

EPHE 245

Memory as Synaptic Strength Worksheet

The key concept to understanding this worksheet, and indeed, this whole topic is to realize that very complex patterns of information can be represented very simply.

Part 1: Binary Numbers

Let's consider a single byte of information, which is comprised of 8 bits.

0 0 1 0 0 1 0 0

In binary, we usually work from right to left (it's a computer thing). To decode the above number, you need the key. Binary numbers are represented as follows:

128 64 32 16 8 4 2 1

If a value is "on" (a binary 1) then you count the number, if a number is "off" (a binary 0), then you do not count the number. So the above number is:

0 0 1 0 0 1 0 0

128 (off) 64 (off) 32 (on) 16 (off) 8 (off) 4 (on) 2 (off) 1 (off) = 36

Try and decode the following binary numbers:

1 0 1 0 0 1 1 0

1 0 1 1 0 1 0 1

0 0 0 0 1 0 0 0

Part 2: Simple Neural Systems

We can use a very similar logic for a neural system. Consider the following two neurons joined by a synapse:



If the synapse is not coloured in, then it is weak, so if the input neuron (the left side) fires, then the output neuron (the right side) will not fire. The arrows designate the direction of neural firing to help you decide which is the input and which is the output.

In binary we would represent a firing neuron as a 1, and a non-firing neuron as a 0. So, for this simple circuit the input and output “matrix” would look like this:

Input	Output
1	0
0	0

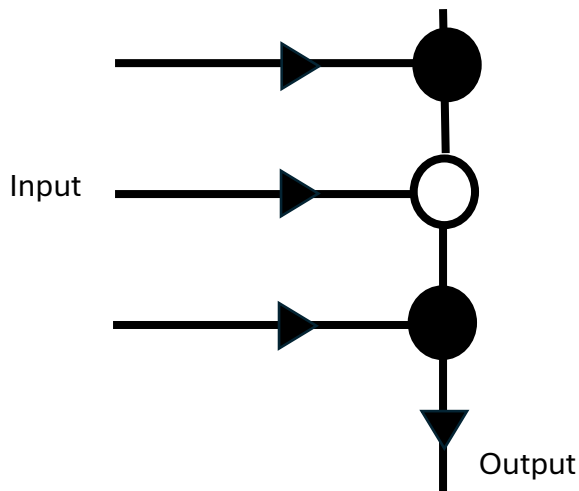
Now, consider this simple neural system.



The input and output matrix would look like this.

Input	Output
1	1
0	0

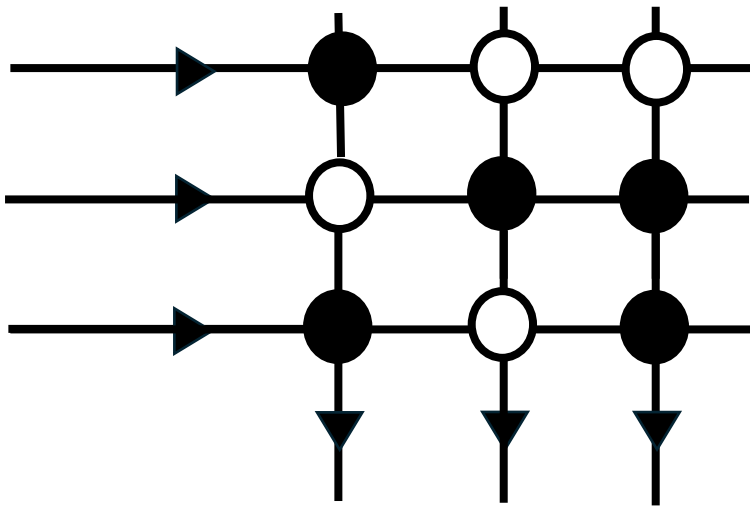
Okay, let’s take this up a level. Let’s consider a neural system where we have three input that are connected to a single output neuron. Given the diagram and strength of connections, complete the input and output matrix.



	Example	Trial 1	Trial 2	Trial 3
Input 1	0	1	1	1
Input 2	0	1	0	0
Input 3	1	1	1	0
Output	1			

Part 3: More Complex Neural Systems

Use the same logic, complete the input and output matrix for this neural system.



	Example	Row 1	Row 2	Row 3
Input 1	1	1	0	0
Input 2	1	0	1	1
Input 3	1	1	0	1
Output		1	0	1

Now try these trials:

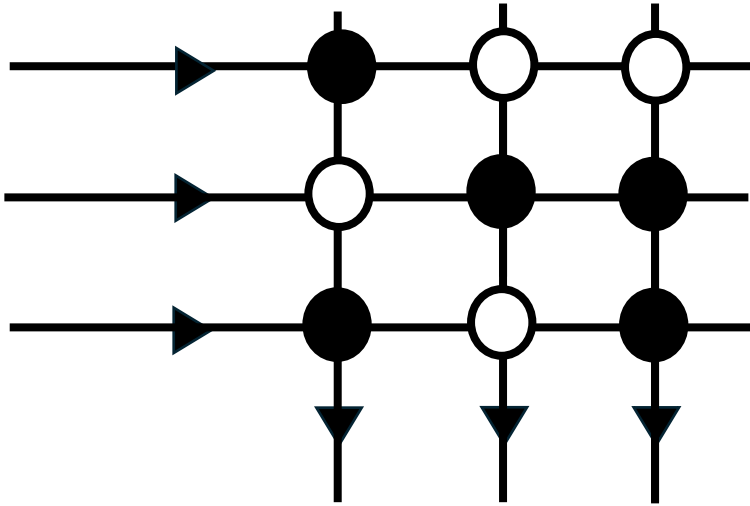
	Trial 1	Row 1	Row 2	Row 3
Input 1	1			
Input 2	0			
Input 3	1			
Output				

	Trial 2	Row 1	Row 2	Row 3
Input 1	0	0	0	0
Input 2	1	0	1 x 1	1 x 1
Input 3	1	1 x 1	1 x 0	1 x 1
Output		1	1	2

	Trial 3	Row 1	Row 2	Row 3
Input 1	0			
Input 2	0			
Input 3	1			
Output				

Part 4: Firing Rules

We will use the same neural circuit as above. BUT, an output neuron will ONLY fire if has a total summed input of 2 or more.



	Example	Row 1	Row 2	Row 3
Input 1	1	1	0	0
Input 2	1	0	1	1
Input 3	1	1	0	1
Output		1	0	1

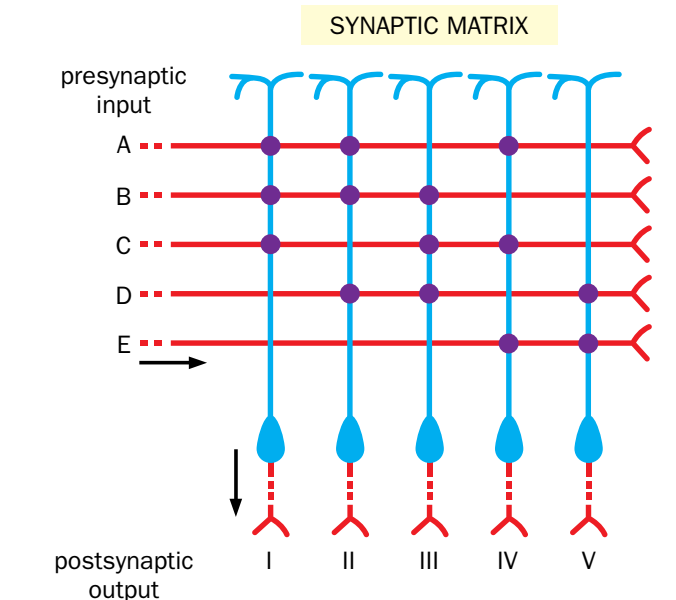
Note, now the output is BINARY!!!! Complete the examples below with the new firing rule.

	Trial 1	Row 1	Row 2	Row 3
Input 1	1			
Input 2	0			
Input 3	1			
Output				

	Trial 2	Row 1	Row 2	Row 3
Input 1	0			
Input 2	1			
Input 3	1			
Output				

	Trial 3	Row 1	Row 2	Row 3
Input 1	0			
Input 2	0			
Input 3	1			
Output				

Part 5. It is Really Just Math



Consider the following. If the input to this synaptic matrix is [1 1 0 0 0] from top to bottom, what would the output be if we assume a firing rule of 1?

Input	Output I	Output II	Output III	Output IV	Output V
1	1 x 1	1 x 1	1 x 0	1 x 1	1 x 0
1	1 x 1	1 x 1	1 x 1	1 x 0	1 x 0
0	0 x 1	0 x 0	0 x 1	0 x 1	0 x 0
0	0 x 0	0 x 1	0 x 1	0 x 0	0 x 1
0	0 x 0	0 x 0	0 x 0	0 x 1	0 x 1
Output	1	1	1	1	0

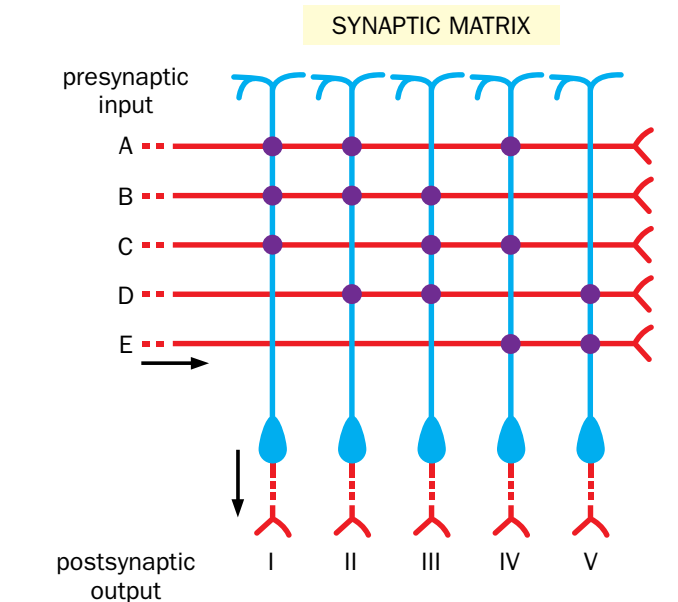
How does this change with a firing rule of 2?

Input	Output I	Output II	Output III	Output IV	Output V
1					
1					
0					
0					
0					
Output					

Now think about this. The output is BINARY. And you can represent anything with binary values! Images, language, music, and motor commands! And a different input gives a different output with the same synaptic strength = FLEXIBILITY!

Part 6. Turning this into Motor Commands

Let's use the same circuit, but just change a few labels.



Input	Forearm Flexor	Forearm Extensor	Biceps	Triceps	Trapezius
0					
1					
0					
1					
1					
Output					

With the given input, and a firing rule of 2, which muscles fire?

How about with this example?

Input	Forearm Flexor	Forearm Extensor	Biceps	Triceps	Trapezius
1					
0					
1					
0					
0					
Output					