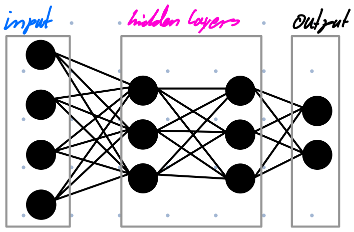
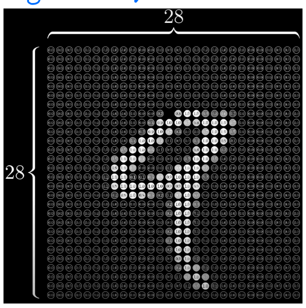
Neural Network = Neuron arranged in layers



input
image 28x28 pixel



Matrix

$$X \begin{bmatrix} \text{---} \; x^{(1)} \; \text{---} \\ \text{---} \; x^{(2)} \; \text{---} \\ \text{---} \; x^{(3)} \; \text{---} \\ \text{---} \; x^{(4)} \; \text{---} \end{bmatrix}$$

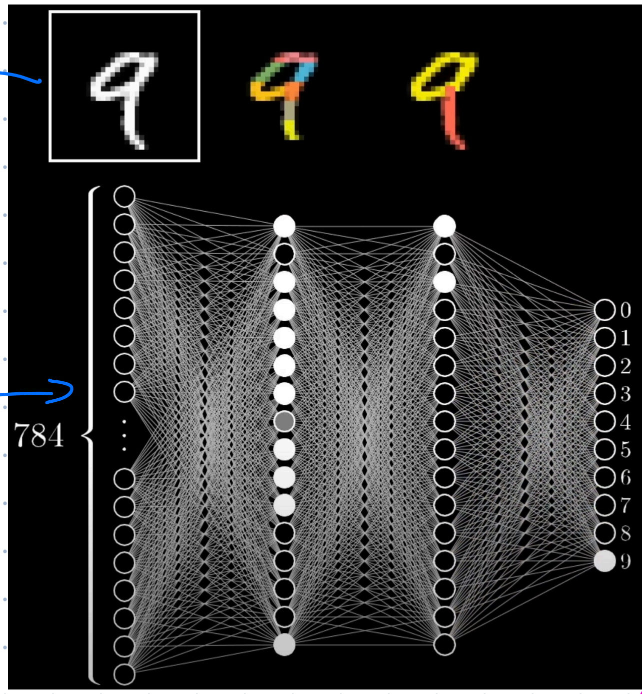
transpose matrix

$$X \begin{bmatrix} | & | & | & | \\ x^{(1)} & x^{(2)} & x^{(3)} & x^{(4)} \\ | & | & | & | \end{bmatrix}$$

hidden layers use the output from prior layer to recognize patterns

recognize writing

recognize components



← output

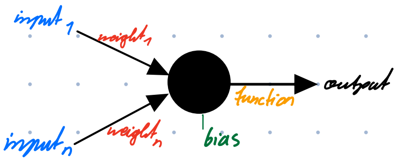
pixels → edges → patterns → digets

similar to human brain one neuron fireing will cause others to do so.

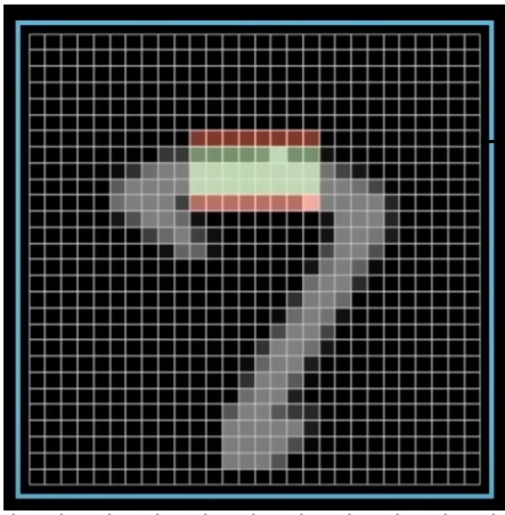
↳ this allows neurons to use prior detected patterns to detect new ones

Edge detection example

Single Neuron



$\longrightarrow$ Output = function($w_1 \cdot i_1 + \dots + w_n \cdot i_n + b$)

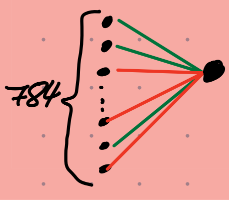


each pixel is one input

→ to recognize the line

↳ add positive weights to pixels where we expect the line

↳ add negative weights where we don't want the line



← this neuron can recognise a line in this specific area.

function

Sigmoid $\sigma(x) = \dfrac{1}{1+e^{-x}}$

↳ negative inputs → close to 0

↳ positive inputs → close to 1

Sigmoid keeps the neurons output between 0–1

bias

↳ neuron only fires if weighted sum exceeds threshold

$$\text{neuron output} = \underbrace{\sigma\Big(\underbrace{w_1\, i_1 + w_2\, i_1 + \dots + w_n\, i_n}_{\text{weighted sum}} - \underbrace{\text{bias}}_{\text{threshold}}\Big)}_{\text{sigmoid function sets output boundries 0-1}}$$

Notation

$$\sigma\left(\overset{\text{weight Matrix}}{\begin{bmatrix} w_{0,0} & w_{0,1} & \cdots & w_{0,n} \\ w_{1,0} & w_{1,1} & \cdots & w_{1,n} \\ \vdots & \vdots & & \vdots \\ w_{k,0} & w_{k,1} & \cdots & w_{k,n} \end{bmatrix}} \overset{\text{input vector}}{\begin{bmatrix} i_0 \\ i_1 \\ \vdots \\ i_n \end{bmatrix}} + \overset{\text{bias vector}}{\begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_k \end{bmatrix}} \right) = \text{neuron output} = \sigma(W\, i + b)$$

Learning → finding the right weights & biases