FIT5186 Intelligent Systems

Sample Exam Questions and Answers for Lectures 1-4.

Question 1

What are the "indecision regions" created by Perceptrons? Explain in words why they might exist in R-category Perceptrons.

(2 marks)

"Indecision regions" are regions where points (data or inputs) are classified into one or more classes, or regions where points (data or inputs) are not classified at all.

R-category Perceptrons separates the input space into regions corresponding to each group by R straight lines and it creates "indecision regions" on the input space.

Question 2

How do you convert the firing threshold of a neuron to be at zero?

(1 mark)

By introducing an additional input to the neuron which always has a value of -1 and whose weight is the value of the firing threshold.

Question 3

Why is it useful to have the firing threshold to be at zero for all neurons?

(1 mark)

It is useful to simplify the model by eliminating the possibility of a different threshold value for each neuron.

Question 4

What is the “credit assignment problem”? How can it be resolved?

(2 marks)

In a multilayered Perceptron, the Perceptrons in the output layer do not know what the effect of the hidden layer weights is (since their input signal comes from a combination of the output layer weights and the outputs of the hidden layer Perceptrons). As such it cannot assign credit or blame to these weights (in order to determine how they should be modified).

The solution is to use a continuous activation function rather than a discrete activation function.

Question 5

Why is it important to reserve some of the data for a test set prior to training supervised neural networks?

(2 marks)

To examine the generalisability of the trained neural network.

It is important that a trained neural network performs equally well on new data that it has not been trained on. That is, the neural network is able to generalize its learning rather than memorising the data.

In order that we have confidence that a trained neural network will perform well on new data, it is important to reserve some data for testing.

Question 6

The Delta rule is not applied to modify the hidden layer weights W in a MFNN. Why is this?

(2 marks)

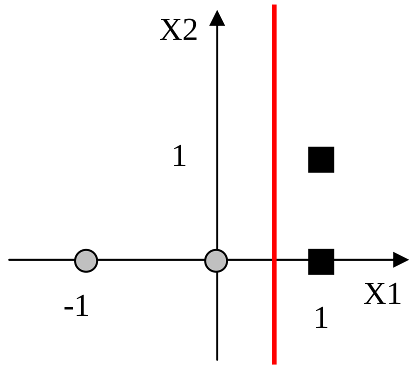
Because we don't know the desired outputs of the hidden neurons.

In order to calculate the weight changes needed for the hidden layer weights, we need to know the desired outputs of the hidden neurons so that we can measure the error at the hidden neurons using the Delta rule. The problem is that we don't know the desired values of the hidden neuron outputs, we only know the desired values of the final neurons. So the Delta rule cannot be directly applied to the hidden layer weights.

Question 7

Suppose we have a two inputs (X1 and X2) Classification problem as shown in the table below. This problem is linearly separable as shown when plotting the four data points in the figure below.

X1	X2	Desired Output
-1	0	0
0	0	0
1	0	1
1	1	1



a) Draw a line (there are infinitely many) on the figure above to dicotomise the two classes.
b) Write down the equation of the line.

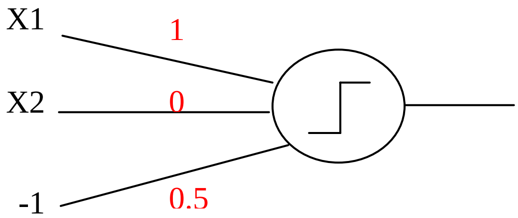
X1 = 0.5 or X1 – 0.5 = 0

c) Rearrange the equation of the line to determine the weights of the Perceptron below, and write the weights on the Perceptron. Be sure the check that the correct output is obtained for the inputs in the table above.

X1 – 0.5 = 0

1 * X1 + 0 * X2 + 0.5 * (-1) = 0

W1 = 1; W2 = 0; T = 0.5

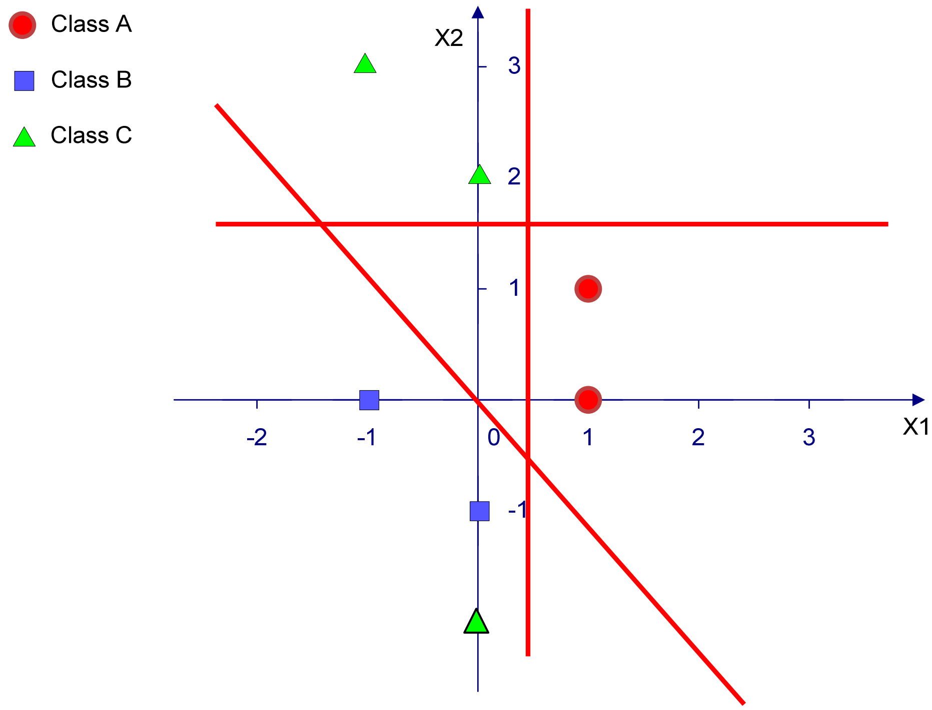


(3 marks)

Question 8

Suppose there are three classes containing the following data points:

CLASS A: (1,0), (1,1)   CLASS B: (-1,0), (0, -1)   CLASS C: (0,2), (-1, 3)



a) Design a classifier (using either equations or Perceptrons) based on the above data.
b) For each of the inputs list the desired output (response) of the network.
c) Suppose an additional data point (0, -2) is added to CLASS C. How does this change the architecture of the classifier? Discuss the consequences of adding this extra data points, including any necessary changes to the learning rule and network characteristics. (You don't need to find the weights of the network for part c)

(7 marks)

(a) Design a classifier based on the above data

Class A:

$X_1 = 0.5 \rightarrow X_1 + 0 * X_2 - 0.5 = 0 \rightarrow (1)*X_1 + (0)*X_2 + (0.5)*(-1) = 0$

Class B:

$X_1 = - X_2 \rightarrow X_1 + X_2 = 0 \rightarrow - X_1 - X_2 = 0 \rightarrow (-1)* X_1 + (-1)* X_2 + (0)*(-1) = 0$

Class C:

$X_2 = 1.5 \rightarrow 0 *X_1 + X_2 - 1.5 = 0 \rightarrow (0) * X_1 + (1)* X_2 + (1.5)*(-1) = 0$

Discriminant functions

$g_A (X_1, X_2) = (1)*X_1 + (0)* X_2 + (0.5)*(-1) = 0$

$g_B (X_1, X_2) = (-1)* X_1 + (-1)* X_2 + (0)*(-1) = 0$

$g_C (X_1, X_2) = (0)* X_1 + (1)* X_2 + (1.5)*(-1) = 0$

(b) For each of the inputs list the desired output (response) of the network.

1	0	-1	1
			0
			0
1	1	-1	1
			0
			0
-1	0	-1	0
			1
			0
0	-1	-1	0
			1
			0
0	2	-1	0
			0
			1
-1	3	-1	0
			0
			1

(c) Suppose an additional data point (0,-2) is added to CLASS C. How does this change the architecture of the classifier? Discuss the consequences of adding this extra data points, including any necessary changes to the learning rule and network characteristics. (You don't need to find the weights of the network for part c)

The problem becomes linearly non-separable. The classifier needs to have a multilayered architecture. To modify the weights of a multilayered classifier, the backpropagation learning rule is required.