

Adeena Ahmed, Suada Demirovic, Yash Dhayal, Jason Swick
CSC 426-01
File Name: D4_Analysis.pdf
Final Project

Observations:

Input Values	Hidden values	Rounded Hidden Values	Output values
1,0,0,0,0,0,0,0	0.61305, 0.00974, 0.01091	1,0,0	1,0,0,0,0,0,0,0
0,1,0,0,0,0,0,0	0.00568, 0.02275, 0.20995	0,0,0	0,1,0,0,0,0,0,0
0,0,1,0,0,0,0,0	0.01702, 0.96126, 0.0793	0,1,0	0,0,1,0,0,0,0,0
0,0,0,1,0,0,0,0	0.00972, 0.77652, 0.98754	0,1,1	0,0,0,1,0,0,0,0
0,0,0,0,1,0,0,0	0.32076, 0.00823, 0.9859	0,0,1	0,0,0,0,1,0,0,0
0,0,0,0,0,1,0,0	0.92429, 0.91205, 0.00814	1,1,0	0,0,0,0,0,1,0,0
0,0,0,0,0,0,1,0	0.99439, 0.08463, 0.7256	1,0,1	0,0,0,0,0,0,1,0
0,0,0,0,0,0,0,1	0.92717, 0.99022, 0.98096	1,1,1	0,0,0,0,0,0,0,1

Analysis:

When we look at the hidden unit values produced for each of the input values after 5000 epochs, we see that our network has learned to encode our input values using a 3-bit binary representation.

Our learned encoding is as follows:

- 1,0,0,0,0,0,0,0 is encoded as 1,0,0
- 0,1,0,0,0,0,0,0 is encoded as 0,0,0
- 0,0,1,0,0,0,0,0 is encoded as 0,1,0
- 0,0,0,1,0,0,0,0 is encoded as 0,1,1
- 0,0,0,0,1,0,0,0 is encoded as 0,0,1
- 0,0,0,0,0,1,0,0 is encoded as 1,1,0
- 0,0,0,0,0,0,1,0 is encoded as 1,0,1
- 0,0,0,0,0,0,0,1 is encoded as 1,1,1

The network then learns to decode these hidden values to return the original input vector. Hence, our network was able to learn a useful representation of the eight input vectors in the hidden layer which is vital to learn the target function successfully.

