Simple Reflex Agent:

```
  -1
(base) suman@gole:~/Documents/AI lab/Artificial-Intelligence-college-course$ python -u
/home/suman/Documents/AI lab/Artificial-Intelligence-college-course/Agents/agents.py"
Vacuum randomly placed at Location B.
{'A': 1, 'B': 0}
Performance Measurement: 0
(base) suman@gole:~/Documents/AI lab/Artificial-Intelligence-college-course$
```

AND gate using:

Hebbian Learning

```
(base) suman@gole:~/Documents/AI lab/Artificial-Intelligence-college-course$ python -u "
/home/suman/Documents/AI lab/Artificial-Intelligence-college-course/And gate /hebbian.py
"
input is:
[[ 1  1  1]
 [ 1 -1  1]
 [-1  1  1]
 [-1 -1  1]]
output for And Gate is:
[[ 1]
 [-1]
 [-1]
 [-1]]
[0. 0. 0.]
/home/suman/Documents/AI lab/Artificial-Intelligence-college-course/And gate /hebbian.py
:31: RuntimeWarning: invalid value encountered in double_scalars
  c=-(weights[2]/weights[0])
Checking after learning selectg a input
Enter the test case no you want to try3
Select a logic you also want  to check
a
[ 2.  2. -2.]
selected input is 3
[-1 -1  1]
-1
(base) suman@gole:~/Documents/AI lab/Artificial-Intelligence-college-course$
```

Perceptron Learning:

```
(base) suman@gole:~/Documents/AI lab/Artificial-Intelligence-college-course$ python -u "
/home/suman/Documents/AI lab/Artificial-Intelligence-college-course/And gate /perceptron
.py"
AND 0, 1 = 0
AND 1, 1 = 1
AND 0, 0 = 0
AND 1, 0 = 0
(base) suman@gole:~/Documents/AI lab/Artificial-Intelligence-college-course$
```

Map Problems:

A* algorithm

```
(base) suman@gole:~/Documents/AI lab/Artificial-Intelligence-college-course$ python -u "
/home/suman/Documents/AI lab/Artificial-Intelligence-college-course/Map problem/astar.py
"
Path found: ['A', 'B', 'D']
(base) suman@gole:~/Documents/AI lab/Artificial-Intelligence-college-course$
```


BFS

```
(base) suman@gole:~/Documents/AI lab/Artificial-Intelligence-college-course$ python -u "
/home/suman/Documents/AI lab/Artificial-Intelligence-college-course/Map problem/bfs.py"
0 1 3 2 8 9
(base) suman@gole:~/Documents/AI lab/Artificial-Intelligence-college-course$
```


Naive Bayes Algorithm for Text Classification

```
print(prob_ham)
print(prob_spam)
print(result(prob_ham, prob_spam))
```

```
Enter the messageyou are out of money call me
0.6666666666666666
0.030303030303030304
spam
```

Puzzle Problems:

BFS

```
(base) suman@gole:~/Documents/AI lab/Artificial-Intelligence-college-course$ python -u "/home/suman/Documents/AI lab/Artificial-Intelligence-college-course/PuzzleProblem/bfs.py"
[[[4, 1, 2], [0, 6, 3], [7, 5, 8]], [[0, 1, 2], [4, 6, 3], [7, 5, 8]], [[1, 0, 2], [4, 6, 3], [7, 5, 8]], [[1, 2, 0], [4, 6, 3], [7, 5, 8]], [[1, 2, 3], [4, 6, 0], [7, 5, 8]], [[1, 2, 3], [4, 0, 6], [7, 5, 8]], [[1, 2, 3], [4, 5, 6], [7, 0, 8]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[4, 1, 2], [7, 5, 3], [8, 0, 6]], [[4, 1, 2], [7, 5, 3], [0, 8, 6]], [[4, 1, 2], [0, 5, 3], [7, 8, 6]], [[0, 1, 2], [4, 5, 3], [7, 8, 6]], [[1, 0, 2], [4, 5, 3], [7, 8, 6]], [[1, 2, 0], [4, 5, 3], [7, 8, 6]], [[1, 2, 3], [4, 5, 0], [7, 8, 6]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[2, 3, 6], [0, 5, 8], [1, 4, 7]], [[2, 3, 6], [1, 5, 8], [0, 4, 7]], [[2, 3, 6], [1, 5, 8], [4, 0, 7]], [[2, 3, 6], [1, 5, 8], [4, 7, 0]], [[2, 3, 6], [1, 5, 0], [4, 7, 8]], [[2, 3, 0], [1, 5, 6], [4, 7, 8]], [[2, 0, 3], [1, 5, 6], [4, 7, 8]], [[0, 2, 3], [1, 5, 6], [4, 7, 8]], [[1, 2, 3], [0, 5, 6], [4, 7, 8]], [[1, 2, 3], [4, 5, 6], [0, 7, 8]], [[1, 2, 3], [4, 5, 6], [7, 0, 8]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[1, 2, 3], [4, 5, 0], [7, 8, 6]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[1, 3, 5], [0, 4, 2], [7, 8, 6]], [[1, 3, 5], [4, 0, 2], [7, 8, 6]], [[1, 3, 5], [4, 2, 0], [7, 8, 6]], [[1, 3, 0], [4, 2, 5], [7, 8, 6]], [[1, 0, 3], [4, 2, 5], [7, 8, 6]], [[1, 2, 3], [4, 0, 5], [7, 8, 6]], [[1, 2, 3], [4, 5, 0], [7, 8, 6]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[1, 0, 2], [7, 4, 3], [8, 6, 5]], [[1, 2, 0], [7, 4, 3], [8, 6, 5]], [[1, 2, 3], [7, 4, 0], [8, 6, 5]], [[1, 2, 3], [7, 4, 5], [8, 6, 0]], [[1, 2, 3], [7, 4, 5], [8, 0, 6]], [[1, 2, 3], [7, 4, 5], [0, 8, 6]], [[1, 2, 3], [0, 4, 5], [7, 8, 6]], [[1, 2, 3], [4, 0, 5], [7, 8, 6]], [[1, 2, 3], [4, 5, 0], [7, 8, 6]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[4, 0, 1], [7, 2, 3], [5, 8, 6]], [[4, 1, 0], [7, 2, 3], [5, 8, 6]], [[4, 1, 3], [7, 2, 0], [5, 8, 6]], [[4, 1, 3], [7, 2, 6], [5, 8, 0]], [[4, 1, 3], [7, 2, 6], [5, 0, 8]], [[4, 1, 3], [7, 2, 6], [0, 5, 8]], [[4, 1, 3], [0, 2, 6], [7, 5, 8]], [[0, 1, 3], [4, 2, 6], [7, 5, 8]], [[1, 0, 3], [4, 2, 6], [7, 5, 8]], [[1, 2, 3], [4, 0, 6], [7, 5, 8]], [[1, 2, 3], [4, 5, 6], [7, 0, 8]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[1, 6, 2], [4, 5, 3], [7, 0, 8]], [[1, 6, 2], [4, 0, 3], [7, 5, 8]], [[1, 0, 2], [4, 6, 3], [7, 5, 8]], [[1, 2, 0], [4, 6, 3], [7, 5, 8]], [[1, 2, 3], [4, 6, 0], [7, 5, 8]], [[1, 2, 3], [4, 0, 6], [7, 5, 8]], [[1, 2, 3], [4, 5, 6], [7, 0, 8]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[1, 5, 2], [7, 3, 0], [8, 4, 6]], [[1, 5, 2], [7, 0, 3], [8, 4, 6]], [[1, 5, 2], [7, 4, 3], [8, 0, 6]], [[1, 5, 2], [7, 4, 3], [0, 8, 6]], [[1, 5, 2], [0, 4, 3], [7, 8, 6]], [[1, 5, 2], [4, 0, 3], [7, 8, 6]], [[1, 0, 2], [4, 5, 3], [7, 8, 6]], [[1, 2, 0], [4, 5, 3], [7, 8, 6]], [[1, 2, 3], [4, 5, 0], [7, 8, 6]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[1, 5, 2], [0, 7, 3], [8, 4, 6]], [[1, 5, 2], [7, 0, 3], [8, 4, 6]], [[1, 5, 2], [7, 4, 3], [8, 0, 6]], [[1, 5, 2], [7, 4, 3], [0, 8, 6]], [[1, 5, 2], [0, 4, 3], [7, 8, 6]], [[1, 5, 2], [4, 0, 3], [7, 8, 6]], [[1, 0, 2], [4, 5, 3], [7, 8, 6]], [[1, 2, 0], [4, 5, 3], [7, 8, 6]], [[1, 2, 3], [4, 5, 0], [7, 8, 6]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
Puzzle solved using breadth depth first search in 0.015179157257080078 seconds.
```

DFS

```
(base) suman@gole:~/Documents/AI lab/Artificial-Intelligence-college-course$ python -u "/home/suman/Documents/AI lab/Artificial-Intelligence-college-course/PuzzleProblem/dfs.py"
[[[1, 0, 3], [4, 2, 5], [7, 8, 6]], [[1, 2, 3], [4, 0, 5], [7, 8, 6]], [[1, 2, 3], [4, 5, 0], [7, 8, 6]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[1, 2, 3], [7, 8, 0], [6, 4, 5]], [[1, 2, 3], [7, 0, 8], [6, 4, 5]], [[1, 2, 3], [7, 4, 8], [6, 0, 5]], [[1, 2, 3], [7, 4, 8], [0, 6, 5]], [[1, 2, 3], [0, 4, 8], [7, 6, 5]], [[1, 2, 3], [4, 0, 8], [7, 6, 5]], [[1, 2, 3], [4, 8, 0], [7, 6, 5]], [[1, 2, 3], [4, 8, 5], [7, 6, 0]], [[1, 2, 3], [4, 8, 5], [7, 0, 6]], [[1, 2, 3], [4, 0, 5], [7, 8, 6]], [[1, 2, 3], [4, 5, 0], [7, 8, 6]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[1, 2, 3], [5, 7, 6], [4, 0, 8]], [[1, 2, 3], [5, 0, 6], [4, 7, 8]], [[1, 2, 3], [0, 5, 6], [4, 7, 8]], [[1, 2, 3], [4, 5, 6], [0, 7, 8]], [[1, 2, 3], [4, 5, 6], [7, 0, 8]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[2, 3, 6], [1, 4, 0], [7, 5, 8]], [[2, 3, 0], [1, 4, 6], [7, 5, 8]], [[2, 0, 3], [1, 4, 6], [7, 5, 8]], [[0, 2, 3], [1, 4, 6], [7, 5, 8]], [[1, 2, 3], [0, 4, 6], [7, 5, 8]], [[1, 2, 3], [4, 0, 6], [7, 5, 8]], [[1, 2, 3], [4, 5, 6], [7, 0, 8]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[4, 2, 3], [8, 7, 1], [5, 0, 6]], [[4, 2, 3], [8, 0, 1], [5, 7, 6]], [[4, 2, 3], [8, 1, 0], [5, 7, 6]], [[4, 2, 0], [8, 1, 3], [5, 7, 6]], [[4, 0, 2], [8, 1, 3], [5, 7, 6]], [[4, 1, 2], [8, 0, 3], [5, 7, 6]], [[4, 1, 2], [0, 8, 3], [5, 7, 6]], [[4, 1, 2], [5, 8, 3], [0, 7, 6]], [[4, 1, 2], [5, 8, 3], [7, 0, 6]], [[4, 1, 2], [5, 0, 3], [7, 8, 6]], [[4, 1, 2], [0, 5, 3], [7, 8, 6]], [[0, 1, 2], [4, 5, 3], [7, 8, 6]], [[1, 0, 2], [4, 5, 3], [7, 8, 6]], [[1, 2, 0], [4, 5, 3], [7, 8, 6]], [[1, 2, 3], [4, 5, 0], [7, 8, 6]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[1, 2, 3], [0, 5, 7], [4, 8, 6]], [[1, 2, 3], [5, 0, 7], [4, 8, 6]], [[1, 2, 3], [5, 7, 0], [4, 8, 6]], [[1, 2, 3], [5, 7, 6], [4, 8, 0]], [[1, 2, 3], [5, 7, 6], [4, 0, 8]], [[1, 2, 3], [5, 0, 6], [4, 7, 8]], [[1, 2, 3], [0, 5, 6], [4, 7, 8]], [[1, 2, 3], [4, 5, 6], [0, 7, 8]], [[1, 2, 3], [4, 5, 6], [7, 0, 8]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[4, 1, 2], [5, 3, 0], [7, 8, 6]], [[4, 1, 2], [5, 0, 3], [7, 8, 6]], [[4, 1, 2], [0, 5, 3], [7, 8, 6]], [[0, 1, 2], [4, 5, 3], [7, 8, 6]], [[1, 0, 2], [4, 5, 3], [7, 8, 6]], [[1, 2, 0], [4, 5, 3], [7, 8, 6]], [[1, 2, 3], [4, 5, 0], [7, 8, 6]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[1, 0, 5], [7, 3, 6], [8, 2, 4]], [[1, 3, 5], [7, 0, 6], [8, 2, 4]], [[1, 3, 5], [7, 2, 6], [8, 0, 4]], [[1, 3, 5], [7, 2, 6], [8, 4, 0]], [[1, 3, 5], [7, 2, 0], [8, 4, 6]], [[1, 3, 0], [7, 2, 5], [8, 4, 6]], [[1, 0, 3], [7, 2, 5], [8, 4, 6]], [[1, 2, 3], [7, 0, 5], [8, 4, 6]], [[1, 2, 3], [7, 4, 5], [8, 0, 6]], [[1, 2, 3], [7, 4, 5], [0, 8, 6]], [[1, 2, 3], [0, 4, 5], [7, 8, 6]], [[1, 2, 3], [4, 0, 5], [7, 8, 6]], [[1, 2, 3], [4, 5, 0], [7, 8, 6]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[2, 3, 6], [1, 7, 0], [4, 8, 5]], [[2, 3, 6], [1, 7, 5], [4, 8, 0]], [[2, 3, 6], [1, 7, 5], [4, 0, 8]], [[2, 3, 6], [1, 0, 5], [4, 7, 8]], [[2, 3, 6], [1, 5, 0], [4, 7, 8]], [[2, 3, 0], [1, 5, 6], [4, 7, 8]], [[2, 0, 3], [1, 5, 6], [4, 7, 8]], [[0, 2, 3], [1, 5, 6], [4, 7, 8]], [[1, 2, 3], [0, 5, 6], [4, 7, 8]], [[1, 2, 3], [4, 5, 6], [0, 7, 8]], [[1, 2, 3], [4, 5, 6], [7, 0, 8]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[2, 0, 3], [1, 4, 8], [7, 6, 5]], [[0, 2, 3], [1, 4, 8], [7, 6, 5]], [[1, 2, 3], [0, 4, 8], [7, 6, 5]], [[1, 2, 3], [4, 0, 8], [7, 6, 5]], [[1, 2, 3], [4, 8, 0], [7, 6, 5]], [[1, 2, 3], [4, 8, 5], [7, 0, 6]], [[1, 2, 3], [4, 0, 5], [7, 8, 6]], [[1, 2, 3], [4, 5, 0], [7, 8, 6]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
Puzzle solved using depth first search in 0.04974865913391113 seconds.
(base) suman@gole:~/Documents/AI lab/Artificial-Intelligence-college-course$
```

Iterative Deepening Search

```
(base) suman@gole:~/Documents/AI lab/Artificial-Intelligence-college-course$ python -u "/home/suman/Documents/AI lab/Artificial-Intelligence-college-course/PuzzleProblem/iterativedeepening.py"
[[[1, 5, 2], [4, 8, 3], [7, 0, 6]], [[1, 5, 2], [4, 0, 3], [7, 8, 6]], [[1, 0, 2], [4, 5, 3], [7, 8, 6]], [[1, 2, 0], [4, 5, 3], [7, 8, 6]], [[1, 2, 3], [4, 5, 0], [7, 8, 6]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[1, 2, 3], [0, 4, 6], [7, 5, 8]], [[1, 2, 3], [4, 0, 6], [7, 5, 8]], [[1, 2, 3], [4, 5, 6], [7, 0, 8]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[1, 5, 2], [4, 8, 3], [7, 0, 6]], [[1, 5, 2], [4, 0, 3], [7, 8, 6]], [[1, 0, 2], [4, 5, 3], [7, 8, 6]], [[1, 2, 0], [4, 5, 3], [7, 8, 6]], [[1, 2, 3], [4, 5, 0], [7, 8, 6]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[2, 6, 5], [1, 7, 3], [4, 0, 8]], [[2, 6, 5], [1, 0, 3], [4, 7, 8]], [[2, 0, 5], [1, 6, 3], [4, 7, 8]], [[2, 5, 0], [1, 6, 3], [4, 7, 8]], [[2, 5, 3], [1, 6, 0], [4, 7, 8]], [[2, 5, 3], [1, 0, 6], [4, 7, 8]], [[2, 0, 3], [1, 5, 6], [4, 7, 8]], [[0, 2, 3], [1, 5, 6], [4, 7, 8]], [[1, 2, 3], [0, 5, 6], [4, 7, 8]], [[1, 2, 3], [4, 5, 6], [0, 7, 8]], [[1, 2, 3], [4, 5, 6], [7, 0, 8]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[4, 0, 3], [2, 1, 5], [7, 8, 6]], [[4, 1, 3], [2, 0, 5], [7, 8, 6]], [[4, 1, 3], [0, 2, 5], [7, 8, 6]], [[0, 1, 3], [4, 2, 5], [7, 8, 6]], [[1, 0, 3], [4, 2, 5], [7, 8, 6]], [[1, 2, 3], [4, 0, 5], [7, 8, 6]], [[1, 2, 3], [4, 5, 0], [7, 8, 6]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[2, 0, 3], [1, 4, 6], [7, 5, 8]], [[0, 2, 3], [1, 4, 6], [7, 5, 8]], [[1, 2, 3], [0, 4, 6], [7, 5, 8]], [[1, 2, 3], [4, 0, 6], [7, 5, 8]], [[1, 2, 3], [4, 5, 6], [7, 0, 8]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[1, 2, 3], [0, 4, 6], [7, 5, 8]], [[1, 2, 3], [4, 0, 6], [7, 5, 8]], [[1, 2, 3], [4, 5, 6], [7, 0, 8]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[1, 3, 6], [4, 5, 0], [7, 8, 2]], [[1, 3, 6], [4, 5, 2], [7, 8, 0]], [[1, 3, 6], [4, 5, 2], [7, 0, 8]], [[1, 3, 6], [4, 0, 2], [7, 5, 8]], [[1, 3, 6], [4, 2, 0], [7, 5, 8]], [[1, 3, 0], [4, 2, 6], [7, 5, 8]], [[1, 0, 3], [4, 2, 6], [7, 5, 8]], [[1, 2, 3], [4, 0, 6], [7, 5, 8]], [[1, 2, 3], [4, 5, 6], [7, 0, 8]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[4, 0, 3], [2, 1, 6], [7, 5, 8]], [[4, 1, 3], [2, 0, 6], [7, 5, 8]], [[4, 1, 3], [0, 2, 6], [7, 5, 8]], [[0, 1, 3], [4, 2, 6], [7, 5, 8]], [[1, 0, 3], [4, 2, 6], [7, 5, 8]], [[1, 2, 3], [4, 0, 6], [7, 5, 8]], [[1, 2, 3], [4, 5, 6], [7, 0, 8]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
[[[4, 1, 3], [7, 2, 5], [8, 0, 6]], [[4, 1, 3], [7, 2, 5], [0, 8, 6]], [[4, 1, 3], [0, 2, 5], [7, 8, 6]], [[0, 1, 3], [4, 2, 5], [7, 8, 6]], [[1, 0, 3], [4, 2, 5], [7, 8, 6]], [[1, 2, 3], [4, 0, 5], [7, 8, 6]], [[1, 2, 3], [4, 5, 0], [7, 8, 6]], [[1, 2, 3], [4, 5, 6], [7, 8, 0]]]
Puzzle solved using iterative depth first search in 0.01089627742767334 seconds.
(base) suman@gole:~/Documents/AI lab/Artificial-Intelligence-college-course$
```

Genetic Algorithm

```
(base) suman@gole:~/Documents/AI lab/Artificial-Intelligence-college-course$ python -u "
/home/suman/Documents/AI lab/Artificial-Intelligence-college-course/geneticalgorithm.py"
>0, new best f([0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 0, 0, 0, 1]) = -9.000
>0, new best f([1, 0, 1, 1, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 1]) = -11.000
>0, new best f([0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 1, 0, 1, 0, 0, 1, 0, 1]) = -12.000
>0, new best f([1, 0, 1, 1, 0, 1, 1, 0, 0, 0, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0]) = -13.000
>0, new best f([0, 1, 1, 1, 0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1]) = -14.000
>0, new best f([1, 0, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1]) = -15.000
>1, new best f([1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 0, 1, 1]) = -16.000
>2, new best f([1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1]) = -17.000
>4, new best f([1, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]) = -18.000
>5, new best f([1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]) = -19.000
>9, new best f([1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]) = -20.000
Done!
f([1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]) = -20.000000
(base) suman@gole:~/Documents/AI lab/Artificial-Intelligence-college-course$
```

Neural Network

```
(base) suman@gole:~/Documents/AI lab/Artificial-Intelligence-college-course$ python -u "
/home/suman/Documents/AI lab/Artificial-Intelligence-college-course/neuralnetwork.py"
>epoch=0, lrate=0.500, error=6.350
>epoch=1, lrate=0.500, error=5.531
>epoch=2, lrate=0.500, error=5.221
>epoch=3, lrate=0.500, error=4.951
>epoch=4, lrate=0.500, error=4.519
>epoch=5, lrate=0.500, error=4.173
>epoch=6, lrate=0.500, error=3.835
>epoch=7, lrate=0.500, error=3.506
>epoch=8, lrate=0.500, error=3.192
>epoch=9, lrate=0.500, error=2.898
>epoch=10, lrate=0.500, error=2.626
>epoch=11, lrate=0.500, error=2.377
>epoch=12, lrate=0.500, error=2.153
>epoch=13, lrate=0.500, error=1.953
>epoch=14, lrate=0.500, error=1.774
>epoch=15, lrate=0.500, error=1.614
>epoch=16, lrate=0.500, error=1.472
>epoch=17, lrate=0.500, error=1.346
>epoch=18, lrate=0.500, error=1.233
>epoch=19, lrate=0.500, error=1.132
[{'weights': [-1.4688375095432327, 1.850887325439514, 1.0858178629550297], 'output': 0.0
29980305604426185, 'delta': 0.0059546604162323625}, {'weights': [0.37711098142462157, -0
.0625909894552989, 0.2765123702642716], 'output': 0.9456229000211323, 'delta': -0.002627
9652850863837}]
[{'weights': [2.515394649397849, -0.3391927502445985, -0.9671565426390275], 'output': 0.
23648794202357587, 'delta': 0.04270059278364587}, {'weights': [-2.5584149848484263, 1.00
36422106209202, 0.42383086467582715], 'output': 0.7790535202438367, 'delta': -0.03803132
596437354}]
(base) suman@gole:~/Documents/AI lab/Artificial-Intelligence-college-course$
```