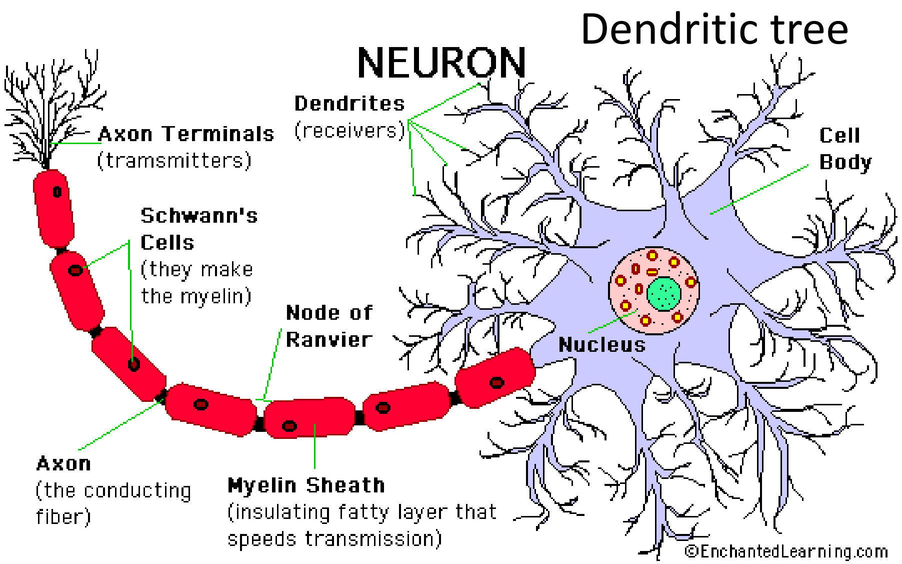
Artificial Intelligence Fundamentals

Learning: Neural Networks

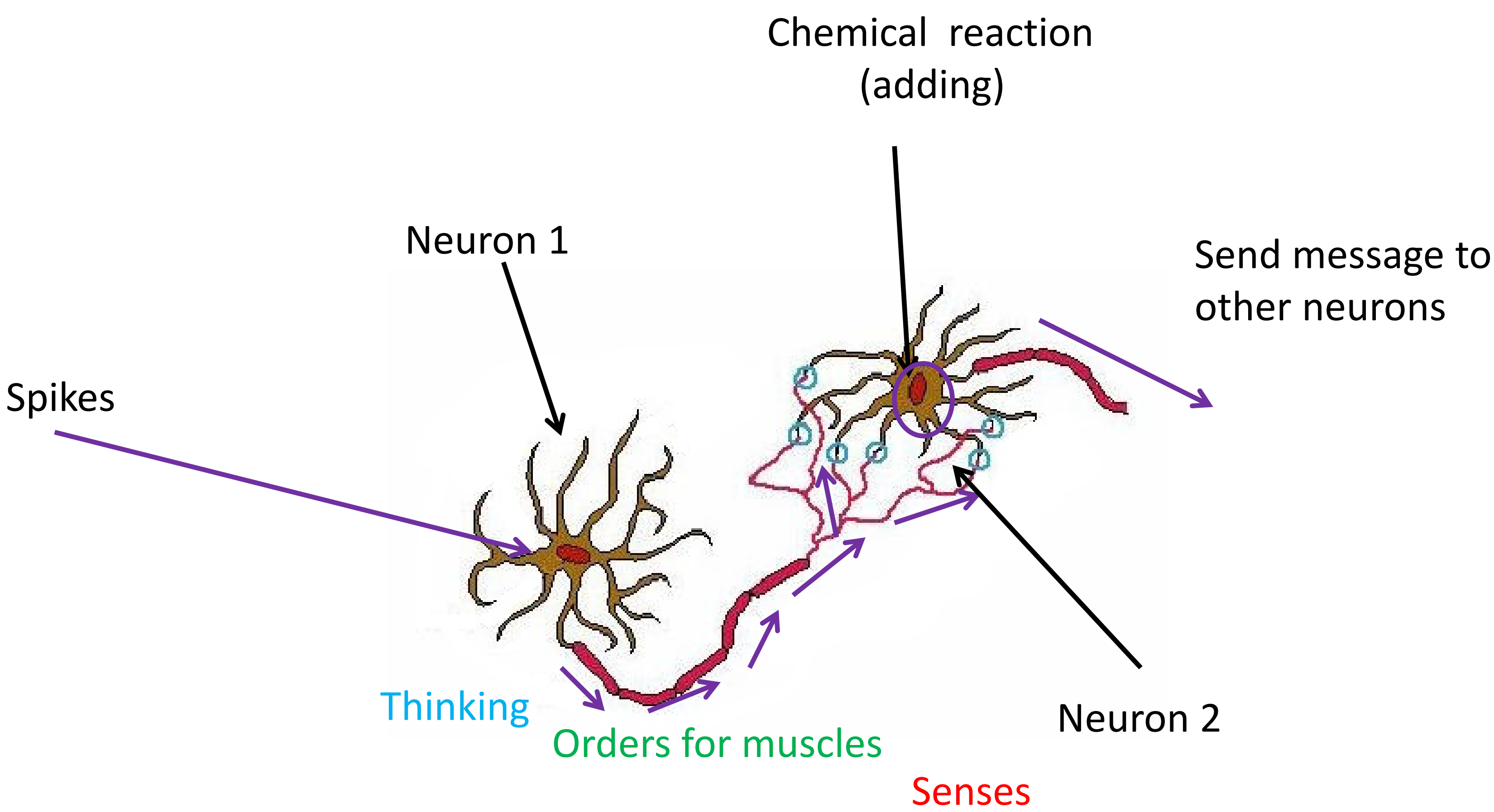
Biology



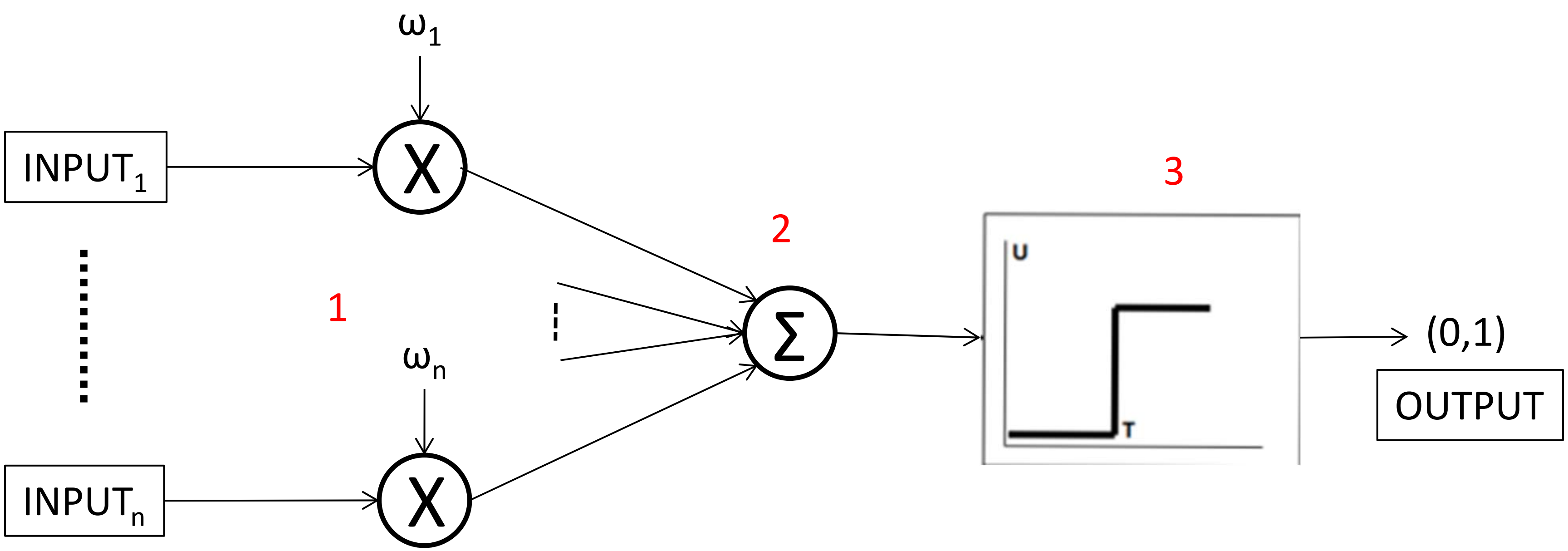
Major observations:

1. Synaptic weights
2. Cumulative effect
3. All or none

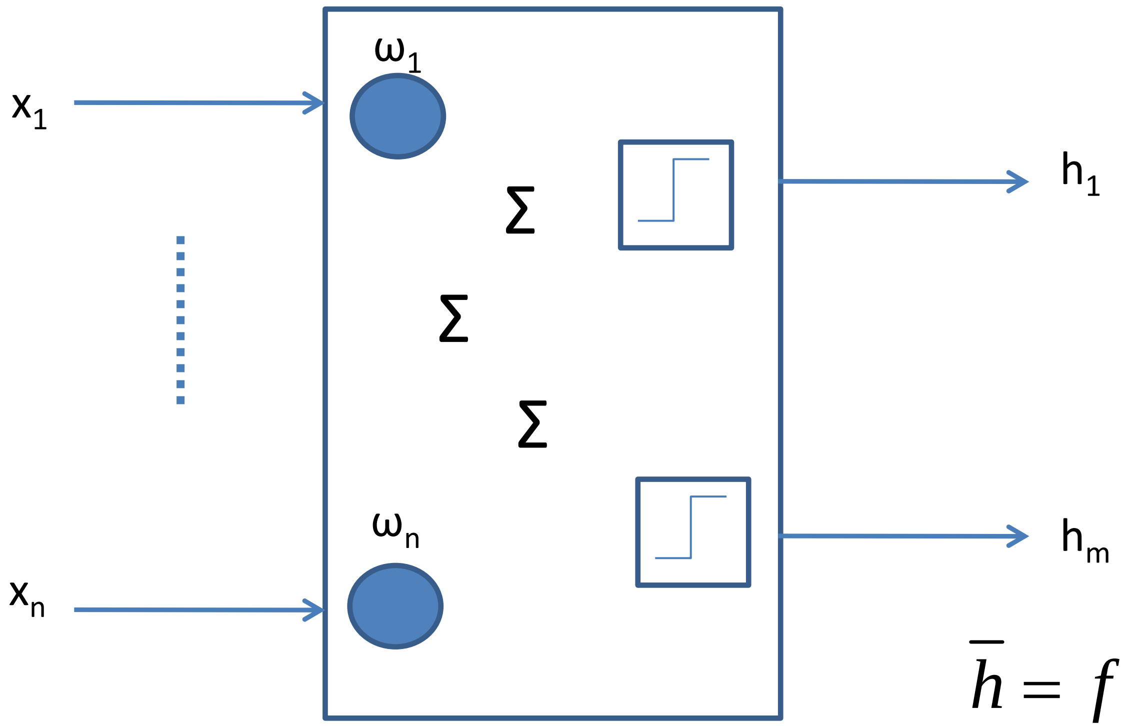
Neurons connectivity



Artificial neuron model



Neural Network



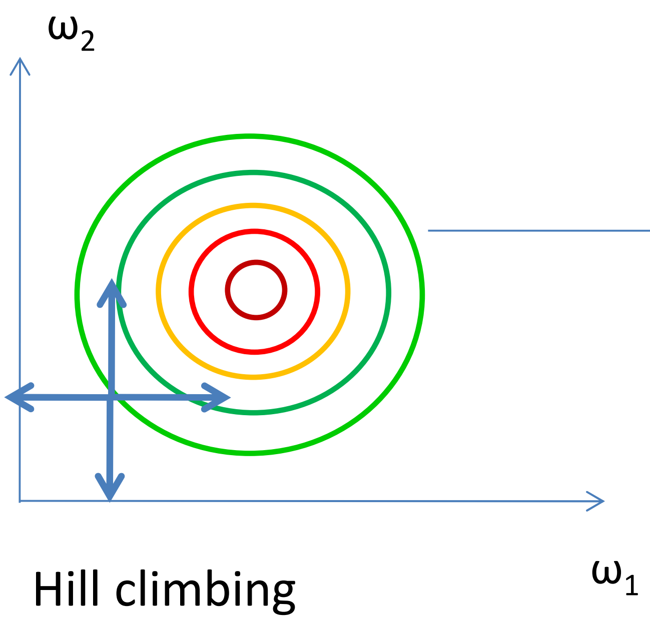
$$\overline{h} = f\left(\overline{x}, \overline{\omega}\right)$$

Activation function $\quad \overline{d} = g\left(\overline{x}\right)$

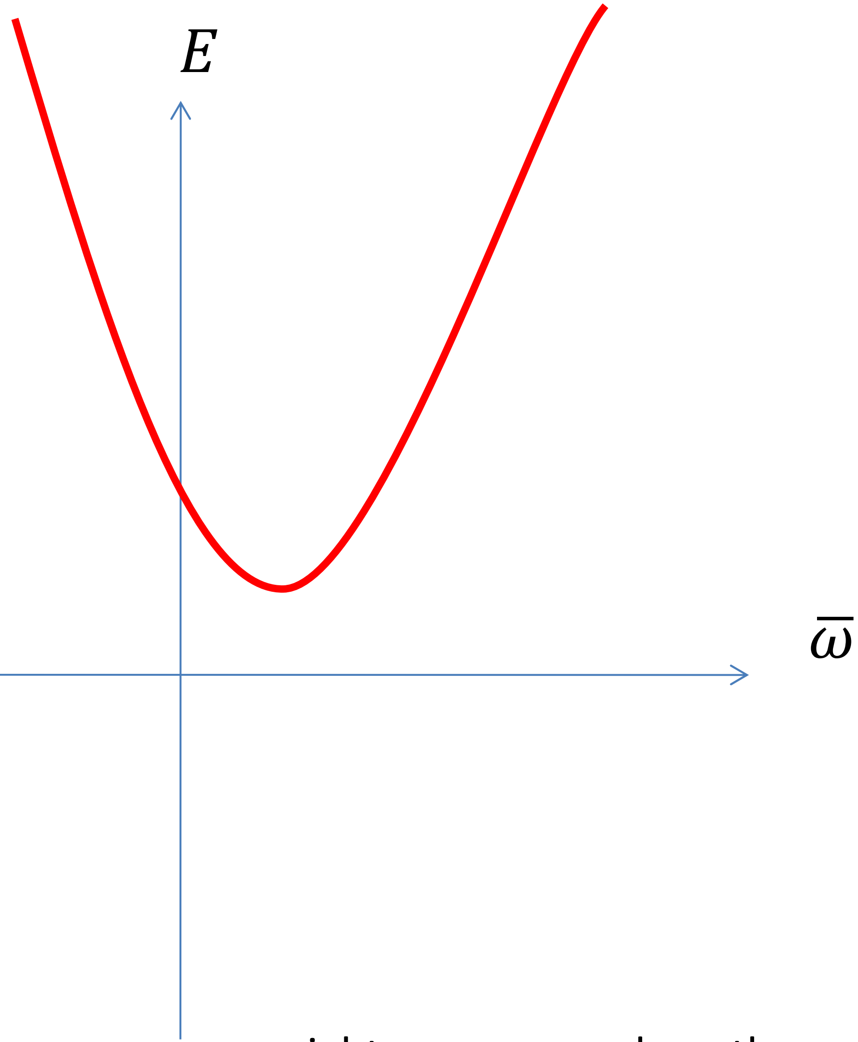
Performance – error function

$$E\left(\overline{d}, \overline{h}\right) = \frac{1}{2}\left\|\overline{d} - \overline{h}\right\|^2$$

Minimize the function

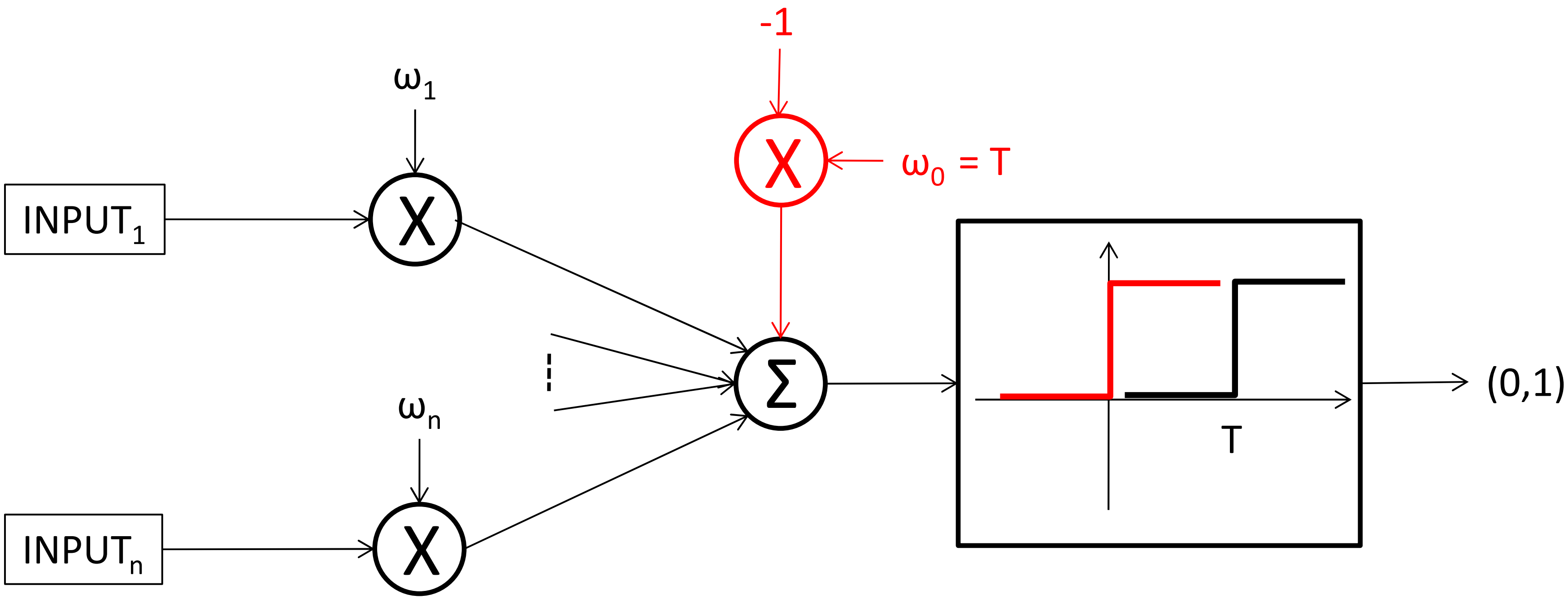


Hill climbing

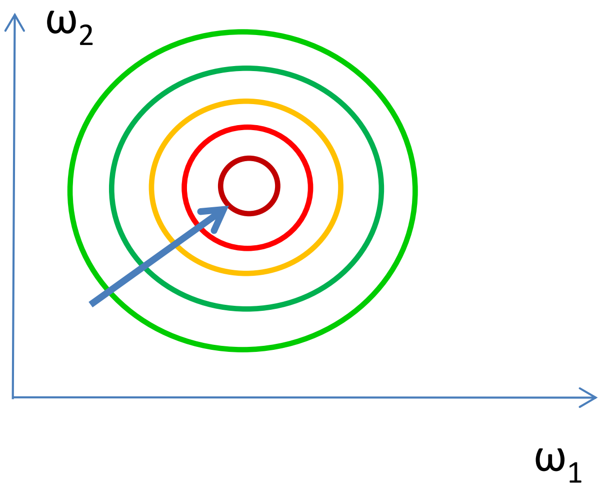


There are many weights, so we need another approach.

The threshold



Gradient descent

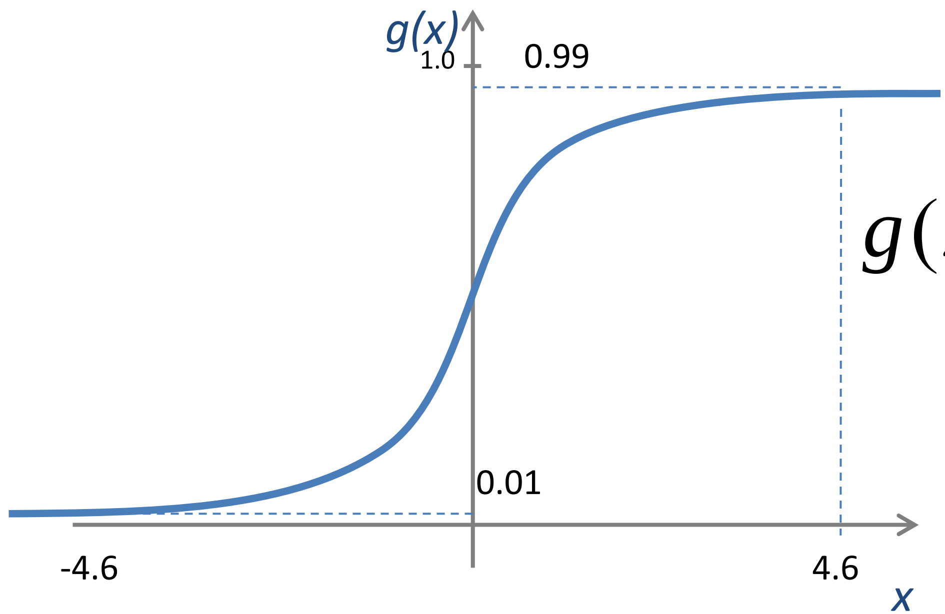


Gradient descent – moving toward the minimum.

learning rate

$$\Delta\omega = \sigma\left(\frac{\partial E}{\partial \omega_1} i + \frac{\partial E}{\partial \omega_2} j\right)$$

We need a continuous function – sigmoid function.

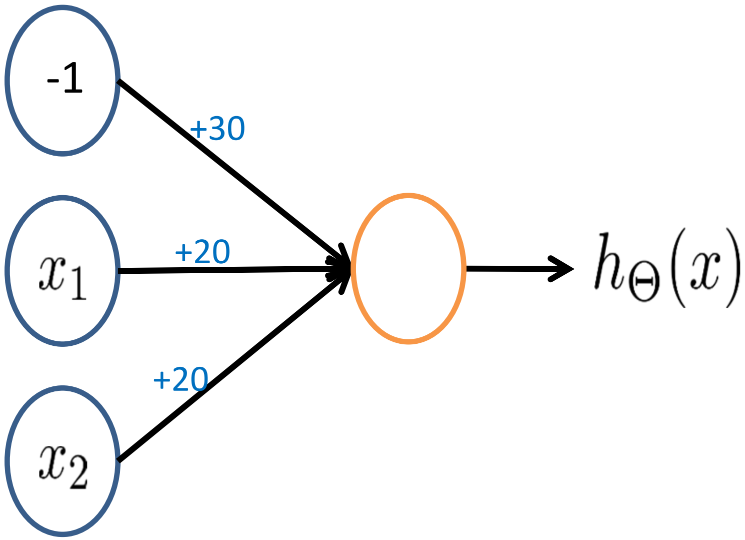


$$g(x) = \frac{1}{1+e^{-x}}$$

Example: logical AND

$x_1, x_2 \in \{0, 1\}$

$y = x_1 \text{ AND } x_2$

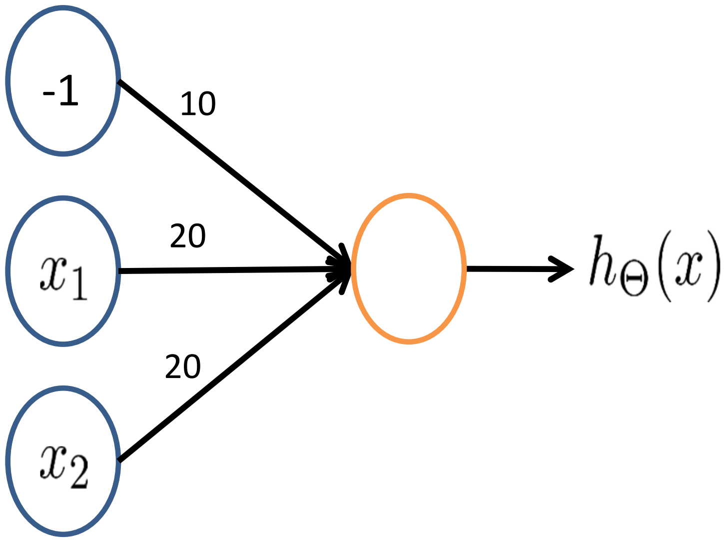


x_1	x_2	$h_\Theta(x)$
0	0	g(-30) ≈ $\boxed{0}$
0	1	g(-10) ≈ $\boxed{0}$
1	0	g(-10) ≈ $\boxed{0}$
1	1	g(10) ≈ $\boxed{1}$

$h(x) = x_1 \text{ AND } x_2$

$h(x) = g(-30 + 20\, x_1 + 20\, x_2)$

Example: logical OR

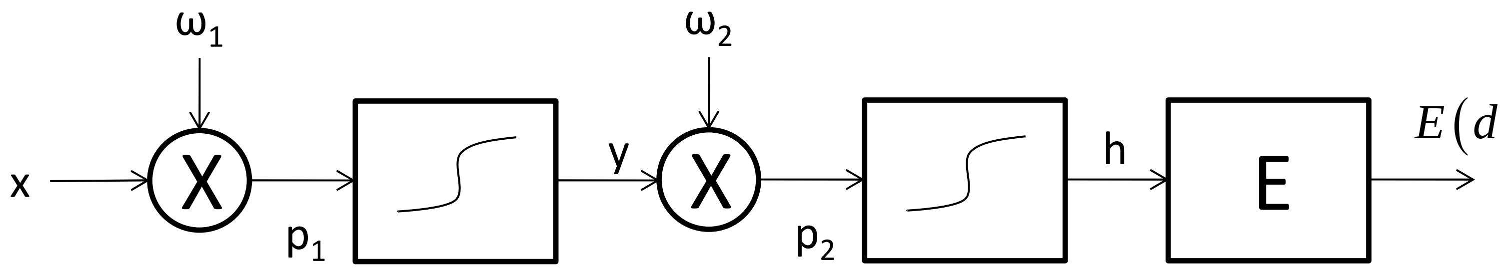


x_1	x_2	$h_\Theta(x)$
0	0	$g(-10) \approx \boxed{0}$
0	1	$g(10) \approx \boxed{1}$
1	0	$g(10) \approx \boxed{1}$
1	1	$g(30) \approx \boxed{1}$

$h(x) = g(-10 + 20\,x_1 + 20\,x_2)$

$h(x) = x_1 \text{ OR } x_2$

A simple neural network



$$E(d,h)=\frac{1}{2}(d-h)^2$$

$$\frac{\partial E}{\partial \omega_2}=\frac{\partial E}{\partial h}\frac{\partial h}{\partial \omega_2}=-(d-h)\frac{\partial h}{\partial \omega_2}=-(d-h)\frac{\partial h}{\partial p_2}\frac{\partial p_2}{\partial \omega_2}=-(d-h)\frac{\partial h}{\partial p_2}y$$

$$\frac{\partial E}{\partial \omega_1}=\frac{\partial E}{\partial h}\frac{\partial h}{\partial \omega_1}=-(d-h)\frac{\partial h}{\partial \omega_1}=-(d-h)\frac{\partial h}{\partial p_2}\frac{\partial p_2}{\partial \omega_1}=-(d-h)\frac{\partial h}{\partial p_2}\frac{\partial p_2}{\partial y}\frac{\partial y}{\partial \omega_1}$$

$$=-(d-h)\frac{\partial h}{\partial p_2}\omega_2\frac{\partial y}{\partial p_1}\frac{\partial p_1}{\partial \omega_1}=-(d-h)\frac{\partial h}{\partial p_2}\omega_2\frac{\partial y}{\partial p_1}x$$

Derivative of the sigmoid

$$g = \frac{1}{1+e^{-x}}$$

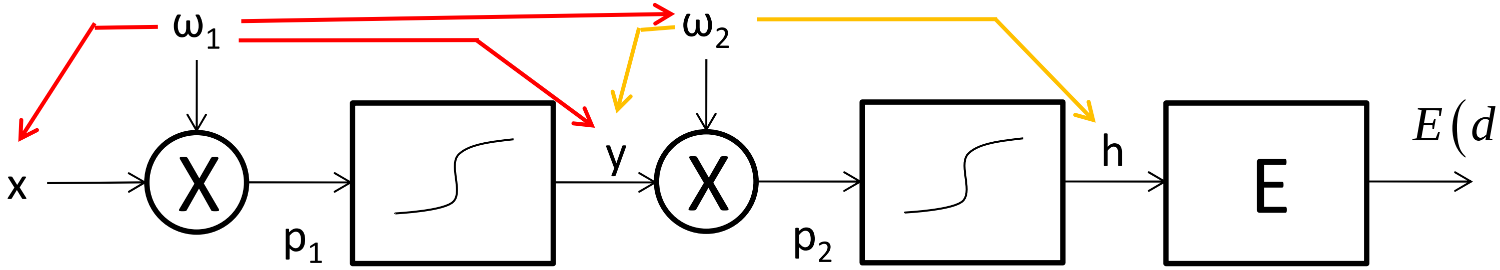
$$\frac{dg}{dx} = \frac{d}{dx}(1+e^{-x})^{-1} = -(1+e^{-x})^{-2}(-e^{-x}) = \frac{e^{-x}}{(1+e^{-x})^{-2}}$$

$$= \frac{1+e^{-x}-1}{(1+e^{-x})^{-2}} = \frac{1}{(1+e^{-x})}\left[\frac{1+e^{-x}}{(1+e^{-x})} - \frac{1}{(1+e^{-x})}\right] = g(1-g)$$

$$\frac{\partial E}{\partial \omega_2} = -(d-h)\frac{\partial h}{\partial p_2} y \quad = -(d-h)h\,(1-h)\,y$$

$$\frac{\partial E}{\partial \omega_1} = -(d-h)\frac{\partial h}{\partial p_2}\omega_2 \frac{\partial y}{\partial p_1} x = -(d-h)h(1-h)\,\omega_2 y(1-y)x$$

Backpropagation algorithm



$$E(d,h)=\frac{1}{2}(d-h)^2$$

$$\frac{\partial E}{\partial \omega_2}=(h-d)h(1-h)y$$

$$\frac{\partial E}{\partial \omega_1}=\left(h-d\right)h(1-h)\omega_2 y(1-y)x$$

It's only a local computation - depends by the staff on the vicinity

Changing the weights

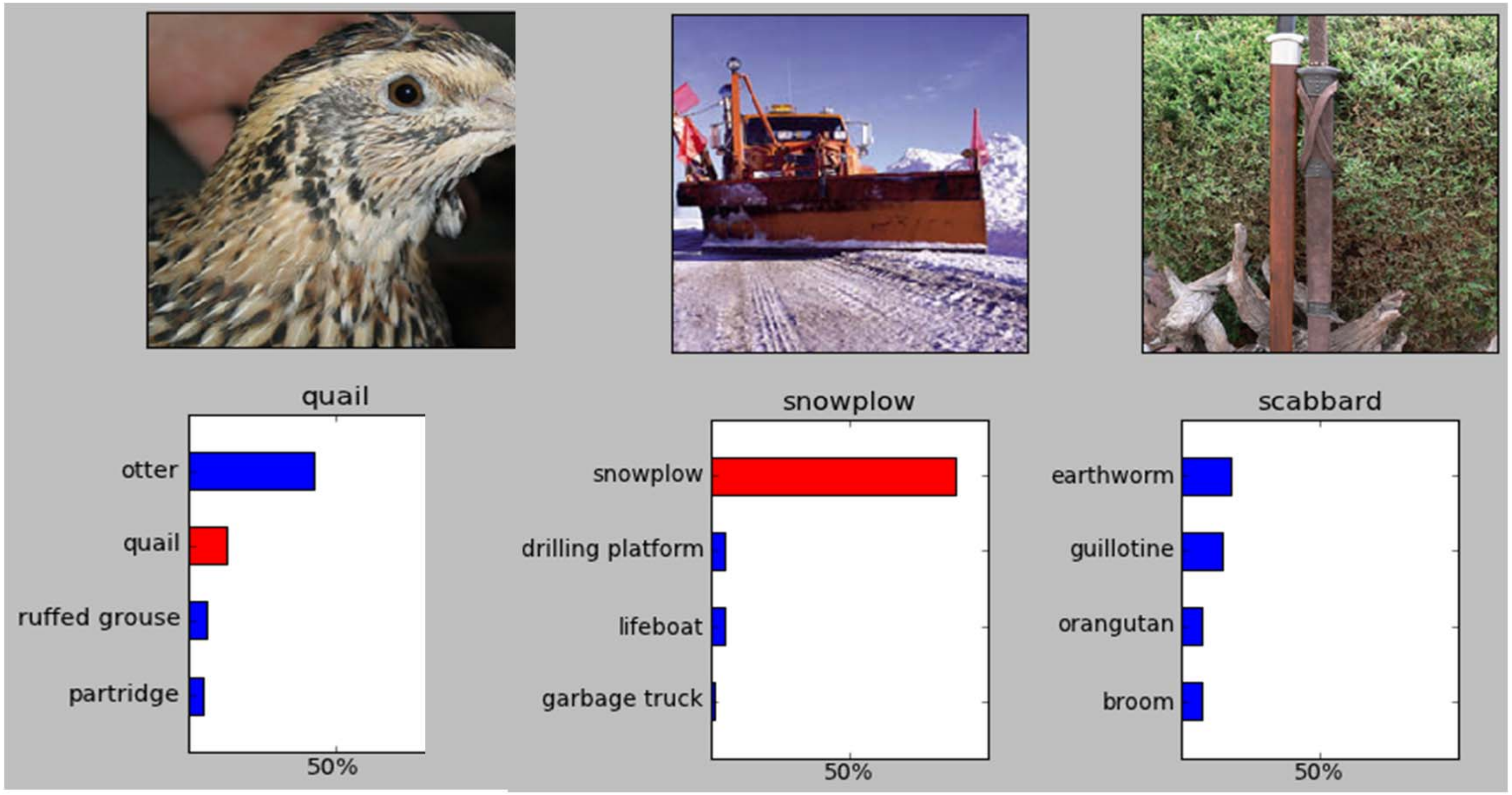
$$\omega_1 = \omega_{1old} + \sigma \frac{\partial E}{\partial \omega_1}$$

$$\omega_2 = \omega_{2old} + \sigma \frac{\partial E}{\partial \omega_2}$$

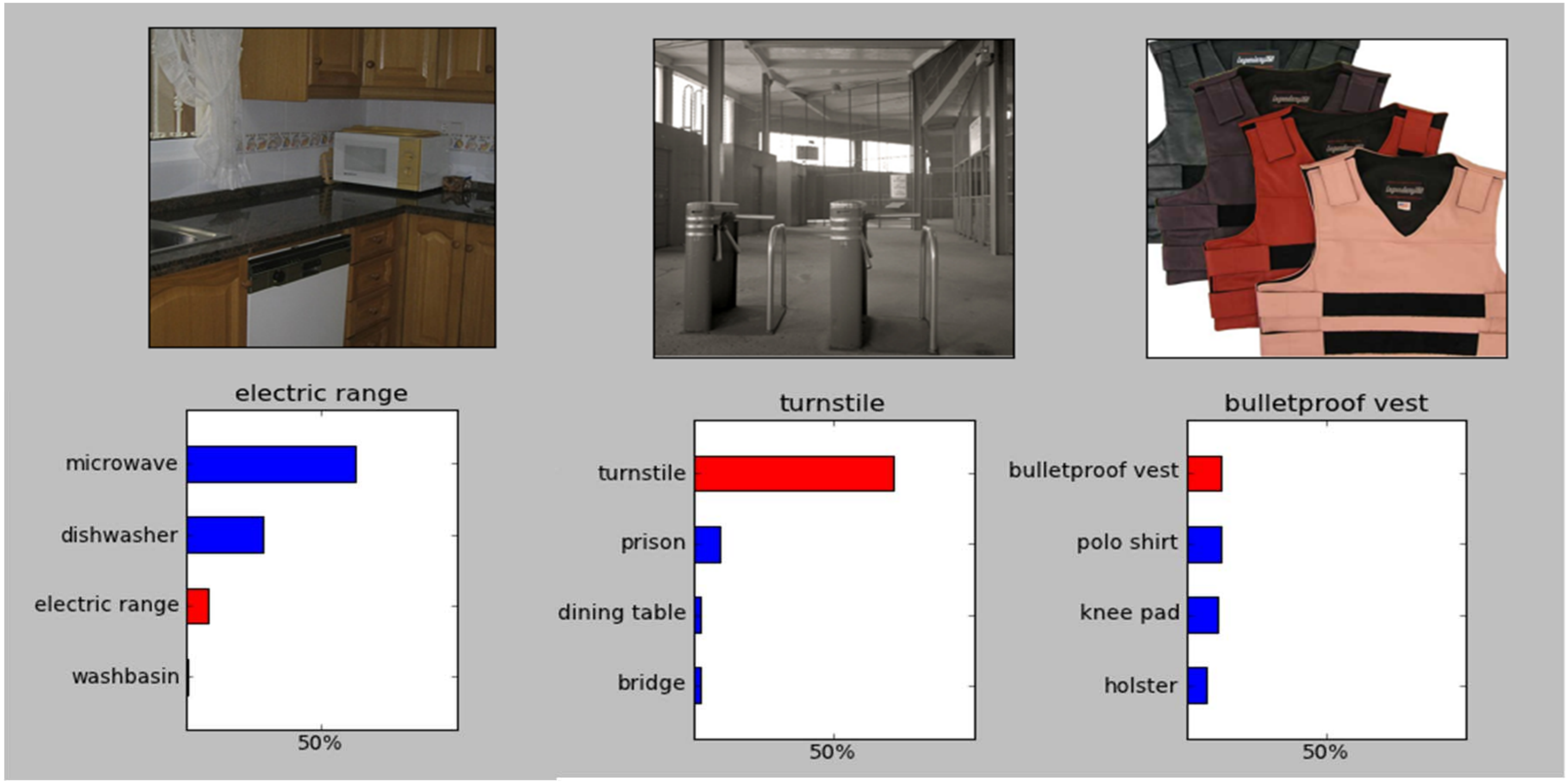
- It’s very important to choose a right learning rate σ (e.g. usually 0.01 or 0.001)

AlexNet

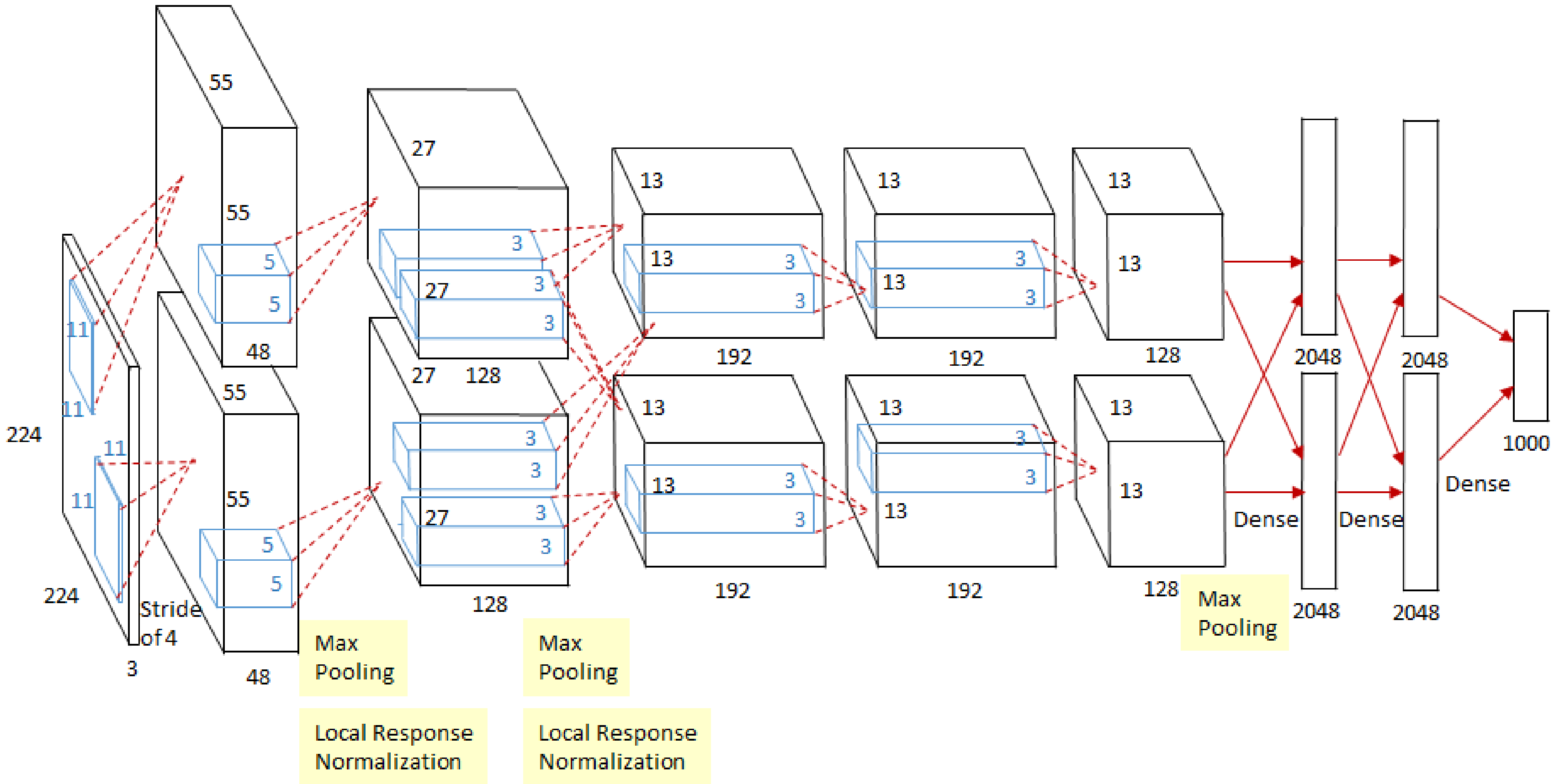
Some examples from an earlier version of the net



It can deal with a wide range of objects



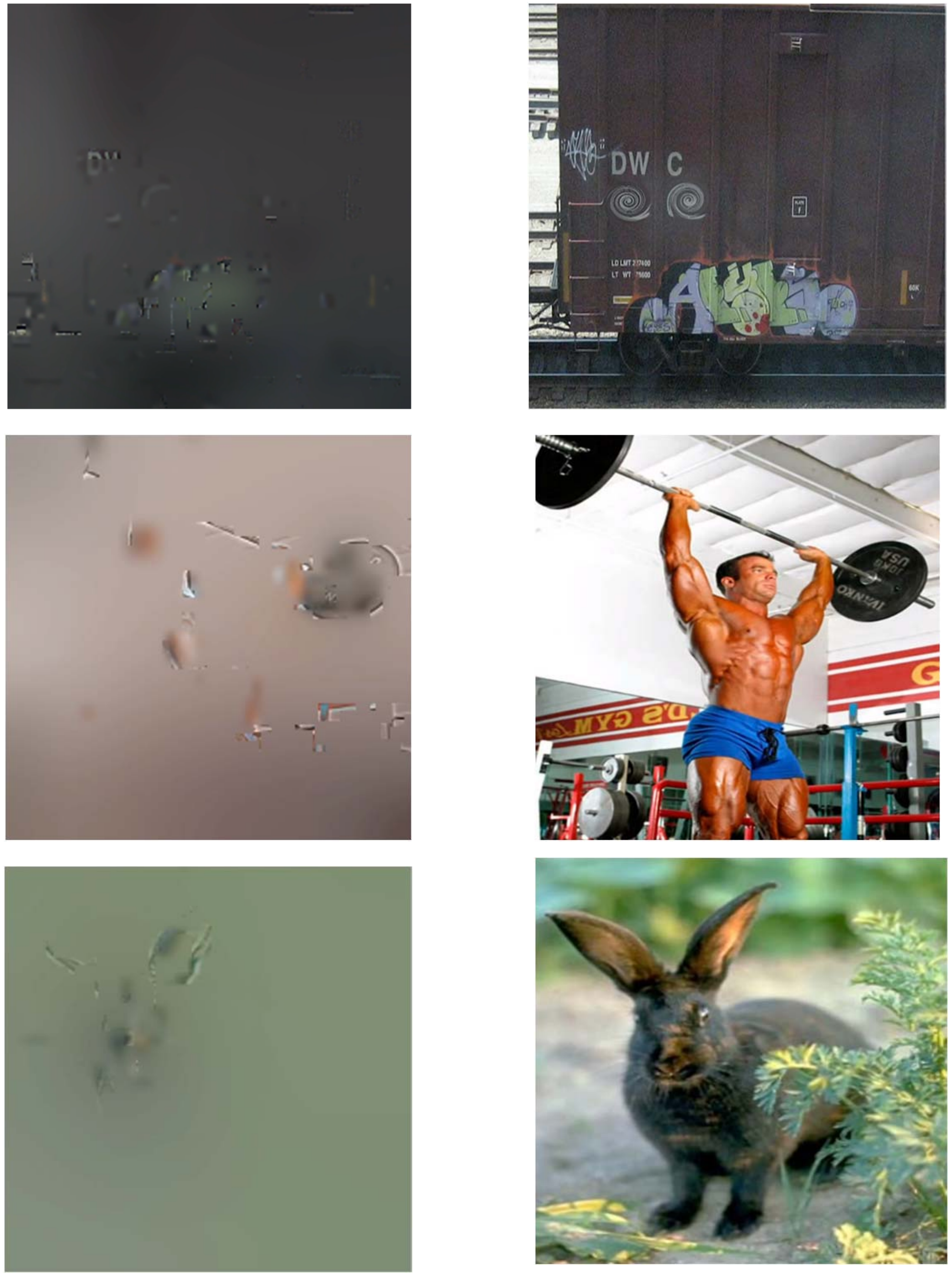
AlexNet



- 60 millions parameters
- 224 x 224 RGB input
- 1000 classes output

Problems

- Fitting a curve. Why it’s better than other methods?
- How we encode the problems parameters ? (e.g. weather prediction – the hard problem)
- Over fitting
- Oscillations


DW C

Related resources

- http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-034-artificial-intelligence-fall-2010/exams/MIT6_034F10_quiz2_2007.pdf

Readings

- Artificial Intelligence (3rd Edition), Patrick Winston, Chapter 12