Edge Hill University

</TECHUP_WOMEN>

# Introduction to Neural Networks

Topic 5, Module 2

Duration: 1 Hour

Durham University

Edge Hill University

The University of Nottingham
UNITED KINGDOM · CHINA · MALAYSIA

UNIVERSITY of York

institute of CODING

Office for Students

# 1. What We'll Cover



This module will introduce…

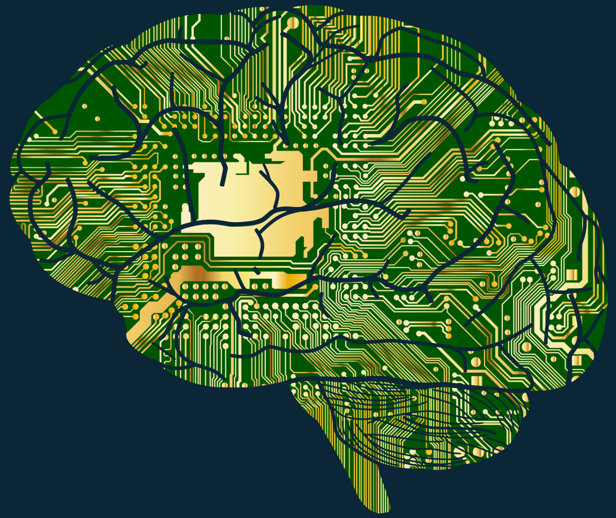
- Useful terminology.
- Key concepts.

- Some basic principles – how biology relates to artificial neural networks.
- How neural networks "learn".

- Examples of how artificial neural networks are applied in practice.

The aim: to help you understand what a neural network is, what it can do, and how it works at a fundamental level. Lets begin by learning about biological neural networks.

# 2. Context



**A. I.**

↓

**Machine Learning**

↓

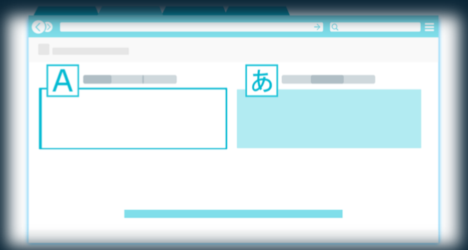
**Artificial Neural Networks**

- Artificial Intelligence (A.I.) is a field of study concerned with reproducing/replicating human intelligence.
- Machine Learning is a branch of A.I., concerned with replicating the human capacity to learn and make decisions.
- Artificial Neural Networks (A.N.N) or just Neural Networks, are a topic of study within machine leaning.

Edge Hill University

# 3. What are Neural Networks?

</TECHUP_WOMEN>



Google Translate

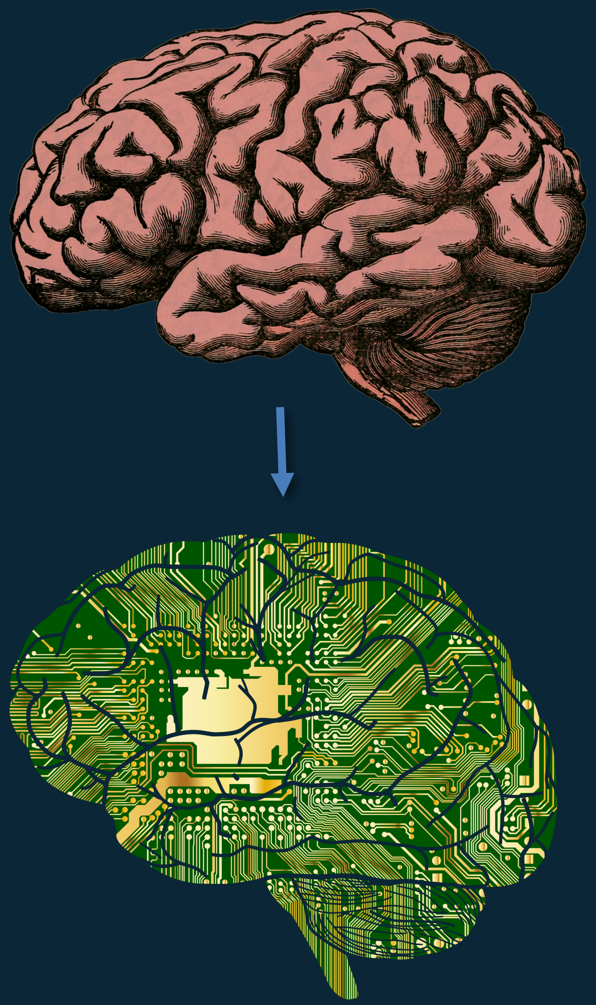
Tesla


Social Media

- Neural networks are sophisticated machine learning algorithms, capable of solving many real-world problems.
- They are able to classify (and make predictions over) complex datasets with high accuracy.
- Neural networks have become increasing popular in recent years.
- In fields such as data science and applied machine learning, they have become favoured. They are used for,
    1. Translation, as in Google Translate.
    2. Autonomous vehicles, e.g. Tesla
    3. Face recognition, as used by Facebook.

# 4. Why so popular?





- Neural networks are popular as they've been shown to work exceptionally well for many problems.
- They work so well due to how they attempt to mimic how our own brains work.
- The logic behind designing systems in this way is clear:

If human brains work so well on complex visual, auditory, and data dominated problems, then by emulating biological learning, we can create automated learning systems as effective as ourselves.

# 5. Human Brain

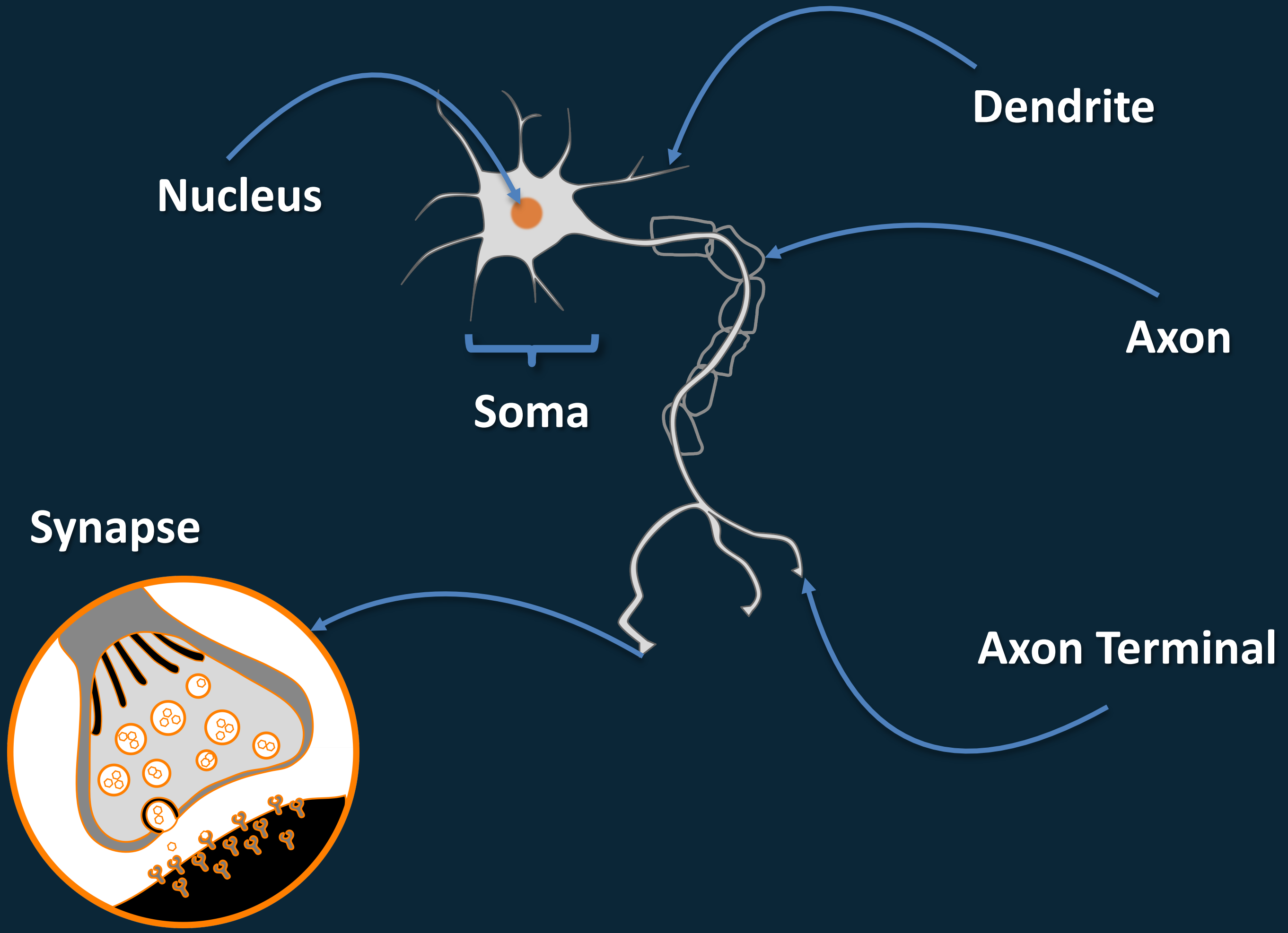
- The human brain is an immensely complex neural network.
- It contains approximately 100 billion individual neurons.
- There are 180 – 320 trillion synapses.
- The neurons work together to process the data arriving from the senses.
- Information is passed between the neurons via the synapses.
- Explaining what neurons and synapses are, may help us to understand why having lots of them, is a good thing (at least for learning).

Edge Hill University

# 6. Biological Neurons

</TECHUP_WOMEN>

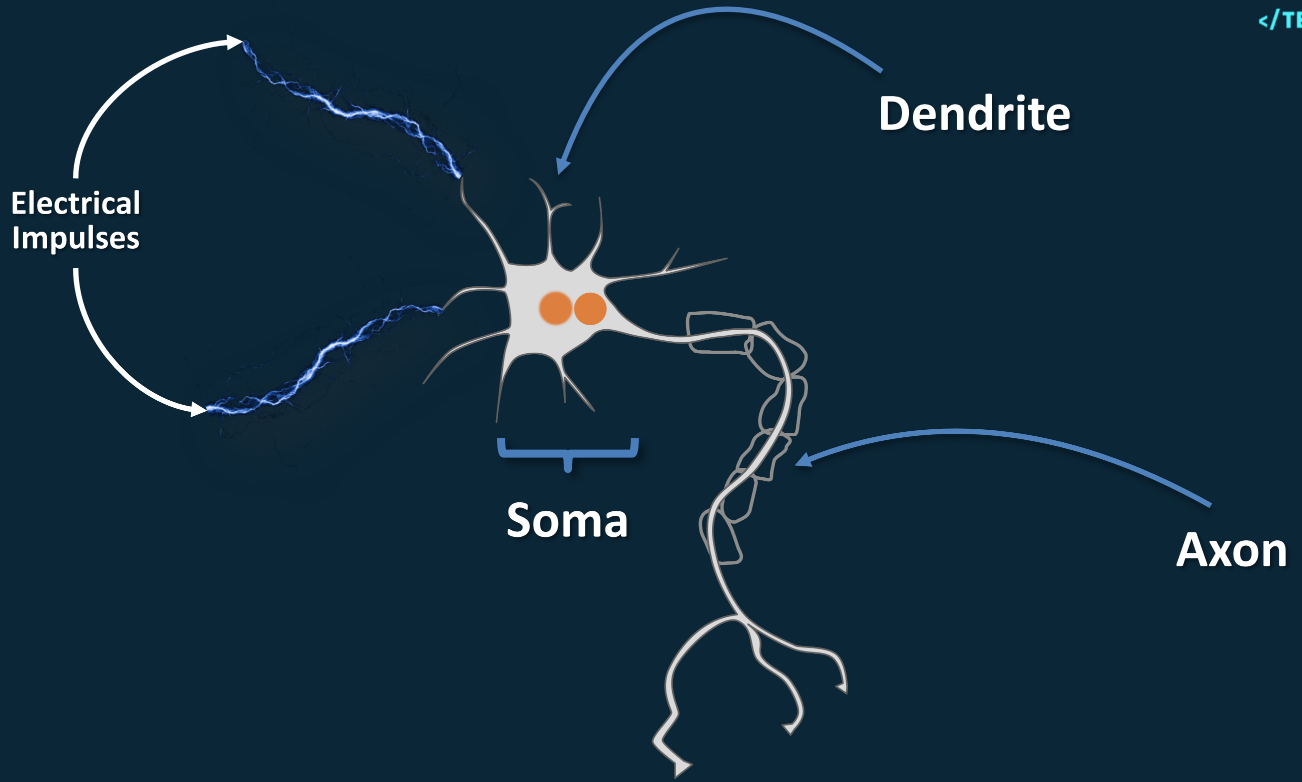
- At a basic level a neuron is just nerve cell.
- These cells are excitable. This means they can produce electrical or chemical signals when encountering external stimuli.
- Neurons have a number of components:
  1. Nucleus.
  2. Dendrites.
  3. Axon.
  4. Axon Terminal.
- The synapse is a structure between neurons, over which electrical impulses can pass.



# 7. How Neurons Work



- When we process sensory data, it initiates chemical reactions that generate electrical impulses.
- These impulses propagate through our brains, and are picked up by the dendrites.
- The dendrites pass the impulse to the Soma, sometimes just called the "cell body", that surrounds the nucleus.
- If the input electrical signal is strong enough, the cell body will begin propagating its own impulse.
- This impulse moves down the axon, to the axon terminals.

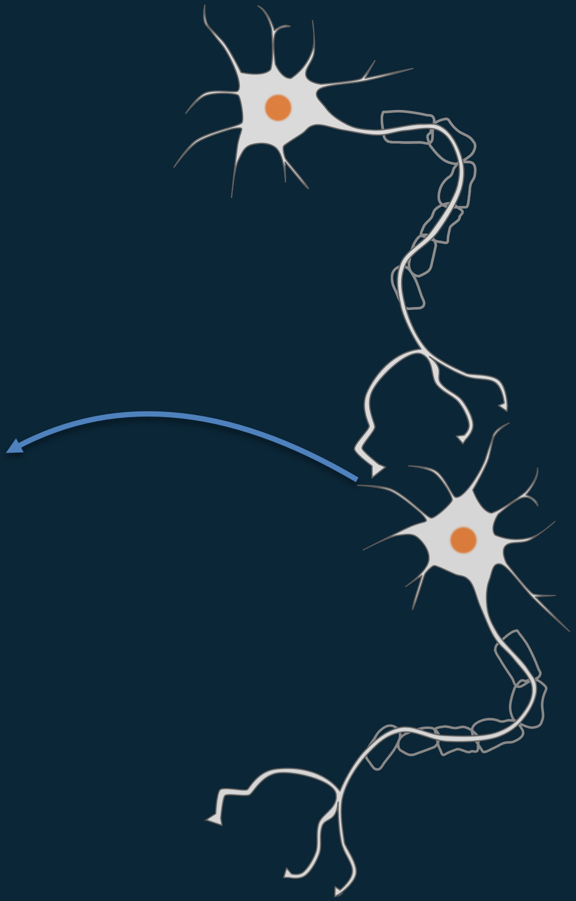


# 8. How Neurons Work



- If the electrical current is strong enough,  the impulse from the axon terminal is transmitted across the synapse, via conductive chemical transmitters.
- The impulses may make it to the dendrites of other nearby neurons, potentially exciting them.
- In this way, waves of impulses can move through our brains from neuron to neuron.
- Our brains interpret the electrical patterns created.
- This allows us to understand what our senses are telling us.

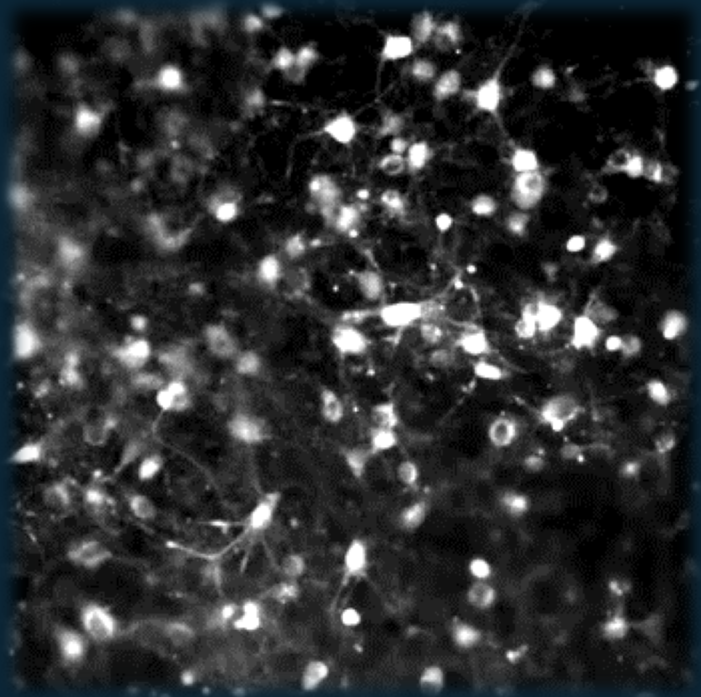
Synapse



# 9. Neurons in Action

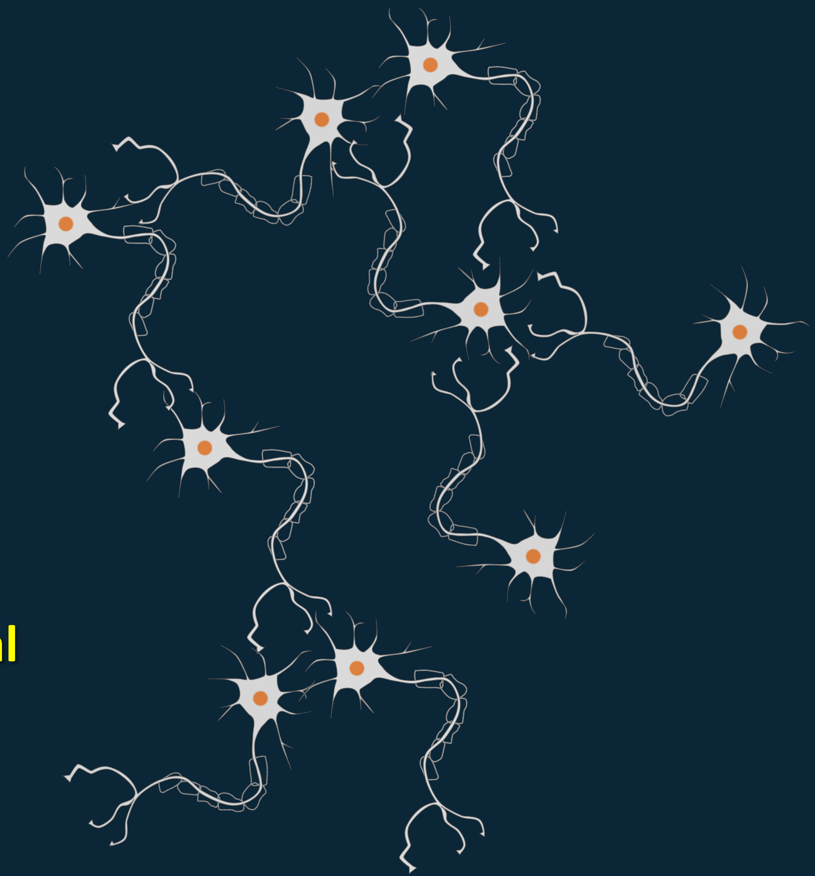
</TECHUP_WOMEN>

**Neurons "Firing"**



Credit: Georgia Tech. NeuroLab

- **We can see how this looks in practice.**
- **The diagram on the right shows how lots of neurons packed together might look.**
- **The video to the left shows how real neurons propagate electrical signals.**

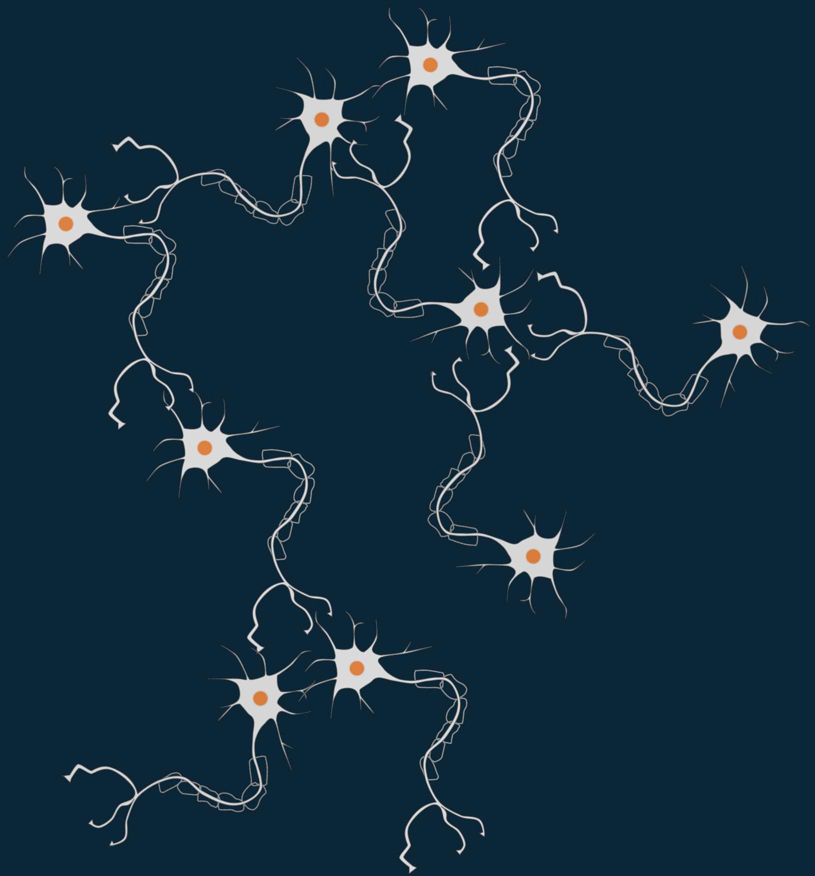


Edge Hill University

# 10. Neurons and Learning

</TECHUP_WOMEN>

- **Interestingly when neurons regularly fire in the brain, chemicals are produced that help increase the strength of their impulses.**
- **When neurons don’t fire, this chemical dissipates.**
- **This causes the impulse strength of neurons that don’t fire, to decrease over time.**
- **Why is this relevant?**



# 11. Neurons and Learning



- When undertaking a common task, the response of the neurons involved is strengthened. This makes the task easier to complete in future. For instance, when learning to play the piano, practice makes perfect, as they say.

- When we avoid a task, the response of the neurons used to complete it weaken. Hence, it can sometimes be difficult getting back up to speed after a long break.

- These points will become more important as we move on.

# 12. Why explain the biology?

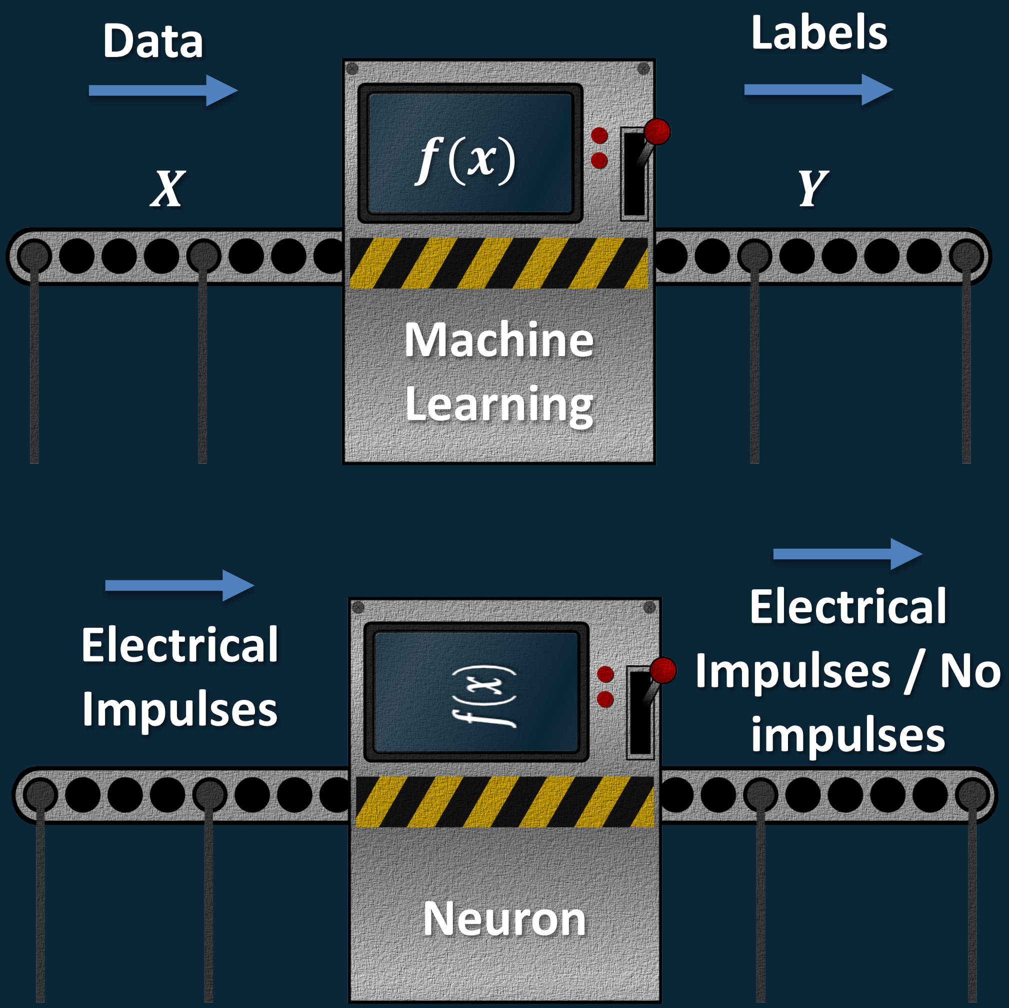


- In the last few slides, we briefly explored how the brain works.
- We discovered that biological neurons receive and send electrical impulses through the brain.
- How neurons firing together, create waves of electrical impulses that we interpret to understand the world around us.
- We spent time gaining this knowledge for an important reason. It allows us to understand neurons in a new way relevant to machine learning – as functions.

# 13. Revisiting Functions

</TECHUP_WOMEN>

- In module 5.1, part 2, we introduced functions.
- Functions are input-output boxes.
- In Module 5.1, we described machine learning algorithms as functions that,
  - Accept input data described via features.
  - Output labels, useful for prediction.
- It may seem strange, but we can think of neurons in a similar way.
- They,
  - Accept electrical impulses as input.
  - Produce electrical impulses as output.

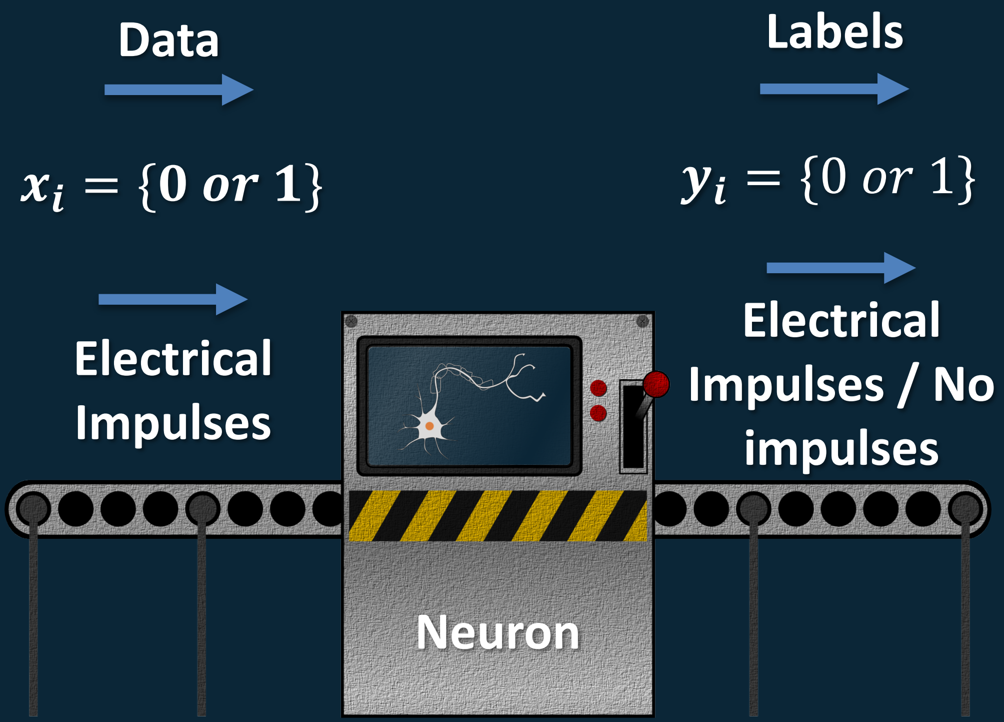



**Functions tutorial:**
https://www.youtube.com/watch?v=52tpYl2tTqk

# 14. Neuron as a Function



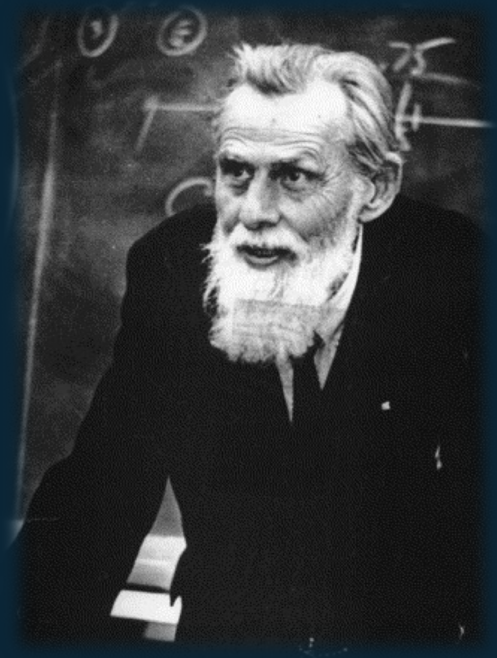
- Suppose we expand things, by describing electrical impulses in a simple way.
- If there is an electrical impulse passed to a neuron as input, we say the input to the neuron is 1.
- If there's no electrical impulse, we say the input is 0.
- Here $x_i$ describes the input to neuron $i$.
- We can do the same for the output. Output 1 if the neuron generates an electrical impulse, otherwise 0.
- Here $y_i$ describes the output of neuron $i$.
- Remember $i$ can represent any number, neuron 1, 2, or 3 e.g. $(i = 3)$ and so on.
- Now the electrical impulses represent data and labels.
- In this way, we can now think of a single neuron as a machine learning function.



# 15. McCulloch and Pitts

</TECHUP_WOMEN>

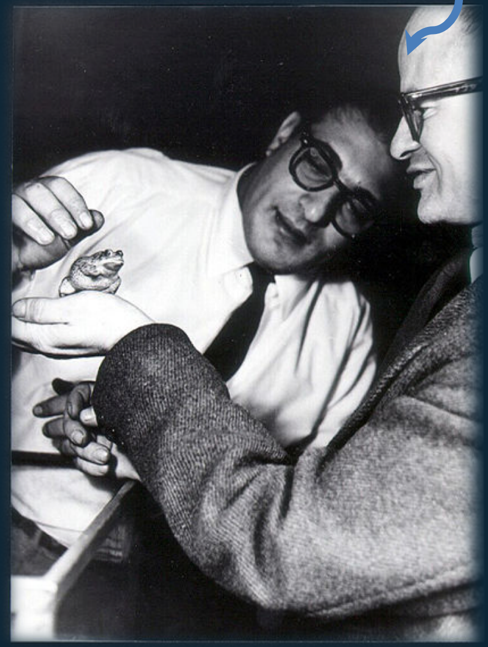
**Warren Strugis McCulloch**



Credit Diaz & Diaz, 2007: https://doi.org/10.1016/j.biosystems.2006.08.010

- In the last few slides, we've actually recreated an idea first suggested in 1943.
- Two scientists, Warren Strugis McCulloch, and Walter Pitts, realized it was possible to model neurons using simple functions.
- They also measured the time it takes for electrical impulses to move through the brain. It wasn't as fast as first thought.
- They realized something tantalizing - we could create an analogue of the biologic neuron, based on functions, that could be constructed.
- They described the first "artificial neuron".
- In principle, this could be recreated in technology – even in 1943.

**Walter Pitts**

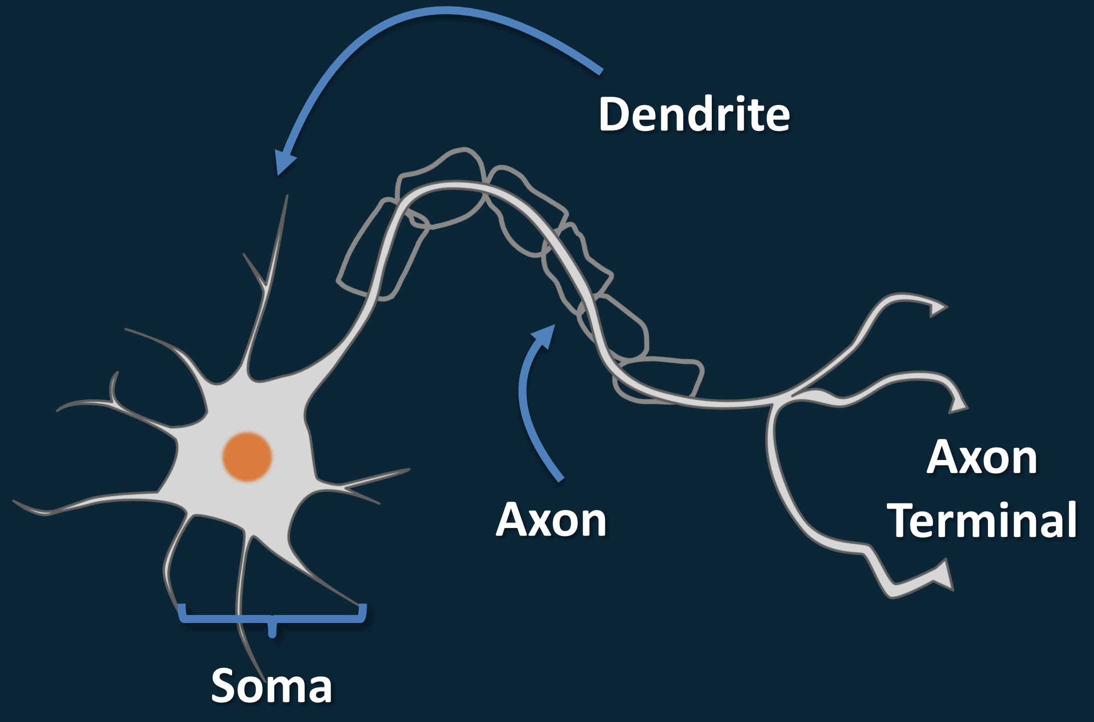


Levin & Pitts Family Album, distributed under a CC BY-SA 3.0.

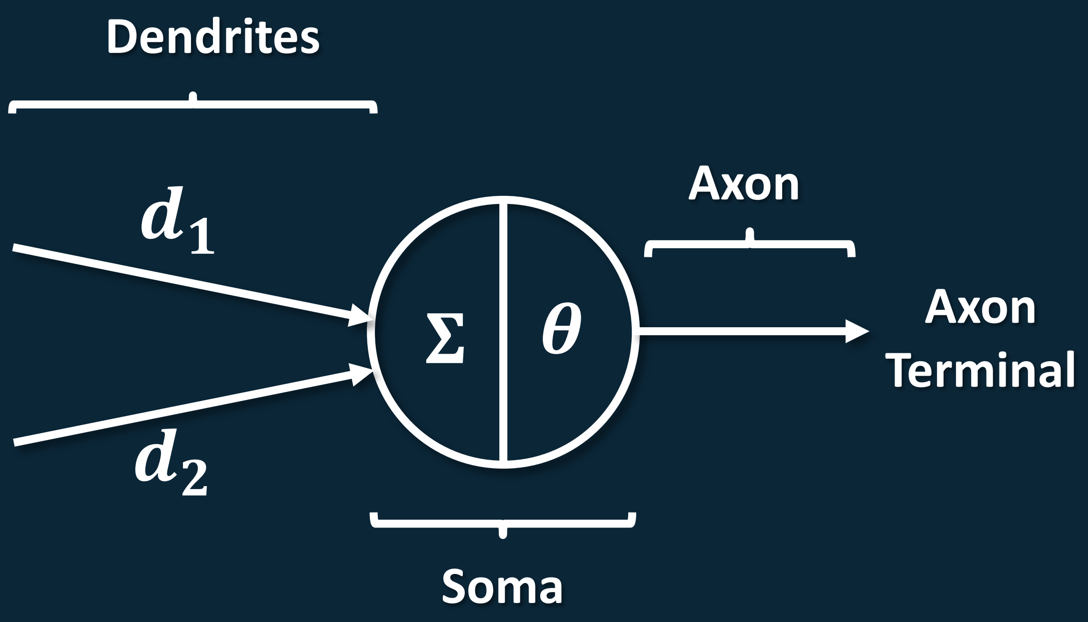
Durham University · Edge Hill University · The University of Nottingham UNITED KINGDOM · CHINA · MALAYSIA · University of York · institute of CODING · Office for Students OfS

# 16. Artificial vs. Biological Neuron



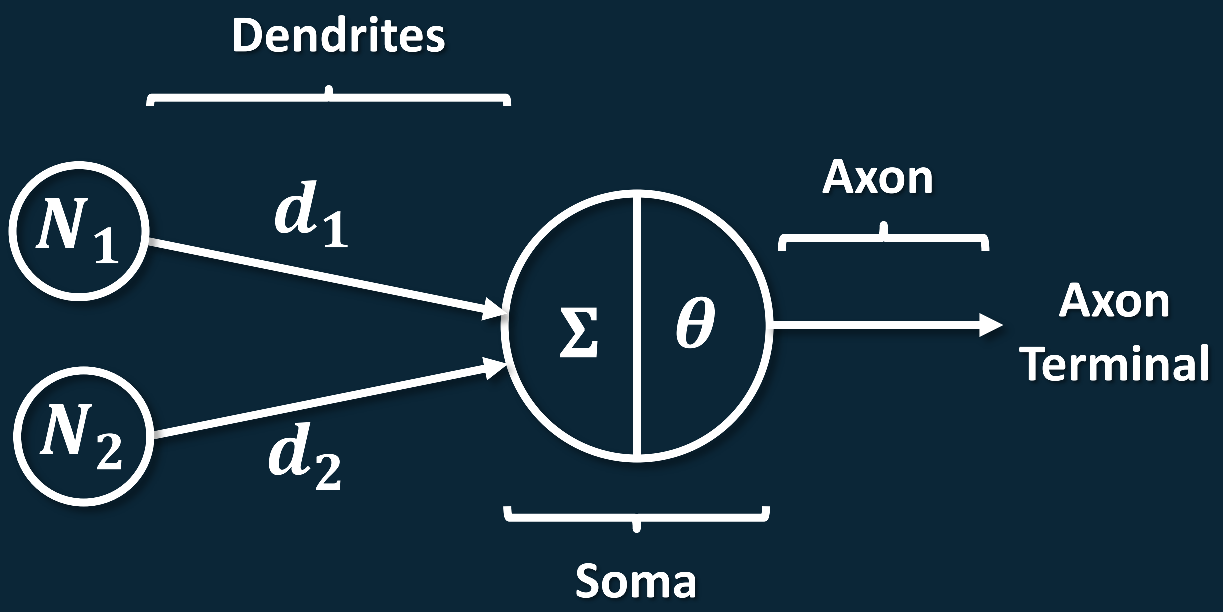



**Biological Neuron.**




**McCulloch-Pitts Neuron with two dendrites.**

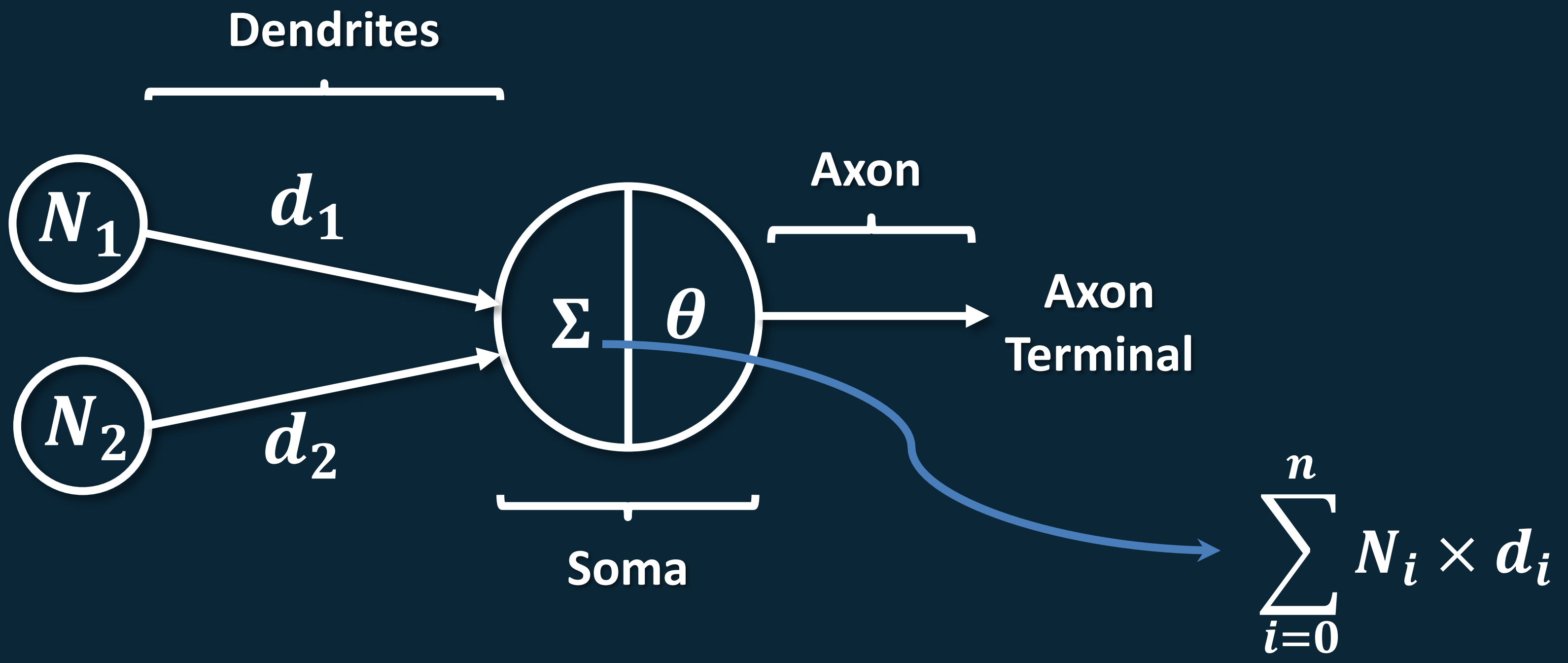
# 17. McCulloch and Pitts Neuron




- The artificial neuron receives two inputs from Neurons $N_1$ and $N_2$ .
- Remember our discussion of neurons getting stronger when used, and weaker when unused? Well $d_1$ and $d_2$ allow us to simulate this. Each dendrite is associated with a numerical weight e.g. $d_1 = 1$ or $d_2 = 5$.
- Connections can be strengthened by increasing the weights, or weakened by decreasing them.

# 18. McCulloch and Pitts Neuron

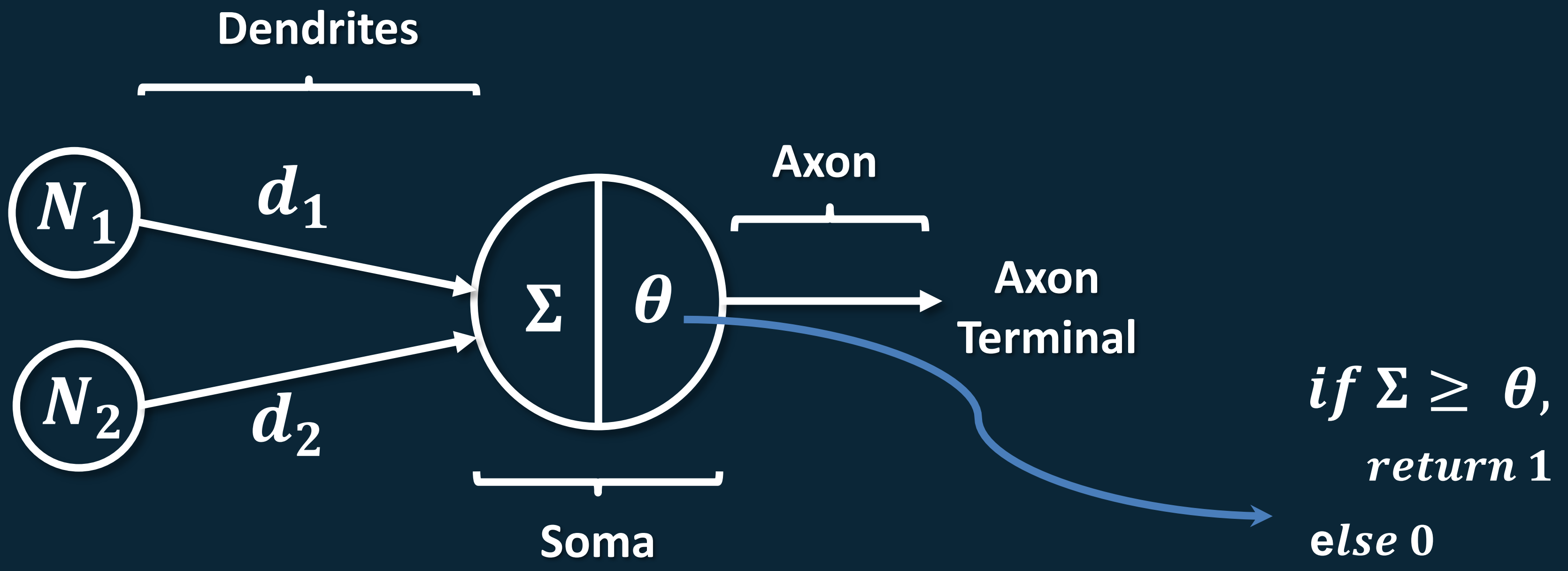





- The Sigma symbol ($\Sigma$), is actually very easy to understand. It is equal to the value of $N_1$ multiplied by $d_1$, plus the value of $N_2$ multiplied by $d_2$ . This is written above.
- We use the $i$ symbol in the formula, as perhaps we'll have more than two dendrites and weights in the future.

# 19. McCulloch and Pitts Neuron






- Finally the theta symbol, $\theta$, represents a function, or even a single value, with an important role.
- If the value of $\Sigma$ is greater than, or equal to the value of $\theta$, then the artificial neuron fires. Otherwise it does not.
- If the neuron "fires", it returns 1, otherwise it returns 0.
- Before proceeding watch this 1 minute video, to help solidify your knowledge.

# 20. Example



- So far what we've described appears complex – but seeing past the symbols, this is not the case.
- We can show this by training an artificial neuron for ourselves.
- We're going to teach a neuron the concept of "AND".
- In logic "AND" represents a simple condition. This is shown in the table below.

| Input $N_1$ | Input $N_2$ | Output $y$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Thus $y$ is only equal to 1, when both $N_1$ and $N_2$ are 1.

In logic 1 = TRUE and 0 = FALSE.

We'll build a neuron that outputs 1, only when $N_1 = 1$ and $N_2 = 1$.

# 21. Why?

</TECHUP_WOMEN>

- Why are we going to try and build a neuron that captures “AND”, for ourselves?
- Well, actually building an artificial neuron at a fundamental level, will help you understand how things scale up.

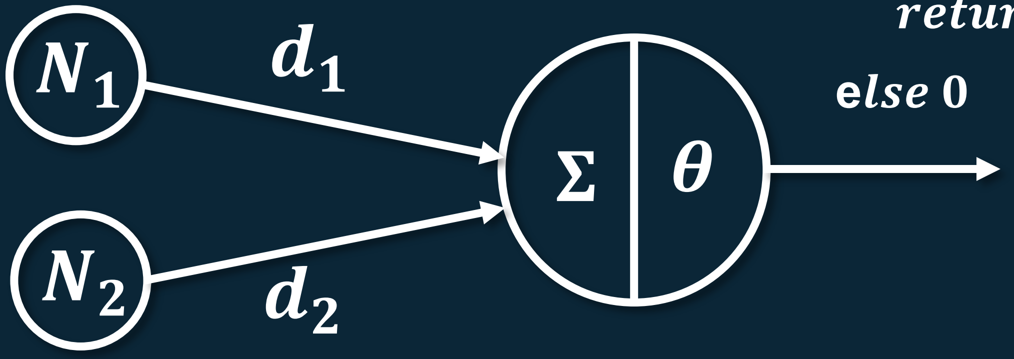
# 22. Example - Starting State

</TECHUP_WOMEN>

| $N_1$ | $N_2$ | $y$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Logical "AND" training data

$$\sum_{i=0}^{n} N_i \times d_i$$




$if\ \Sigma \geq \theta,$

$return\ 1$

$else\ 0$

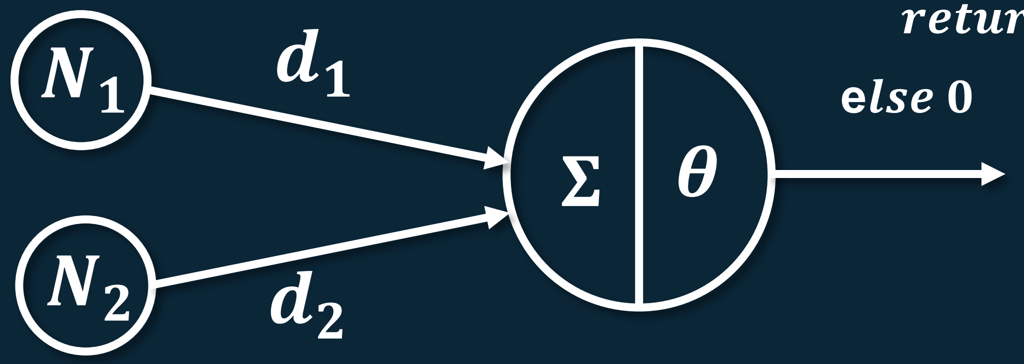
**FAILED TO LEARN "AND"!**

| | $N_1$ | $d_1$ | $N_2$ | $d_2$ | $\Sigma$ | $\theta$ | Output |
|---|---|---|---|---|---|---|---|
| Input 1 | 0 | 0 | 0 | 0 | (0 x 0) + (0 x 0) = **0** | 1 | 0 |
| Input 2 | 0 | 0 | 1 | 0 | (0 x 0) + (1 x 0) = **0** | 1 | 0 |
| Input 3 | 1 | 0 | 0 | 0 | (1 x 0) + (0 x 0) = **0** | 1 | 0 |
| Input 4 | 1 | 0 | 1 | 0 | (1 x 0) + (1 x 0) = **0** | 1 | 0 |

# 23. Example - Updating Weights for $d_1$

| $N_1$ | $N_2$ | $y$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Logical "AND" training data




$if\ \Sigma \geq \theta,$
$return\ 1$
$else\ 0$

**FAILED TO LEARN "AND"!**

| | $N_1$ | $d_1$ | $N_2$ | $d_2$ | $\Sigma$ | $\theta$ | Output |
|---|---|---|---|---|---|---|---|
| Input 1 | 0 | **1** | 0 | 0 | (0 x 1) + (0 x 0) = **0** | 1 | 0 |
| Input 2 | 0 | **1** | 1 | 0 | (0 x 1) + (1 x 0) = **0** | 1 | 0 |
| Input 3 | 1 | **1** | 0 | 0 | (1 x 1) + (0 x 0) = **1** | 1 | 1 |
| Input 4 | 1 | **1** | 1 | 0 | (1 x 1) + (1 x 0) = **1** | 1 | 1 |

# 24. Example - Updating Weights for $d_2$



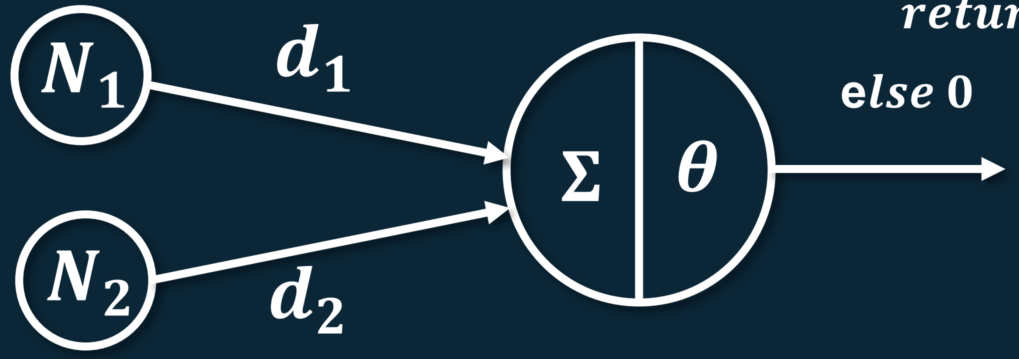
| $N_1$ | $N_2$ | $y$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Logical "AND" training data

$if\ \Sigma \geq \theta,$

$return\ 1$

$else\ 0$




**FAILED TO LEARN "AND"!**

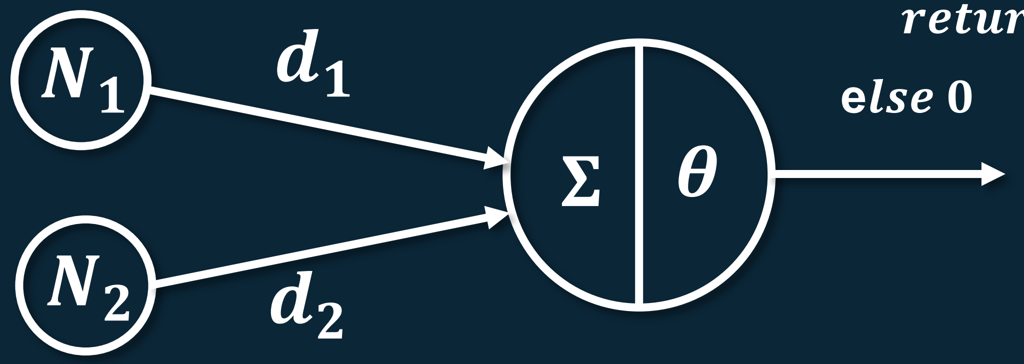
| | $N_1$ | $d_1$ | $N_2$ | $d_2$ | $\Sigma$ | $\theta$ | Output |
|---|---|---|---|---|---|---|---|
| Input 1 | 0 | 1 | 0 | **1** | (0 x 1) + (0 x 1) = **0** | 1 | 0 |
| Input 2 | 0 | 1 | 1 | **1** | (0 x 1) + (1 x 1) = **1** | 1 | 1 |
| Input 3 | 1 | 1 | 0 | **1** | (1 x 1) + (0 x 1) = **1** | 1 | 1 |
| Input 4 | 1 | 1 | 1 | **1** | (1 x 1) + (1 x 1) = **2** | 1 | 1 |

# 25. Example - Updating Theta ($\theta$)



| $N_1$ | $N_2$ | $y$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

**Logical "AND" training data**




$if\ \Sigma \geq \theta,$

$return\ 1$

$else\ 0$

**Finally Learned "AND"!**

| | $N_1$ | $d_1$ | $N_2$ | $d_2$ | $\Sigma$ | $\theta$ | Output |
|---|---|---|---|---|---|---|---|
| Input 1 | 0 | 1 | 0 | 1 | (0 x 1) + (0 x 1) = **0** | **2** | 0 |
| Input 2 | 0 | 1 | 1 | 1 | (0 x 1) + (1 x 1) = **1** | **2** | 0 |
| Input 3 | 1 | 1 | 0 | 1 | (1 x 1) + (0 x 1) = **1** | **2** | 0 |
| Input 4 | 1 | 1 | 1 | 1 | (1 x 1) + (1 x 1) = **2** | **2** | 1 |

# 26. Learning via Optimizing

</TECHUP_WOMEN>

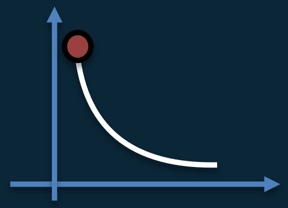
**Inputs**

| $N_1$ | $N_2$ |
|---|---|
| 0 | 0 |
| 0 | 1 |
| 1 | 0 |
| 1 | 1 |

**Outputs**

| $y$ |
|---|
| 0 |
| 0 |
| 0 |
| 1 |

Error



- Given some inputs and desired outputs, it was able to learn to recognize “AND”.
- The basic neuron “learned” via finding optimal weight values, and the optimal value for theta ($\theta$).
- This is similar to how we described learning in Module 1 – as a search for parameters that minimise some error rate.

**You may also hear this process being called: Pattern Recognition**

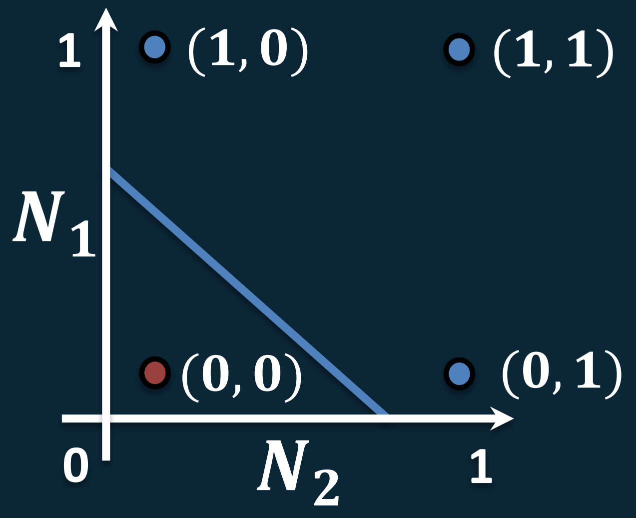
# 27. Decision Boundaries

- In module 1, we learned about linear separators, and decision boundaries.
- A single artificial neuron is capable of forming a linear boundary, thus it can be used for classification problems.
- This is shown here for Logical “OR” .
- Also for Logical “AND”.

Return 0
Return 1

| $N_1$ | $N_2$ | $y$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

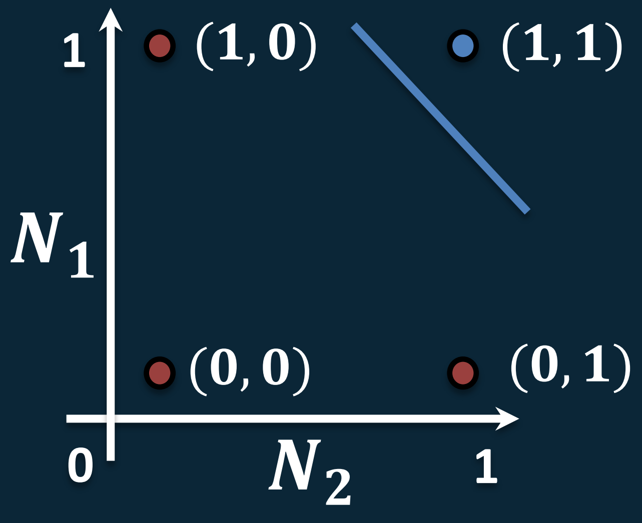
Logical “OR” training data




OR

| $N_1$ | $N_2$ | $y$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Logical “AND” training data




AND

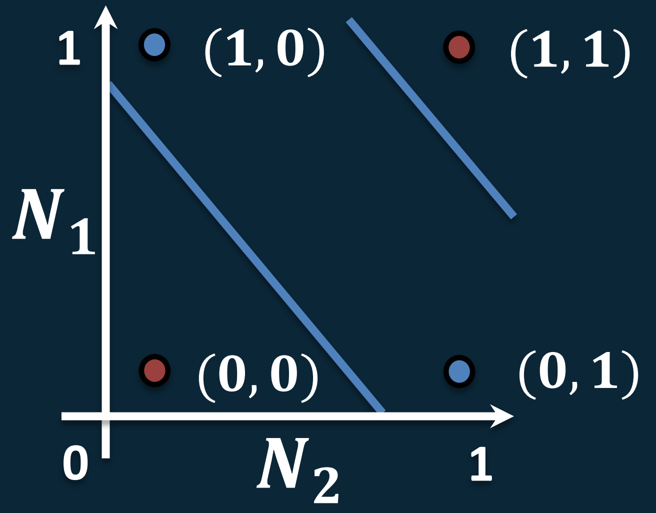
# 28. More Complex Boundaries



- Not all simple logical truths can be captured by our current artificial neuron.
- Consider the truth known as “Exclusive OR” (or XOR). It only returns true only when either $N_1$ or $N_2$ are TRUE.

| $N_1$ | $N_2$ | $y$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Logical “XOR” training data




XOR

- Return 0
- Return 1

# 29. Solving More Complex Problems

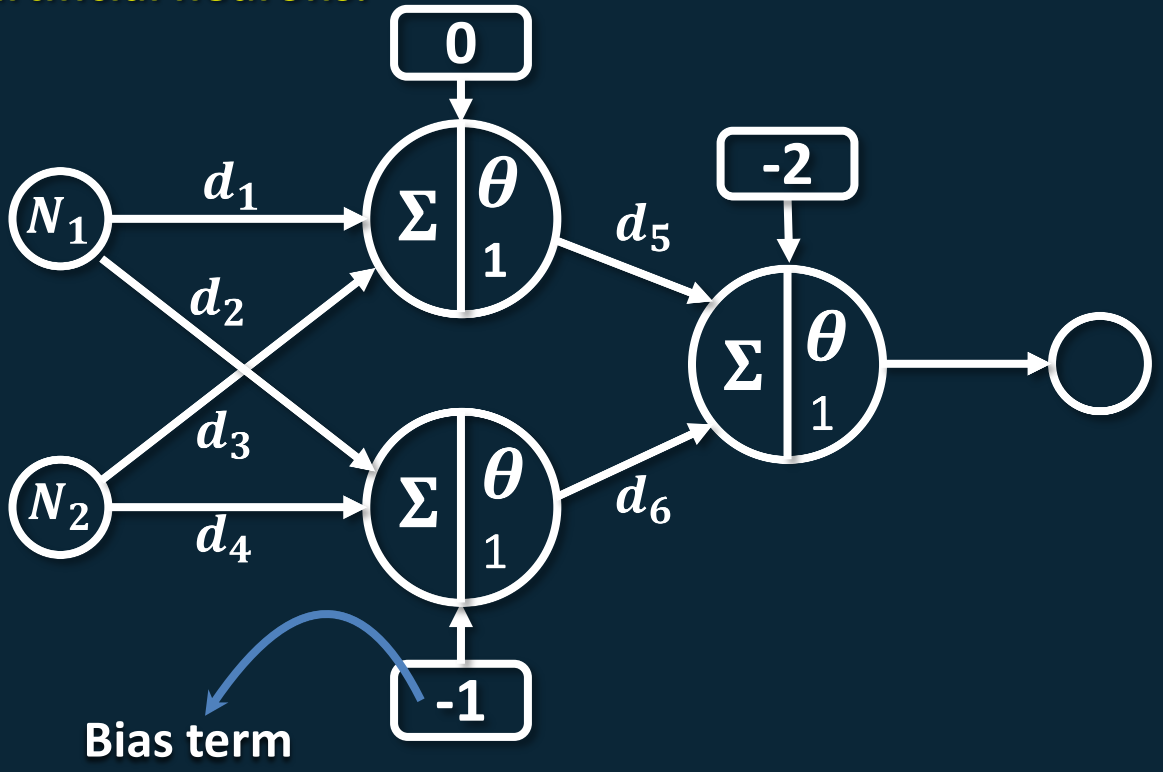
</TECHUP_WOMEN>

- To solve this, we copy what evolution did to the human brain – add more neurons.
- By combining neurons, we can create systems capable of recognising extremely complex patterns.
- Here's how we solve XOR using three artificial neurons.

There are bias terms that get added. These are added to $\Sigma$.

$$\sum_{i=0}^{n}(N_i \times d_i) - bias\ term$$

Why not try working it out?



# 30. Example – "XOR" Neuron 2

| $N_1$ | $N_2$ | $y$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

"XOR"

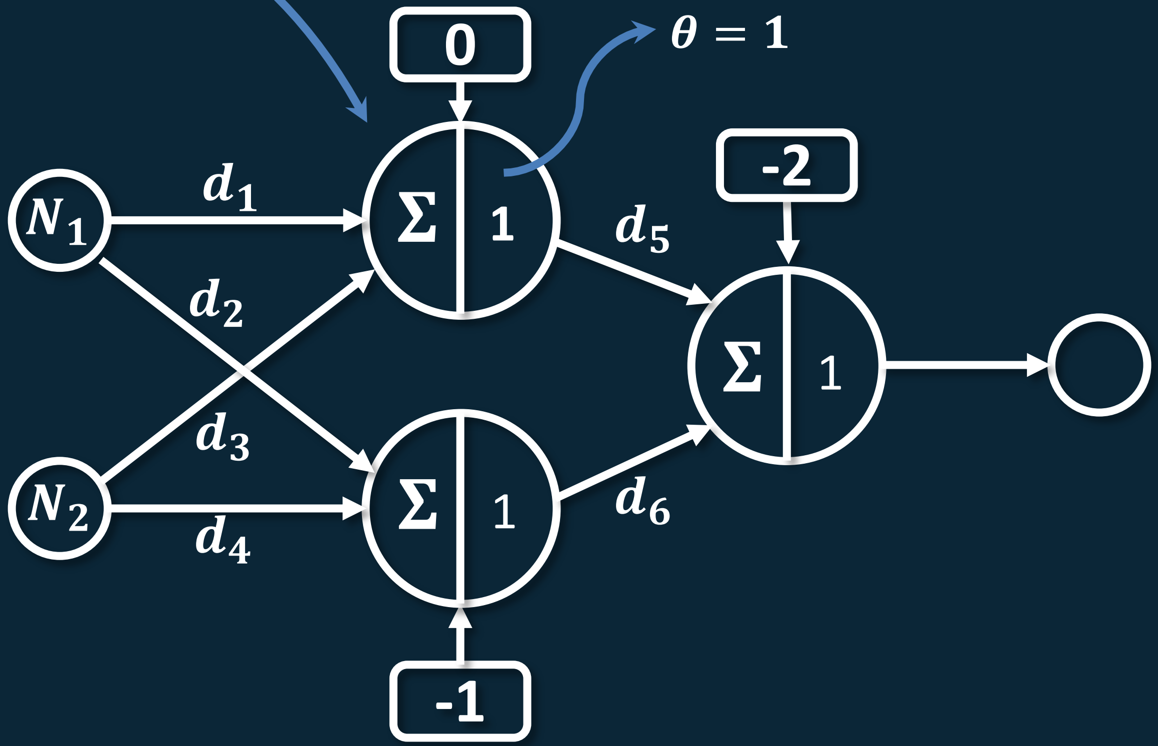
**Weights**

| | |
|---|---|
| $d_1$ | 1 |
| $d_2$ | 1 |
| $d_3$ | 1 |
| $d_4$ | 1 |
| $d_5$ | 3 |
| $d_6$ | -2 |

$$(N_1 \times d_1) + (N_2 \times d_3) + 0$$
$$(0 \times 1) + (1 \times 1) + 0$$
$$= \; 0 + 1 + 0 = 1$$

$$1 \geq \; \theta \; thus \ldots$$

**This Neuron "Fires"**



# 31. Example – “XOR” Neuron 2



| $N_1$ | $N_2$ | $y$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

“XOR”

**Weights**

| | |
|---|---|
| $d_1$ | 1 |
| $d_2$ | 1 |
| $d_3$ | 1 |
| $d_4$ | 1 |
| $d_5$ | 3 |
| $d_6$ | -2 |

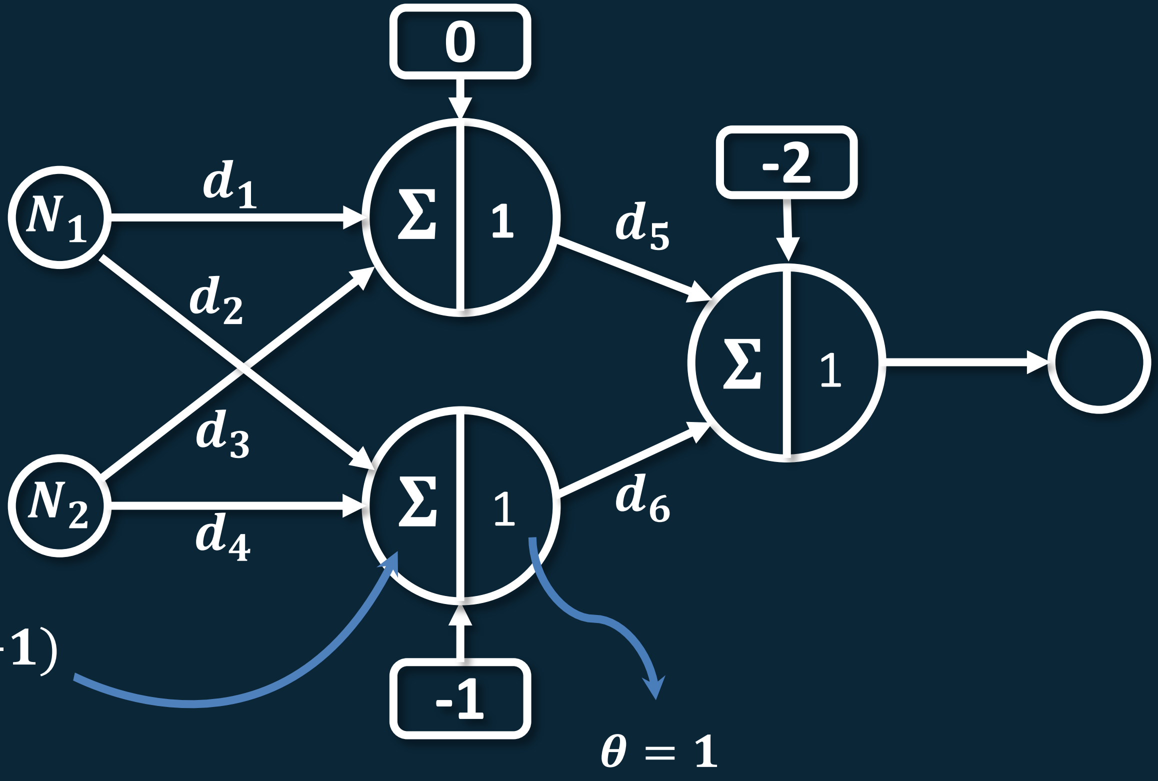
$0 < \theta \; thus \ldots$

**This Neuron does NOT “Fire”**

$$(N_1 \times d_2) + (N_2 \times d_4) + (-1)$$

$$(0 \times 1) + (1 \times 1) + (-1)$$

$$= \; 0 + 1 + (-1) = 0$$



Edge Hill University

# 32. Example – “XOR” Neuron 3

</TECHUP_WOMEN>

| $N_1$ | $N_2$ | $y$ |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

“XOR”

**Weights**

| | |
|---|---|
| $d_1$ | 1 |
| $d_2$ | 1 |
| $d_3$ | 1 |
| $d_4$ | 1 |
| $d_5$ | 3 |
| $d_6$ | -2 |

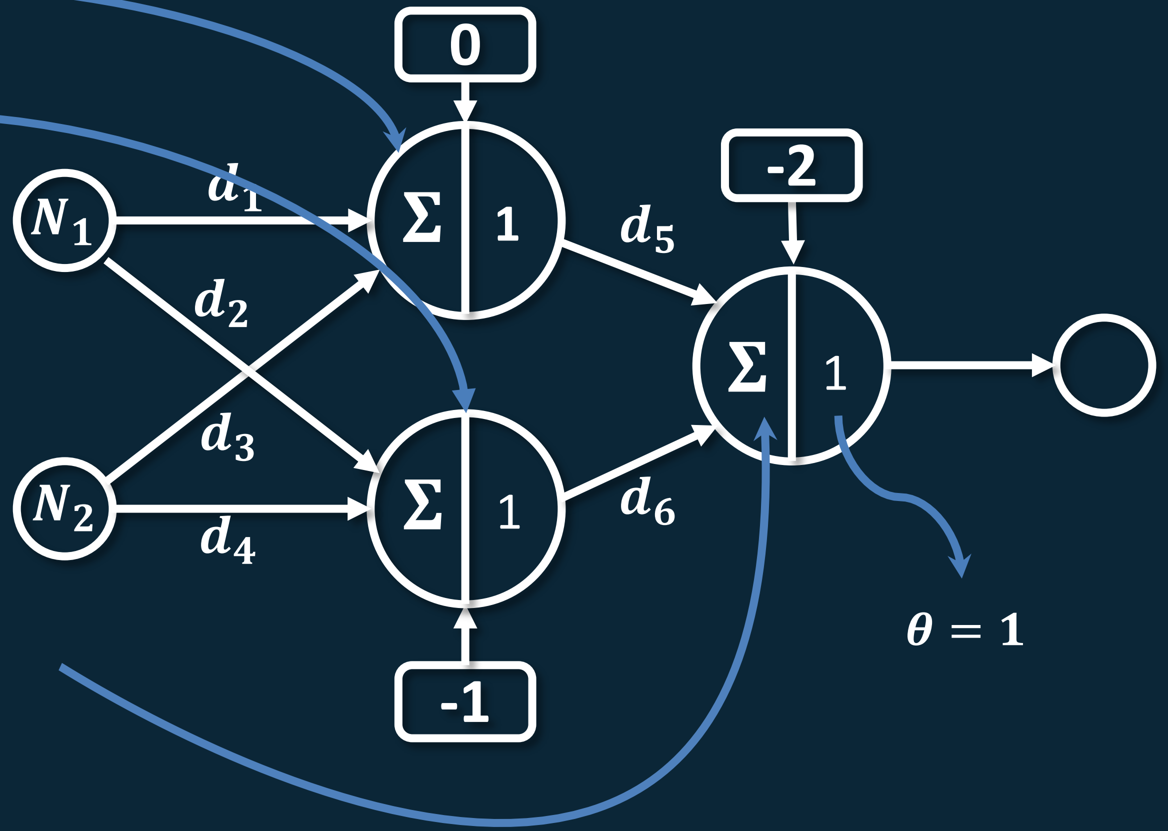
**This Neuron output 1**

**This Neuron output 0**

$1 \geq \theta$ ***thus …***

**This Neuron “Fires”**
**It recognised “XOR”!**

$$(1 \times d_5) + (0 \times d_6) + (-2)$$
$$(1 \times 3) + (0 \times -2) + (-2)$$
$$= 3 + 0 + (-2) = 1$$



# 33. Reflection



- Reflecting on what we've learned at this stage is useful.
- We've learned what an artificial neuron is.
- We've learned that by combining multiple neurons, we can solve problems of increasing complexity.
- We know that we train neural networks by attuning weights and thresholds so that the values input to the network, creating the outputs we expect.
- Neural networks can read in our machine learning features, and output predictive labels.
- So how do these ideas apply to real-world problems?

# 34. Real-world Example – Facial Recognition



- So far, we learned to build basic networks capable of recognizing simple logical conditions.
- It is much harder to learn to recognize faces or language.
- Yet the principles are the same, and you already understand them, perhaps without realizing.
- To prove it, suppose we want to identify faces in images.
- We collect a set of faces and we digitize them, representing the pixels in each image as numbers.

Edge Hill University

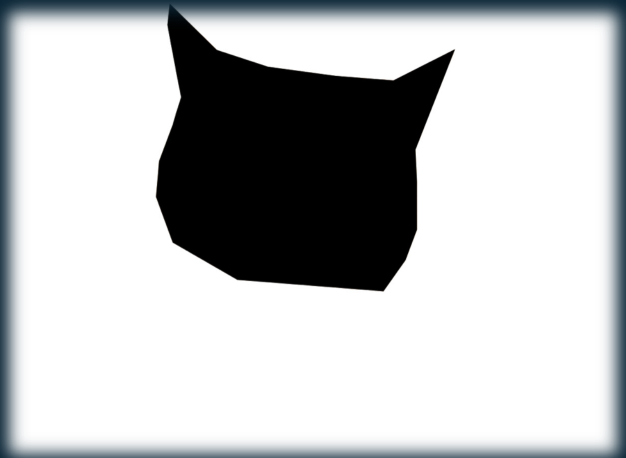
# 35. Processing the Data

</TECHUP_WOMEN>

- Suppose we have the example image shown.
- We annotate the pixels in the image.
- This helps us create a mask.
- Anything outside the mask should be labelled as 0 = No Face, 1 = Face.
- We use the mask to label the pixels, in each image in the training set. Each image will need it’s own mask.

Original

Annotated



Resulting Mask

Edge Hill University

# 36. Labelling the Pixels

</TECHUP_WOMEN>

- Here’s an example mask for the image shown on the last slide.
- Each pixel is represented by a zero or a one. The pixels containing the face are shaded blue for clarity.
- Now each pixel in the image is linked to a true class label.

| | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Edge Hill University

# 37. Training Set

</TECHUP_WOMEN>

- Now the images themselves, and the labels obtained via the mask, form a training set.
- The input image is 2-dimensional, thus the pixels can be described as a table.
- Each row can be linked to an output label from the mask.
- Neural networks can be trained to recognize parts of faces in each row, using the $y$-values provided.
- We can then test them on independent test sets, the concept of "face", has been learned.
- We could do the same for the columns – remember this isn't exactly how it's done in practice, but the process is the same in the real world.

| | $Pixel\ 1$ | | | $Pixel\ n$ | $y$ |
|---|---|---|---|---|---|
| Row 1 | 23 | 17 | 22 | 30 | 0 |
| . | 26 | 26 | 28 | 29 | 1 |
| . | 40 | 25 | 14 | 22 | 1 |
| Row $m$ | 22 | 16 | 34 | 19 | 0 |

Row 1 – no face

Row 50 – face

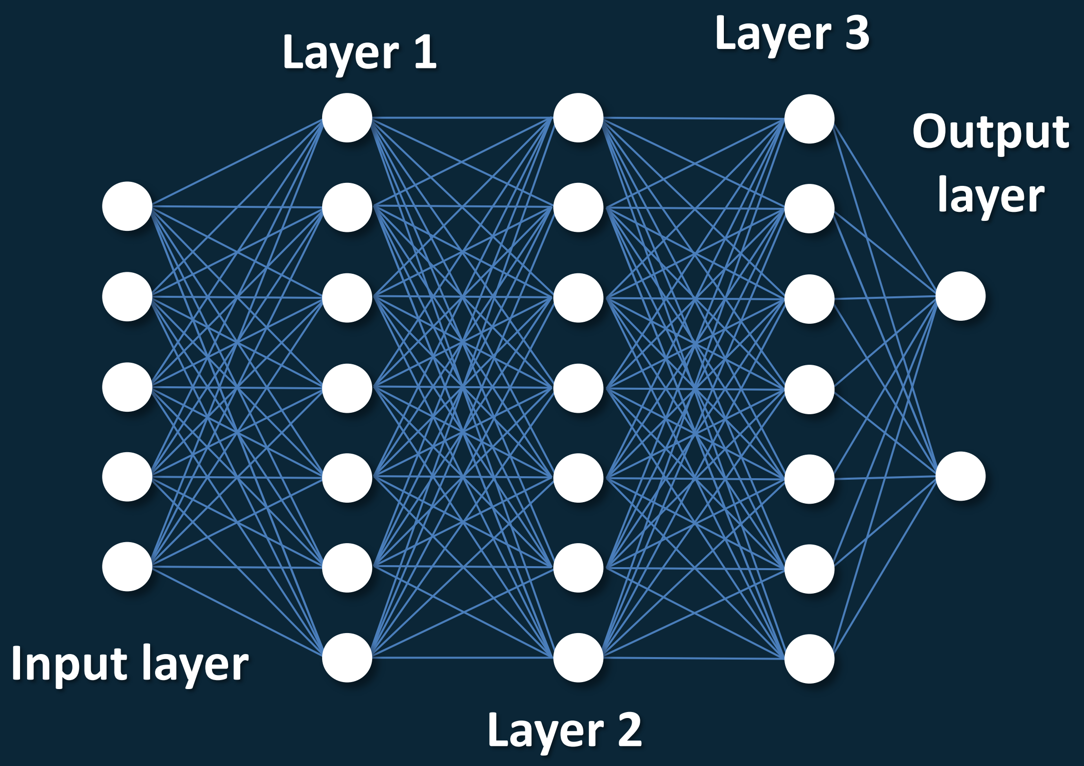
# 38. More Complex Problems

</TECHUP_WOMEN>

- Neural networks become extremely large as the problems we need them to solve increase in complexity.
- There is a link between problem complexity, and the number of artificial neurons required to solve it.
- The result is that these networks often need to become “deep”. – containing thousands, or even millions of neurons.
- To build such networks, we now understand that this means computing, and optimising, millions of connection weights and threshold values.
- This presents a challenge - training large networks becomes,
    - Time consuming.
    - Computationally expensive.
    - Costly in terms of energy.
- For a long time, this slowed the development of neural networks.
- In recent years, advances in hardware have made it possible to overcome some of these hurdles.

# 39. Present day “Deep” Learning

</TECHUP_WOMEN>



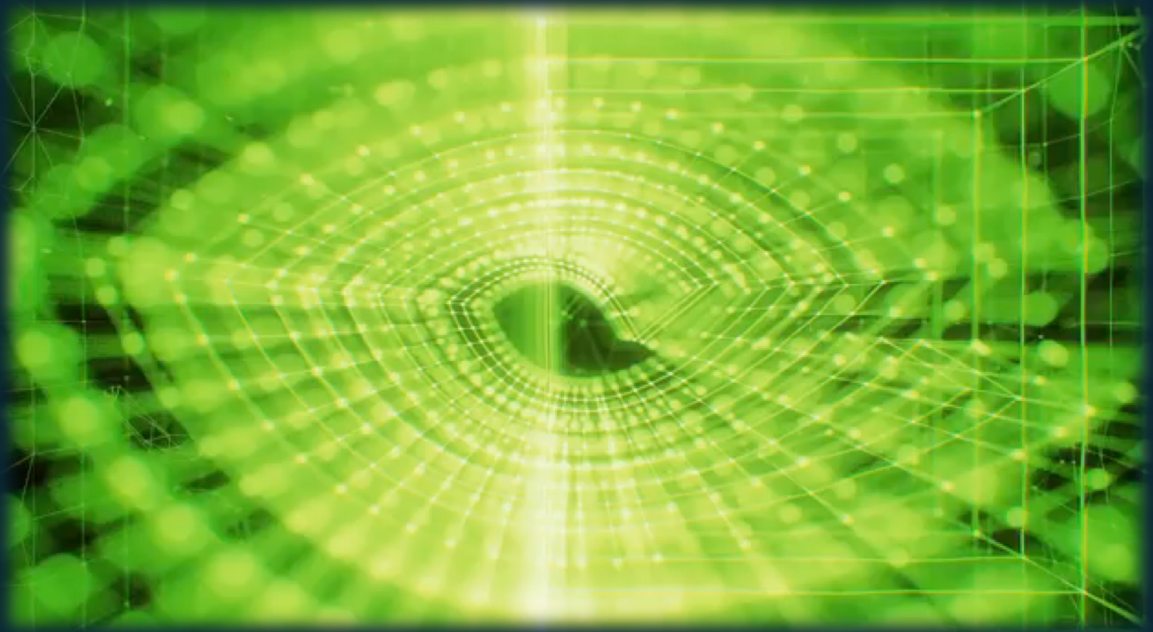

- Deep networks possess,
  - Many input neurons.
  - Many layers of neurons.
  - Possibly many output neurons.
- Please take the time to watch this 5-minute video – it will help explain these topics further.
- The network to the left is tiny by modern standards - modern networks created by companies such as Google contain countless artificial neurons.

Edge Hill University

# 40. Deep Neural Networks

</TECHUP_WOMEN>

- Neural networks can also be used for truly remarkable purposes.
- In recent years, an approach was developed that could be used to generate new data based on what it learned.
- That is, if you give it paintings by the worlds top artists, it could generate a new piece of artwork inspired by their styles.
- Another use, was the generate of "fake" faces. In the video shown to the right, we see new faces created from those of existing people.
- It's hard to believe these faces aren't real.



Credit: Nvidia

Durham University · Edge Hill University · The University of Nottingham UNITED KINGDOM · CHINA · MALAYSIA · University of York · institute of CODING · Office for Students OfS

# 41. What does the future hold?

</TECHUP_WOMEN>

- As hardware improves, we'll continue to build and develop larger and more accurate neural networks.
- How this will impact our lives is yet to be seen. There are likely to be advantages and disadvantages to their continual improvement.
- What way are things heading?
  - Self-driving cars are likely to be on U.K. roads by 2020-2021. All made possible via neural networks.
  - The job market will become more difficult for some, and easier for others, as the adoption of machine learning continues.
  - Simulations of the human brain are on the horizon, using specialist hardware and neural networks: https://www.humanbrainproject.eu/en/
  - Neural networks are being applied in medicine, with the potential to vastly improve treatment and patient outcomes.
- It's an exciting time to be learning about neural networks.
- Only time will tell how things will truly play out.

Edge Hill University

# 42. Summary

</TECHUP_WOMEN>

During this module we've introduced,

- Biological learning.
- Neurons and synapses.
- The artificial analogue of the biological neuron.
- How artificial neural networks are structured.
- How artificial neural networks are trained.
- How they can be used to solve real-world problems.
- Deep learning.

We hope the ideas introduced piqued your curiosity. Perhaps you'd like to continue learning more about artificial neural networks in the future.

Durham University · Edge Hill University · The University of Nottingham UNITED KINGDOM · CHINA · MALAYSIA · UNIVERSITY of York · institute of CODING · Office for Students OfS

# 43. Resources



**Useful links:**

- **Neural Networks Tutorial Playlist:**
  https://www.youtube.com/playlist?list=PLqcwDw9tMf6gmk0kTvAE0FIZs1S09zJnE
- **“Deep Learning” by Ian Goodfellow, Yoshua Bengio & Aaron Courville, 2016, MIT Press.**
- **Machine Learning Course by Andrew Ng (Stanford):** https://www.coursera.org/learn/machine-learning.