The University of Adelaide
School of Computer Science

Artificial Intelligence, 2019
Worksheet 3

Question 1 (Question 20.21 in AIMA 2ed)
Consider the problem of separating N data points into positive and negative examples using a linear separator. Clearly, this can always be done for $N = 2$ points on a line of dimension $d = 1$, regardless of how the points are labeled or where they are located (unless the points are in the same place).

1. Show that it can always be done for $N = 3$ points on a plane of dimension $d = 2$, unless they are collinear.
2. Show that it cannot always be done for $N = 4$ points on a plane of dimension $d = 2$.
3. Show that it can always be done for $N = 4$ points in a space of dimension $d = 3$, unless they are coplanar.
4. Show that it cannot always be done for $N = 5$ points in a space of dimension $d = 3$.

Question 2 (Question 20.11 in AIMA 2ed)
Construct by hand a neural network that computes the XOR function of two inputs. Make sure to specify what sort of units you are using.

Question 3 (Question 20.19 in AIMA 2ed)
Suppose that that a training set contains only a single example, repeated 100 times. In 80 of the 100 cases, the single output value is 1; in the other 20, it is 0. What will a back-propagation network predict for this example, assuming it has been trained and reaches a global optimum? (Hint: to find the global optimum, differentiate the error function and set to zero.)

Question 4

Let $X = \{x_1, x_2, \dots x_n\}$ be a dataset of n samples with 2 features ($x \in \mathbb{R}^2$). The samples are categorized by 2 labels $y \in \{1, 0\}$, and we want to train a neural network to perform binary classification on the dataset using the network shown below in Figure 1. Derive the gradient descent update function for the weights of the network (i.e. θ_{ji} and θ_{kj}) with the assumption that the learning rate is α. For the loss function, use l_1 loss and for the activation function use a ReLu. Note that the derivative of the l_1 loss function is just the sign of the loss. The derivative of ReLu is 1 when the input is greater than zero, and zero otherwise which can be written as a Heaviside step function $H(x) = \begin{cases} 1, \text{if} x > 0 \\ 0, x \leq 0 \end{cases}$

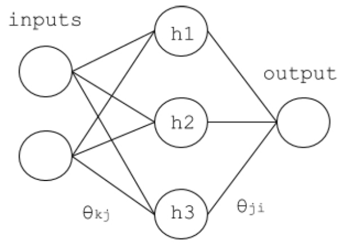

Figure 1: Network architecture.