

CENG 463 Spring 2017

Programming Exercise 3: Two-Layer Neural Network:

Forward Pass, Backpropagation and Weight Update

Introduction

In this exercise, you will develop a neural network with one fully-connected hidden layer to perform a number of forward and backward passes to train the network parameters. The ultimate goal of the network will be to make a 5-class classification.

Files included in this exercise

[?] NN.py - Python script that will help you through the exercise.

? indicates the files you will need to complete and submit

Throughout the exercise, you will be using the script NN.py. This script will help you through implementation of the neural network. For this homework, you will need to modify and submit NN.py. Please also write your name and student number into NN.py.

Submission

You will submit the homework via CMS. There is a homework submission link, clicking to which will lead you a file browser that you can choose your file. Please zip your files before submission and name the file as yourstudentno-hw3.zip. Submit the homework until 31 May 2017 23:55.

Network Architecture

The neural network will consist of an input layer of size 200, a hidden layer of 30 neurons and an output layer. It will be fully connected. The sigmoid function will be used as the activation function.

Weight Initialization

The weights will be initialized randomly, by sampling from a standard Normal distribution of 0 mean and unit variance. Biases will be initialized to zero.

At each iteration:

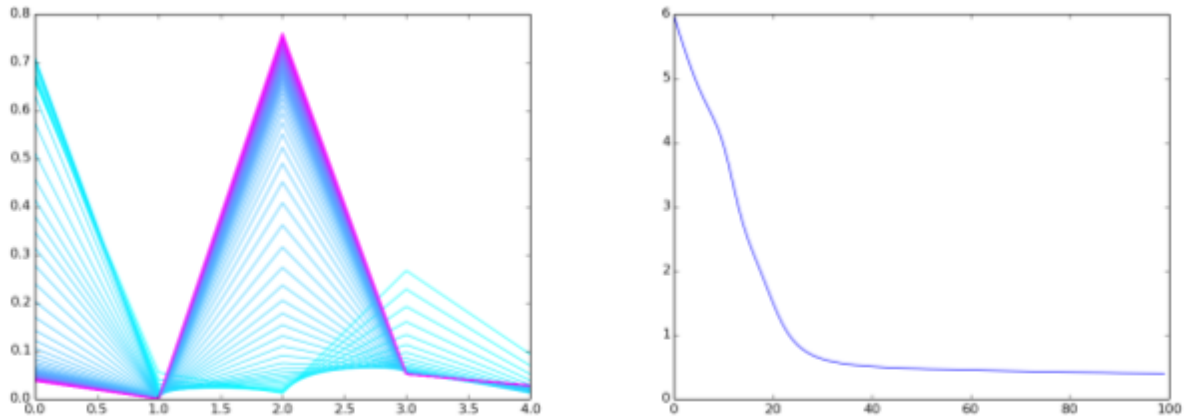
Forward Pass and Loss Calculation

You will perform the forward pass, computing the class scores for the input. Next, you will calculate the cost. Both class scores and cost will be recored in a list for plotting purposes.

Backpropagation

The error will be backpropagated and the gradients for the network parameters (weights and biases) will be calculated. The parameters will be updated using gradient descent.

You are expected to obtain the following plots at the end:



You can find more details in the Python script.

- All through the implementation, both forward and backward propagation have to be done in **VECTORIZED** fashion.
- You are allowed to use the for loop **ONLY** for **calcCost** function.
- Add your code between **## YOUR CODE HERE ##** and **#####** lines. Do NOT change anything elsewhere.