Solutions to Chapter 9
NEURAL NETWORKS
Prepared by James Cunningham, Graduate Assistant

2. **Clearly describe each of these characteristics of a neural network:**

   a. **Layered**
   The neural network is composed to two or more layers of nodes and typically includes an input layer, one or more hidden layers, and an output layer.

   b. **Feedforward**
   This characteristic implies that information flows in a single direction and does not loop or cycle.

   c. **Completely connected**
   Each node in a given layer connects to all nodes in the next layer.

3. **What is the sole function of the nodes in the input layer?**
   The nodes in the input layer accept the data from the input variables. The data values are passed directly to nodes in the hidden layer.

4. **Should we prefer a large hidden layer or a small one? Describe the benefits and drawbacks of each.**
   As the number of nodes in the hidden layer becomes overly large, the neural network becomes increasingly likely to overfit the data. In contrast, using too few hidden nodes may lead to poor accuracy. Therefore, a balance must be struck when configuring the hidden layer that strives to deliver the best accuracy while also generalizing to the data set.

6. Explain why the updating term for the current weight includes the negative of the sign of the derivative (slope).

The gradient decent method is used to determine the direction (negative or positive) that each of the weights for the vector ***w*** should be adjusted, together, to decrease SSE. Because each partial derivative $\frac{\partial SSE}{\partial w_i}$ is the slope of the SSE curve at w_i, we negate the sign of slope, such that adjusting each w_i moves it towards the optimal value w_i^*, the value that minimizes SSE.

7. Adjust the weights W0B, W1B, W2B, and W3B from the example on back-propagation in the text.

We are given that $\delta_B = -0.0011$ and $\eta = 0.1$, therefore

For W_{1B} we get

$$\Delta W_{1B} = \eta \delta_B x_1 = (0.1)(-0.0011)(0.4) = -0.000044$$

$$W_{1B,NEW} = W_{1B,CURRENT} + \Delta W_{1B} = 0.9 - 0.000044 = 0.899956$$

For W_{2B} we get

$$\Delta W_{2B} = \eta \delta_B x_2 = (0.1)(-0.0011)(0.2) = -0.000022$$

$$W_{2B,NEW} = W_{2B,CURRENT} + \Delta W_{2B} = 0.8 - 0.000022 = 0.799978$$

For W_{3B} we get

$$\Delta W_{3B} = \eta \delta_B x_3 = (0.1)(-0.0011)(0.7) = -0.000077$$

$$W_{3B,NEW} = W_{3B,CURRENT} + \Delta W_{3B} = 0.4 - 0.000077 = 0.399923$$

For W_{0B} we get

$$\Delta W_{0B} = \eta \delta_B x_0 = (0.1)(-0.0011)(1) = -0.00011$$

$$W_{0B,NEW} = W_{0B,CURRENT} + \Delta W_{0B} = 0.7 - 0.00011 = 0.69989$$

8. Refer to Exercise 7. Show that the adjusted weights result in a smaller prediction error.

Now that Exercise 7 is complete we have all the adjusted weights after back propagation has occurred and so we compute a new predicted valued:

$$net_A = W_{0A}(1) + W_{1A}x_1 + W_{2A}x_2 + W_{3A}x_3$$

$$(0.499877)(1)+(0.5999508)(0.4) + (0.7999754)(0.2) + (0.5999139)(0.7) = 1.3198$$

$$f(net_A) = \frac{1}{1+e^{-1.3198}} = 0.7891$$

$$net_B = W_{0B}(1) + W_{1B}x_1 + W_{2B}x_2 + W_{3B}x_3$$

$$(0.69989)(1)+(0.899956)(0.4) + (0.799978)(0.2) + (0.399923)(0.7) = 1.4998$$

$$f(net_A) = \frac{1}{1+e^{-1.4998}} = 0.8175$$

$$net_Z = W_{0Z}(1) + W_{AZ}x_{AZ} + W_{BZ}x_{BZ}$$

$$(0.49918)(1)+(0.899353)(0.7891) + (0.89933)(0.8175) = 1.9441$$

$$f(net_Z) = \frac{1}{1+e^{-1.9441}} = 0.8745$$

Now, the prediction error = actual – output = 0.8 – 0.8745 = –0.0745. Recall that the initial prediction, before back propagation was 0.8 – 0.8750 = –0.075. Thus, the prediction error after back propagation has indeed decreased.

9. True or false: Neural networks are valuable because of their capacity for always finding the global minimum of the SSE.

A neural network model is not guaranteed to find the global minimum of the SSE; however, the algorithm will find a local minimum that often represents a good solution to the problem at hand. Learning rate and momentum parameters are often adjusted to search local minima for near-optimal solutions.

10. Describe the benefits and drawbacks of using large or small values for the learning rate.

Choosing very small values for the learning rate will tend to cause the algorithm to take an unacceptably long time to converge on a solution. Conversely, setting the learning rate to too large a value may cause the neural network oscillate and have trouble converging a solution. Ideally, the neural network should begin by using a larger value for the learning rate, and gradually decrease the value as the algorithm approaches a solution.