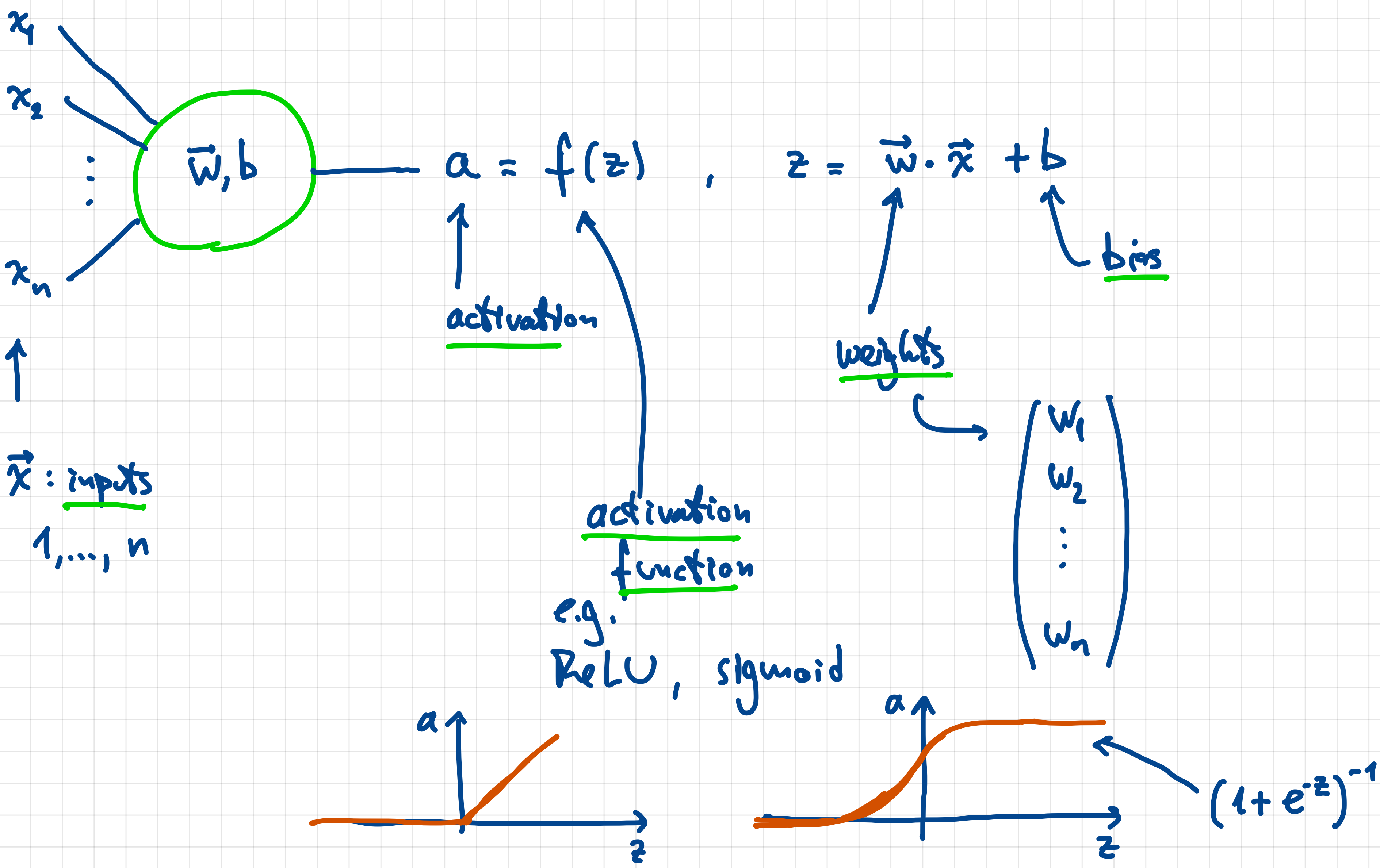
Short Takes 331

Machine learning and linear algebra

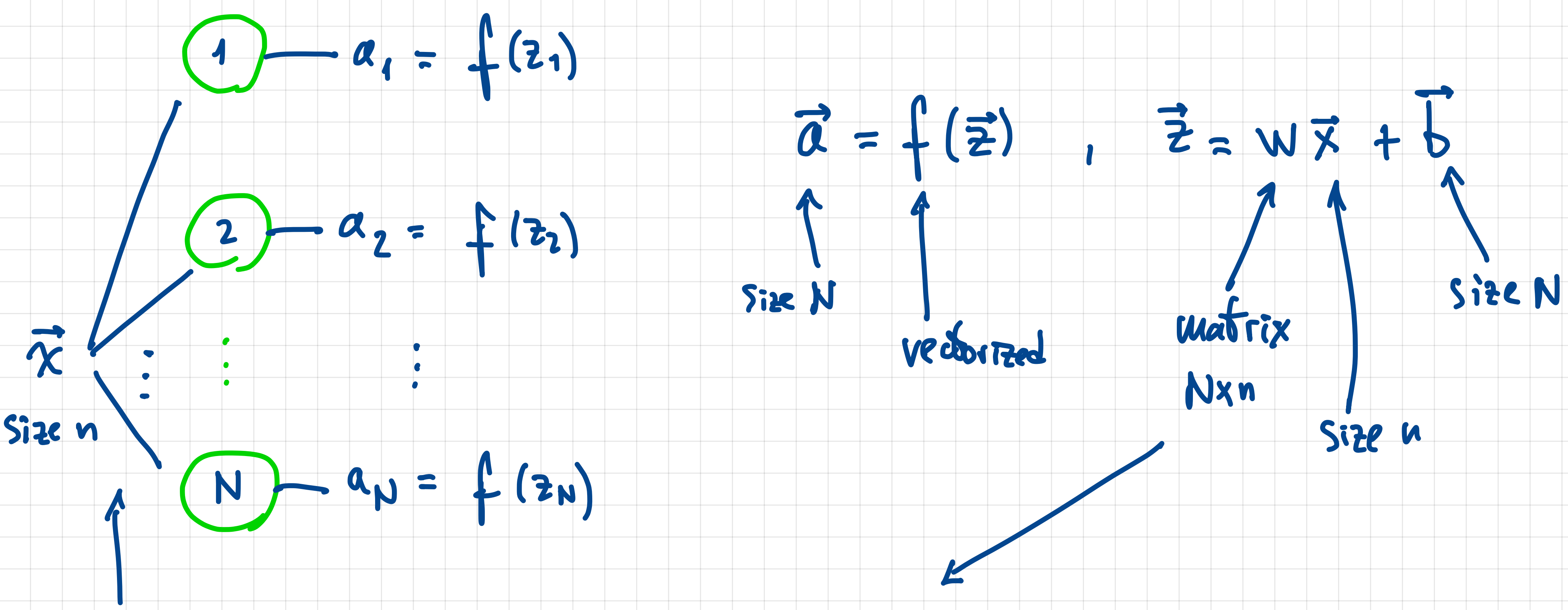
Machine learning & linear algebra

At its basic level, ML is about training a computer program to identify & predict patterns.

This is accomplished via artificial neural networks, which are collections of interlinked artificial neurons...



- If we have many neurons...

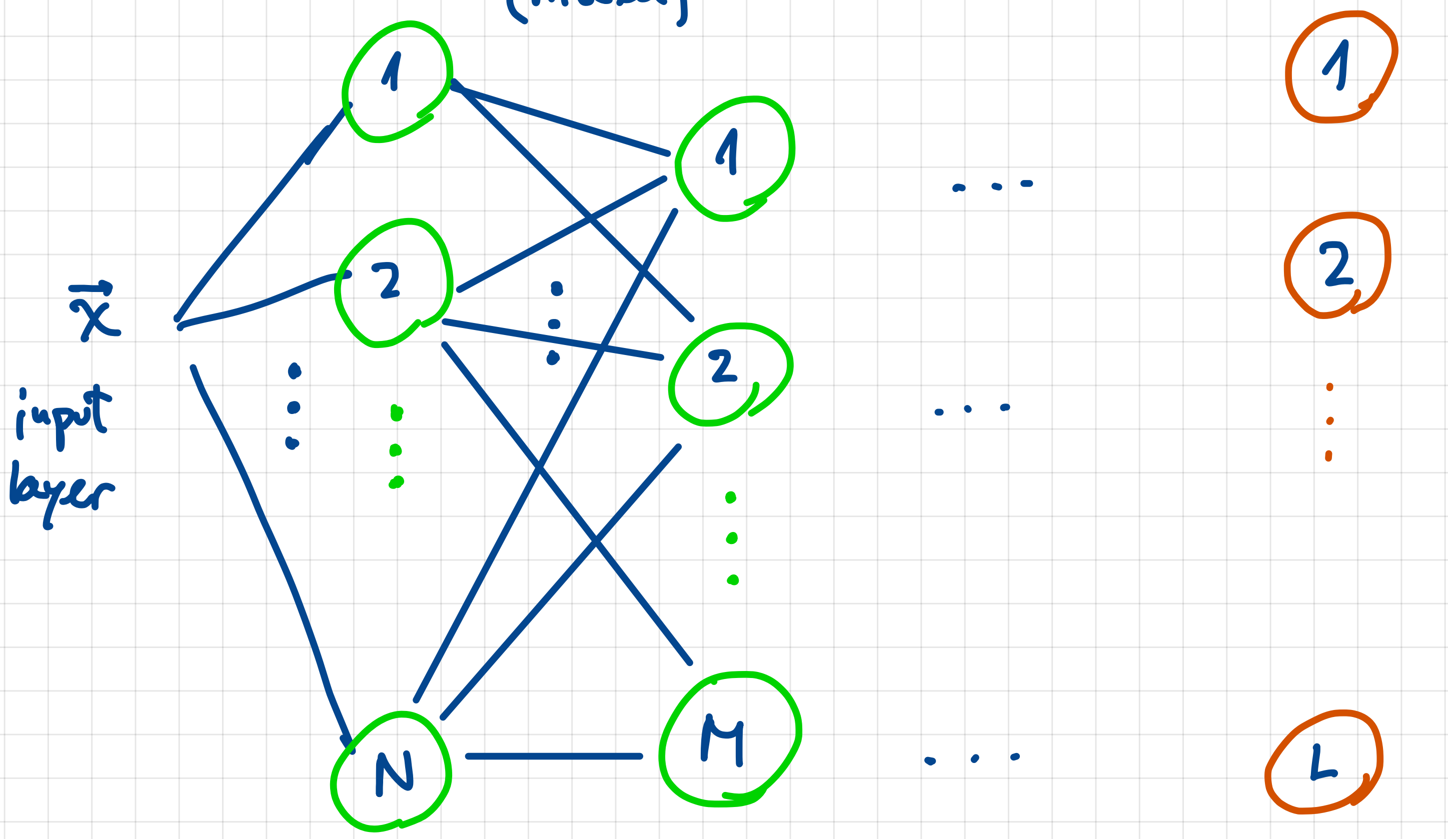


Note: same input $\vec{x}$ goes into all neurons.

$$W = \begin{pmatrix} w_{11} & w_{12} & \dots & w_{1n} \\ w_{21} & w_{22} & \dots & w_{2n} \\ \vdots & & & \\ w_{N1} & w_{N2} & \dots & w_{Nn} \end{pmatrix}$$

Use matrix-vector multiplication!

- In practice... many (hidden) layers!



Layer 1	Layer 2	...	output layer
$\vec{z}^{(1)} = W^{(1)}\vec{x} + \vec{b}^{(1)}$	$\vec{z}^{(2)} = W^{(2)}\vec{a}^{(1)} + \vec{b}^{(2)}$		$\vec{a}^{(L)}$
$\vec{a}^{(1)} = f(\vec{z}^{(1)})$	$\vec{a}^{(2)} = f(\vec{z}^{(2)})$		
$W^{(1)}$ is $N \times n$	$W^{(2)}$ is $M \times N$		
$\vec{x}$ is size n	$\vec{a}^{(2)}, \vec{z}^{(2)}$ is size M		
$\vec{a}^{(1)}, \vec{z}^{(1)}$ is size N			

$$\Rightarrow \quad \vec{a}^{(L)} = f(\dots \quad f(W^{(2)} f(W^{(1)}\vec{x} + \vec{b}^{(1)}) + \vec{b}^{(2)}) + \vec{b}^{(3)} \dots)$$

The weights & biases define the neural network (along with f).
They are fit to minimize a given cost function on training data.

The process of optimizing these parameters is where "learning" happens and usually involves stochastic gradient descent.
