IOC Topic 5.2 - Introduction to Neural Networks

Transcript & Notes

Author: Dr. Robert Lyon

Contact: robert.lyon@edgehill.ac.uk (www.scienceguyrob.com)

Institution: Edge Hill University

Version: 1.0

Topic 5, Module 2

Introduction Slide

Hello and welcome to Topic 5, Module 2, Introduction to neural networks. It will introduce ideas and concepts that may unfamiliar to you. My name is Dr. Robert Lyon, and I'll be taking you through this module. Notes will be provided that accompany this content. I advise that you keep those nearby as we move through the slides. Each slide is numbered, and this number corresponds to a location in the notes.

Slide 2
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.

Slide 3
- 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.

Slide 4
- 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,
- Translation, as in Google Translate.

- o Autonomous vehicles, e.g. Tesla
- o Face recognition, as used by Facebook.

Additional Notes:

This article describes some further examples where neural networks have been applied in recent years:

https://www.forbes.com/sites/bernardmarr/2018/08/20/10-amazing-examples-of-how-deep-learning-ai-is-used-in-practice/#2cd4001df98a

Slide 5

- 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:
- o 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.

Slide 6

- 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).

Slide 7

- 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. A nucleus, which sits in the cell body known as the Soma.
2. Dendrites.
3. An Axon.
4. An Axon Terminal.
- The synapse is a structure between neurons, over which electrical impulses can pass.

Slide 8

- 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.

Slide 9

- 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.

Slide 10

- 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.

Slide 11
- 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?

Slide 12
- 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.

Slide 13
- 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.

Slide 14

- 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.
- I advise that you follow the link to the tutorial video (11 minutes long), if unfamiliar with functions. It will help you understand this link a little better: https://www.youtube.com/watch?v=52tpYl2tTqk

Slide 15
- 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.

Slide 16
- 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 they could create an analogue of the biologic neuron, based on functions, which could be constructed.
- They suggested the first "artificial neuron".
- In principle, this could be recreated in technology – even in 1943.

Slide 17
Here on the left of the slide, we have a diagram of a biological neuron.

While on the right, we have a diagram describing the first artificial neuron. This is known as the McCulloch-Pitts Neuron. This artificial neuron has two dendrites, d_1 and d_2. These are connected to the Soma, which is connected to an axon. There is some unusual notation at the centre of the artificial neuron that will be unfamiliar. Not to worry, we'll explain this in more detail next. What's important is that you understand that this artificial neuron is simply a function.

<u>**Slide 18**</u>

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.

<u>**Slide 19**</u>
- The Sigma symbol (Σ), 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.

<u>**Slide 20**</u>

- Finally, the theta symbol ($\boldsymbol{\theta}$), represents a function, or even a single value, with an important role.
- If the value of Σ is greater than, or equal to the value of $\boldsymbol{\theta}$, 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.

<u>**Slide 21**</u>
- 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.

We have two inputs in the table N_1 and N_2, plus an output called y. Here y assumes its value, based on the values of N_1 and N_2.
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 $N_2 = 1$.

<u>**Slide 22**</u>
- 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.

Slide 23

First, we can see our model of a neuron. It receives input from two other neurons, and produces one output. Here we have a table representing logical "AND". The table shows there are two inputs, N_1 and N_2. There is a single output variable y.

The next table shows the four possible combinations of inputs for N_1 and N_2. You can also see here that the weights d_1 and d_2 are initially set to zero in all cases.

The next table shows how we compute the value of Sigma (Σ) using the formula shown on the slide. We can see that N_1 and N_2 are both equal to zero, Sigma is also equal zero. In fact, for these inputs, Sigma is always zero.

This table shows the threshold value, Theta, that we use for each input. We keep this the same.

Finally, this table shows the output produced by the neuron, for each input.

We can see that when N_1 and N_2 both equal 1, the neuron produces an output of zero. Remember this is because the neuron only fires when the value of sigma, is greater than, or equal to, the value of theta.

 It has therefore failed to produce the y value that we desired. In other words, it hasn't learned the concept of "AND".

Slide 24

Now we've made a change. We've updated the values of d_1 - we've change them from 0, to 1 (d_1 = 1). This means we have strengthened this connection. What affect does this have on the output of the neuron?

We recompute the value of sigma (Σ) using the updated value for d_1. This has certainly changed things. But what is the end result?

We keep the threshold value at 1 for now.

Keeping the threshold the same, but changing the weights, has improved things. Now when N_1 and N_2 are both equal to one, sigma equals one ($\Sigma = 1$). As sigma is greater than or equal to theta (i.e. $\Sigma \geq \theta = 1$), the neuron fires. This means it has correctly recognised a part of "AND". But things aren't perfect.

We can see that when N_1=1 and N_2=0, sigma is also one ($\Sigma = 1$). Again, sigma is greater than or equal to theta (i.e. $\Sigma \geq \theta = 1$), which causes the neuron to incorrectly fire. It has again failed to learn the concept of "AND". Out of the four input patterns, it has managed to get three out of four (3/4) correct.

Slide 25

This time we've updated the values of d_2 - we've changed them from 0, to 1 (d_2 = 1). This means we have strengthened this connection too. What affect does this have on the output of the neuron?

We recompute the value of sigma (Σ) using the updated value for d_2.

We keep the threshold value again at 1 for now.

Keeping the threshold the same, but with changing the weights, has made things worse. Now only two out of four outputs are correct! The neuron is now firing when either N_1 or N_2 are equal to 1.

This shows that altering weights alone, doesn't always improve things.

Slide 26

Let's update theta this time. We set theta equal to 2 ($\theta = 2$).

Again, we compute the values for the inputs.

Then recompute the value of sigma (Σ). We can see now that things look promising.

The neuron only fires when both N_1 or N_2 are equal to 1. It doesn't fire under any other circumstances.

This is because the value of sigma is only equal to the value of theta, when N_1 or N_2 are equal to 1.

This configuration of a simple neuron has successfully captured the concept of "AND"!

We now understand that to represent "AND", we had to attune the weights and threshold values, to get the desired output from the neuron. The neuron only fires when its weighted inputs are greater than or equal to theta. This is similar to how neurons in the brain work – they will only "fire" when the electrical impulse reaching them, is strong enough.

Slide 27

- Given some inputs and desired outputs, a simple neuron was able to learn to recognize "AND".
- The basic neuron "learned" via finding optimal weight values, and the optimal value for theta (θ).
- 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

Slide 28

- 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 linearly separable classification problems.
- This is shown here for Logical "OR". First, we can see the table describing logical "OR". Now we can create a plot to show this information graphically. Each outcome can be graphed as a pair on an x, y plane. When both N_1 and N_2 are zero, we obtain this red point here. Noting that red points indicate a return value of zero, and blue points a return value of one. We can plot the remaining three outcomes using blue points like so. It may not be obvious, but there is a decision boundary separating these outcomes.
- We can do the same for Logical "AND". The outcomes in this case are also easily separable.
  If necessary, pause at this slide, and check that you fully understand what you're seeing.

Slide 29

- 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. When N_1 and N_2 are both equal to zero, we obtain a return value of zero. When both N_1 and N_2 are equal to one, again we obtain a return value of zero. But if either N_1 or N_2 are equal to one, but not together, we obtain a return value of one. We can see that this problem can't be solved by a single linear separator. Thus, linear models cannot solve it. Unfortunately, a single artificial neuron is a linear model.

Slide 30

- To solve this "exclusive or" problem, 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 to sigma. Why don't we try applying this now.

Slide 31

- Here we have XOR and our first artificial neural network.
- This is the pattern we'll use to test the network.
- These are the connection weights we'll use.
- Let's compute sigma at the first neuron in the network.
- We fill in the values.
- And obtain our answer - sigma is equal to 1.
- Now we check if sigma is greater than or equal to Theta.
- Since both have a value of 1, the neuron fires, producing an output of 1.

Slide 32

- Let's compute sigma at the second neuron in the network.
- We fill in the values.
- And obtain our answer - sigma is equal to 0.
- Now we check if sigma is greater than or equal to theta.
- Sigma is less than theta, so the neuron does not fire, producing an output of 0.

Slide 33

- Let's compute sigma at the third and final neuron in the network.
- We have the inputs 1 and 0 from the two neurons its connected to.
- We fill in the values.
- And obtain our answer - sigma is equal to 1.
- Now we check if sigma is greater than or equal to theta.
- Sigma is equal to theta, so the neuron fires, producing an output of 1. It has successfully recognised this XOR pattern. I leave you to test the other three patterns we didn't try here, in your own time.

Slide 34

- 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?

Slide 35

- 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.

Slide 36

- Suppose we have the example image shown.
- We annotate the pixels in the image. This involves identifying which pixels belong to objects or concepts under consideration. For instance, which pixels are in the foreground, which are in the background, which belong to the cat, etc.
- This helps us create a mask, as shown. This shows us the region occupied by the face.
- The pixels falling outside the mask, should be labelled zero, which means they are not part of the face. The rest should be labelled one, indicating that they are part of the face.
- We use the mask to label the pixels, in each image in the training set. Each image will need its own mask.

Slide 37

- 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.

Slide 38

- 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. For instance, row 1 contains no face, but row 50 contains a face.
- We can then test algorithms trained this way on independent test sets, to ensure 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.

Slide 39

- 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 them.
- 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.

- o Computationally expensive.
- o 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.

Slide 40
- 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.

Additional Notes:

Also see this website for a tutorial on a specific type of deep neural networks called, Convolution Neural Networks: See this excellent website for more details: https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/

Slide 41
- 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 produced by the world's top artists, it could generate a new piece of artwork inspired by their styles.
- Another use was the generation 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.

Slide 42
- 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?
- o Self-driving cars are likely to be on U.K. roads by 2020-2021. All made possible via neural networks.
- o The job market will become more difficult for some, and easier for others, as the adoption of machine learning continues.
- o Simulations of the human brain are on the horizon, using specialist hardware and neural networks: https://www.humanbrainproject.eu/en/
- o 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.

Slide 43

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.

There's so much more to learn about neural networks – we've barely scratched the surface here. Nonetheless, we hope the ideas introduced piqued your curiosity. Perhaps you'd like to continue learning more about this topic in the future.

Slide 44

- There are some useful links that I wanted to share before I go.
- There is a series of tutorials on YouTube that will help you understand the content of this module.
- There is a book called "Deep Learning", which I highly recommend you look in to.
- Finally, there is a full machine learning course available for you to pursue, in your own time, online.
- Best of look with the rest of your studies.