**CSC 527 Homework#2**

**Lan Nguyen**

**Problem 1:**

We assumed all the weight =1 therefore I will write the sum as (x1 + x2) + bk.

a) bk=0

|  |  |  |  |
| --- | --- | --- | --- |
| x1-excitatory | x2-inhibitory | yk | Explain |
| 0 | 0 | 1 | x1+x2+bk = 0+0+0 = 0 🡪 yk=1 |
| 1 | 0 | 1 | x1+x2+bk = 1+0+0 = 0 🡪 yk=1 |
| 0 | -1 | 0 | x1+x2+bk = 0-1+0 =-1 🡪 yk=0 |
| 1 | -1 | 1 | x1+x2+bk = 1-1+0 =0 🡪 yk=1 |

b) bk=1

|  |  |  |  |
| --- | --- | --- | --- |
| x1-excitatory | x2-inhibitory | yk | Explain |
| 0 | 0 | 1 | x1+x2+bk = 0+0+1 = 1 🡪 yk=1 |
| 1 | 0 | 1 | x1+x2+bk = 1+0+1 = 2 🡪 yk=1 |
| 0 | -1 | 0 | x1+x2+bk = 0-1+1 = 0 🡪 yk=1 |
| 1 | -1 | 0 | x1+x2+bk = 1-1+1 = 1 🡪 yk=1 |

c) There would be 3 combinations of excitatory and inhibitory signals.

They are: both are excitatory signals, both are inhibitory signals or 1 excitatory and 1 inhibitory signal.

d)

Let s = sum(wixi) + bk

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| x1 | x2 | s | Value of bk to get s >=0 | Value of bk to get s < 0 |
| 0 | 0 | 0+0+bk | bk>=0 | bk<0 |
| 0 | 1 | 0+1+bk | bk>=-1 | bk<-1 |
| 1 | 0 | 1+0+bk | bk>=-1 | bk<-1 |
| 0 | -1 | 0-1+bk | bk>=1 | bk<1 |
| -1 | 0 | -1+0+bk | bk>=1 | bk<1 |
| 1 | -1 | 1-1+bk | bk>=0 | bk<0 |
| -1 | 1 | -1+1+bk | bk>=0 | bk<0 |
| 1 | 1 | 1+1+bk | bk>=-2 | bk<-2 |
| -1 | -1 | -1-1+bk | bk>=2 | bk<2 |

As we can see from the above table, the neuron will always fire if bk >= 2

and will not fire if bk <-2. On the other hand, we need to adjust bk in the range [-2; 2) with respect to each combination of two inputs in order to get the precisely outcome yk.

e)

If both signals are excitatory then we can generate an OR MP neuron.

If both signals are inhibitory then we can generate a NOR MP neuron.

If 1 excitatory and 1 inhibitory then we can generate an OR NOT MP neuron.

**Problem 2:**

a) bk=1

|  |  |  |
| --- | --- | --- |
| x1 | x2 | yk |
| 0 | 0 | 0 |
| 1 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 1 | 0 |

b)

With bk=1 the program in HW1 will output yk=1 either x1 or x2 = 1. Also the neuron from HW1 always fire if bk > -1. Here bk=0 but we still got yk = 0.So we cannot implement this neuron using the previous program.

c) Let x1 is excitatory, x2 is inhibitory

bk=0

|  |  |  |
| --- | --- | --- |
| x1 | x2 | yk |
| 0 | 0 | 0 |
| 1 | 0 | 1 |
| 0 | -1 | 0 |
| 1 | -1 | 0 |

We cannot implement this neuron using the previous program because we don’t have option for input value = -1. Also with bk=0 the program in HW1 will output yk=0 (x1=0, x2 =0)

d) Yes. The properties of the networks mentioned is XOR logic gate. In order to generate this gate we cannot using a single MP neuron but using multiple neurons.

Step1: Generate neuron N1 as OR gate.

Step2: Generate neuron N2 as NAND gate.

Step3: Using output of N1 and N2 for neuron N3 as AND gate.

Model of N1(OR) with w1=1, w2=1, bk=-1:

|  |  |  |
| --- | --- | --- |
| x1 | x2 | yk(N1) |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

Model of N1(NAND) with w1=-1, w2=-1, bk=2:

|  |  |  |
| --- | --- | --- |
| x1 | x2 | yk(N2) |
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Combine two Model to get N3 with w1=1, w2=1, bk=-2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| x1 | x2 | yk(N1) | yk(N2) | yk(N3) |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 1 | 0 | 0 |