Exercise neural networks BIOS13

You should by now be familiar with at least simple programming in R. The very simplest artificial neural network (ANN) is the single layer perceptron. If you have forgotten what it looks like, check the lecture notes. The exercise is divided into two levels, i) basic, ii) advanced, and a extracurricular part using MATLAB for ones that are interested. It might be good to work in pairs so that you can discuss the questions with someone.

---

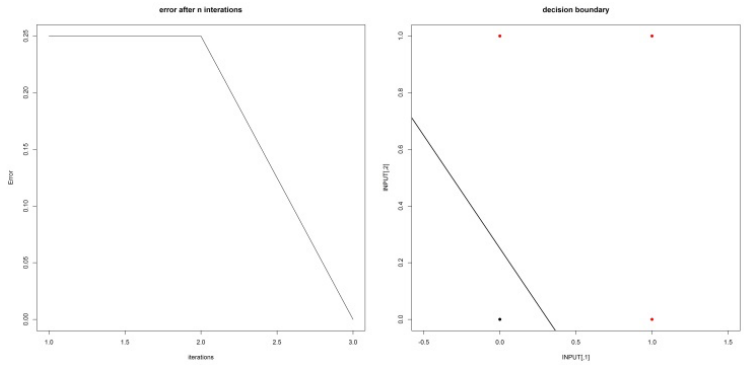
i) Basic level

1. Download the file Perceptron.R from the course web page and save it in a suitable directory. Open it in Rstudio (R: File > open script) or in R as a script: File > Open File).

This file is an Rscript that will run the perceptron algorithm on a batch of four sets of two inputs of 1s and 0s. In order to run a script or a function in R, use the source command in the R console (make sure you have the right working directory).

```
> source('Perceptron.R')
```

2. Run the script in R and observe the output. You should see a **plot** similar to the one below (your lines will be different). If your script made a plot, it is working properly.



The left pane shows the mean error of the perceptron as it starts from the first random setting of the weights and how it changes with new weight values. The perceptron should stop when the mean error is 0. When the mean error is 0 the four points (0 0, 1 0, 0 1, 1 1) are correctly separated according to the logical (boolean) operator. The line that did this separation is shown in the right pane.

2. A good strategy for understanding the code and answering the questions below will be to tinker with variables and inserting strategic “print()” statements into the code. You can also use the Debugging feature.

3. What is the training task (logical operator) that the perceptron is trained on in the script? Hint: Look at the initiation (=first creation) of the vector called CORRECT_OUTPUT

4. Do you understand the format of the input? Compare the matrix called INPUT with the lecture notes.

5. How do we measure the success of the algorithm?
Hint: what is stored in the variable TOTAL_ERROR and how are these values converted into the number that is stored in SaveOutput vector?

6. Run the script. How many generations did the perceptron need to find the solution (=to reach zero error)?

7. Run the script again. Did you get the same answer as in question 6? If so, run it again. Why may the number of iterations differ for the same problem?
Hint: what is stored in the W vector when it is initiated?

8. Run the code 10 times, what is the average generation time for solving this task?
Hint: now might be a good time write a for loop…

9. Change the code so that the perceptron is trained on the AND operator.
Hint: change the numbers in the CORRECT_OUTPUT variable.

10. Run the code 10 times, what is the average generation time?

11. Which task takes longest to solve ”OR” or ”AND”? Why? The task is the same, to draw a line somewhere in the coordinate system for x and y between 0 and 1.
Hint: in what range are the weights first set?

ii) Advanced level

12. What learning rule is used?

13. Look up the NAND (Not AND) operator and write it down as a truth table. How does it compare to the AND operator?

14. Now train the perceptron on the XOR operator. How good was the perceptron? What happens, why does the mean error reach 1 and why does it not stay at the minimal error value (0.25)? Hint: if you look at the code, the loop will stop after 25 generations. What is the alternative stop condition?

iii) Extracurricular

15. If you have a student laptop from the ecology building, open MATLAB. If you have your own computer Download MATLAB if you have not already got it, it is free for LU students. You can download it with your STIL identity from: http://program.ddg.lth.se/

Programming in MATLAB is very similar to programming in R. Type *nnd4db* at the command prompt. This should produce a pedagogic illustration of the perceptron.

a) Move the handles of the separation line so that it separates the black dot from the three white ones. Which operator is the perceptron acting on?

b) Grab the black dot to the left of the yellow pane and change the training operator to OR. In what direction would you move the separation line so it solves the OR condition?

c) Move the handles of the separation line at the same time as you look on the line created by the weight values (called w). How do these relate to each other?

d) Move the separation line so that it passes exactly through the point x = 0 and y = 0. What is the value of the bias (b) now? Can the perceptron solve the conditions AND and OR without a bias?