

Human Psychophysics

Submitted by Selma Boudissa

13/05/2018

Contents

I	Construction of the network	3
II	Training data	4
III	Personnal test set	4
IV	Bonus	4

Introduction

For this project I was suppose to create a classification control-patient using Kohonen network, to accomplish this task I code using Matlab.

Part I

Construction of the network

The first step consist on the construction of the network using a toy example, as you see in my code they are declared under the variables name 'Tr' and 'Te' forllowing the guidelines given in the pdf file "Part 1" on my guithub repository.[1]

For this part I created a function called 'kohonen_network.m'.

Question 1: Training with the vectors

The goal was to train the kohonen network with the vectors [1 1 0 0; 1 0 0 0; 0 0 0 1; 0 0 1 1] in order to carry out the classification of this one, after several test I fixed the learning rate at 0.72 (0.9 * 0.8) cause this value allows a convergence.

At the end the vector (1 1 0 0) and (1 0 0 0) should fall in class I while the others should fall in class II. The Figure 1 below show the correct result I obtain in my command window.

```
Train Data for Question 1:

Tr =

    1    1    0    0
    1    0    0    0
    0    0    0    1
    0    0    1    1

Results:
[1  1  0  0 ] This Vector Belongs to Class 1
[1  0  0  0 ] This Vector Belongs to Class 1
[0  0  0  1 ] This Vector Belongs to Class 2
[0  0  1  1 ] This Vector Belongs to Class 2
```

Figure 1: Training the vectors

Question 2: Test with the vectors

Once I train the network is time for test using another set of vectors [0 0 0 0.9; 0 0 0.8 0.9; 0.7 0 0 0; 0.7 0.9 0 0] theoritically the result should be the vectors (0 0 0 0.9) and (0 0 0.8 0.9) should fall in class II while the rest in class I, as you can see I obtain the correct result in Figure 2.

```
Test Data for Question 2:

Te =

      0      0      0      0.9000
      0      0      0.8000      0.9000
    0.7000      0      0      0
    0.7000      0.9000      0      0

Results:
[0      0      0      0.9 ] This Vector Belongs to Class 2
[0      0      0.8      0.9 ] This Vector Belongs to Class 2
[0.7      0      0      0 ] This Vector Belongs to Class 1
[0.7      0.9      0      0 ] This Vector Belongs to Class 1
```

Figure 2: Test the vectors

Part II

Training data

Now that I train and test my kohonen network with the toy examples vectors I have to do it using the training dataset 'control.txt' and 'patient.txt'. Basically I will use the same code for the first part and load my two text files, you will found it on 'kohonen_network2.m'.

Part III

Personnal test set

The last part of the project each student get an individual test dataset called 'test_five.txt' to be able to classify each 4 subject as control or patient. For this I right a function called 'kohonen_network3' my function is able to determine for each test if it's a control or a patient. In the Figure 3 below of my command windows you will see the resulted output of my personnal test dataset.

```
Part 3:
Result
Test Vector 1 Belongs to Patient
Test Vector 2 Belongs to Control
Test Vector 3 Belongs to Control
Test Vector 4 Belongs to Control
```

Figure 3: Classification patient-control on my individual dataset

Part IV

Bonus

Question:

What are the principle differences if any between a bio-inspired algorithm like the Kohonen SOM and two other well-known similar clustering algorithms – the kmeans and k nearest neighbour algorithm?

Answer:

There are several differences regarding what you are interested in.

- One of the difference is that you need to specify in advance the number of clusters you desire with th Kmeans algorithms compare to Kohonen SOM where you don't need too.
- SOM provide more robust learning and can be implemented in various ways
- Kmeans is more sensitive to the noise in the dataset compared to SOM.

References

1. Guithub repository: <https://github.com/SelmaBoubou/Human-Psychophysic-Project--Classification>
2. Virtual reality as sharp as the human eye can see: <http://www.bbc.com/news/business-42963408>