COMS 4030A

Adaptive Computation and Machine Learning

Assignment 1

This assignment counts 10% towards your final mark.
The submission is to be done on moodle and will be graded by the autograder which also uses a plagiarism checker.
Submissions will be accepted until Monday 17 April at 17h00.

In this assignment you are required to create a Python program that does the following:
(Note: You may **not** use any Python machine learning libraries.)

(a) Implement a neural network with 3 layers with the following specifications:

the input layer has 4 nodes
the hidden layer has 6 nodes
the output layer has 3 nodes
all nodes in the hidden layer and output layer use sigmoid activation function
all weights are initialised to 1
all bias values are initalised to 1

(b) You need to implement the feedforward step to compute the output of the network for some given inputs.

(c) You need to implement the sum-of-squares loss computation between the output and target.
(Recall: sum-of-squares loss is $L(\boldsymbol{y}, \boldsymbol{t}) = \frac{1}{2}\sum_{j=1}^{k}(y_j - t_j)^2$.)

(d) You need to implement the backpropagation method for updating the weights and biases of the network. Use a learning rate of 0.1.

Your Python **submission** to moodle must do the following:

(1) Read in from standard input a list of seven numbers, such as
-3
2
1
-1
1
0
0
The first 4 values are the input to the network and the last 3 are the corresponding targets.

(2) Feedforward the input values to obtain output values and compute the sum-of-squares loss with respect to the target values.

(3) Perform one iteration of backpropagation.

(4) After that, feedforward the same input values into the updated network to get new output values, and compute a new loss value.

(5) The following values must be output using standard output:
the loss before training and the loss after training, rounded to 4 decimal places.
For the above input, the output should be:
0.9645
0.9636

Here is another example:
input
1
0
-2
-3
0
0.7
0.3

output
0.4269
0.4235