3 Neural Networks

1. **Give brief definitions of the following terms:**

- **neuron**

  an information processing unit.

- **action potential**

  the signal outputted by a biological neuron.

- **firing rate**

  the number of action potentials emitted during a defined time-period.

- **synapse**

  the connection between two neurons.

- **an artificial neural network**

  a parallel architecture composed of many simple processing elements interconnected to achieve certain collective computational capabilities.

2. **A neuron has a transfer function which is a linear weighted sum of its inputs and an activation function that is the Heaviside function. If the weights are $\mathrm{w} = [0.1, -0.5, 0.4]$, and the threshold zero, what is the output of this neuron when the input is: $\mathrm{x}_1 = [0.1, -0.5, 0.4]^t$ and $\mathrm{x}_2 = [0.1, 0.5, 0.4]^t$?**

Output of neuron is defined as:

$$y = H(\mathbf{wx} - \theta)$$

$$y_1 = H(\mathbf{wx}_1 - \theta) = H\left((0.1 \times 0.1) + (-0.5 \times -0.5) + (0.4 \times 0.4) - 0\right) = H(0.42) = 1$$

$$y_2 = H(\mathbf{wx}_2 - \theta) = H\left((0.1 \times 0.1) + (-0.5 \times 0.5) + (0.4 \times 0.4) - 0\right) = H(-0.08) = 0$$

3. **A Linear Threshold Unit has one input, x_1, that can take binary values. Apply the sequential Delta learning rule so that the output of this neuron, y, is equal to $\bar{x_1}$ (or NOT(x_1)), *i.e.*, such that:**

x_1	y
0	1
1	0

Assume initial values of $\theta = 1.5$ and $w_1 = 2$, and use a learning rate of 1.

Using Augmented notation, $y = H(\mathbf{wx})$ where $\mathbf{w} = [-\theta, w_1]$, and $\mathbf{x} = [1, x_1]^T$.

For the Delta rule, weights are updated such that: $\mathbf{w} \leftarrow \mathbf{w} + \eta(t-y)\mathbf{x}^t$

Initial $\mathbf{w} = [-1.5, 2]$

$\mathbf{x}^t$	t	$y = H(\mathbf{wx})$	$t-y$	$\eta(t-y)\mathbf{x}^t$	$\mathbf{w}$
$(1,0)$	1	$H(-1.5 \times 1 + 2 \times 0) = H(-1.5) = 0$	1	$[1,0]$	$[-0.5,2]$
$(1,1)$	0	$H(-0.5 \times 1 + 2 \times 1) = H(1.5) = 1$	-1	$[-1,-1]$	$[-1.5,1]$
$(1,0)$	1	$H(-1.5 \times 1 + 1 \times 0) = H(-1.5) = 0$	1	$[1,0]$	$[-0.5,1]$
$(1,1)$	0	$H(-0.5 \times 1 + 1 \times 1) = H(0.5) = 1$	-1	$[-1,-1]$	$[-1.5,0]$
$(1,0)$	1	$H(-1.5 \times 1 + 0 \times 0) = H(-1.5) = 0$	1	$[1,0]$	$[-0.5,0]$
$(1,1)$	0	$H(-0.5 \times 1 + 0 \times 1) = H(-0.5) = 0$	0	$[0,0]$	$[-0.5,0]$
$(1,0)$	1	$H(-0.5 \times 1 + 0 \times 0) = H(-0.5) = 0$	1	$[1,0]$	$[0.5,0]$
$(1,1)$	0	$H(0.5 \times 1 + 0 \times 1) = H(0.5) = 1$	-1	$[-1,-1]$	$[-0.5,-1]$
$(1,0)$	1	$H(-0.5 \times 1 - 1 \times 0) = H(-0.5) = 0$	1	$[1,0]$	$[0.5,-1]$
$(1,1)$	0	$H(0.5 \times 1 + -1 \times 1) = H(-0.5) = 0$	0	$[0,0]$	$[0.5,-1]$
$(1,0)$	1	$H(0.5 \times 1 - 1 \times 0) = H(0.5) = 1$	0	$[0,0]$	$[0.5,-1]$

Learning has converged, so required weights are $\mathbf{w} = [0.5, -1]$*, or equivalently* $\theta = -0.5$, $w_1 = -1$.

4. Repeat the above question using the batch Delta learning rule.

Epoch 1, initial $\mathbf{w} = [-1.5, 2]$

$\mathbf{x}^t$	t	$y = H(\mathbf{wx})$	$t-y$	$\eta(t-y)\mathbf{x}^t$	$\mathbf{w}$
$(1,0)$	1	$H(-1.5 \times 1 + 2 \times 0) = H(-1.5) = 0$	1	$[1,0]$	
$(1,1)$	0	$H(-1.5 \times 1 + 2 \times 1) = H(0.5) = 1$	-1	$[-1,-1]$	
			total weight change	$[0,-1]$	$[-1.5,1]$

Epoch 2, initial $\mathbf{w} = [-1.5, 1]$

$\mathbf{x}^t$	t	$y = H(\mathbf{wx})$	$t-y$	$\eta(t-y)\mathbf{x}^t$	$\mathbf{w}$
$(1,0)$	1	$H(-1.5 \times 1 + 1 \times 0) = H(-1.5) = 0$	1	$[1,0]$	
$(1,1)$	0	$H(-1.5 \times 1 + 1 \times 1) = H(-0.5) = 0$	0	$[0,0]$	
			total weight change	$[1,0]$	$[-0.5,1]$

Epoch 3, initial $\mathbf{w} = [-0.5, 1]$

$\mathbf{x}^t$	t	$y = H(\mathbf{wx})$	$t-y$	$\eta(t-y)\mathbf{x}^t$	$\mathbf{w}$
$(1,0)$	1	$H(-0.5 \times 1 + 1 \times 0) = H(-0.5) = 0$	1	$[1,0]$	
$(1,1)$	0	$H(-0.5 \times 1 + 1 \times 1) = H(0.5) = 1$	-1	$[-1,-1]$	
			total weight change	$[0,-1]$	$[-0.5,0]$

Epoch 4, initial $\mathbf{w} = [-0.5, 0]$

$\mathbf{x}^t$	t	$y = H(\mathbf{wx})$	$t-y$	$\eta(t-y)\mathbf{x}^t$	$\mathbf{w}$
$(1,0)$	1	$H(-0.5 \times 1 + 0 \times 0) = H(-0.5) = 0$	1	$[1,0]$	
$(1,1)$	0	$H(-0.5 \times 1 + 0 \times 1) = H(-0.5) = 0$	0	$[0,0]$	
			total weight change	$[1,0]$	$[0.5,0]$

Epoch 5, initial $\mathbf{w} = [0.5, 0]$

$\mathbf{x}^t$	t	$y = H(\mathbf{wx})$	$t - y$	$\eta(t-y)\mathbf{x}^t$	$\mathbf{w}$
$(1,0)$	1	$H(0.5 \times 1 + 0 \times 0) = H(0.5) = 1$	0	$[0,0]$	
$(1,1)$	0	$H(0.5 \times 1 + 0 \times 1) = H(0.5) = 1$	-1	$[-1,-1]$	
			total weight change	$[-1,-1]$	$[-0.5,-1]$

Epoch 6, initial $\mathbf{w} = [-0.5, -1]$

$\mathbf{x}^t$	t	$y = H(\mathbf{wx})$	$t - y$	$\eta(t-y)\mathbf{x}^t$	$\mathbf{w}$
$(1,0)$	1	$H(-0.5 \times 1 - 1 \times 0) = H(-0.5) = 0$	1	$[1,0]$	
$(1,1)$	0	$H(-0.5 \times 1 - 1 \times 1) = H(-1.5) = 0$	0	$[0,0]$	
			total weight change	$[1,0]$	$[0.5,-1]$

Epoch 7, initial $\mathbf{w} = [0.5, -1]$

$\mathbf{x}^t$	t	$y = H(\mathbf{wx})$	$t - y$	$\eta(t-y)\mathbf{x}^t$	$\mathbf{w}$
$(1,0)$	1	$H(0.5 \times 1 - 1 \times 0) = H(0.5) = 1$	0	$[0,0]$	
$(1,1)$	0	$H(0.5 \times 1 - 1 \times 1) = H(-0.5) = 0$	0	$[0,0]$	
			total weight change	$[0,0]$	$[0.5,-1]$

Learning has converged, so required weights are $\mathbf{w} = [0.5, -1]$*, or equivalently* $\theta = -0.5$*,* $w_1 = -1$*.*

5. A Linear Threshold Unit has two inputs, x_1 and x_2, that can take binary values. Apply the sequential Delta learning rule so that the output of this neuron, y, is equal to x_1AND x_2, *i.e.*, such that:

x_1	x_2	y
0	0	0
0	1	0
1	0	0
1	1	1

Assume initial values of $\theta = -0.5$, $w_1 = 1$ and $w_2 = 1$, and use a learning rate of 1.

Using Augmented notation, $y = H(\mathbf{wx})$ *where* $\mathbf{w} = [-\theta, w_1, w_2]$*, and* $\mathbf{x} = [1, x_1, x_2]^t$*.*

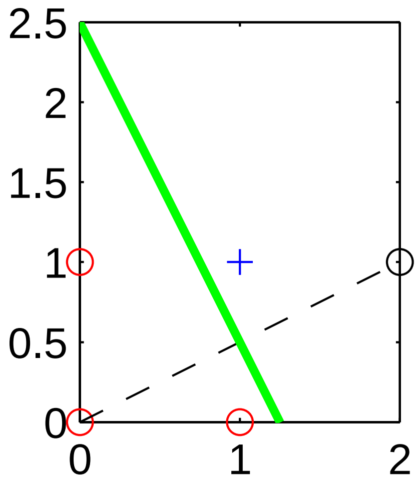
For the Delta rule, weights are updated such that: $\mathbf{w} \leftarrow \mathbf{w} + \eta(t - y)\mathbf{x}^t$

Initial w=[0.5,1,1]

$\mathbf{x}^t$	t	$y = H(\mathbf{wx})$	$t - y$	$\eta(t-y)\mathbf{x}^t$	$\mathbf{w}$
$(1,0,0)$	0	$H(0.5 \times 1 + 1 \times 0 + 1 \times 0) = H(0.5) = 1$	-1	$[-1,0,0]$	$[-0.5,1,1]$
$(1,0,1)$	0	$H(-0.5 \times 1 + 1 \times 0 + 1 \times 1) = H(0.5) = 1$	-1	$[-1,0,-1]$	$[-1.5,1,0]$
$(1,1,0)$	0	$H(-1.5 \times 1 + 1 \times 1 + 0 \times 0) = H(-0.5) = 0$	0	$[0,0,0]$	$[-1.5,1,0]$
$(1,1,1)$	1	$H(-1.5 \times 1 + 1 \times 1 + 0 \times 1) = H(-0.5) = 0$	1	$[1,1,1]$	$[-0.5,2,1]$
$(1,0,0)$	0	$H(-0.5 \times 1 + 2 \times 0 + 1 \times 0) = H(-0.5) = 0$	0	$[0,0,0]$	$[-0.5,2,1]$
$(1,0,1)$	0	$H(-0.5 \times 1 + 2 \times 0 + 1 \times 1) = H(0.5) = 1$	-1	$[-1,0,-1]$	$[-1.5,2,0]$
$(1,1,0)$	0	$H(-1.5 \times 1 + 2 \times 1 + 0 \times 0) = H(0.5) = 1$	-1	$[-1,-1,0]$	$[-2.5,1,0]$
$(1,1,1)$	1	$H(-2.5 \times 1 + 1 \times 1 + 0 \times 1) = H(-1.5) = 0$	1	$[1,1,1]$	$[-1.5,2,1]$
$(1,0,0)$	0	$H(-1.5 \times 1 + 2 \times 0 + 1 \times 0) = H(-1.5) = 0$	0	$[0,0,0]$	$[-1.5,2,1]$
$(1,0,1)$	0	$H(-1.5 \times 1 + 2 \times 0 + 1 \times 1) = H(-0.5) = 0$	0	$[0,0,0]$	$[-1.5,2,1]$
$(1,1,0)$	0	$H(-1.5 \times 1 + 2 \times 1 + 1 \times 0) = H(0.5) = 1$	-1	$[-1,-1,0]$	$[-2.5,1,1]$
$(1,1,1)$	1	$H(-2.5 \times 1 + 1 \times 1 + 1 \times 1) = H(-0.5) = 0$	1	$[1,1,1]$	$[-1.5,2,2]$
$(1,0,0)$	0	$H(-1.5 \times 1 + 2 \times 0 + 2 \times 0) = H(-1.5) = 0$	0	$[0,0,0]$	$[-1.5,2,2]$
$(1,0,1)$	0	$H(-1.5 \times 1 + 2 \times 0 + 2 \times 1) = H(0.5) = 1$	-1	$[-1,0,-1]$	$[-2.5,2,1]$
$(1,1,0)$	0	$H(-2.5 \times 1 + 2 \times 1 + 1 \times 0) = H(-0.5) = 0$	0	$[0,0,0]$	$[-2.5,2,1]$
$(1,1,1)$	1	$H(-2.5 \times 1 + 2 \times 1 + 1 \times 1) = H(0.5) = 1$	0	$[0,0,0]$	$[-2.5,2,1]$
$(1,0,0)$	0	$H(-2.5 \times 1 + 2 \times 0 + 1 \times 0) = H(-2.5) = 0$	0	$[0,0,0]$	$[-2.5,2,1]$
$(1,0,1)$	0	$H(-2.5 \times 1 + 2 \times 0 + 1 \times 1) = H(-1.5) = 0$	0	$[0,0,0]$	$[-2.5,2,1]$

Learning has converged, so required weights are $\mathbf{w} = [-2.5, 2, 1]$, *or equivalently* $\theta = 2.5$, $w_1 = 2$, $w_2 = 1$.

The decision surface looks like this:



6. Consider the following linearly separable data set.

$\mathbf{x}^t$	$class$
$(0,2)$	1
$(1,2)$	1
$(2,1)$	1
$(-3,1)$	0
$(-2,-1)$	0
$(-3,-2)$	0

Apply the Sequential Delta Learning Algorithm to find the parameters of a linear threshold neuron that will correctly classify this data. Assume initial values of $\theta = -1$, $w_1 = 0$ and $w_2 = 0$, and a learning rate of 1.

Using Augmented notation, $y = H(\mathbf{wx})$ where $\mathbf{w} = [-\theta, w_1, w_2]$, and $\mathbf{x} = [1, x_1, x_2]^T$. So initial weight values are $\mathbf{w} = [1, 0, 0]$ and the dataset is:

$\mathbf{x}^t$	t
$(1, 0, 2)$	1
$(1, 1, 2)$	1
$(1, 2, 1)$	1
$(1, -3, 1)$	0
$(1, -2, -1)$	0
$(1, -3, -2)$	0

For the Sequential Delta Learning Algorithm, weights are updated such that: $\mathbf{w} \leftarrow \mathbf{w} + \eta(t - y)\mathbf{x}^t$. Here, $\eta = 1$.

iteration	$\mathbf{w}$	$\mathbf{x}^t$	$y = H(\mathbf{wx})$	t	$\mathbf{w} \leftarrow \mathbf{w} + (t - y)\mathbf{x}^t$
1	$[1, 0, 0]$	$[1, 0, 2]$	1	1	$[1, 0, 0]$
2	$[1, 0, 0]$	$[1, 1, 2]$	1	1	$[1, 0, 0]$
3	$[1, 0, 0]$	$[1, 2, 1]$	1	1	$[1, 0, 0]$
4	$[1, 0, 0]$	$[1, -3, 1]$	1	0	$[1, 0, 0] - [1, -3, 1] = [0, 3, -1]$
5	$[0, 3, -1]$	$[1, -2, -1]$	0	0	$[0, 3, -1]$
6	$[0, 3, -1]$	$[1, -3, -2]$	0	0	$[0, 3, -1]$
7	$[0, 3, -1]$	$[1, 0, 2]$	0	1	$[0, 3, -1] + [1, 0, 2] = [1, 3, 1]$
8	$[1, 3, 1]$	$[1, 1, 2]$	1	1	$[1, 3, 1]$
9	$[1, 3, 1]$	$[1, 2, 1]$	1	1	$[1, 3, 1]$
10	$[1, 3, 1]$	$[1, -3, 1]$	0	0	$[1, 3, 1]$
11	$[1, 3, 1]$	$[1, -2, -1]$	0	0	$[1, 3, 1]$
12	$[1, 3, 1]$	$[1, -3, -2]$	0	0	$[1, 3, 1]$
13	$[1, 3, 1]$	$[1, 0, 2]$	1	1	$[1, 3, 1]$

Learning has converged (we have gone through all the data without needing to update the weights), so required parameters are $\mathbf{w} = (1, 3, 1)$.

7. A negative feedback network has three inputs and two output neurons, that are connected with weights $W = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix}$. Determine the activation of the output neurons after 5 iterations when the input is $\mathbf{x} = (1, 1, 0)^T$, assuming that the output neurons are updated using parameter $\alpha = 0.25$, and the activations of the output neurons are initialised to be all zero.

The activation of a negative feedback network is determined by iteratively evaluating the following equations:

$$\mathbf{e} = \mathbf{x} - \mathbf{W}^T\mathbf{y}$$

$$\mathbf{y} \leftarrow \mathbf{y} + \alpha\mathbf{W}\mathbf{e}$$

iteration	$\mathbf{e}^T$	$(\mathbf{We})^T$	$\mathbf{y}^T$	$(\mathbf{W}^T\mathbf{y})^T$
1	$(1, 1, 0)$	$(2, 2)$	$(0.5, 0.5)$	$(1, 1, 0.5)$
2	$(0, 0, -0.5)$	$(0, -0.5)$	$(0.5, 0.375)$	$(0.875, 0.875, 0.375)$
3	$(0.125, 0.125, -0.375)$	$(0.25, -0.125)$	$(0.5625, 0.34375)$	$(0.90625, 0.90625, 0.34375)$
4	$(0.09375, 0.09375, -0.34375)$	$(0.1875, -0.15625)$	$(0.60938, 0.30469)$	$(0.91406, 0.91406, 0.30469)$
5	$(0.085938, 0.085938, -0.30469)$	$(0.17188, -0.13281)$	$(0.65234, 0.27148)$	$(0.92383, 0.92383, 0.27148)$

So output is $\begin{pmatrix} 0.65234 \\ 0.27148 \end{pmatrix}$

Note, competition results in the first neuron increasing its output, while the output of the second neuron is suppressed. Also, note that the vector $\mathbf{W}^T\mathbf{y}$ *becomes similar to the input* $\mathbf{x}$*.* $\mathbf{W}^T\mathbf{y}$ *converges towards a reconstruction of the input.*

8. Repeat the previous question using a value of $\alpha = 0.5$.

$$\mathbf{e} = \mathbf{x} - \mathbf{W}^T\mathbf{y}$$

$$\mathbf{y} \leftarrow \mathbf{y} + \alpha\mathbf{W}\mathbf{e}$$

iteration	$\mathbf{e}^T$	$(\mathbf{W}\mathbf{e})^T$	$\mathbf{y}^T$	$(\mathbf{W}^T\mathbf{y})^T$
1	$(1, 1, 0)$	$(2, 2)$	$(1, 1)$	$(2, 2, 1)$
2	$(-1, -1, -1)$	$(-2, -3)$	$(0, -0.5)$	$(-0.5, -0.5, -0.5)$
3	$(1.5, 1.5, 0.5)$	$(3, 3.5)$	$(1.5, 1.25)$	$(2.75, 2.75, 1.25)$
4	$(-1.75, -1.75, -1.25)$	$(-3.5, -4.75)$	$(-0.25, -1.125)$	$(-1.375, -1.375, -1.125)$
5	$(2.375, 2.375, 1.125)$	$(4.75, 5.875)$	$(2.125, 1.8125)$	$(3.9375, 3.9375, 1.8125)$

So output is $\begin{pmatrix} 2.125 \\ 1.8125 \end{pmatrix}$

Note, competition results in oscillatory responses. If α *is too large the network becomes unstable. Instability is a common problem with recurrent neural networks.*

9. A more stable method of calculating the activations in a negative feedback network is to use the following update rules:

$$\mathbf{e} = \mathbf{x} \oslash \left[\mathbf{W}^T\mathbf{y}\right]_{\epsilon_2}$$

$$\mathbf{y} \leftarrow [\mathbf{y}]_{\epsilon_1} \odot \tilde{\mathbf{W}}\mathbf{e}$$

where $[v]_\epsilon = \mathtt{max}(\epsilon, v)$; ϵ_1 and ϵ_2 are parameters; $\tilde{\mathbf{W}}$ is equal to $\mathbf{W}$ but with each row normalised to sum to one; and $\oslash$ and $\odot$ indicate element-wise division and multiplication respectively.

This is called Regulatory Feedback or Divisive Input Modulation.

Determine the activation of the output neurons after 5 iterations when the input is $\mathbf{x} = (1, 1, 0)^T$, and $\mathbf{W} = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix}$, assuming that $\epsilon_1 = \epsilon_2 = 0.01$, and the activations of the output neurons are initialised to be all zero.

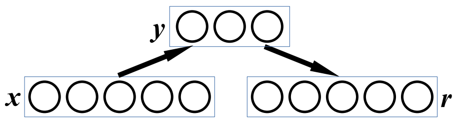
$$\tilde{\mathbf{W}} = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix}$$

iteration	$\mathbf{e}^T$	$(\tilde{\mathbf{W}}\mathbf{e})^T$	$\mathbf{y}^T$	$(\mathbf{W}^T\mathbf{y})^T$
1	$(100, 100, 0)$	$(100, 66.66667)$	$(1, 0.66667)$	$(1.6667, 1.6667, 0.66667)$
2	$(0.6, 0.6, 0)$	$(0.6, 0.4)$	$(0.6, 0.26667)$	$(0.86667, 0.86667, 0.26667)$
3	$(1.1538, 1.1538, 0)$	$(1.1538, 0.76923)$	$(0.69231, 0.20513)$	$(0.89744, 0.89744, 0.20513)$
4	$(1.1143, 1.1143, 0)$	$(1.1143, 0.74286)$	$(0.77143, 0.15238)$	$(0.92381, 0.92381, 0.15238)$
5	$(1.0825, 1.0825, 0)$	$(1.0825, 0.72165)$	$(0.83505, 0.10997)$	$(0.94502, 0.94502, 0.10997)$

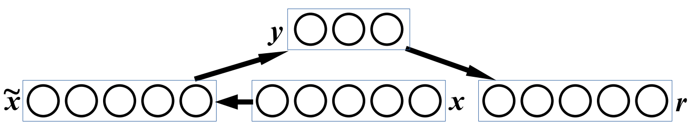
So output is $\begin{pmatrix} 0.83505 \\ 0.10997 \end{pmatrix}$

Note, competition results in the first neuron increasing its output, while the output of the second neuron is suppressed. Also, note that the vector $\mathbf{W}^T\mathbf{y}$ *becomes similar to the input* $\mathbf{x}$*.* $\mathbf{W}^T\mathbf{y}$ *converges towards a reconstruction of the input.*

10. The figure below shows an autoencoder neural network.



Draw a diagram of a de-noising autoencoder and briefly explain how a de-noising autoencoder is trained.



The network is trained so that the output, $\mathbf{r}$*, reconstructs the input,* $\mathbf{x}$*. However, before encoding is performed the input is corrupted with noise. This mitigates overfitting.*