ECE/CS 559 - Fall 2016 - Midterm Part #1.

Full Name: **ID Number:**

- **Q1 (24 pts):** In the following, we follow the convention that $x_0 = 1$.

  Recall that the step-activation function $u : \mathbb{R} \to \{0,1\}$ is defined as $u(v) = 1$ if $v \geq 0$, and $u(v) = 0$, otherwise. Then, for n inputs $x_1, \ldots, x_n$, a **perceptron** can be defined via the input-output relationship

  $$y' = u\left(\textstyle\sum_{i=0}^{n} w_i x_i\right) = u(w_0 + w_1 x_1 + \cdots + w_n x_n),$$

  where y' is the perceptron output, $w_1, \ldots, w_n$ are the synaptic weights, and w_0 is the bias term.

  We define a new type of neuron, namely, a **sauron**, via the input-output relationship

  $$y = u\left(\textstyle\prod_{i=0}^{n} (w_i + x_i)\right) = u((w_0 + 1)(w_1 + x_1) \cdots (w_n + x_n)),$$

  where y is called the sauron output.

  Let the real number 1 represent a `TRUE`, and the real number 0 represent a `FALSE`.

  (a) **(8 pts):** Let $n = 1$. Does there exist w_0, w_1 such that $y = 1 - x_1$ for $x_1 \in \{0,1\}$? In other words, can a single sauron implement the `NOT` gate? If your answer is "Yes," find specific w_0, w_1 such that the sauron implements the `NOT` gate. If your answer is "No," prove that no choice for w_0, w_1 can result in a sauron that implements the `NOT` gate.

  (b) **(8 pts):** Let $n = 2$. Does there exist w_0, w_1, w_2 such that $y = x_1 x_2$ for $x_1, x_2 \in \{0,1\}$? In other words, can a single sauron implement the `AND` gate? Justify your answer as in (a).

  (c) **(8 pts):** Let $n = 2$. Does there exist w_0, w_1, w_2 such that $y = ((x_1 + x_2) \mod 2)$ for $x_1, x_2 \in \{0,1\}$? In other words, can a single sauron implement the `XOR` gate? Justify your answer as in (a).

- **Q2 (26 pts):** In the following, consider only neurons with the step-activation function $u(\cdot)$.

  (a) **(13 pts):** Let $\mathcal{C}_0 = \{\left[\begin{smallmatrix}1\\1\end{smallmatrix}\right], \left[\begin{smallmatrix}-1\\1\end{smallmatrix}\right], \left[\begin{smallmatrix}0\\4\end{smallmatrix}\right]\}$ and $\mathcal{C}_1 = \{\left[\begin{smallmatrix}0\\2\end{smallmatrix}\right], \left[\begin{smallmatrix}0\\-1\end{smallmatrix}\right], \left[\begin{smallmatrix}0\\-2\end{smallmatrix}\right]\}$ as illustrated in Figure (i). Members of classes $\mathcal{C}_0$ and $\mathcal{C}_1$ are represented by black disks and crosses, respectively.

  [I] **(7 pts):** We wish to design a perceptron $y = u(w_0 + w_1 x_1 + w_2 x_2)$ such that $y = 0$ if $\left[\begin{smallmatrix}x_1\\x_2\end{smallmatrix}\right] \in \mathcal{C}_0$ and $y = 1$ if $\left[\begin{smallmatrix}x_1\\x_2\end{smallmatrix}\right] \in \mathcal{C}_1$. Suppose that we use the perceptron training algorithm for this purpose with initial weights $w_0 = 1$, $w_1 = 0$, $w_2 = 1$ and learning rate $\eta = 1$. Either prove that the algorithm will converge, or prove that it will not converge. You may use the perceptron convergence theorem.

  [II] **(6 pts):** Design a neural network (single-layer or multi-layer) such that the network provides an output of 0 if $\left[\begin{smallmatrix}x_1\\x_2\end{smallmatrix}\right] \in \mathcal{C}_0$ and an output of 1 if $\left[\begin{smallmatrix}x_1\\x_2\end{smallmatrix}\right] \in \mathcal{C}_1$.

  (b) **(13 pts)** Repeat (a) for classes $\mathcal{C}_0 = \{\left[\begin{smallmatrix}1\\1\end{smallmatrix}\right], \left[\begin{smallmatrix}-1\\1\end{smallmatrix}\right], \left[\begin{smallmatrix}0\\4\end{smallmatrix}\right]\}$ and $\mathcal{C}_1 = \{\left[\begin{smallmatrix}0\\0\end{smallmatrix}\right], \left[\begin{smallmatrix}0\\-1\end{smallmatrix}\right], \left[\begin{smallmatrix}0\\-2\end{smallmatrix}\right]\}$ illustrated in Figure (ii). Note that the only difference is that now, instead of the point $\left[\begin{smallmatrix}0\\2\end{smallmatrix}\right]$, we have the point $\left[\begin{smallmatrix}0\\0\end{smallmatrix}\right]$ in class $\mathcal{C}_1$.

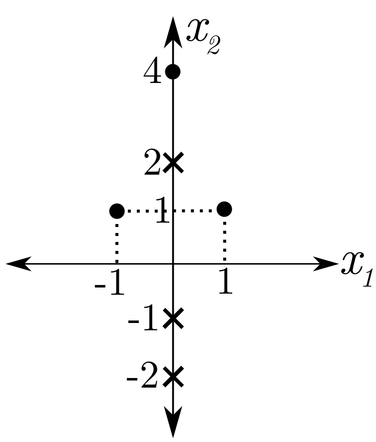

Figure (i)

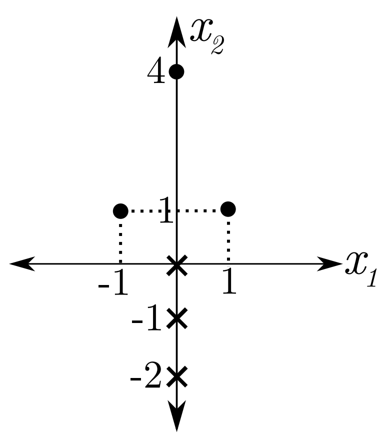

Figure (ii)