

Assignment 3: Neural Networks

CS786 – Spring 2012

Out: Tuesday, July 17, 2012

Due: Monday, July 30, 2012, at midnight

This assignment consists entirely of programming questions to be done in the Java programming language. You will implement some basic functions to evaluate and train feed forward neural networks. A skeleton jar file that includes the definition of a Neural Net class and the signature of the methods that you need to implement will be posted on the course website shortly. Your job is to fill in the methods described below. The Marmoset server will automatically compile your code and run various tests to verify its correctness. Your grade will be the number of release tests passed by your code.

1. **outputs = neuralNetwork.eval(inputs):** function that computes the outputs of a feed forward neural network for a set of inputs.
2. **outputAtEachUnit = neuralNetwork.forwardPhase(inputs):** function that propagates forward the outputs at each unit of a neural network for a set of inputs.
3. **deltas = neuralNetwork.backwardPhase(outputAtEachUnit,desiredOutputs):** function that propagates backwards the deltas at each unit based on the output at each unit and the desired outputs of the neural network.
4. **partialDerivatives = neuralNetwork.gradient(inputs,desiredOutputs):** function that computes the gradient of the weights for a data point. This function should use the forwardPhase and backwardPhase functions.
5. **updatedWeights = neuralNetwork.backProp(dataset,nIterations,stepLength):** function that trains a feed forward neural network by error back propagation. This function should use the gradient function. Since convergence is difficult to detect, nIterations are performed where in each iteration the weights are updated by taking a step in the direction of the gradient for each instance of the dataset.