CSE - 5526

Homework 2

Submitted by

Utkarsh Pratap Singh Jadon (500711257)
jadon.1@osu.edu

Contents

1 Question 1

For the following training samples:

$\mathbf{x}_1 = (0,0)^T \in C_1$
$\mathbf{x}_2 = (0,1)^T \in C_1$
$\mathbf{x}_3 = (1,0)^T \in C_2$
$\mathbf{x}_4 = (1,1)^T \in C_2$

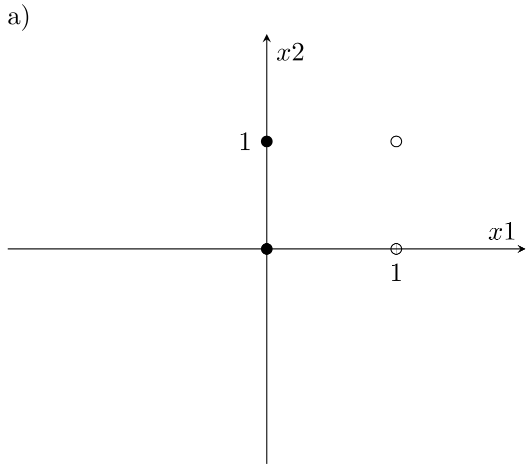
(a) Plot them in input space.

(b) Apply the perceptron learning rule to the above samples one-at-a-time to obtain weights that separate the training samples. Set η to 0.5. Work in the space with the bias as another input element. Use $\mathbf{w}(0) = (0,0,0)^T$. Show values for $\mathbf{w}$ after it is updated for each training sample.

(c) Write the expression for the resulting decision boundary from (b).

(d) For $\mathbf{x}_2, \mathbf{x}_3 \in C_1$ and $\mathbf{x}_1, \mathbf{x}_4 \in C_2$, describe your observation when you apply the perceptron learning rule following the same procedure as in (a)-(c).

Solution
a)



b) Given that $\eta = 0.5$ and $w = (0,0,0)$ including bias. Let us assume C_1 as -1 and C_2 as 1.

Weight update as per the perceptron rule is given by:

$$w(n+1) = w(n) + \nabla w(n) = w(n) + \eta[d(n) - y(n)]x(n) \tag{1}$$

Since a signum activation function is used, following table is generated which consists of old weights, input, actual output (Y), desired output (D), Error, and new weights. Last bit in **Input** represents the bias, and last bit in **Old Weight** represents the weight for that bias.

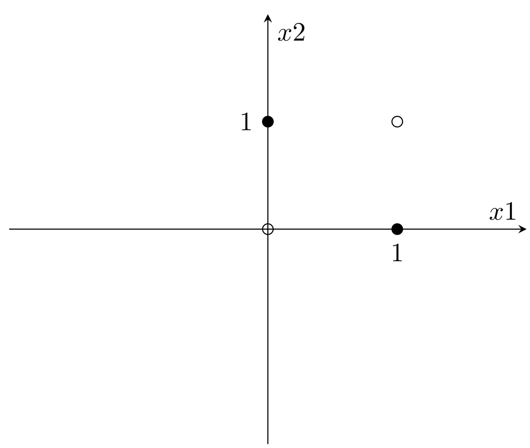
Table 1: Weight Updates fort part (b)

Old Weight	**Input**	**Y**	**D**	**D-Y**	**New Weight**
(0, 0, 0)	(0, 0, 1)	1	-1	-2	(0, 0, 0) + 0.5(-2)(0, 0, 1) = (0, 0, -1)
(0, 0, -1)	(0, 1, 1)	-1	-1	0	(0, 0, -1) + 0.5(0)(0, 1, 1) = (0, 0, -1)
(0, 0, -1)	(1, 0, 1)	-1	1	2	(0, 0, -1) + 0.5(2)(1, 0, 1) = (1, 0, 0)
(1, 0, 0)	(1, 1, 1)	1	1	0	(1, 0, 0) + 0.5(0)(1, 1, 1) = (1, 0, 0)
(1, 0, 0)	(0, 0, 1)	1	-1	-2	(1, 0, 0) + 0.5(-2)(0, 0, 1) = (1, 0, -1)
(1, 0, -1)	(0, 1, 1)	-1	-1	0	(1, 0, -1) + 0.5(0)(0, 1, 1) = (1, 0, -1)
(1, 0, -1)	(1, 0, 1)	1	1	0	(1, 0, -1) + 0.5(0)(1, 0, 1) = (1, 0, -1)
(1, 0, -1)	(1, 1, 1)	1	1	0	(1, 0, -1) + 0.5(0)(1, 1, 1) = (1, 0, -1)
(1, 0, -1)	(0, 0, 1)	-1	-1	0	(1, 0, -1) + 0.5(0)(0, 0, 1) = (1, 0, -1)

c) Decision boundary is given by: $w^T x = 0$
$w_1 x_1 + w_2 x_2 + w_3 b \Rightarrow (1)\mathrm{x}_1 + (0)x_2 + (-1)(1) \Rightarrow (1)\mathrm{x}_1 + 0 - 1 = 0$

Therefore, the decision boundary is given by: $x_1 = 1$

d) For $x2$, $x3 \in \mathrm{C}_1$ and $x1$, $x4 \in \mathrm{C}_2$, following is the plot in input space:



Given that $\eta = 0.5$ and w = (0,0,0) including bias. Let us assume C_1 as 1 and C_2 as -1.

Weight update as per the perceptron rule is given by equation 1, and following table is generated which consists of old weights, input, actual output (Y), desired output (D), Error, and new weights. Last bit in **Input** represents the bias, and last bit in **Old Weight** represents the weight for that bias.

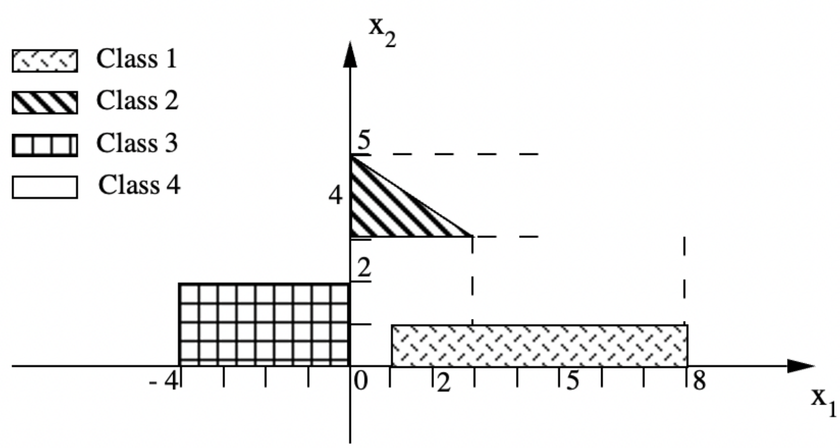
Table 2: Weight Updates for part (d)

Old Weight	**Input**	**Y**	**D**	**D-Y**	**New Weight**
(0, 0, 0)	(0, 0, 1)	1	-1	-2	(0, 0, 0) + 0.5(-2)(0, 0, 1) = (0, 0, -1)
(0, 0, -1)	(0, 1, 1)	-1	1	2	(0, 0, -1) + 0.5(2)(0, 1, 1) = (0, 1, 0)
(0, 1, 0)	(1, 0, 1)	1	1	0	(0, 1, 0) + 0.5(0)(1, 0, 1) = (0, 1, 0)
(0, 1, 0)	(1, 1, 1)	1	-1	-2	(0, 1, 0) + 0.5(-2)(1, 1, 1) = (-1, 0, -1)
(-1, 0, -1)	(0, 0, 1)	-1	-1	0	(-1, 0, -1) + 0.5(0)(0, 0, 1) = (-1, 0, -1)
(-1, 0, -1)	(0, 1, 1)	-1	1	2	(1, 0, -1) + 0.5(2)(0, 1, 1) = (-1, 1, 0)
(-1, 1, 0)	(1, 0, 1)	-1	1	2	(-1, 1, 0) + 0.5(2)(1, 0, 1) = (0, 1, 1)
(0, 1, 1)	(1, 1, 1)	1	-1	-2	(0, 1, 1) + 0.5(-2)(1, 1, 1) = (-1, 0, 0)
(-1, 0, 0)	(0, 0, 1)	1	-1	-2	(-1, 0, 0) + 0.5(-2)(0, 0, 1) = (-1, 0, -1)
(-1, 0, -1)	(0, 1, 1)	-1	1	2	(-1, 0, -1) + 0.5(2)(0, 1, 1) = (-1, 1, 0)
(-1, 1, 0)	(1, 0, 1)	-1	1	2	(-1, 1, 0) + 0.5(2)(1, 0, 1) = (0, 1, 1)
(0, 1, 1)	(1, 1, 1)	1	-1	-2	(0, 1, 1) + 0.5(-2)(1, 1, 1) = (-1, 0, 0)
(-1, 0, 0)	(0, 0, 1)	1	-1	-2	(-1, 0, 0) + 0.5(-2)(0, 0, 1) = (-1, 0, -1)

Here, we can see that the weights don't converge and PLA runs indefinitely. It can be seen visually from the graph as well that the data is now not linearly separable. Hence, the perceptron learning rule is violated here.

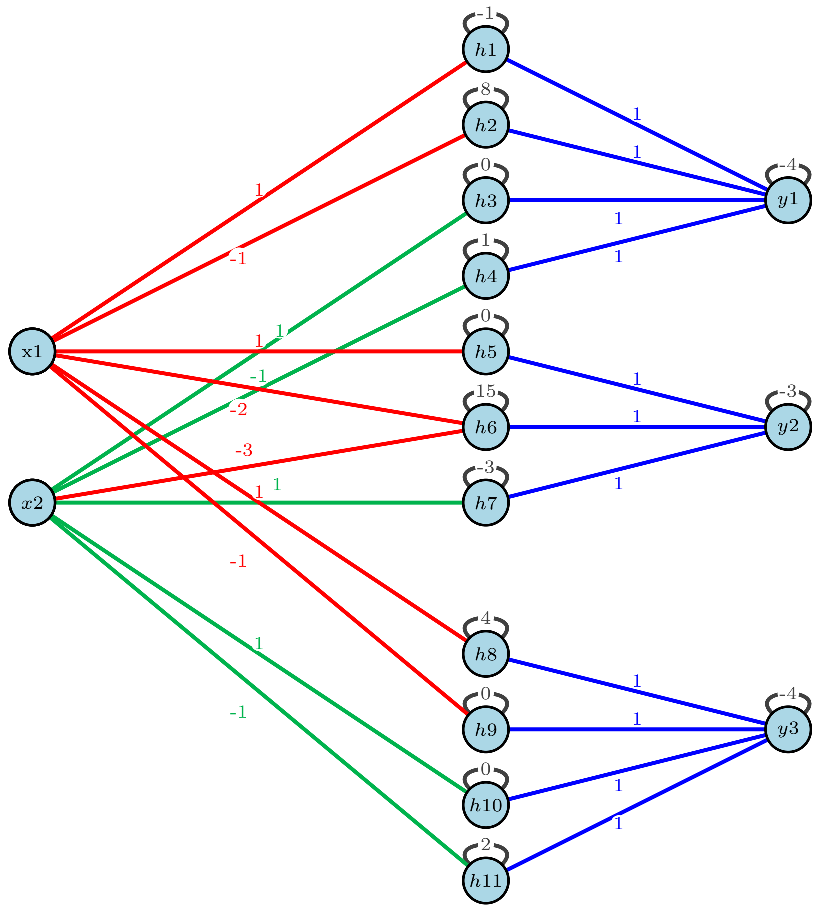
2 Question 2

The following figure shows the decision regions of four classes. Design a neural network that correctly classifies points from each class, using a network of M-P neurons with three output units. For class $i\,(1 \leq i \leq 3)$, classification requires that $y_i = 1$, while $y_j = -1$ for $j \neq i$; Class 4 is recognized when $y_i = -1$ for $1 \leq i \leq 3$ (HINT: try a two-layer feedforward network). Be sure to show the network and non-zero weight values.



Solution

From the given figure, we can deduce that the ranges for three classes is given as; $C_1 : x1 \in [1, 8]$ & $x2 \in [0, 1]$, $C_2 : x1 \in [0, 3]$ & $x2 \in [3, 5]$ & $15 - 2x_1 - 3x_2 >= 0$, and $C_3 : x1 \in [-4, 0]$ & $x2 \in [0, 2]$



First layer is the input layer consisting of two neurons for given two inputs x_1 and x_2. Middle layer is a hidden layer consisting of 12 neurons. And final layer is the output layer consisting of 3 neurons.

In the first set of 4 neurons in hidden layer:
h_1 will output 1 when $x_1 >= 1$ and -1 otherwise
h_2 will output 1 when $x_1 <= 8$ and -1 otherwise
h_3 will output 1 when $x_2 >= 0$ and -1 otherwise
h_4 will output 1 when $x_2 <= 1$ and -1 otherwise

From above conditions, it can be inferred that the first neuron in output layer, y_1 will output 1 when the first four neurons in hidden layer, h_1, h_2, h_3, and h_4 will all output 1, and -1 otherwise. This satisfies the condition for region $C_1 : x1 \in [1, 8]$ & $x2 \in [0, 1]$.

Similarly, next three neurons in hidden layer will satisfy the condition $C_2 : x1 \in [0, 3]$ & $x2 \in [3, 5]$ & $15 - 2x_1 - 3x_2 >= 0$. And last four neurons in hidden layer will satisfy the condition $C_3 : x1 \in [-4, 0]$ & $x2 \in [0, 2]$.

In all other cases, $y_1 = y_2 = y_3 =$ -1, which is the condition for C_4.

3 Question 3

Given the following inputs, X, and corresponding desired outputs, D:

$X = \{-0.5, -0.2, -0.1, 0.3, 0.4, 0.5, 0.7\}$
$D = \{-1, 1, 2, 3.2, 3.5, 5, 6\}$

write down the cost function with respect to w (setting the bias to zero). Compute the gradient at the point w = 2 using both direct differentiation and LMS approximation (average for all data samples in both cases), and see if they agree.

Solution

Error term E is given by:

$$E = \frac{1}{2}\sum_{i=1}^{N}(d_i - y_i)^2 = \frac{1}{2}\sum_{i=1}^{N}(d_i^2 + x_i^2 w^2 - 2d_i x_i w) \tag{2}$$

where $y = w^T x$

We substitute the values for x and d from the question, and reduce E to a function of w, setting bias as zero, as follows:

$$E = \frac{1}{2}[1.29w^2 - 18.32w + 89.49] \tag{3}$$

Mean Error can be obtained by dividing the Error obtained above, by number of samples, i.e. N = 7:

$$\overline{E} = \frac{1}{14}[1.29w^2 - 18.32w + 89.49] \tag{4}$$

Gradient using direct differentiation

Direct differentiation of mean error $\overline{E}$ obtained from 4 is given as:

$$\frac{\mathrm{d}\overline{E}}{\mathrm{d}w} = \frac{1}{14}[(1.29 * 2)w - 18.32]$$

$$\frac{\mathrm{d}\overline{E}}{\mathrm{d}w} = \frac{1}{14}[2.58w - 18.32]$$

At w = 2

$$\frac{\mathrm{d}\overline{E}}{\mathrm{d}w} = \frac{1}{14}[2.58 * 2 - 18.32]$$

$$\frac{\mathrm{d}\overline{E}}{\mathrm{d}w} = \textbf{- 0.94}$$

Gradient using LMS approximation

LMS approximation of E obtained in 2 is given by:

$$\nabla E_w = \frac{\mathrm{d}E}{\mathrm{d}w} = \frac{1}{2}\frac{\mathrm{d}(d-y)^2}{\mathrm{d}w} = \frac{1}{2}\frac{\mathrm{d}(d-wx)^2}{\mathrm{d}w}$$

$$\Rightarrow \nabla E_w = (\mathrm{d} - \mathrm{wx})(-\mathrm{x})$$

Following table is generated calculating error value for each sample at $w = 2$, where X is input, Y is actual output, and D is desired output:

Table 3: LMS Approximation

X	Y	D	D - Y	∇E
-0.5	-1	-1	0	0
-0.2	-0.4	1	1.4	0.28
-0.1	-0.2	2	2.2	0.22
0.3	0.6	3.2	2.6	-0.78
0.4	0.8	3.5	2.7	-1.08
0.5	1	5	4	-2
0.7	1.4	6	4.6	-3.22

Therefore,

$\nabla\overline{(E)} = (1/7) * [0 + 0.28 + 0.22 - 0.78 - 1.08 - 2 - 3.22]$

$\Rightarrow \nabla\overline{(E)} =$ **- 0.94**

From results of direct differentiation and LMS approximation, we can see that both results agree with each other, and compute same gradient value at point $w = 2$.