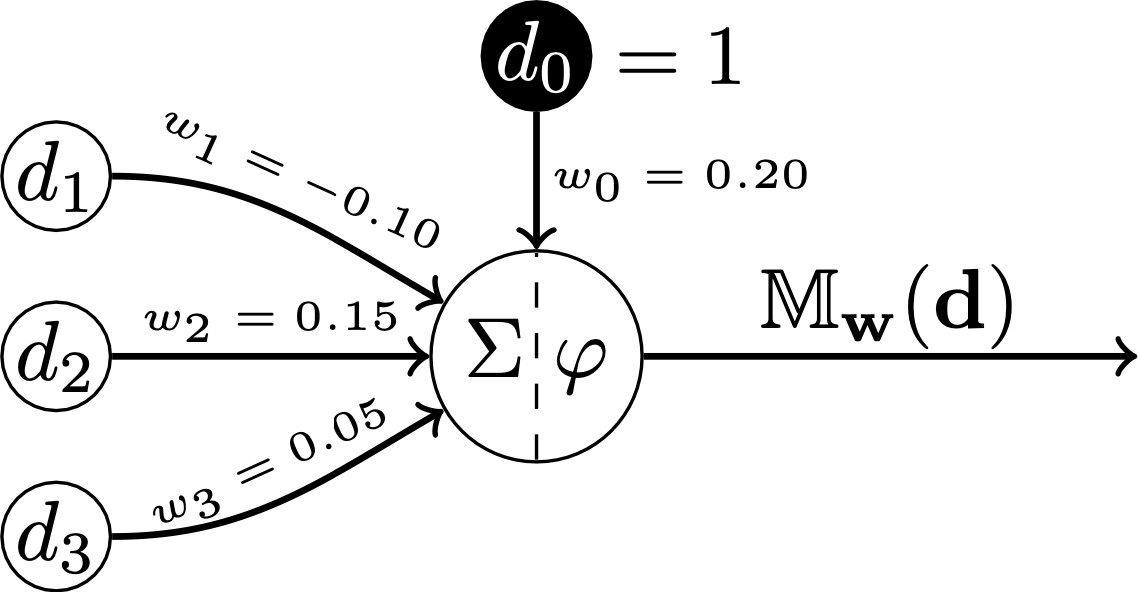
# DSBA 6156

**Assignment #3**

# Module 8

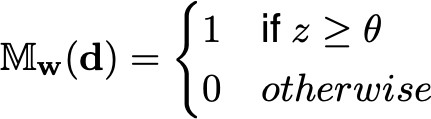
1. The following image shows an artificial neuron that takes 3 inputs: (4 points)



* 1. Calculate the weighted sum for this neuron for the input vector: d = [0.2, 0.5, 0.7]

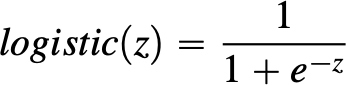
Z = (1\*.2) + (.2 \*-1) + (.5\*.15) + (.7 \*.05) =

.2 + -.02 + .075 + .035 = .29

* 1. What would be the output from this neuron if the activation function φ is a threshold activation with θ = 1?

It would be 0 based on the activation above, our Z value from part a is .29, we know our theta is 1, so Mw(d) = 0

* 1. What would be the output from this neuron if the activation function φ is the logistic function?

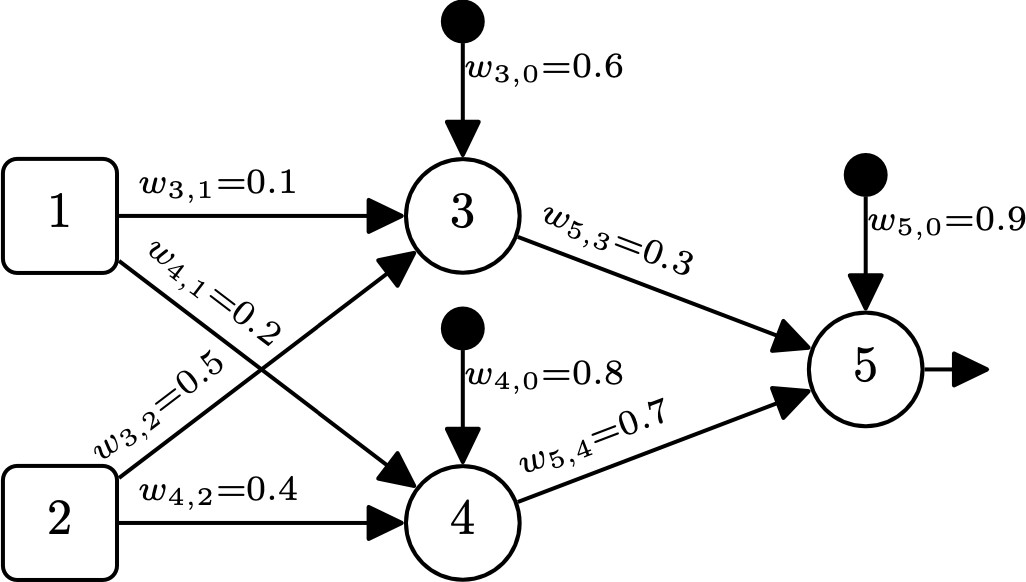


1/ 1+e^-(.29) = .5719961329

* 1. What would be the output from this neuron if the activation function φ is the rectified linear function?

The output would .29 as the max of a rectifier from 0 to Z would be the max of .29

1. The following image shows an artificial neural network with two sensing neurons (Neurons 1 and 2) and 3 processing neurons (Neurons 3, 4, and 5)



* 1. Assuming that the processing neurons in this network use a logistic activation function, what would be the output of Neuron 5 if the network received the input vector: Neuron 1 = 0.7 and Neuron 2 = 0.3? (2 points)

Z3 =(.6 \*1) +(.1\*.7) + (.5 \*.3)

=(.6) + (.07) + (.15)

= .82

Z4 = (.8\*1) + (.2 \*.7) +(.4 \*.3)

= (.8) + (.14) + (.12)

= 1.06

Logistic Function Z3: 1/1 + e^-(.82) = .6942363401

Logistic Function Z4: 1/1 + e^-(1.06) = .7426905453

Z5 = (.9 \*1) + (.3 \* .6942363401) + (.7 \* .7426905453)

= (.9) + (.208270902) + (.519883382)

= 1.6281542837

Logistic Function Z5 = 1/1+e^-(1.6281542837) = .835916637

* 1. Assuming that the processing neurons in this network use a ReLU activation function, what would be the output of Neuron 5 if the network received the input vector: Neuron 1 = 0.7 and Neuron 2 = 0.3? (1 point)

We would use the weighted calculations above with the rectifier Max(0,Z) because both of the values are greater than 0

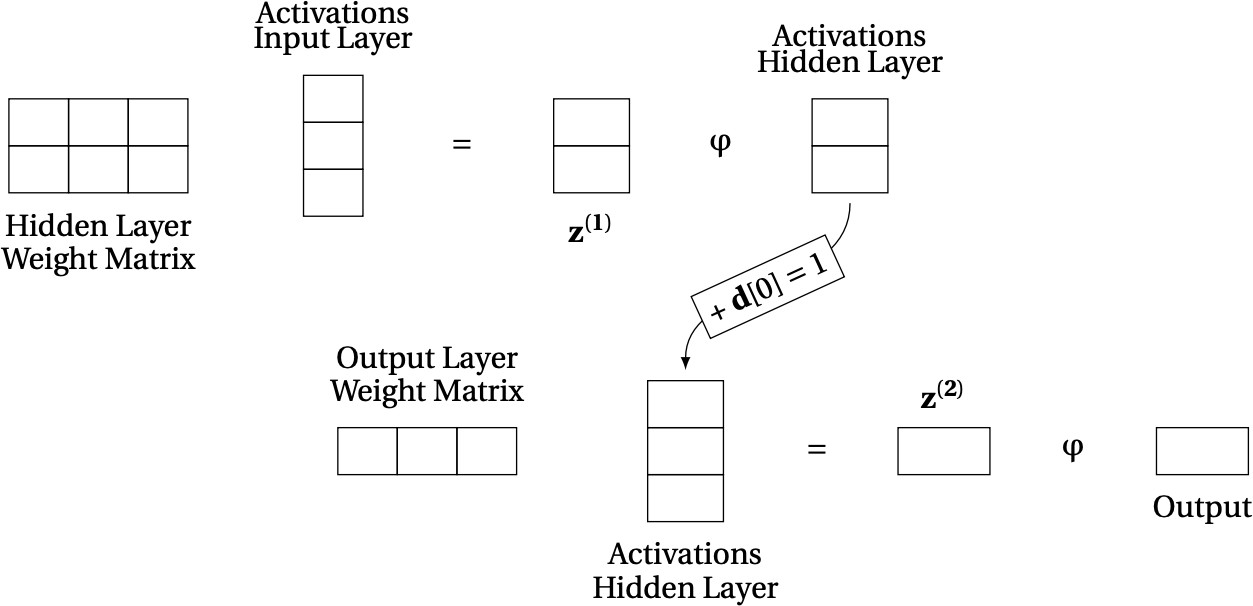
Z5 = (.9 \* 1) + (.3 \*.82) + (.7 \*1.06)

= (.9) + (.246) + (.742)

= 1.888

The final output would use the same equation with the others, so the activation would be for Neuron 5 is 1.888

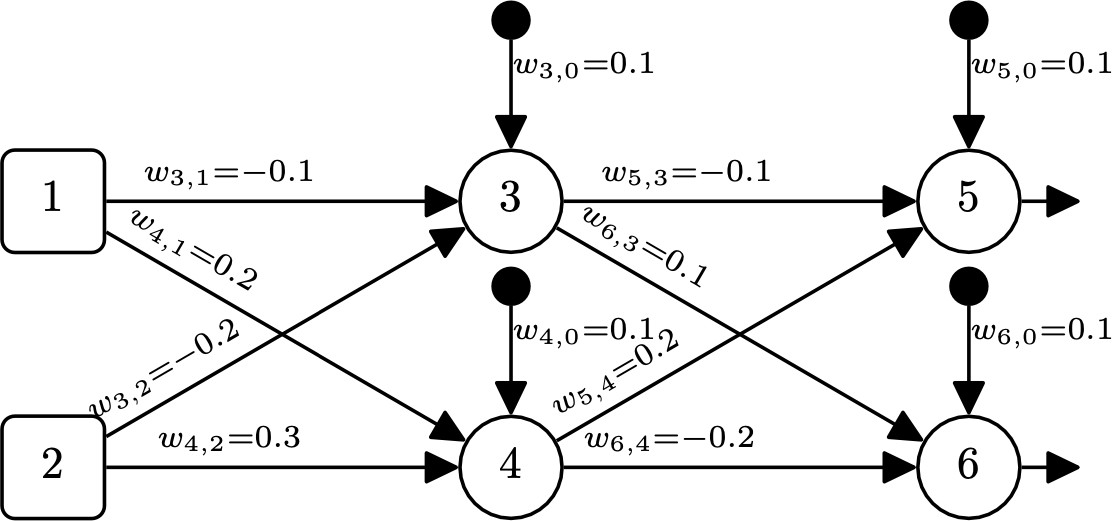
* 1. The following image provides a template diagram for the sequence of matrix operations that our neural network would use to process the input vector Neuron 1 = 0.7 and Neuron 2 = 0.3. Assuming that the processing neurons in the network use a ReLU activation function, fill in the diagram (Excel) with the appropriate weights, bias terms, weighted sum values, and activations. (4 points)



A graph with a line and numbers

Description automatically generated with medium confidence

1. The following image illustrates the topology of a feedforward neural network that has two sensing neurons (Neurons 1 and 2), two hidden processing neurons (Neurons 3, and 4), and two processing output neurons (Neurons 5 and 6).



* 1. Assuming that the processing neurons use a rectifier activation functions, that the input to the network is Neuron 1 = 0.3 and Neuron 2 = 0.6 and that the desired output for this input is Neuron 5 = 0.7 and Neuron 6 = 0.4:
     1. Calculate the output generated in response to this input. (2 points)

Z3 = (.1\*1) + (-.1 \*.3) + (-.2 \*.6) = -.05

Z4 = (.1\*1) + (.2 \* .3) +(.3 \* .6) = .34

0 and .34

Z5 = (.1\*1) + (-.1 \*0) +(.2 \*.34) = .168

Z6 = (.1\*1) + (.1\*0) + (-.2 \*.34) = .032

* + 1. Calculate the sum of squared errors for this network in this example. (1 point)

Z5 Error = (Desired – Total) ^2 = (.7 -.168) ^2 = .283024

Z6 Error = (Desired – Total) ^2 = (.4 - .032) ^2 = .135424

Sum of Squared Errors = .418448

* + 1. Calculate the δ values for each of the processing neurons in the network (i.e., δ6, δ5, δ4, δ3). (2 points)

δ3 = ((−0.1⋅0.532) +(0.1⋅0.368)) \* 1 = 0 because it was negative

δ4 = ((0.2⋅0.532) +(−0.2⋅0.368)) \* 1 = 0.0328

δ5 = -(.7 -.168) \* 1 = .532

δ6 = -(.4 - .032) \* 1 = -0.368

* + 1. Using the δ values you calculated above, calculate the sensitivity of the error of the network to changes in each of the weights of the network i.e.

(3 points)

w6\_4: .368 \* .34 = 0.12512,

w6\_3: .368 \* 0 = 0.0,

w6\_0: .368 \* 1 = 0.368,

w5\_4: .532 \* .34 = 0.180880,

w5\_3: .532 \* 0 = 0.0,

w5\_0: .532 \* 1 = 0.532,

w4\_2: .0328 \* .6 = .0.019680,

w4\_1: .0328 \* .3 = 0.009840,

w4\_0: .0328 \* 1 = 0.03280,

w3\_2: 0.0,

w3\_1: 0.0,

w3\_0: 0.0

* + 1. Assuming a learning rate of α = 0. 1, calculate the updated values for each of the weights in the network

(𝑤6,4, 𝑤6,3, 𝑤6,0, 𝑤5,4, 𝑤5,3, 𝑤5,0, 𝑤4,2, 𝑤4,1, 𝑤4,0, 𝑤3,2, 𝑤3,1, 𝑤3,0) after the processing of this single training example.(3 points)

w6\_4: −0.2 − 0.1 \* 0.12512 = −0.2 − 0.012512 = −0.212512

w6\_3: Remains unchanged because its sensitivity is 0, so new value = 0.1

w6\_0: 0.1 − 0.1 \* 0.368 = 0.1 − 0.0368 = 0.0632

w5\_4: 0.2 − 0.1 \* 0.18088 = 0.2 − 0.018088 = 0.181912

w5\_3: Remains unchanged because its sensitivity is 0, so new value = -0.1

w5\_0: 0.1 − 0.1 \* 0.532 = 0.1 − 0.0532 = 0.0468

w4\_2: 0.3 − 0.1 \* 0.01968 = 0.3 − 0.001968 = 0.298032

w4\_1: 0.2 − 0.1 \* 0.00984 = 0.2 − 0.000984 = 0.199016

w4\_0: 0.1 − 0.1 \* 0.0328 = 0.1 − 0.00328 = 0.09672

w3\_2: Remains unchanged because its sensitivity is 0, so new value = -0.2

w3\_1: Remains unchanged because its sensitivity is 0, so new value = -0.1

w3\_0: Remains unchanged because its sensitivity is 0, so new value = 0.1

* + 1. Calculate the reduction in the error of the network for this example using the new weights, compared with using the original weights.(3 points)

Z3 = (.1\*1) + (-.1 \*.3) + (-.2 \*.6) = -1.13 aka 0

Z4 = (0.09672\*1) + (0.199016\* .3) +(0.298032\* .6) = 0.335244

Z5 = 0.0468+ (-0.1\* -1.13) + (0.181912\* 0.335244) = .22078490652

Z6 = 0.0632 + (.1 \* 0) + (−0.212512 \* 0.335244) = -0.00804337292 aka 0