

INF3490 Oblig 2

joakikt

October 2018

1 The program

Made with python 3

Requirements:

numpy

pandas (For matrix)

scikit-learn (For accuracy function)

Running:

```
python3 movements.py
```

2 Tasks

I implemented using sequential training and the output is a list of values where the index of the neuron with the highest value is the motion class. I created a class Neuron that holds the weights to the next layer and the value of the neuron itself, and a class Layer that contains all neurons + bias neuron.

6 hidden neurons:

Model accuracy: 0.7657657657657657

Predicted	0	1	2	3	4	5	6	7	All
Actual									
0	14	0	0	0	0	0	0	1	15
1	0	15	0	0	0	0	0	0	15
2	0	0	12	0	0	0	0	0	12
3	0	1	1	4	6	0	1	2	15
4	1	0	0	0	12	0	0	0	13
5	0	1	0	0	0	3	6	0	10
6	0	1	0	0	0	2	12	0	15
7	2	0	0	1	0	0	0	13	16
All	17	18	13	5	18	5	19	16	111

8 hidden neurons:

Model accuracy: 0.7927927927927928

Predicted	0	1	2	3	4	5	6	7	All
Actual									
0	10	0	1	0	1	0	0	2	14
1	0	17	0	0	0	0	0	1	18
2	0	0	13	0	0	0	0	0	13
3	0	0	1	5	0	0	0	1	7
4	0	0	0	2	12	0	0	0	14
5	0	1	0	0	0	12	5	0	18
6	0	0	0	0	0	2	6	0	8
7	1	1	0	4	0	0	0	13	19
All	11	19	15	11	13	14	11	17	111

12 hidden neurons:

Model accuracy: 0.8558558558558559

Predicted	0	1	2	3	4	5	6	7	All
Actual									
0	8	0	0	0	0	0	0	0	8
1	0	18	0	0	0	0	0	0	18
2	0	0	10	0	0	0	0	0	10
3	0	0	2	8	3	0	0	4	17
4	0	0	0	0	16	0	0	0	16
5	0	1	0	0	0	9	1	0	11
6	0	0	1	0	0	1	14	0	16
7	1	0	0	2	0	0	0	12	15
All	9	19	13	10	19	10	15	16	111

How many nodes do you find sufficient for the network to classify well?

Judging by these 3 runs it seems more hidden nodes will increase the accuracy of the MLP

By only looking at your reported confusion tables, which classes were likely to be mistaken for each other?

Class 3 and 4 could be confused with each other. In 6 neurons hidden they have almost the same predictions, and even as more neurons are added both 3 and 4 gets mixed up