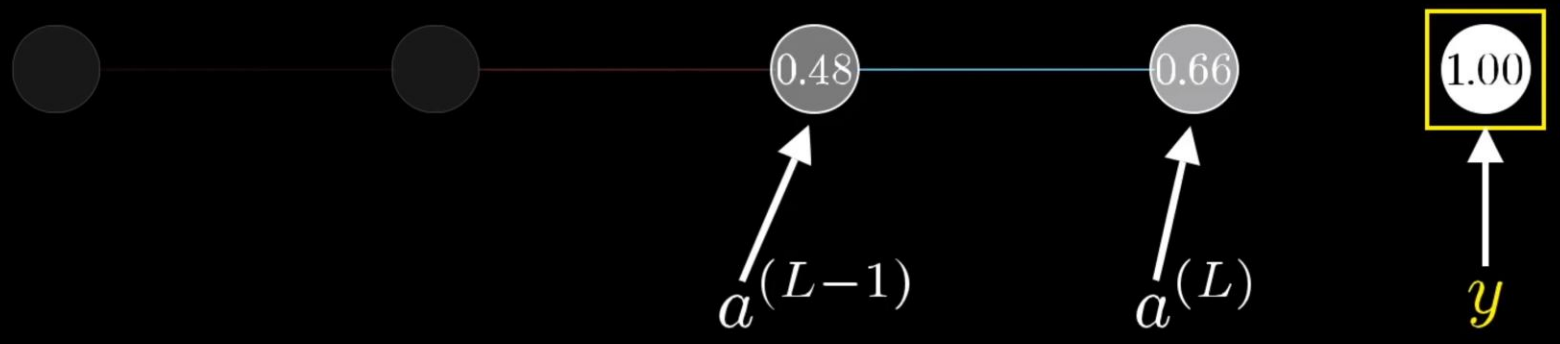
Mass Street University
PER EDUCATIONEM PROGRESSUS

Cost Function

$$\text{Cost} \longrightarrow C_0(\ldots) = (a^{(L)} - y)^2$$

$$a^{(L)} = \sigma(w^{(L)} a^{(L-1)} + b^{(L)})$$

Desired output



Chain Rule of SGD

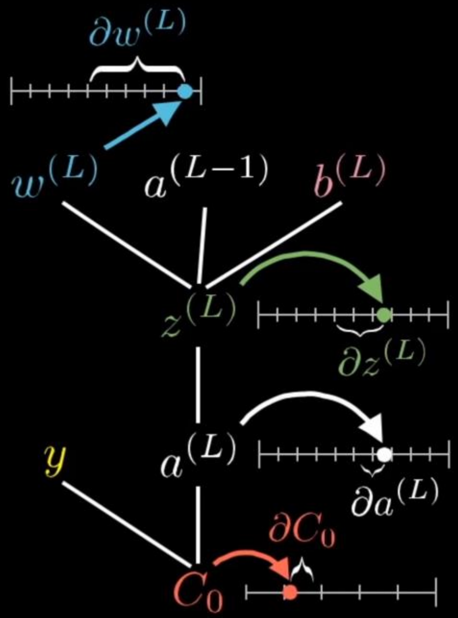
$$\boxed{\frac{\partial C_0}{\partial w^{(L)}} = \frac{\partial z^{(L)}}{\partial w^{(L)}} \frac{\partial a^{(L)}}{\partial z^{(L)}} \frac{\partial C0}{\partial a^{(L)}}}$$

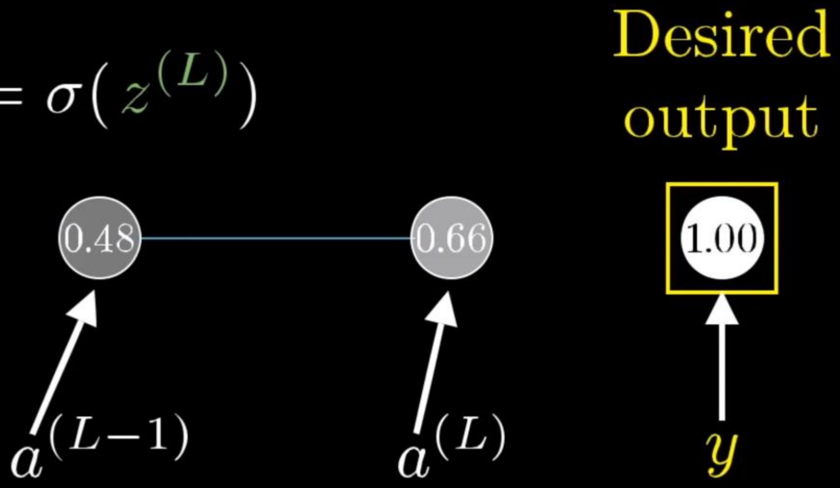
Chain rule

$$C_0(\ldots) = (a^{(L)} - y)^2$$

$$z^{(L)} = w^{(L)} a^{(L-1)} + b^{(L)}$$

$$a^{(L)} = \sigma(z^{(L)})$$





Values of the derivatives

$$\frac{\partial C_0}{\partial w^{(L)}} = \frac{\partial z^{(L)}}{\partial w^{(L)}} \frac{\partial a^{(L)}}{\partial z^{(L)}} \frac{\partial C0}{\partial a^{(L)}}$$

$$\frac{\partial C0}{\partial a^{(L)}} = 2(a^{(L)} - y)$$

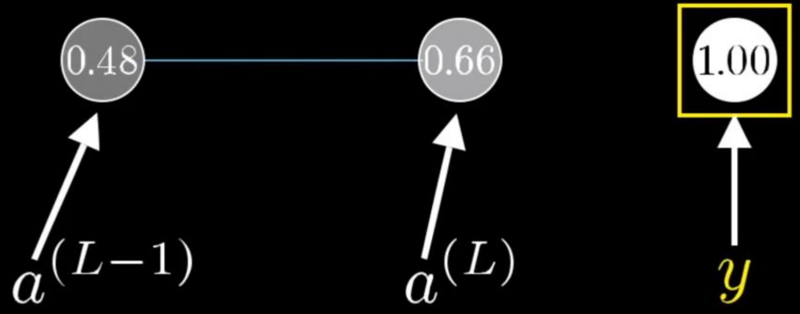
$$\frac{\partial a^{(L)}}{\partial z^{(L)}} = \sigma'(z^{(L)})$$

$$\frac{\partial z^{(L)}}{\partial w^{(L)}} = a^{(L-1)}$$

$$C_0 = (a^{(L)} - y)^2$$

$$z^{(L)} = w^{(L)} a^{(L-1)} + b^{(L)}$$

$$a^{(L)} = \sigma(z^{(L)})$$



Final Equation

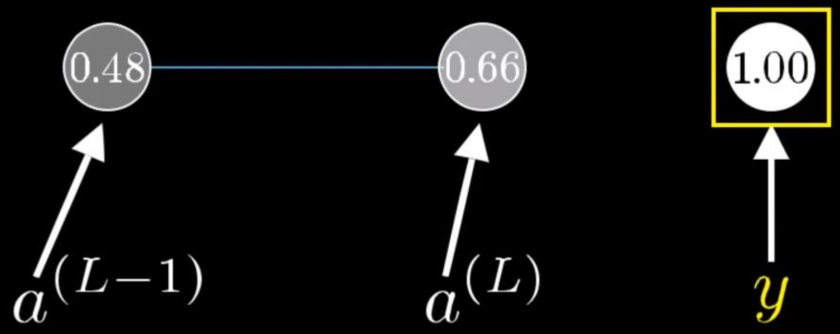
$$\frac{\partial C_0}{\partial w^{(L)}} = \frac{\partial z^{(L)}}{\partial w^{(L)}} \frac{\partial a^{(L)}}{\partial z^{(L)}} \frac{\partial C0}{\partial a^{(L)}} = a^{(L-1)} \sigma'(z^{(L)}) 2(a^{(L)} - y)$$

$$C_0 = (a^{(L)} - y)^2$$

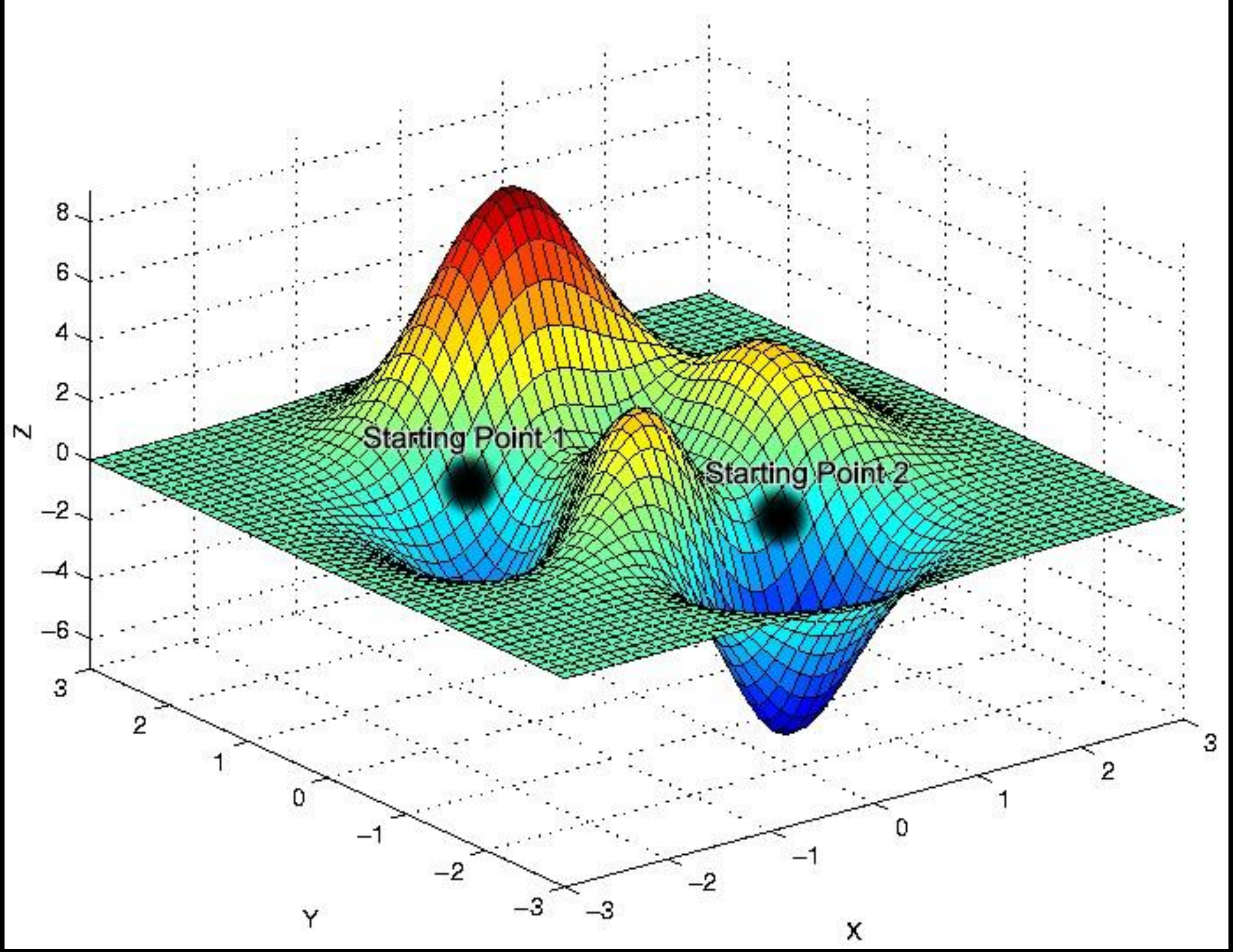
$$z^{(L)} = w^{(L)} a^{(L-1)} + b^{(L)}$$

$$a^{(L)} = \sigma(z^{(L)})$$

$$\underbrace{\frac{\partial C}{\partial w^{(L)}}}_{\text{Derivative of full cost function}} = \overbrace{\frac{1}{n} \sum_{k=0}^{n-1} \frac{\partial C_k}{\partial w^{(L)}}}^{\text{Average of all training examples}}$$

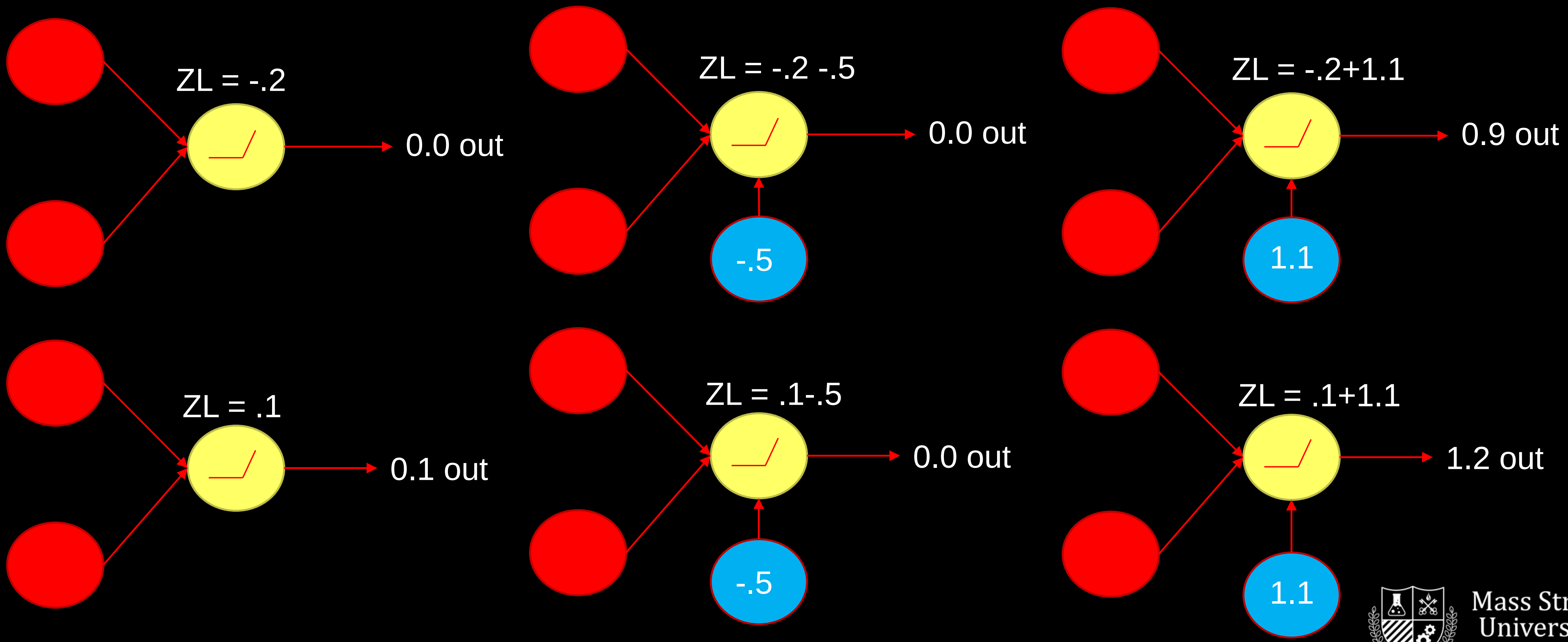


3D model



Mass Street University
PER EDUCATIONEM PROGRESSUS

How Bias and Activations Work



Mass Street University
PER EDUCATIONEM PROGRESSUS