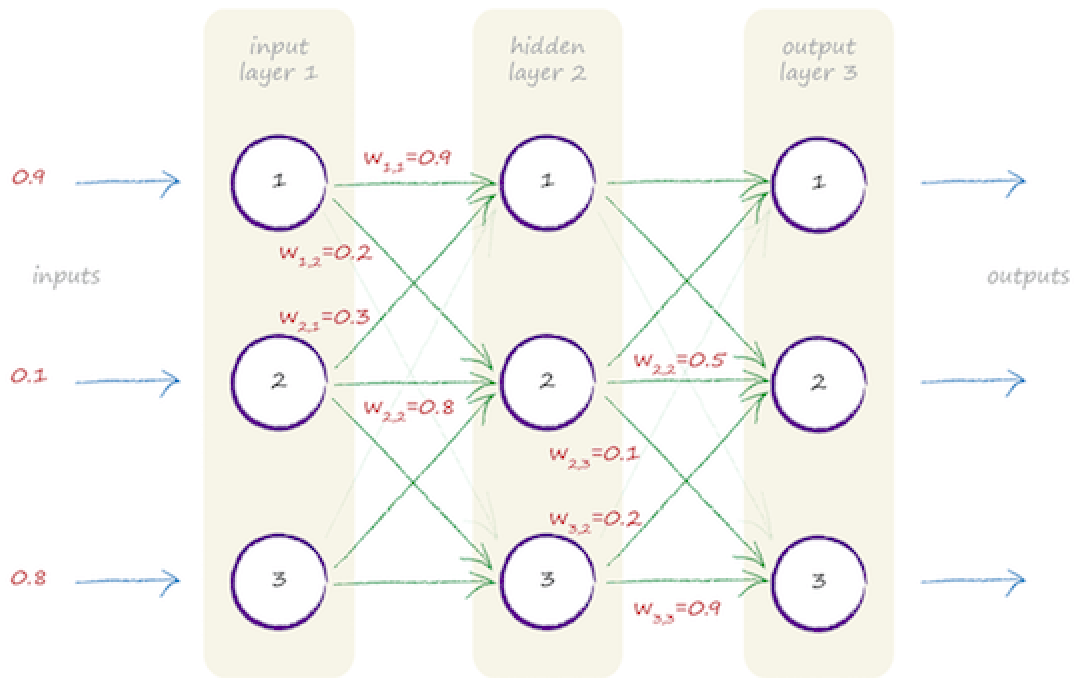
# A Three Layer Example: Working on Input Layer

In this lesson, we will take a look at a three-layered Neural Network example and calculate the values for input layer.

We haven't worked through feeding signals through a neural network using matrices to do the calculations. We also haven't worked through an example with more than two layers, which is interesting because we need to see how we treat the outputs of the middle layer as inputs to the final third layer. The following diagram shows an example neural network with three layers, each with three nodes. To keep the diagram clear, not all the weights are marked.




We'll introduce some of the commonly used terminology here too. The first layer is the *input layer*, as we know. The final layer is the *output layer*, as we also know. The middle layer is called the *hidden layer*. That sounds mysterious and dark, but sadly there isn't a mysterious dark reason for it. The name just stuck because the outputs of the middle layer are not necessarily made apparent as outputs, so are "hidden". Yes, that's a bit lame, but there really isn't a better reason for the name. Let's work through that example network, illustrated in that diagram. We can see the three inputs are $0.9$, $0.1$ and $0.8$. So the input matrix $\mathbf{I}$ is:

$$I = \begin{pmatrix} 0.9 \\ 0.1 \\ 0.8 \end{pmatrix}$$

That was easy. That's the first input layer done because that's all the input layer does - it merely represents the input. Next is the middle hidden layer. Here we need to work out the combined (and moderated) signals to each node in this middle layer. Remember each node in this middle hidden layer is connected to by every node in the input layer, so it gets some portion of each input signal. We don't want to go through the many calculations as we did earlier, we want to try this matrix method.

As we just saw, the combined and moderated inputs into this middle layer are $X = W.I$ where **I** is the matrix of input signals, and **W** is the matrix of weights. We have **I** but what is **W**? Well, the diagram shows some of the (made up) weights for this example but not all of them. The following shows all of them again made up randomly. There's nothing particularly special about them in this example.

$$W_{input\_hidden} = \begin{pmatrix} 0.9 & 0.3 & 0.4 \\ 0.2 & 0.8 & 0.2 \\ 0.1 & 0.5 & 0.6 \end{pmatrix}$$

You can see the weight between the first input node, and the first node of the middle hidden layer is $w_{1,1} = 0.9$, just as in the network diagram above. Similarly, you can see the weight for the link between the second node of the input and the second node of the hidden layer is $w_{2,2} = 0.8$, as shown in the diagram. The diagram doesn't show the link between the third input node and the first hidden layer node, which we made up as $w_{3,1} = 0.4$.

But wait — why have we written "input_hidden" next to that **W**? It's because $W_{input\_hidden}$ are the weights between the input and hidden layers. We need another matrix of weights for the links between the hidden and output layers, and we can call it $W_{hidden\_output}$. The following shows this second matrix $W_{hidden\_output}$ with the weights filled in as before. Again you should be able to see, for example, the link between the third hidden node and the third output node as $w_{3,3} = 0.9$.

$$W_{hidden\_output} = \begin{pmatrix} 0.3 & 0.7 & 0.5 \\ 0.6 & 0.5 & 0.2 \\ 0.8 & 0.1 & 0.9 \end{pmatrix}$$

Great, we have got the weight matrices sorted. Let's get on with working out the combined moderated input into the hidden layer. We should also give it a descriptive name, so we know it refers to the combined input to the middle layer and not the final layer. Let's call it $X_{hidden}$.

$$X_{hidden} = W_{input\_hidden} . I$$

We are not going to do the entire matrix multiplication here because that was the whole point of using matrices. We want computers to do the laborious number crunching. The answer is worked out as shown below.

$$X_{hidden} = \begin{pmatrix} 0.9 & 0.3 & 0.4 \\ 0.2 & 0.8 & 0.2 \\ 0.1 & 0.5 & 0.6 \end{pmatrix} \cdot \begin{pmatrix} 0.9 \\ 0.1 \\ 0.8 \end{pmatrix}$$

$$X_{hidden} = \begin{pmatrix} 1.16 \\ 0.42 \\ 0.62 \end{pmatrix}$$

I used a computer to work this out, and we will learn to do this together using the Python programming language in part 2 of this guide. We won't do it now as we don't want to get distracted by computer software just yet. So we have the combined moderated inputs into the middle hidden layer, and they are $1.16$, $0.42$ and $0.62$. And we used matrices to do the hard work for us. That's an achievement to be proud of!