CIS 419/519: Homework 5

Jiatong Sun

03/13/2020

Although the solutions are entirely my own, I consulted with the following people and sources while working on this homework: $YuchenSun$, $JunfanPan$
https://stackoverflow.com/questions/50994504/how-to-put-figure-between-items-on-enumerate-list

1 Logical Functions with Neural Nets

a.

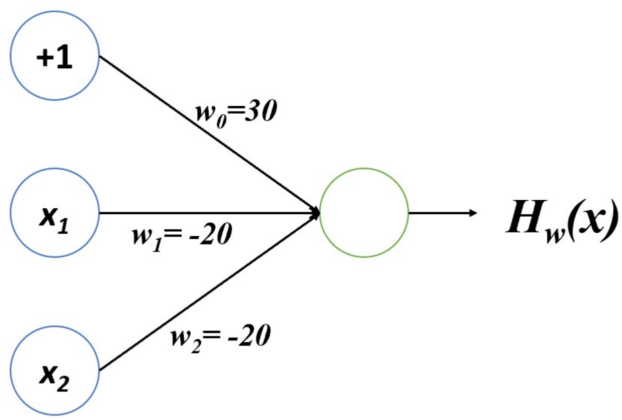

Figure 1: NAND

Truth Table		
x_1	x_2	$H(x)$
0	0	$\sigma(30) = 1$
0	1	$\sigma(10) = 1$
1	0	$\sigma(10) = 1$
1	1	$\sigma(-10) = 0$

Table 1: NAND

b.

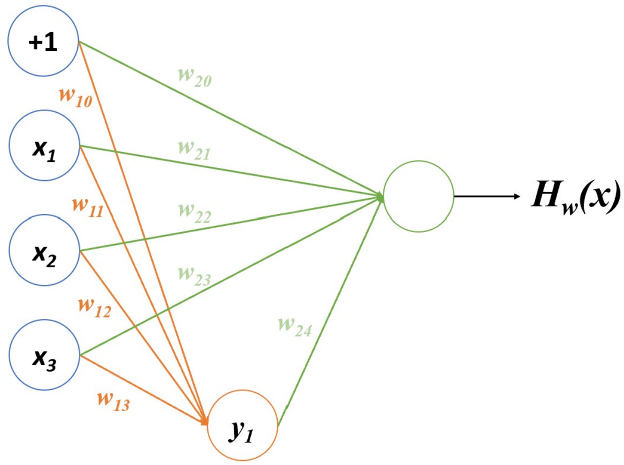

Figure 2: NAND

Truth Table								
w_{10}	w_{11}	w_{12}	w_{13}	w_{20}	w_{21}	w_{22}	w_{23}	w_{24}
-30	20	20	20	-10	20	20	20	-40

Table 2: Weight table

Truth Table				
x_1	x_2	x_3	y_1	$H(x)$
0	0	0	$\sigma(-30) = 0$	$\sigma(-10) = 0$
0	0	1	$\sigma(-10) = 0$	$\sigma(10) = 1$
0	1	0	$\sigma(-10) = 0$	$\sigma(10) = 1$
0	1	1	$\sigma(-10) = 0$	$\sigma(10) = 1$
1	0	0	$\sigma(10) = 1$	$\sigma(-10) = 0$
1	0	1	$\sigma(10) = 1$	$\sigma(-10) = 0$
1	1	0	$\sigma(10) = 1$	$\sigma(-10) = 0$
1	1	1	$\sigma(30) = 1$	$\sigma(10) = 1$

Table 3: Parity

2 Calculating Backprop by Hand

Since the hidden layer uses a sign function and the output layer uses a sigmoid function, the output can be expressed as follows:

$$a_2(\boldsymbol{X}) = Sigmoid(b_2(\boldsymbol{X})) = \frac{1}{1+e^{-b_2(\boldsymbol{X})}} = \frac{1}{1+e^{-\boldsymbol{W^2}a_1(\boldsymbol{X})}} = \frac{1}{1+e^{-\boldsymbol{W^2}\cdot Sign(\boldsymbol{W^1}\boldsymbol{X})}}$$

where

$$a_1(\boldsymbol{X}) = Sign(b_1(\boldsymbol{X})) = Sign(\boldsymbol{W^1}\boldsymbol{X}) = Sign(\begin{bmatrix} 0.1 & 0.2 \\ -0.4 & 0.3 \end{bmatrix}\begin{bmatrix} 5 \\ 4 \end{bmatrix}) = Sign(\begin{bmatrix} 1.3 \\ -0.8 \end{bmatrix}) = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

and

$$a_2(\boldsymbol{X}) = \frac{1}{1+e^{-\boldsymbol{W^2}a_1(\boldsymbol{X})}} = \frac{1}{1+e^{-\begin{bmatrix} 0.1 & 0.2 \end{bmatrix}\begin{bmatrix} 1 \\ -1 \end{bmatrix}}} = \frac{1}{1+e^{0.1}} = 0.475$$

To calculate the output gradient with respect to each of the weights, we can use the chain rule and thus, we know that

$$\frac{da_2(\boldsymbol{X})}{d\boldsymbol{W^2}} = \frac{da_2(\boldsymbol{X})}{db_2(\boldsymbol{X})}\frac{db_2(\boldsymbol{X})}{d\boldsymbol{W^2}} = [a_2(\boldsymbol{X})(1-a_2(\boldsymbol{X}))] \cdot a_1(\boldsymbol{X}) = 0.475 \cdot (1-0.475) \cdot \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \begin{bmatrix} 0.249 \\ -0.249 \end{bmatrix}$$

$$\frac{da_2(\boldsymbol{X})}{d\boldsymbol{W^1}} = \frac{da_2(\boldsymbol{X})}{db_2(\boldsymbol{X})}\frac{db_2(\boldsymbol{X})}{da_1(\boldsymbol{X})}\frac{da_1(\boldsymbol{X})}{db_1(\boldsymbol{X})}\frac{db_1(\boldsymbol{X})}{d\boldsymbol{W^1}} = 0.249 \cdot \boldsymbol{W^2} \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix} \cdot \boldsymbol{W^1} = 0.249 \cdot \begin{bmatrix} 0.1 & 0.2 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix} \cdot \begin{bmatrix} 0.1 & 0.2 \\ -0.4 & 0.3 \end{bmatrix} = \begin{bmatrix} 0.005 & 0.01 \\ -0.02 & 0.015 \end{bmatrix}$$

3 Neural Nets in SuperTuxKart