Assignment8 – Learning

Given: June 20   Due: June 25

Problem 8.1 (Support Vectors)

Consider the following 2-dimensional dataset

support vector	*classification*
$\mathbf{x}_1 = \langle 0, 0 \rangle$	$\mathbf{y}_1 = -1$
$\mathbf{x}_2 = \langle 0, 0.5 \rangle$	$\mathbf{y}_2 = -1$
$\mathbf{x}_3 = \langle 0.5, 0 \rangle$	$\mathbf{y}_3 = -1$
$\mathbf{x}_4 = \langle 1, 1 \rangle$	$\mathbf{y}_4 = 1$
$\mathbf{x}_5 = \langle 2, 2 \rangle$	$\mathbf{y}_5 = -1$

1. Give a *linear separator* in the form $h(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b$ for the dataset containing only the examples for $\mathbf{x}_1$ to $\mathbf{x}_4$.

   Solution: Many solutions, e.g., $\mathbf{w} = \langle 1, 1 \rangle$ and $b = -1$.

2. Explain informally why no linear separator exists for the full dataset of all 5 vectors.

   Solution: The points $\mathbf{x}_1, \mathbf{x}_4, \mathbf{x}_5$ lie on a line and the middle one has a different classification than the others. No line can have $\mathbf{x}_1$ and $\mathbf{x}_5$ on one side and $\mathbf{x}_4$ on the other.

3. Transform the dataset into a 3-dimensional dataset by applying the function $F(\langle u, v \rangle) = \langle u^2, v^2, u + v \rangle$.

   Solution:

support vector $\mathbf{x}$	$F(\mathbf{x})$	*classification*
$\mathbf{x}_1$	$\langle 0, 0, 0 \rangle$	$\mathbf{y}_1 = -1$
$\mathbf{x}_2$	$\langle 0, 0.25, 0.5 \rangle$	$\mathbf{y}_2 = -1$
$\mathbf{x}_3$	$\langle 0.25, 0, 0.5 \rangle$	$\mathbf{y}_3 = -1$
$\mathbf{x}_4$	$\langle 1, 1, 2 \rangle$	$\mathbf{y}_4 = 1$
$\mathbf{x}_5$	$\langle 4, 4, 4 \rangle$	$\mathbf{y}_5 = -1$

4. Give a *linear separator* for the transformed full dataset in the form $h(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b$.

   Solution: Many solutions, e.g., $\mathbf{w} = \langle -1, -1, 2 \rangle$ and $b = -1$.

Problem 8.2 (Weight Updates)

We're trying to find a linear separator. Our examples are the set

Example number	$\mathbf{x}_1$	$\mathbf{x}_2$	y
1	2	0	2
2	3	1	2

Our hypothesis space contains the functions $h_{\mathbf{w}}(\mathbf{x}) = A(\mathbf{w}\cdot\mathbf{x})$ for 2+1-dimensional vectors $\mathbf{w}, \mathbf{x}$ (using the trick $\mathbf{x}_0 = 1$ to allow for the constant term $\mathbf{w}_0$) and some fixed function A.

As the initial weights, we use $\mathbf{w}_0 = \mathbf{w}_1 = \mathbf{w}_2 = 0$.

For each of the following cases, iterate the respective weight update rule once for each example (using the examples in the order listed). Use learning rate $\alpha = 1$.

1. Using the threshold function $A(z) = \mathcal{T}(z)$, i.e., $A(z) = 1$ if $z > 0$ and $A(z) = 0$ otherwise. Here we cannot do gradient descent, so we have to use the perceptron learning rule.

Solution: The update rule is $\mathbf{w}_i \leftarrow \mathbf{w}_i + \alpha(y - h_{\mathbf{w}}(\mathbf{x}))\mathbf{x}_i$. Using the examples, we obtain:

- Example 1: $y - h_{\mathbf{w}}(\mathbf{x}) = 2 - \mathcal{T}((0,0,0)\cdot(1,2,0)) = 2$, i.e., $\mathbf{w}_i \leftarrow \mathbf{w}_i + 1\mathbf{x}_i$. Thus, $\mathbf{w} \leftarrow (2,4,0)$.
- Example 2: $y - h_{\mathbf{w}}(\mathbf{x}) = 2 - \mathcal{T}((2,4,0)\cdot(1,3,1)) = 1$, i.e., $\mathbf{w}_i \leftarrow \mathbf{w}_i + 1\mathbf{x}_i$. Thus, $\mathbf{w} \leftarrow (3,7,1)$.

---

2. Using the logistic function $A(z) = 1/(1+e^{-x})$. Here we use gradient descent.

Solution: The update rule is $\mathbf{w}_i \leftarrow \mathbf{w}_i + \alpha(y - h_{\mathbf{w}}(\mathbf{x}))h_{\mathbf{w}}(\mathbf{x})(1 - h_{\mathbf{w}}(\mathbf{x}))\mathbf{x}_i$. Using the examples, we obtain:

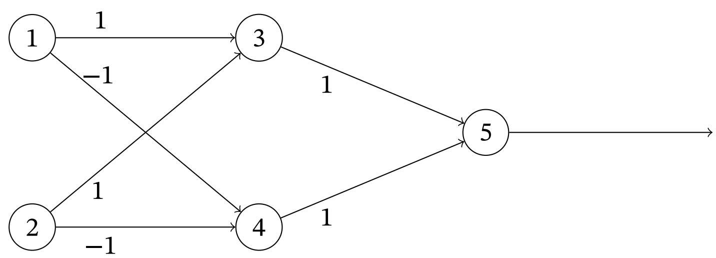
- Example 1: $h_{\mathbf{w}}(\mathbf{x}) = 1/(1+e^0) = 1/2$, i.e., $\mathbf{w}_i \leftarrow \mathbf{w}_i + 1/4(2 - 1/2)\mathbf{x}_i$. Thus, $\mathbf{w} \leftarrow (3/8, 3/4, 0)$.
- Example 2: $h_{\mathbf{w}}(\mathbf{x}) = 1/(1+e^{-((3/8,3/4,0)\cdot(1,3,1))}) = 1/(1+e^{-21/8})$, i.e., (after rounding) $\mathbf{w}_i \leftarrow \mathbf{w}_i + 0.07\mathbf{x}_i$. Thus, $\mathbf{w} \leftarrow (0.44, 0.95, 0.07)$.

Problem 8.3 (XOR Neural Network)

Consider the following neural network with

- inputs a_1 and a_2
- units 3, 4, 5 with activation functions such that $a_i \leftarrow \begin{cases} 1 & \text{if } \Sigma_j w_{ji} a_j > b_i \\ 0 & \text{otherwise} \end{cases}$

- weights w_{ij} as given by the labels on the edges



1. Assume $b_3 = b_4 = b_5 = 0$ and inputs $a_1 = a_2 = 1$. What are the resulting activations a_3, a_4, and a_5?

Solution: $a_3 = 1$, $a_4 = 0$, $a_5 = 1$

---

2. Choose appropriate values for b_3, b_4, and b_5 such that the network *implements* the XOR function.

Solution: E.g., $b_3 = 0.5$, $b_4 = -1.5$, $b_5 = 1.5$. More generally, any values work that satisfy $0 \leq b3 < 1$, $-2 \leq b4 < -1$, and $1 \leq b5 < 2$.
