ASSIGNMENT 7, due December 1

Homeworks are due in class or in my mail box by 2pm on the due date. LFD is the class textbook

1. **(200) Gradient Descent on a "Simple" Function**

   Consider the function $f(x, y) = x^2 + 2y^2 + 2\sin(2\pi x)\sin(2\pi y)$.

   (a) Implement gradient descent to minimize this function. Let the initial values be $x_0 = 0.1; y_0 = 0.1$, let the learning rate be $\eta = 0.01$ and let the number of iterations be 50; Give a plot of the how the function value drops with the number of iterations performed.

   Repeat this problem for a learning rate of $\eta = 0.1$. What happened?

   (b) Obtain the "minimum" value and the location of the minimum you get for gradient descent using the same η and number of iterations as in part (a), starting from the following initial points: $(0.1, 0.1), (1, 1), (-0.5, -0.5), (-1, -1)$. A table with the location of the minimum and the minimum values will suffice. You should now appreciate why finding the "true" global minimum is a hard problem.

2. **(200) Neural Networks and Backpropagation**

   Write a program in your favorite language (even matlab code would be fine) to implement gradient descent for a 2 input ($d^{(0)} = 2$), N_H-hidden unit ($d^{(1)} = N_H$), 1 output sigmoidal neural network (L=2). For the output node activation function, allow for both $S(x) = x$ and $S(x) = \tanh(x)$. Implement gradient decent on the squared error $E_{\text{in}}(\mathbf{w}) = \frac{1}{4N}\sum_{n=1}^{N}\left(h(x_n, \mathbf{w}) - y_n\right)^2$, and check your gradient calculation as follows:

   (a) Use a network with $N_H = 2$. Set all the weights to 0.25 and consider a data set with 1 point: $x_1 = x_2 = 1$; $y = 1$. For both the identity and $\tanh(\cdot)$ output node transformation functions, obtain the gradient of $E_{\text{in}}(\mathbf{w})$ using the backpropagation algorithm. Report this result - there should be as many numbers as parameters in this network.

   (b) Now, obtain the gradient numerically by peturbing each weight in turn by 0.0001. Report this result. (Hint: If this result is not similar to your previous result then there is something wrong with your backpropagtion gradient calculation.)

3. **(400) Neural Network for Digits**

   Use your neural network implementation with 10 hidden units to build a classifier for separating the digits 1 from 5. Use the two features and data you developed in a previous assignment. Sample 500 training data points randomly and initialize the weights to small random values. For this problem, train with the linear output transformation function but use the sign threshold on the output node for actually classifying the data.

   (a) Plot $E_{\text{in}}(\mathbf{w})$ versus iteration for the variable learning rate gradient descent heuristic and 2×10^6 iterations, for both $S(x) = x$ and $S(x) = \tanh(x)$. Also show the resulting classifiers.

   (b) Now use weight decay with $\lambda = 0.01/N$ and use variable learning rate gradient descent to minimize the augmented error. Show the resulting classifier.

   (c) Now use early stopping with a validation set of size 50 and training set of size 450, and show the resulting classifier.