# Artificial Intelligence

ACTL3143 & ACTL5111 Deep Learning for Actuaries

Patrick Laub

UNSW SYDNEY

# Lecture Outline

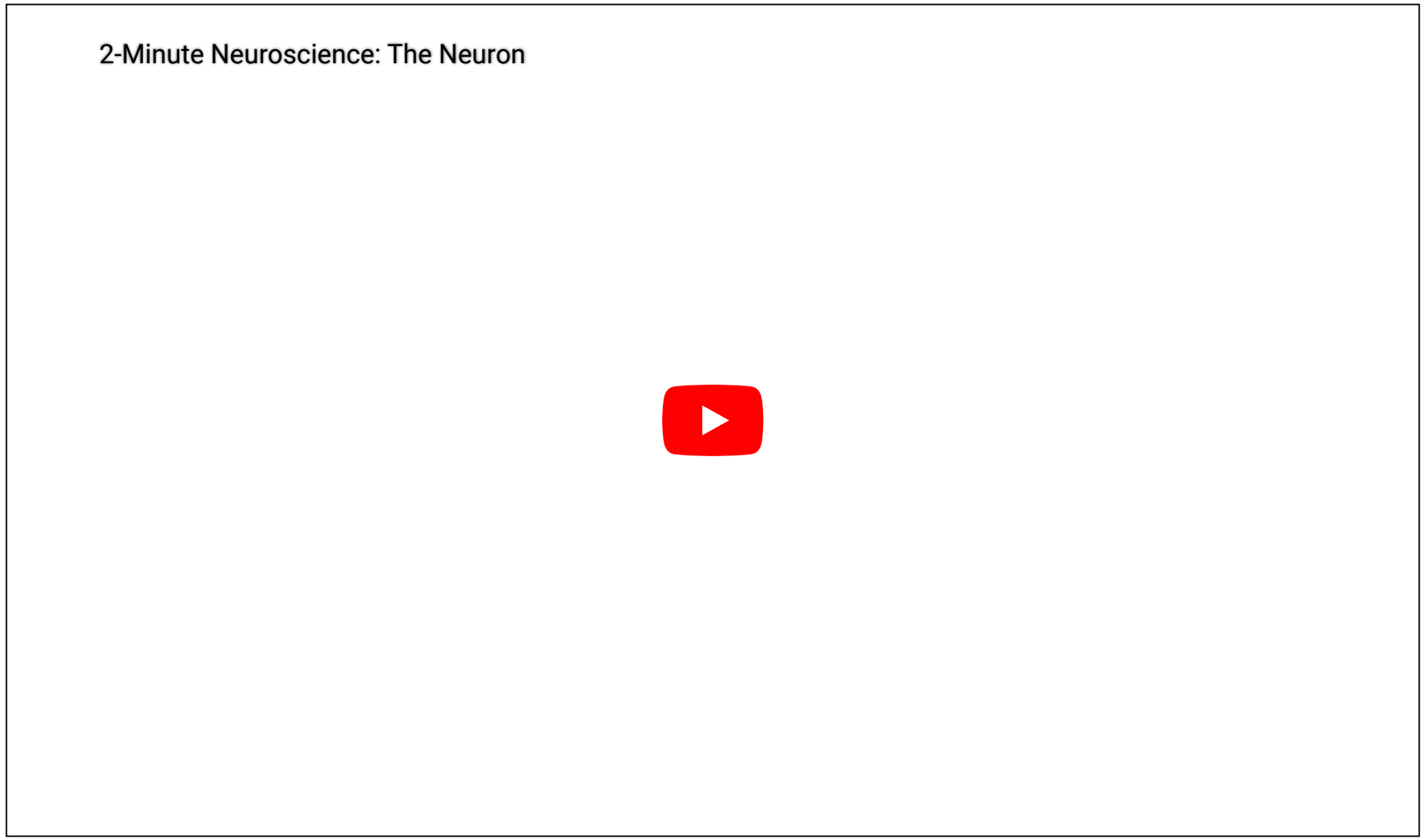
- **Artificial Intelligence**
- Neural Networks

# Lecture Outline

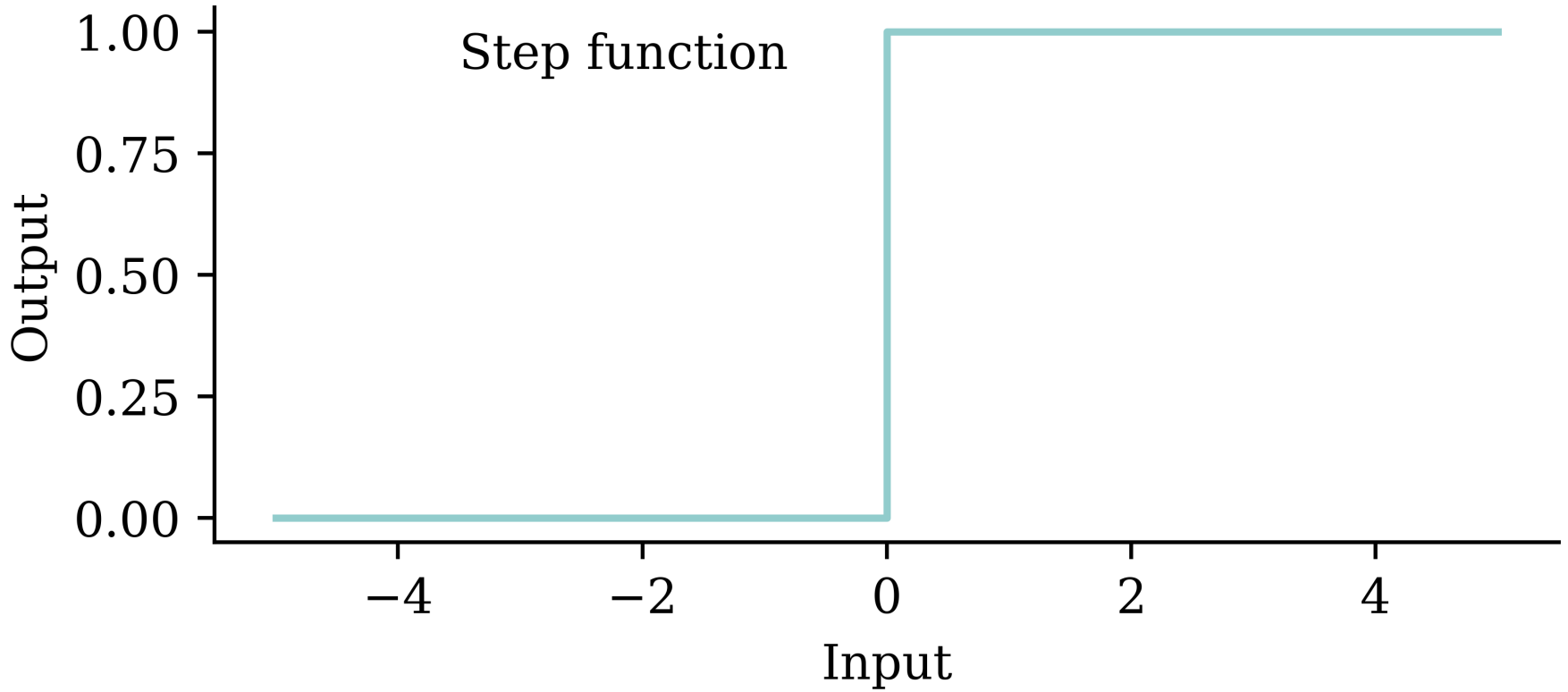
- Artificial Intelligence
- **Neural Networks**

UNSW SYDNEY

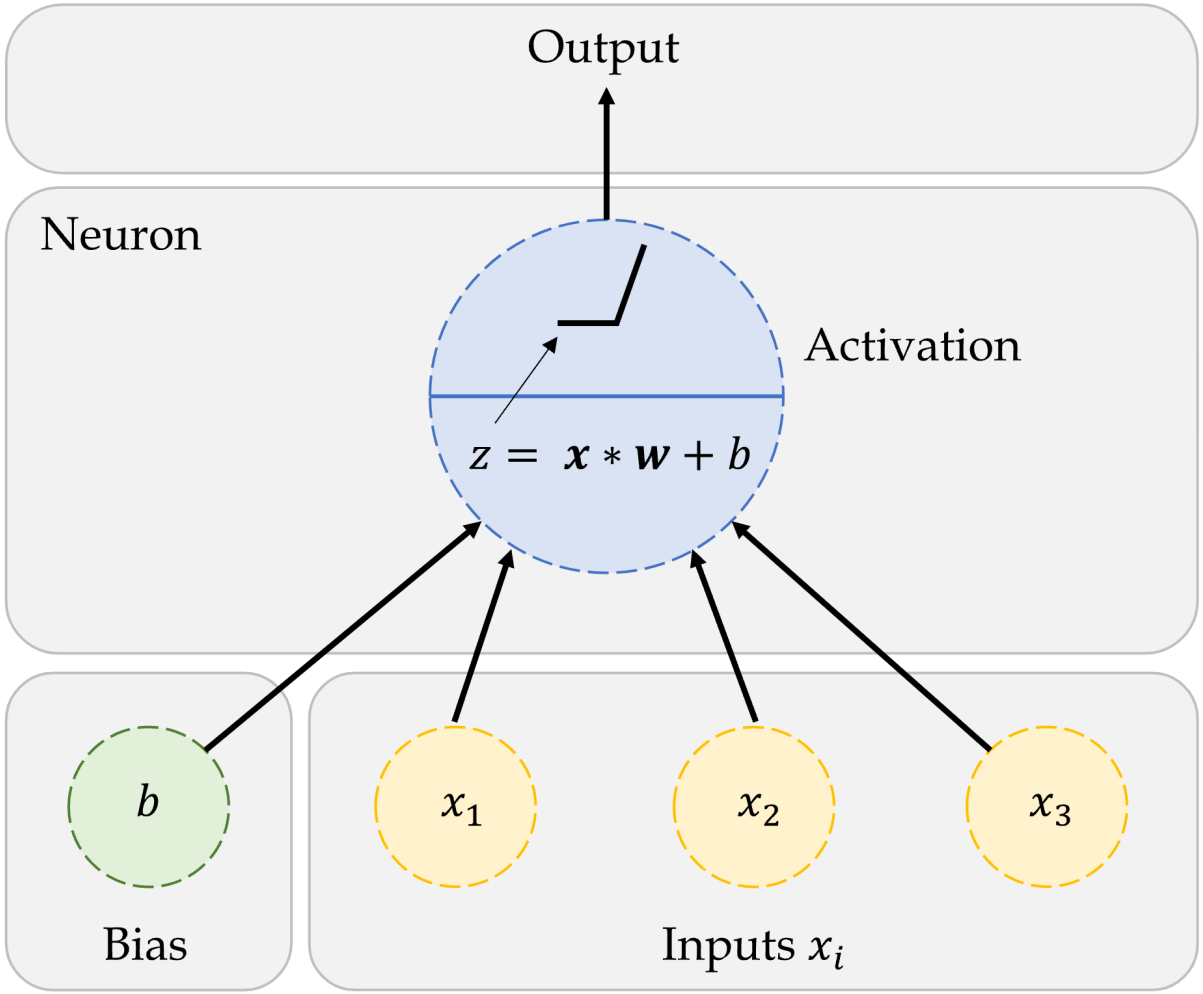
# How do real neurons work?



# A neuron ‘firing’



# An artificial neuron




A neuron in a neural network with a ReLU activation.

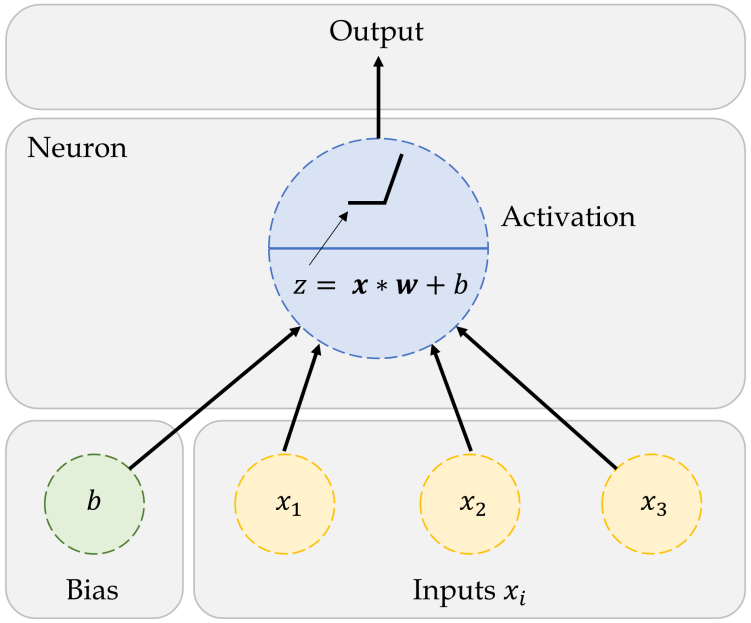
Source: Marcus Lautier (2022).

# One neuron

$$
\begin{aligned}
z &= x_1 \times w_1 + \\
&\quad x_2 \times w_2 + \\
&\quad x_3 \times w_3.
\end{aligned}
$$

$$
a = \begin{cases} z & \text{if } z > 0 \\ 0 & \text{if } z \leq 0 \end{cases}
$$

Here, $x_1, x_2, x_3$ is just some fixed data.




A neuron in a neural network with a ReLU activation.

The weights $w_1, w_2, w_3$ should be 'learned'.

Source: Marcus Lautier (2022).

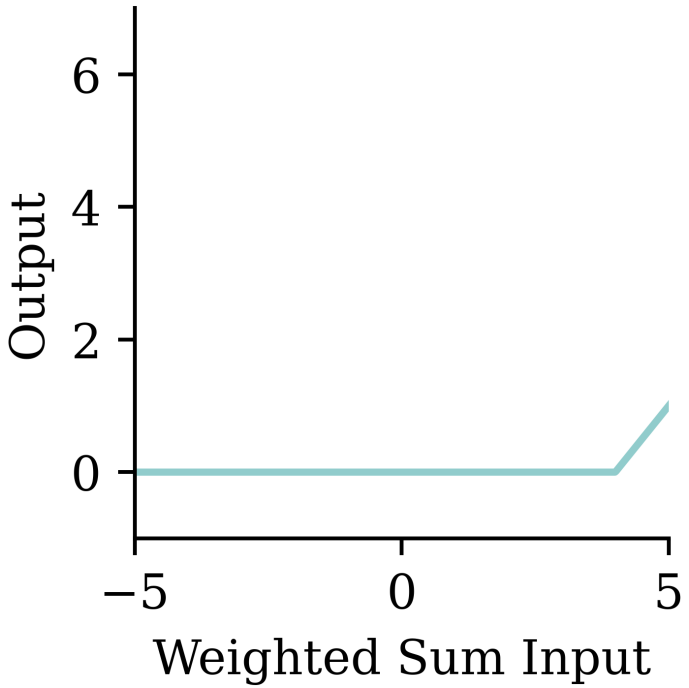
# One neuron with bias

$$
\begin{aligned}
z &= x_1 \times w_1 + \\
&\quad x_2 \times w_2 + \\
&\quad x_3 \times w_3 + b.
\end{aligned}
$$

$$
a = \begin{cases} z & \text{if } z > 0 \\ 0 & \text{if } z \leq 0 \end{cases}
$$

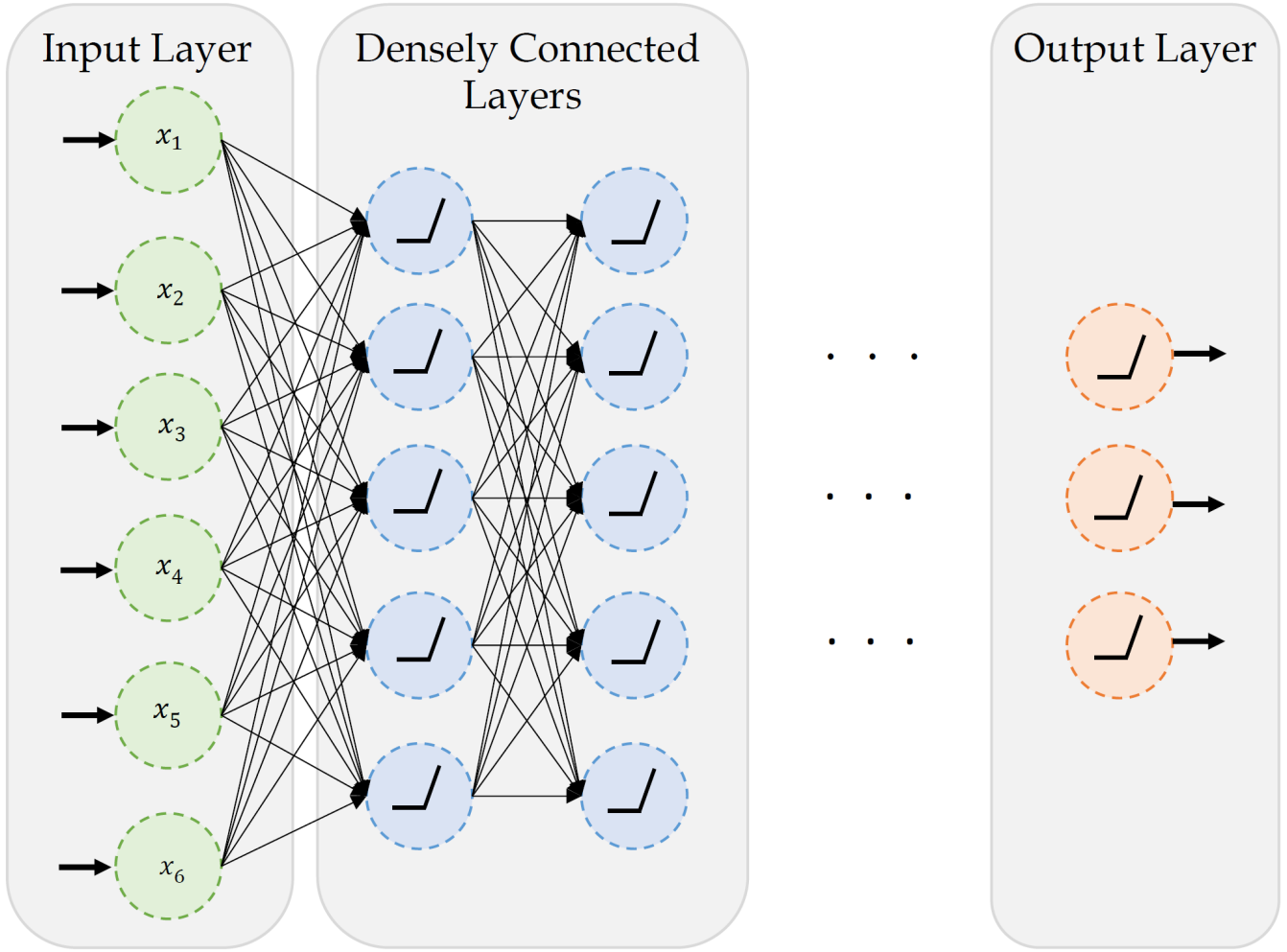
The weights $w_1$, $w_2$, $w_3$ and bias $b$ should be 'learned'.

Bias = -4 0 4




UNSW SYDNEY

# A basic neural network




A basic fully-connected/dense network.

Source: Marcus Lautier (2022).

# Step-function activation

## Perceptrons

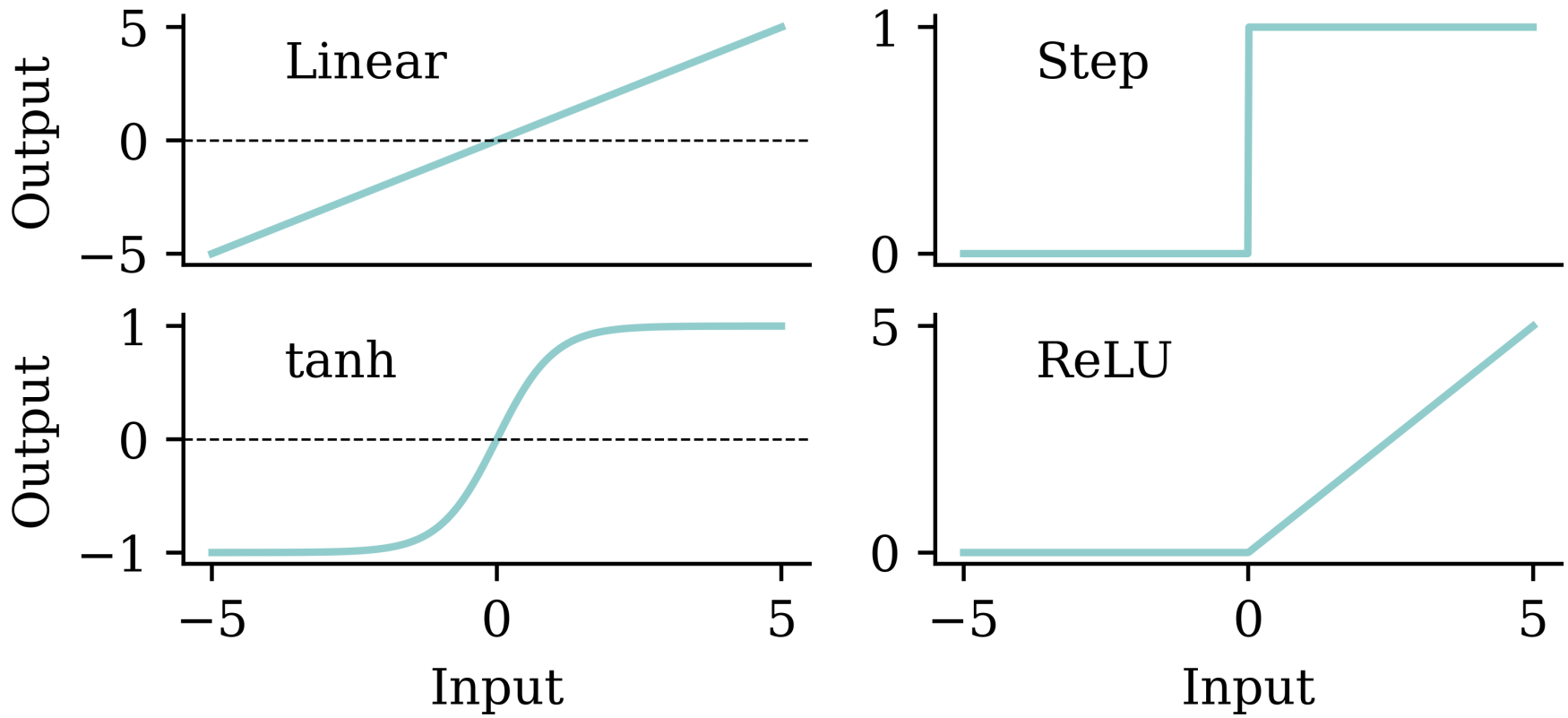
Brains and computers are binary, so make a perceptron with binary data. Seemed reasonable, impossible to train.

## Modern neural network

Replace binary state with continuous state. Still rather slow to train.

> **Note**
>
> It's a neur**al** network made of neur**on**s, not a "neuron network".

# Try different activation functions

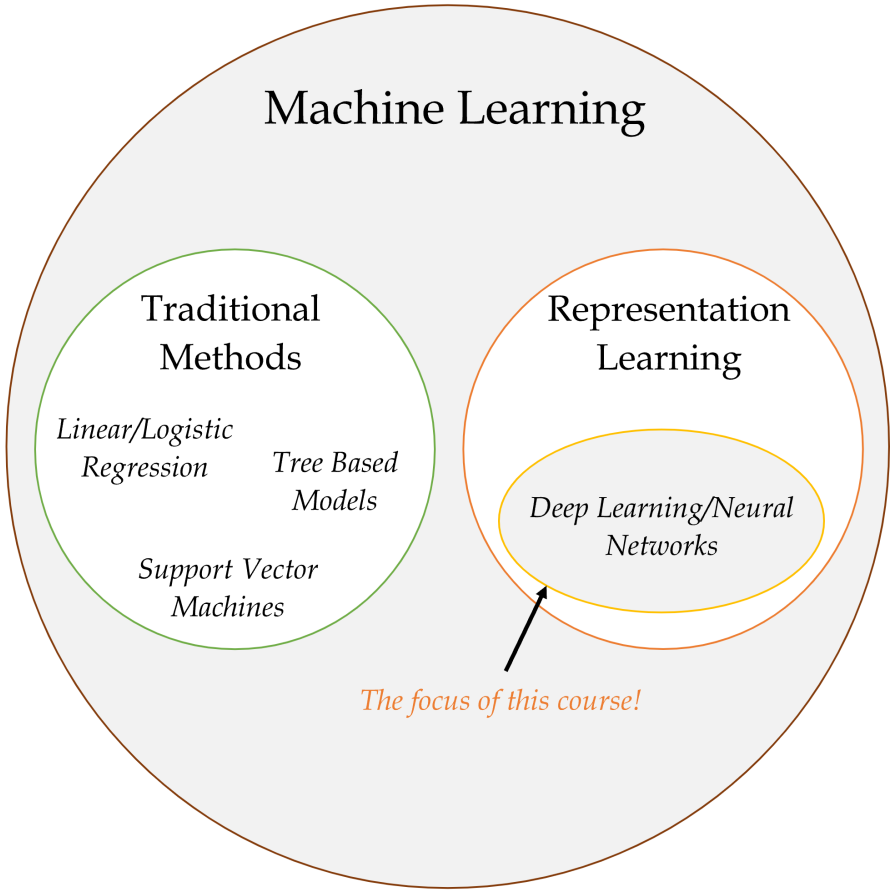


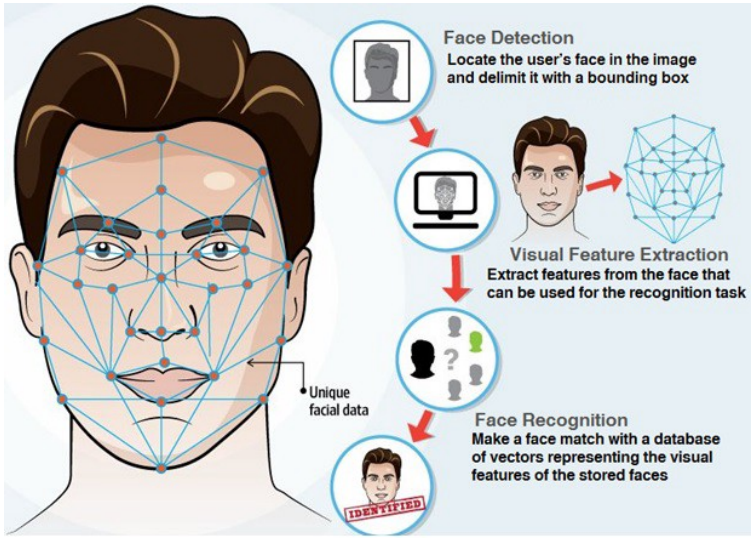
# Flexible

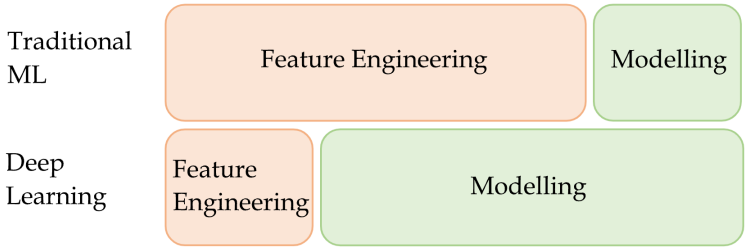
> One can show that an MLP is a **universal approximator**, meaning it can model any suitably smooth function, given enough hidden units, to any desired level of accuracy (Hornik 1991). One can either make the model be “wide” or “deep”; the latter has some advantages…

Source: Murphy (2012), Machine Learning: A Probabilistic Perspective, 1st Ed, p. 566.

# Feature engineering




Sources: Marcus Lautier (2022) & Fenjiro (2019), *Face Id: Deep Learning for Face Recognition*, Medium.

# The deep learning hammer

## Deep learning is not always the answer!


The map of data science.

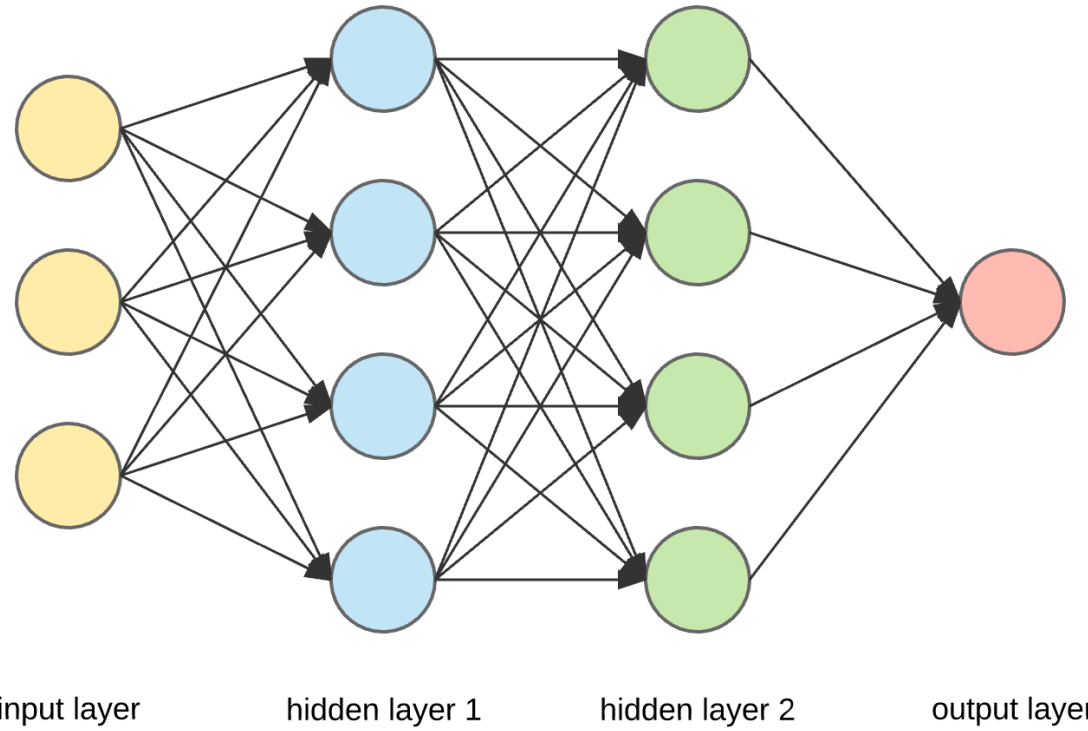
Source: Serge Masis (2022), LinkedIn post.

# Quiz

In this ANN, how many of the following are there:

- features,
- targets,
- weights,
- biases, and
- parameters?

What is the depth?




An artificial neural network.

Source: Dertat (2017), *Applied Deep Learning - Part 1: Artificial Neural Networks*, Medium.

# Package Versions

```
1 from watermark import watermark
2 print(watermark(python=True, packages="keras,matplotlib,numpy,pandas,seaborn,scipy,torch
```

```
Python implementation: CPython
Python version       : 3.11.9
IPython version      : 8.24.0

keras     : 3.3.3
matplotlib: 3.8.4
numpy     : 1.26.4
pandas    : 2.2.2
seaborn   : 0.13.2
scipy     : 1.11.0
torch     : 2.0.1
tensorflow: 2.16.1
tf_keras  : 2.16.0
```

UNSW SYDNEY

# Glossary

- activations, activation function
- artificial neural network
- biases (in neurons)
- classification problem
- deep network, network depth
- dense or fully-connected layer
- feed-forward neural network
- labelled/unlabelled data
- machine learning
- neural network architecture
- perceptron
- ReLU
- representation learning
- sigmoid activation function
- targets
- training/test split
- weights (in a neuron)