[**Jj2196@nyu.edu**](mailto:Jj2196@nyu.edu)

**Joby Joy**

**Project 3**

As part of the project, we develop a discrete Hopfield Neural Network using the training samples of:

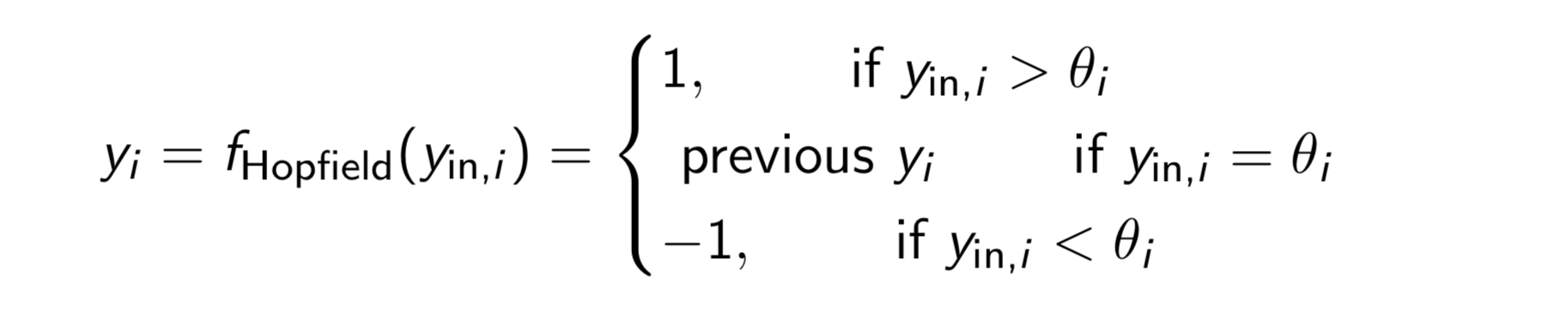
[1,1,-1,-1,-1,1],

[1,-1,-1,1,-1,-1],

[-1,-1,1,1,1,-1],

[-1,1,1,-1,1,1]

We use the activation function:



Using the Hebb’s rule we find the Weight Matrix to be equal to:

Weight Matrix = W

[[ 0. 0. -4. 0. -4. 0.]

[ 0. 0. 0. -4. 0. 4.]

[-4. 0. 0. 0. 4. 0.]

[ 0. -4. 0. 0. 0. -4.]

[-4. 0. 4. 0. 0. 0.]

[ 0. 4. 0. -4. 0. 0.]]

We generated a 64 bit bipolar text vector using the training sample: [1,1,-1,-1,-1,1]

We performed series of experiments on our Hopfield Neural Network to infer the solutions to the given question.

All the experiments were performed on both the training samples or the stored state vector and the 64 bipolar input test vector generated

**Experiment one:**

As part of finding the equilibrium state, we used a non-repeating random neuron to be activated i.e., the order by which each neuron can be selected is = [1,2,3,4,5,6] , [2,1,3,4,6,5],[5,6,3,1,2,4],……..

**Experiment two:**

Here we used a fixed order i.e., for every test vector, we only select the neuron in the following order [1,2,5,3,4,6]

**Experiment three:**

Here we used a repeating random order of selecting neuron for activation i.e.,

[2,2,3,4,2,6], [1,1,3,4,1,1], [1,1,1,1,1,1] and so on:

**Observations:**

**Experiment one:**

1. Yes, we find that all the stored patterns are in equilibrium state:

Results:

stable.. y\_old [1, 1, -1, -1, -1, 1],y\_changed [1, 1, -1, -1, -1, 1], x\_input [1, 1, -1, -1, -1, 1]

stable.. y\_old [1, -1, -1, 1, -1, -1],y\_changed [1, -1, -1, 1, -1, -1], x\_input [1, -1, -1, 1, -1, -1]

stable.. y\_old [-1, -1, 1, 1, 1, -1],y\_changed [-1, -1, 1, 1, 1, -1], x\_input [-1, -1, 1, 1, 1, -1]

stable.. y\_old [-1, 1, 1, -1, 1, 1],y\_changed [-1, 1, 1, -1, 1, 1], x\_input [-1, 1, 1, -1, 1, 1]

2—

Upon trying to find the assoications for the entire test vectors(64 bipolar input test vectors), we find that all the vectors reach an equilibrium state.

So we get 64 equilbrium states .

We find that all the 64 equilibrium states also converge to the stored states. And thus we do not find any spurious vectors.

Some results from the test:

**stable..** y\_old [1, 1, -1, -1, -1, 1],y\_changed [1, 1, -1, -1, -1, 1], x\_input [1, 1, -1, -1, -1, 1]

unstable.. y\_old [-1, 1, -1, -1, -1, 1],y\_changed [1, 1, -1, -1, -1, 1], x\_input [-1, 1, -1, -1, -1, 1]

**stable..** y\_old [1, 1, -1, -1, -1, 1],y\_changed [1, 1, -1, -1, -1, 1], x\_input [-1, 1, -1, -1, -1, 1]

unstable.. y\_old [1, -1, -1, -1, -1, 1],y\_changed [1, 1, -1, -1, -1, 1], x\_input [1, -1, -1, -1, -1, 1]

**stable..** y\_old [1, 1, -1, -1, -1, 1],y\_changed [1, 1, -1, -1, -1, 1], x\_input [1, -1, -1, -1, -1, 1]

For every unstable state we see that there is a stable state where the y\_old will be the same as y\_changed and hence its an equilbrium.

Additionally each of the equilbrium state converges to a stored state:

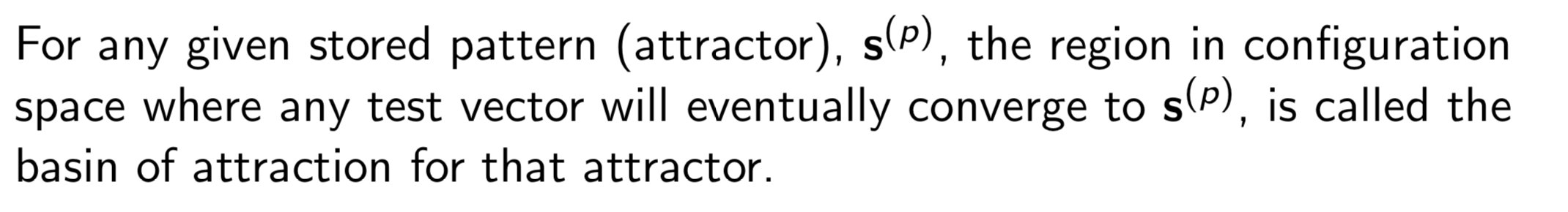
Total number of states in the equilbrium state 64

Total number of states in the spurious state 0

\*Since we see the same pattern for the rest of the bipolar vectors we are not publishing it in the report.

1. – We based our result on the definition of basin to analyze our

Basin of attraction that is,



On these lines, we known that out of the 64 bipolar vectors we get 4 vectors that are alrteady the stored vector and the equilbrium is already established and constitute to the 4 attractor of the stored vectors. Thus we try to find the basin of attraction to the other 60 vectors in the bipolar vector list.

Here we assume that the convergence space is the vectors that were created as part of the activation that lead up to the stored vector. (results following the discussion)

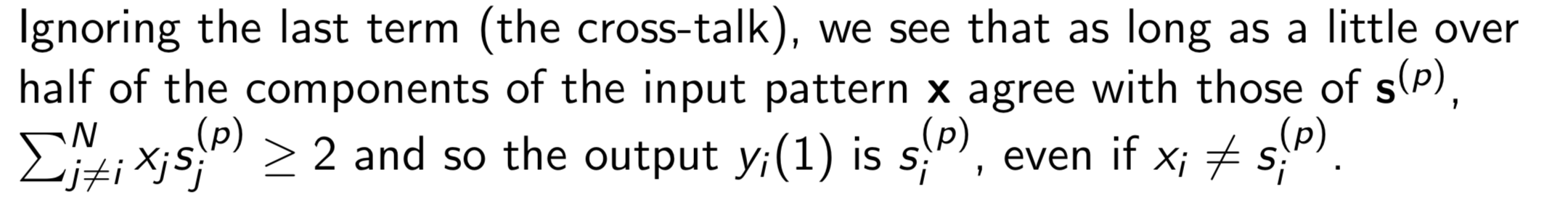
Post convergence we try to find how many of the 60 vectors are attracted to the stored vectors. For this analysis we found the count of all the equilibrium states that attractes to the stored vector.

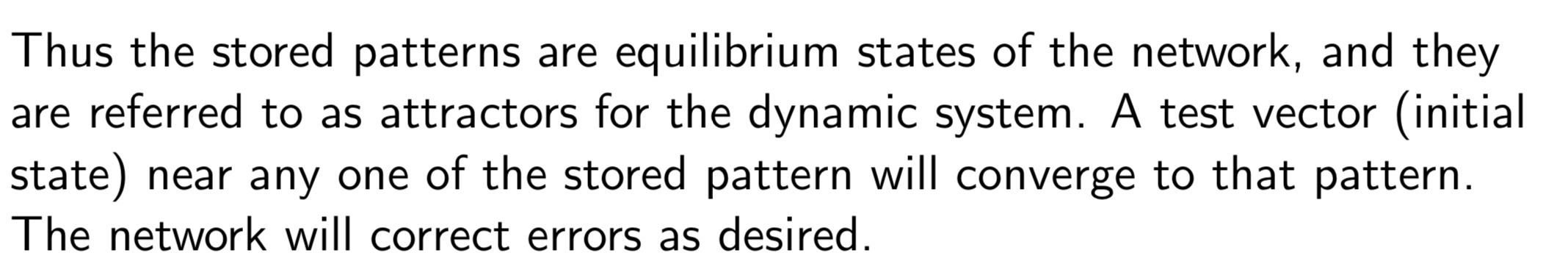
{S1: 15, S2: 15, S3: 15, S4: 15}

Here we see that all the equlibrium states have the same count of attractors. In other words 15 are attracted to S1 stored vector, 15 S2, 15 S3, 15 S4. And the other four vectors have been the stored vectors. In summary we have

{S1: 16, S2: 16, S3: 16, S4: 16} considering the stored vector samples.

1. – For this experiment we based our analysis to prove the theorm:





We calculate the maxiumn hamming distance for each of the patterns from the stored states:

We see that the max distance can be at 2 which is <= N/2 and hence as per the theorem the chance that an input pattern doesn’t associate with any stored pattern is 0.

Results:

Hamming distances:

{'-1 -1 -1 -1 -1 -1': 2, <- max distance

'-1 -1 -1 -1 -1 1': 2,

'-1 -1 -1 -1 1 -1': 2,

'-1 -1 -1 -1 1 1': 2,

'-1 -1 -1 1 -1 -1': 1,

'-1 -1 -1 1 -1 1': 2,

'-1 -1 -1 1 1 -1': 1,

'-1 -1 -1 1 1 1': 2,

'-1 -1 1 -1 -1 -1': 2,

'-1 -1 1 -1 -1 1': 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': 2,

'-1 -1 1 1 1 -1': 0,

'-1 -1 1 1 1 1': 1,

'-1 1 -1 -1 -1 -1': 2,

'-1 1 -1 -1 -1 1': 1,

'-1 1 -1 -1 1 -1': 2,

'-1 1 -1 -1 1 1': 1,

'-1 1 -1 1 -1 -1': 2,

'-1 1 -1 1 -1 1': 2,

'-1 1 -1 1 1 -1': 2,

'-1 1 -1 1 1 1': 2,

'-1 1 1 -1 -1 -1': 2,

'-1 1 1 -1 -1 1': 1,

'-1 1 1 -1 1 -1': 1,

'-1 1 1 -1 1 1': 0,

'-1 1 1 1 -1 -1': 2,

'-1 1 1 1 -1 1': 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 1 -1': 2,

'1 -1 -1 -1 1 1': 2,

'1 -1 -1 1 -1 -1': 0,

'1 -1 -1 1 -1 1': 1,

'1 -1 -1 1 1 -1': 1,

'1 -1 -1 1 1 1': 2,

'1 -1 1 -1 -1 -1': 2,

'1 -1 1 -1 -1 1': 2,

'1 -1 1 -1 1 -1': 2,

'1 -1 1 -1 1 1': 2,

'1 -1 1 1 -1 -1': 1,

'1 -1 1 1 -1 1': 2,

'1 -1 1 1 1 -1': 1,

'1 -1 1 1 1 1': 2,

'1 1 -1 -1 -1 -1': 1,

'1 1 -1 -1 -1 1': 0,

'1 1 -1 -1 1 -1': 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': 2,

'1 1 -1 1 1 1': 2,

'1 1 1 -1 -1 -1': 2,

'1 1 1 -1 -1 1': 1,

'1 1 1 -1 1 -1': 2,

'1 1 1 -1 1 1': 1,

'1 1 1 1 -1 -1': 2,

'1 1 1 1 -1 1': 2,

'1 1 1 1 1 -1': 2,

'1 1 1 1 1 1': 2

Configuration Space(Basin of Attraction):

1 - Index

[1, 1, -1, -1, -1, 1] – Stored Vector

[[1, 1, -1, -1, -1, 1], [1, 1, -1, -1, -1, 1], [1, 1, -1, -1, -1, 1], [1, 1, -1, -1, -1, 1]] – Convergence space

2

[1, 1, -1, -1, -1, 1]

[[1, 1, -1, -1, -1, 1]]

3

[1, 1, -1, -1, -1, 1]

[[1, 1, -1, -1, -1, 1]]

4

[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]]

5

[1, 1, -1, -1, -1, 1]

[[1, 1, -1, -1, -1, 1]]

6

[1, 1, -1, -1, -1, 1]

[[1, 1, -1, -1, -1, 1], [1, 1, -1, -1, -1, 1], [1, 1, -1, -1, -1, 1]]

7

[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, -1, -1, -1, 1], [1, 1, -1, -1, -1, 1]]

8

[-1, 1, 1, -1, 1, 1]

[[-1, 1, 1, -1, 1, 1]]

9

[1, 1, -1, -1, -1, 1]

[[-1, 1, -1, -1, -1, 1], [-1, 1, -1, -1, -1, 1], [1, 1, -1, -1, -1, 1]]

10

[-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, 1, -1, 1, 1], [-1, 1, 1, -1, 1, 1]]

11

[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, -1, -1, -1, 1]]

12

[1, 1, -1, -1, -1, 1]

[[1, -1, -1, -1, -1, 1], [1, -1, -1, -1, -1, 1], [1, 1, -1, -1, -1, 1]]

13

[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, -1, 1, -1, -1]]

14

[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, -1, -1, -1, 1]]

15

[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, -1, 1, -1, -1]]

16

[1, 1, -1, -1, -1, 1]

[[1, 1, 1, -1, -1, 1], [1, 1, -1, -1, -1, 1]]

17

[-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, 1, -1, 1, 1], [-1, 1, 1, -1, 1, 1]]

18

[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, -1, -1, -1, -1], [1, 1, -1, -1, -1, 1]]

19

[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, -1, -1, -1, 1]]

20

[1, -1, -1, 1, -1, -1]

[[1, -1, -1, 1, -1, -1]]

21

[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]]

22

[-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]]

23

[1, -1, -1, 1, -1, -1]

[[-1, -1, -1, 1, -1, -1], [1, -1, -1, 1, -1, -1], [1, -1, -1, 1, -1, -1]]

24

[-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]]

25

[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, -1, 1, -1, -1], [1, -1, -1, 1, -1, -1]]

26

[-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]]

28

[-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]]

29

[-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, 1, -1, 1, 1]]

30

[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, -1, 1, -1, -1], [1, -1, -1, 1, -1, -1]]

31

[-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, 1, -1, 1, 1]]

32

[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, -1, 1, -1, -1], [1, -1, -1, 1, -1, -1]]

33

[-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, 1, -1, 1, 1]]

34

[1, -1, -1, 1, -1, -1]

[[1, -1, -1, -1, -1, -1], [1, -1, -1, 1, -1, -1], [1, -1, -1, 1, -1, -1]]

35

[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, -1, 1, -1, -1]]

37

[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, -1, 1, -1, -1], [1, -1, -1, 1, -1, -1]]

38

[-1, 1, 1, -1, 1, 1]

[[-1, 1, 1, 1, 1, 1], [-1, 1, 1, -1, 1, 1], [-1, 1, 1, -1, 1, 1]]

39

[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]]

40

[-1, 1, 1, -1, 1, 1]

[[-1, 1, 1, -1, 1, -1], [-1, 1, 1, -1, 1, -1], [-1, 1, 1, -1, 1, 1]]

41

[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, -1, 1, -1, -1]]

42

[-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]]

43

[-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, 1, -1, 1, 1]]

44

[-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, 1, 1, 1, -1]]

45

[-1, -1, 1, 1, 1, -1]

[[-1, -1, -1, 1, 1, -1], [-1, -1, 1, 1, 1, -1]]

46

[1, -1, -1, 1, -1, -1]

[[1, -1, -1, 1, -1, -1], [1, -1, -1, 1, -1, -1]]

47

[-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, 1, 1, 1, -1], [-1, -1, 1, 1, 1, -1]]

48

[-1, 1, 1, -1, 1, 1]

[[-1, 1, 1, -1, 1, 1]]

49

[-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]]

50

[-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, 1, -1, 1, 1]]

51

[-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, -1, 1, 1, -1], [-1, -1, 1, 1, 1, -1]]

52

[-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]]

53

[1, -1, -1, 1, -1, -1]

[[1, -1, -1, 1, -1, -1], [1, -1, -1, 1, -1, -1], [1, -1, -1, 1, -1, -1]]

54

[-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, 1, -1, 1, -1], [-1, -1, 1, 1, 1, -1]]

55

[1, -1, -1, 1, -1, -1]

[[1, -1, -1, 1, -1, -1], [1, -1, -1, 1, -1, -1], [1, -1, -1, 1, -1, -1]]

56

[-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]]

57

[-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, 1, 1, 1, -1], [-1, -1, 1, 1, 1, -1]]

58

[-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, 1, 1, 1, -1]]

59

[-1, -1, 1, 1, 1, -1]

[[-1, -1, 1, 1, 1, -1], [-1, -1, 1, 1, 1, -1]]

60

[-1, -1, 1, 1, 1, -1]

[[-1, -1, 1, 1, 1, -1], [-1, -1, 1, 1, 1, -1]]

61

[-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]]

62

[-1, -1, 1, 1, 1, -1]

[[-1, -1, 1, 1, 1, -1]]

Experiment 2: Fixed order of activations - [4,5,0,1,2,3]

1. We find that the stored vectors are in euilibrium state.
2. stable.. y\_old [1, 1, -1, -1, -1, 1],y\_changed [1, 1, -1, -1, -1, 1], x\_input [1, 1, -1, -1, -1, 1]
3. stable.. y\_old [1, -1, -1, 1, -1, -1],y\_changed [1, -1, -1, 1, -1, -1], x\_input [1, -1, -1, 1, -1, -1]
4. stable.. y\_old [-1, -1, 1, 1, 1, -1],y\_changed [-1, -1, 1, 1, 1, -1], x\_input [-1, -1, 1, 1, 1, -1]
5. stable.. y\_old [-1, 1, 1, -1, 1, 1],y\_changed [-1, 1, 1, -1, 1, 1], x\_input [-1, 1, 1, -1, 1, 1]
6. All the 64 bipolar vectors are in euilibrium states and they all converge to the stored vectors and no spurious state were to be found.

Total number of states in the equilbrium state 64

Total number of states in the spurious state 0

1. The results for the basin of attraction are the same as experiement one except for the order in which the activations happen.

we see that all the equlibrium states have the same count of attractors. In other words 15 are attracted to S1 stored vector, 15 S2, 15 S3, 15 S4. And the other four vectors have been the stored vectors. In summary we have

{S1: 16, S2: 16, S3: 16, S4: 16} considering the stored vector samples.

1. There is no chance that an input pattern wouldn’t associate with the stored pattern, same calculations as experiement one.

Experiment 3:

Here we used a repeating random order of selecting neuron for activation i.e.,

[2,2,3,4,2,6], [1,1,3,4,1,1], [1,1,1,1,1,1] and so on:

**Observations:**

1. The stored vectors continue to be in euilibrium state.

stable.. y\_old [1, 1, -1, -1, -1, 1],y\_changed [1, 1, -1, -1, -1, 1], x\_input [1, 1, -1, -1, -1, 1]

stable.. y\_old [1, -1, -1, 1, -1, -1],y\_changed [1, -1, -1, 1, -1, -1], x\_input [1, -1, -1, 1, -1, -1]

stable.. y\_old [-1, -1, 1, 1, 1, -1],y\_changed [-1, -1, 1, 1, 1, -1], x\_input [-1, -1, 1, 1, 1, -1]

stable.. y\_old [-1, 1, 1, -1, 1, 1],y\_changed [-1, 1, 1, -1, 1, 1], x\_input [-1, 1, 1, -1, 1, 1]

2-

In this test, we could see that the bipolar input vectors still reach an equilibrium state for all the vectors, however they do not converge to the stored vector i.e, there are spurious vectors

stable.. y\_old [1, 1, -1, -1, -1, 1],y\_changed [1, 1, -1, -1, -1, 1], x\_input [1, 1, -1, -1, -1, 1]

unstable.. y\_old [-1, 1, -1, -1, -1, 1],y\_changed [1, 1, -1, -1, -1, 1], x\_input [-1, 1, -1, -1, -1, 1]

stable.. y\_old [1, 1, -1, -1, -1, 1],y\_changed [1, 1, -1, -1, -1, 1], x\_input [-1, 1, -1, -1, -1, 1]

unstable.. y\_old [1, -1, -1, -1, -1, 1],y\_changed [1, 1, -1, -1, -1, 1], x\_input [1, -1, -1, -1, -1, 1]

stable.. y\_old [1, 1, -1, -1, -1, 1],y\_changed [1, 1, -1, -1, -1, 1], x\_input [1, -1, -1, -1, -1, 1]

unstable.. y\_old [1, 1, 1, -1, -1, 1],y\_changed [1, 1, -1, -1, -1, 1], x\_input [1, 1, 1, -1, -1, 1]

stable.. y\_old [1, 1, -1, -1, -1, 1],y\_changed [1, 1, -1, -1, -1, 1], x\_input [1, 1, 1, -1, -1, 1]

unstable.. y\_old [1, 1, -1, 1, -1, 1],y\_changed [1, 1, -1, -1, -1, 1], x\_input [1, 1, -1, 1, -1, 1]

stable.. y\_old [1, 1, -1, -1, -1, 1],y\_changed [1, 1, -1, -1, -1, 1], x\_input [1, 1, -1, 1, -1, 1]

In this test we found that we have

17 spurious states and

47 equlibriuum states that converges to the stored vector.

Of the 17 spurious we find zero reversed states that could converge to the stored vector.

Additionally the count of spurious states vary with every iteration of the test.

3 – We find that for the 47 equilbrium state which has 43 bipolar input text vectors that are not equal to the training or the stored vectors, the basin of attractions are distributed as

{S1: 13, S2: 13, S3: 10, S4: 7} which can varies with iteration of the test

When we consider all 47 equilibrium states:

{S1: 14, S2: 14, S3: 14, S4: 14}

The output for the configuration space or basin of attraction attached towards the end.

1. 17/64 = 26.6% where 17 is the total number of spurious vectors. So we can estimate that there is a 26.6% probablity of the input pattern not associating with any of the stored patterns.

Thus this experiment leads to a failure in yielding associations for the network .

**In summary:**

**Experiment one and two makes the associations for all the input combimations**

**Experiments three brings the network to equilibrium state however yields spurious states and makes invalid assoications.**

Basin of attration from experiment 3:

{1: [[1, 1, -1, -1, -1, 1]],

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]],

3: [[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]],

4: [[1, 1, -1, -1, -1, 1],

[1, 1, -1, -1, -1, 1],

[1, 1, -1, -1, -1, 1],

[1, 1, -1, -1, -1, 1]],

5: [[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]],

6: [[1, 1, -1, -1, -1, 1]],

7: [[-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]],

8: [[-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]],

9: [[1, 1, -1, 1, -1, 1],

[1, 1, -1, -1, -1, 1],

[1, 1, -1, -1, -1, 1],

[1, 1, -1, -1, -1, 1]],

10: [[-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]],

11: [[1, 1, -1, -1, -1, -1],

[1, 1, -1, -1, -1, 1],

[1, 1, -1, -1, -1, 1],

[1, 1, -1, -1, -1, 1]],

12: [[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, -1, -1, -1, 1]],

14: [[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]],

15: [[1, -1, -1, 1, -1, -1]],

16: [[1, 1, -1, 1, -1, 1]],

18: [[1, 1, -1, -1, -1, -1],

[1, 1, -1, -1, -1, 1],

[1, 1, -1, -1, -1, 1],

[1, 1, -1, -1, -1, 1]],

19: [[1, 1, -1, -1, 1, 1], [1, 1, -1, -1, 1, 1]],

20: [[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]],

21: [[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, -1, -1, -1, 1]],

23: [[-1, -1, -1, 1, -1, -1],

[-1, -1, -1, 1, -1, -1],

[-1, -1, -1, 1, -1, -1],

[1, -1, -1, 1, -1, -1]],

25: [[1, -1, -1, -1, -1, -1],

[1, -1, -1, -1, -1, -1],

[1, -1, -1, 1, -1, -1]],

26: [[-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, 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]],

28: [[-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]],

29: [[-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]],

30: [[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, -1, 1, -1, -1]],

31: [[-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, 1, -1, 1, 1],

[-1, 1, 1, -1, 1, 1],

[-1, 1, 1, -1, 1, 1]],

32: [[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, 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, -1, -1],

[1, -1, -1, 1, -1, -1]],

34: [[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]],

35: [[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, -1, 1, -1, -1]],

37: [[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, -1, 1, -1, -1]],

38: [[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, 1, -1, 1, 1]],

39: [[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, -1, 1, -1, -1]],

40: [[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, 1, -1, 1, 1],

[-1, 1, 1, -1, 1, 1],

[-1, 1, 1, -1, 1, 1]],

41: [[1, -1, -1, 1, 1, -1],

[1, -1, -1, 1, 1, -1],

[1, -1, -1, 1, -1, -1],

[1, -1, -1, 1, -1, -1]],

42: [[-1, -1, 1, 1, 1, 1]],

44: [[-1, -1, 1, 1, -1, -1], [-1, -1, 1, 1, 1, -1]],

45: [[-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]],

46: [[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]],

47: [[-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, -1, 1, 1, -1],

[-1, -1, 1, 1, 1, -1],

[-1, -1, 1, 1, 1, -1]],

49: [[-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, 1, 1, 1, -1]],

51: [[-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, 1, 1, 1, -1]],

52: [[1, -1, 1, 1, 1, -1], [1, -1, 1, 1, 1, -1]],

53: [[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]],

54: [[1, -1, 1, 1, 1, -1], [1, -1, 1, 1, 1, -1]],

56: [[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, 1, 1, 1, -1],

[-1, -1, 1, 1, 1, -1],

[-1, -1, 1, 1, 1, -1],

[-1, -1, 1, 1, 1, -1]],

57: [[-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, 1, 1, 1, -1]],

58: [[-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, 1, 1, 1, -1]],

59: [[-1, -1, 1, 1, 1, -1], [-1, -1, 1, 1, 1, -1], [-1, -1, 1, 1, 1, -1]],

60: [[-1, -1, 1, 1, 1, -1]],

61: [[-1, -1, 1, 1, 1, -1],

[-1, -1, 1, 1, 1, -1],

[-1, -1, 1, 1, 1, -1],

[-1, -1, 1, 1, 1, -1]]}