CS557 Homework One (Ungraded)

Andrew Struthers

April 2023

1. Train a discrete bipolar perceptron using the Perceptron Learning Rule, taking:

   - Learning rate = 1
   - $\mathbf{w}_1 = [0, 1, 0]^t$ (initial weights)
   - $(\mathbf{x}_1 = [2, 1, -1]^t, t_1 = -1)$
   - $(\mathbf{x}_2 = [0, -1, -1]^t, t_2 = 1)$

   Repeat the $(\mathbf{x}_1, t1), (\mathbf{x}_2, t_2)$ sequence until you obtain the correct outputs. Do not use a computer (except for typing).

   (a) First, we will calculate net^1 with $\mathbf{w}_1$ and $\mathbf{x}_1$:

   $$\begin{aligned} net^1 &= \mathbf{w}_1^t \mathbf{x}_1 \\ &= \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 2 \\ 1 \\ -1 \end{bmatrix} \\ &= (0)(2) + (1)(1) + (0)(-1) \\ &= 0 + 1 + 0 \\ &= 1 \end{aligned} \tag{1}$$

   Now we can test this result against the bipolar activation function:

   $$sgn(1) = 1 \tag{2}$$

Since this does not equal t_1, we need to adjust the weights based off of the error:

$$\begin{aligned}
\mathbf{w}_2 &= \mathbf{w}_1 + n \cdot (t_1 - y_1)\mathbf{x}_1 \\
&= \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} + 1(-1-1) \cdot \begin{bmatrix} 2 \\ 1 \\ -1 \end{bmatrix} \\
&= \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} - 2 \cdot \begin{bmatrix} 2 \\ 1 \\ -1 \end{bmatrix} \\
&= \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} + \begin{bmatrix} -4 \\ -2 \\ 2 \end{bmatrix} \\
&= \begin{bmatrix} -4 \\ -1 \\ 2 \end{bmatrix}
\end{aligned} \tag{3}$$

Now that we have $\mathbf{w}_2$, we can move on to net^2 and $\mathbf{x}_2$.

$$\begin{aligned}
net^2 &= \mathbf{w}_2^t \mathbf{x}_2 \\
&= \begin{bmatrix} -4 & -1 & 2 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ -1 \\ -1 \end{bmatrix} \\
&= (-4)(0) + (-1)(-1) + (2)(-1) \\
&= 0 + 1 - 2 \\
&= -1
\end{aligned} \tag{4}$$

The result of the bipolar activation function

$$sgn(-1) = -1 \tag{5}$$

does not coincide with our desired t_2, so we must calculate new weights based off of the new error.

$$\begin{aligned}
\mathbf{w}_3 &= \mathbf{w}_2 + n \cdot (t_2 - y_2)\mathbf{x}_2 \\
&= \begin{bmatrix} -4 \\ -1 \\ 2 \end{bmatrix} + 1(1-(-1)) \cdot \begin{bmatrix} 0 \\ -1 \\ -1 \end{bmatrix} \\
&= \begin{bmatrix} -4 \\ -1 \\ 2 \end{bmatrix} + 2 \cdot \begin{bmatrix} 0 \\ -1 \\ -1 \end{bmatrix} \\
&= \begin{bmatrix} -4 \\ -1 \\ 2 \end{bmatrix} + \begin{bmatrix} 0 \\ -2 \\ -2 \end{bmatrix} \\
&= \begin{bmatrix} -4 \\ -3 \\ 0 \end{bmatrix}
\end{aligned} \tag{6}$$

We have now done one full iteration of $\mathbf{x}_1$ and $\mathbf{x}_2$, but we do not yet have our desired results. We will now rerun the calculations with our last calculated weights, and repeat this process until we get the results we want.

$$\begin{aligned} net^1 &= \mathbf{w}_3^t \mathbf{x}_1 \\ &= \begin{bmatrix} -4 & -3 & 0 \end{bmatrix} \cdot \begin{bmatrix} 2 \\ 1 \\ -1 \end{bmatrix} \\ &= (-4)(2) + (-3)(1) + (0)(-1) \\ &= -8 - 3 + 0 \\ &= -11 \end{aligned} \tag{7}$$

Using the bipolar activation function, we get

$$sgn(-11) = -1 \tag{8}$$

which is the result we wanted from t_1. Since this weight vector has given us the desired outcome, no correction is necessary. Now to check net^2:

$$\begin{aligned} net^2 &= \mathbf{w}_3^t \mathbf{x}_2 \\ &= \begin{bmatrix} -4 & -3 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ -1 \\ -1 \end{bmatrix} \\ &= (-4)(0) + (-3)(-1) + (0)(-1) \\ &= 0 + 3 + 0 \\ &= 3 \end{aligned} \tag{9}$$

With the bipolar activation function

$$sgn(3) = 1 \tag{10}$$

we see that this returns t_2, so now we have a weight vector such that we get t_1 and t_2 from our input vectors. We have repeated the Perceptron Learning Rule using the bipolar activation function and a learning rate of 1 to find the weights that return the correct outputs. Therefore, the solution is:

$$\boxed{\mathrm{w} = \begin{bmatrix} -4 \\ -3 \\ 0 \end{bmatrix}}$$