Department of Electrical and Electronics Engineering

EEE 443: Neural Networks

Class Project I Report

Name and Surname: Ali Aral Takak

Student ID: 22001758

Department: EEE

Instructor: Erdem Koyuncu

Problem I

Design a two-layer neural network with the **signum activation function** such that the network implements the logic gate $f(x_1, x_2, x_3) = \overline{x_1}x_2x_3 + x_1\overline{x_2}$. Assume that the input of -1 is used to represent a **FALSE**, and an input of 1 is used to represent a **TRUE**.

Let us first develop the truth table for the given logic function, in order to visualize and better understand the output of our neural network. The table given below displays the truth table of the logic function provided in the problem description:

$\boldsymbol{x_1}$	$\boldsymbol{x_2}$	$\boldsymbol{x_3}$	$\boldsymbol{\overline{x_1}x_2x_3 + x_1\overline{x_2}}$
-1	-1	-1	-1
-1	-1	1	-1
-1	1	-1	-1
-1	1	1	1
1	-1	-1	1
1	-1	1	1
1	1	-1	-1
1	1	1	-1

Using the table, we observe that the Boolean function outputs 1 in three cases, where:

$$x_1 = -1, x_2 = 1, x_3 = 1$$

$$x_1 = 1, x_2 = -1, x_3 = -1$$

$$x_1 = 1, x_2 = -1, x_3 = 1$$

Hence, we decompose the function into two terms, given as:

$$h_1 = \overline{x_1}x_2x_3$$

$$h_2 = x_1\overline{x_2}$$

Now, we follow a simple approach to our design of the neural network. The first layer of the network will consist of two neurons, in order to compute the functions decomposed above. The second and the output layer of the network, then, compute the OR operation for the outputs of the hidden layer neurons. First, let's design the hidden layer neurons. Each hidden neuron computes a term in the Boolean function, followed by a signum activation function.

The first neuron computes the function $h_1 = \overline{x_1} x_2 x_3$. Since we are working with the bipolar encoding of TRUE and FALSE, negation operation is equivalent to multiplication with -1. Hence, we have:

$$h_1 = sgn(-x_1 + x_2 + x_3 + b_1)$$

For the activation of the first neuron, we must have:

$$-x_1 + x_2 + x_3 + b_1 > 0$$

Using the input values that provide an output of 1, we determine b_1 as $b_1 > -3$. The minimum integer that satisfies this condition is -2.

Moving on to our second neuron in the hidden layer, it computes the function $h_2 = x_1 \overline{x_2}$. Hence, we have:

$$h_2 = sgn(x_1 - x_2 + b_2)$$

For the activation of the second neuron, we must have:

$$x_1 - x_2 + b_2 > 0$$

Using the values obtained from the truth table and solving for the inequality yields that $b_2 > -2$. The minimum integer that satisfies this condition is -1.

Proceeding with the output neuron, it computes the function:

$$f(x_1, x_2, x_3) = sgn(h_1 + h_2 + b_3)$$

In this case, the function should output 1 when at least one of the h_1 or h_2 is 1. Hence, the function should hold:

$$1 + 0 + b_3 > 0$$

Then, the bias should satisfy the inequality $b_3 > -1$. The minimum integer that satisfies this condition is 0.

Hence, the finals weights and biases can be stated as:

$$w_{11} = -1, w_{12} = 1$$

$$w_{21} = 1, w_{22} = -1$$

$$w_{31} = 1, w_{32} = 0$$

$$b_1 = -2, b_2 = -1, b_3 = 0$$

$$w_{h_1 3} = 1, w_{h_2 3} = 1$$

The figure provided below displays the diagram of the neural network architecture:

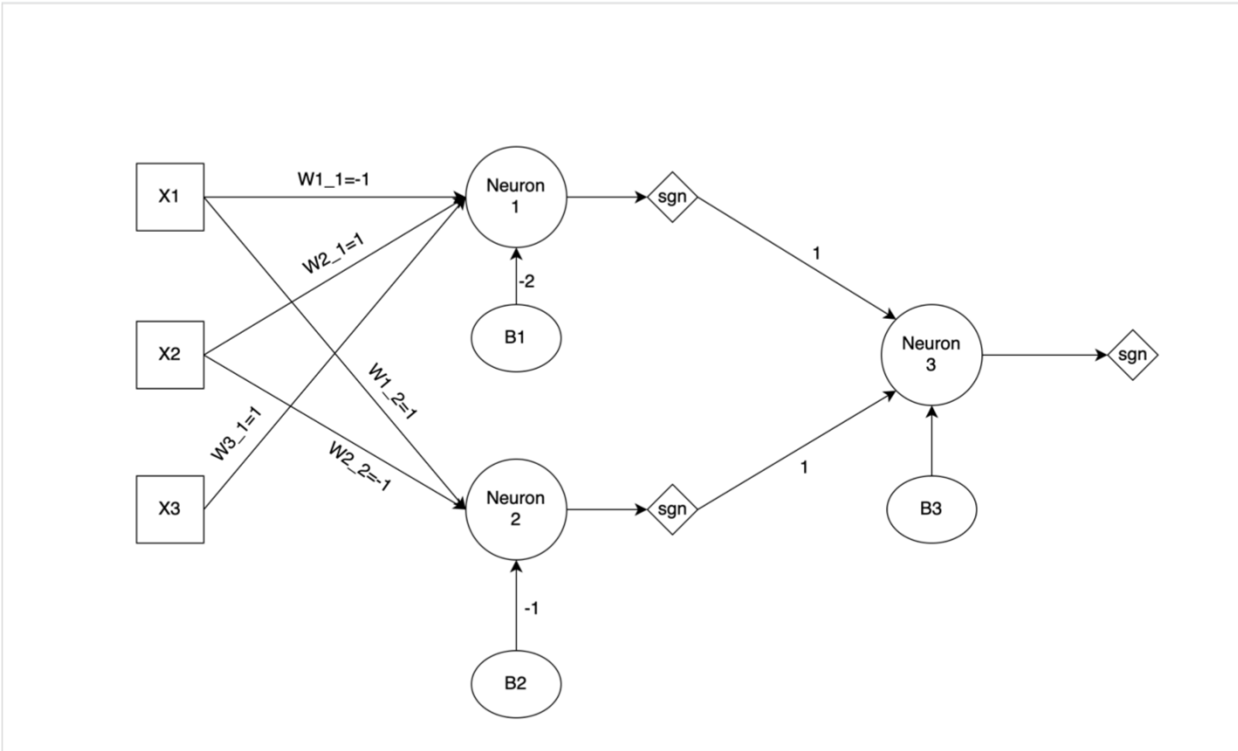

Figure 1: *Architecture diagram of the neural network.*

Problem II

Consider the network in the figure provided below. Write a python program that draws 1000 points uniformly at random from the square $[-2,2]^2$ and feeds each point to the neural network. If the network output is 0, plot the corresponding point as a blue point, and otherwise as a red point. Provide the plot in the report. Provide your estimate of the decision region that separates the output of 0 from an output of 1.

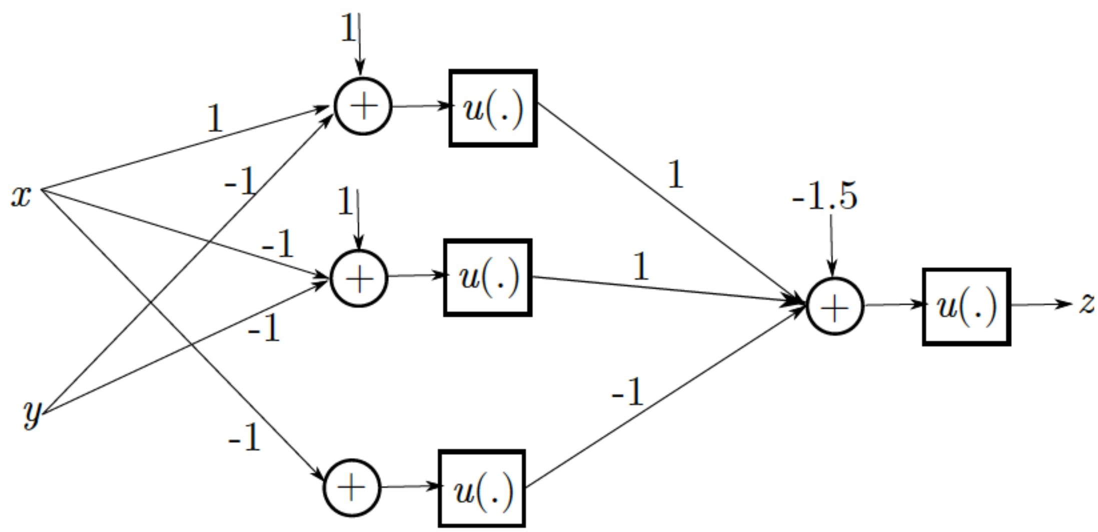

Figure 2: Provided neural network architecture.

Let us first start by examining the properties of the neural network provided. The network has two inputs, x and y, which are connected to three neurons. The biases and the biases of the neurons can be stated as:

$$Neuron\ 1 = [1, 1, -1]$$

$$Neuron\ 2 = [1, -1, -1]$$

$$Neuron\ 3 = [0, -1, 0]$$

Hence, for the first layer of the network, we obtain the mathematical equations:

$$H_1 = U(1 + x - y)$$

$$H_2 = U(1 - x - y)$$

$$H_3 = U(0 - x)$$

Before proceeding further, recall the output of the step activation function. Step activation function takes a single real number as input, and outputs a 0 or a 1, depending on whether the input is greater than or equal to a threshold value. In this case, the threshold value is zero. Mathematically, we can represent the step activation function as:

$$U(x) = \begin{cases} 1 \; if \; x \geq 0 \\ 0 \; o.w. \end{cases}$$

Using our combined knowledge of the step activation function and the mathematical representations of the neurons, we can represent the neuron output regions as:

$$H_1 = \begin{cases} 1, if \; 1 + x - y \geq 0 \\ 0, otherwise \end{cases}$$

$$H_2 = \begin{cases} 1, if \; 1 - x - y \geq 0 \\ 0, otherwise \end{cases}$$

$$H_3 = \begin{cases} 1, if \; -x \geq 0 \\ 0, otherwise \end{cases}$$

Moving further, the output layer neuron has the bias and the weights:

$$Output \; Neuron = [-1.5, 1, 1, -1]$$

The neuron can be mathematically represented as:

$$H_{out} = U(-1.5 + H_1 + H_2 - H_3)$$

Combining the behavior of the neuron and the step activation function, one can obtain:

$$H_{out} = \begin{cases} 1, if(-1.5 + H_1 + H_2 - H_3) \geq 0 \\ 0, otherwise \end{cases}$$

Since the step activation function gives a binary outcome, we can determine the possible region of interest by using a truth table:

H_1	H_2	H_3	$-1.5 + H_1 + H_2 - H_3$	H_{out}
0	0	0	-1.5	0
0	0	1	-2.5	0
0	1	0	-0.5	0
0	1	1	-1.5	0
1	0	0	-0.5	0
1	0	1	-1.5	0
1	1	0	0.5	1
1	1	1	-0.5	0

Hence, as the truth table displays, for $H_{out} = 1$, we need $H_1 = 1, H_2 = 1$ and $H_3 = 0$. By using the previous conditions, we observe that the decision region is a triangular region with borders defined as:

$$y \leq x + 1$$

$$y \geq -x + 1$$

$$x > 0$$

The visualization and decisions of the scattered points across the given space can be observed in the figure provided below:

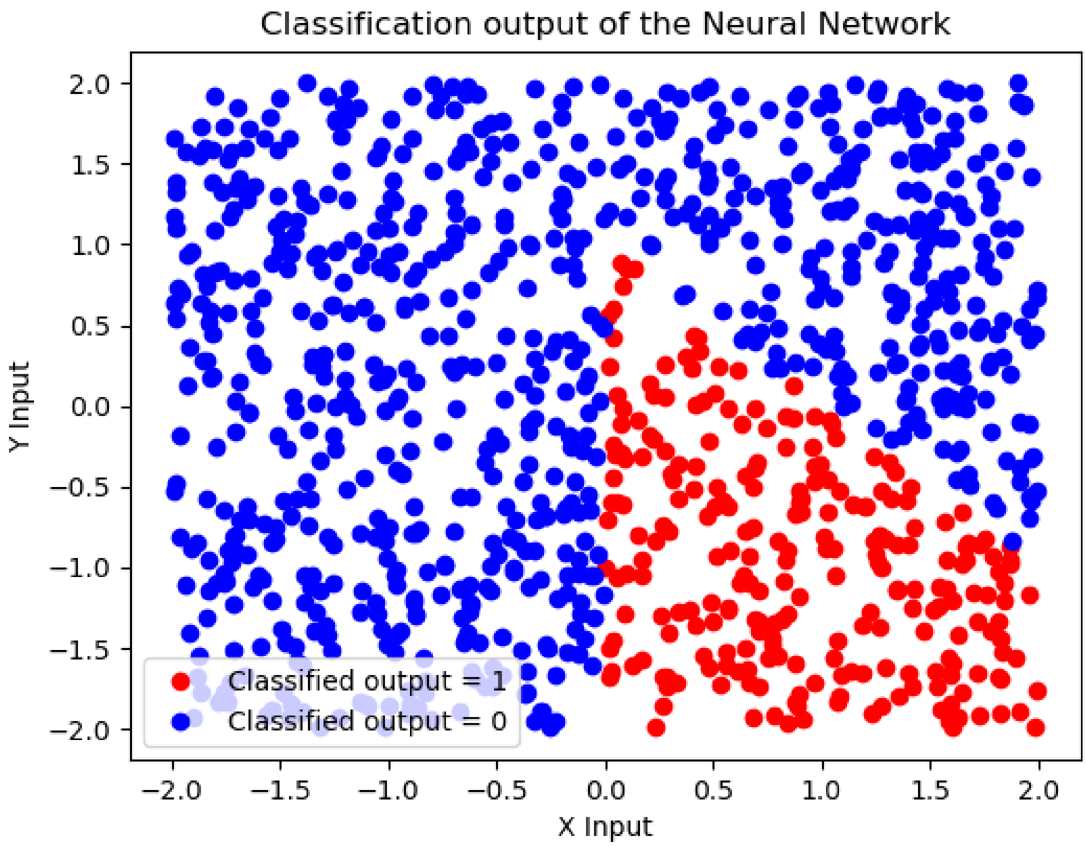

Figure 3: *Classification and decision output of the Neural Network.*