Machine Learning (Homework #3)

Due date: 1/8

(Notice: Please upload only two files. One is your source code (compressed file), and the other is your paper. The filename of your paper should name as "studentID_name_HW3.pdf")

1. Logistic Regression

You are given the Wine data set (x.mat t.mat). This data set contains 3 classes. The first 3 dimensions in x.mat is the values of 1-of-K coding for a target. In this exercise, you will implement the ***Newton-Raphson*** algorithm to construct a multiclass logistic regression model with the softmax transformation as.

$p(C_k|\phi) = y_k(\phi) = \frac{\exp(a_k)}{\Sigma_j \exp(a_j)}$ The error function is formed by using the cross-entropy error function as. $E(\mathbf{w}) = -\sum_{n=1}^{N} \sum_{k=1}^{k} \{t_{nk} \ln y_{nk}\}$

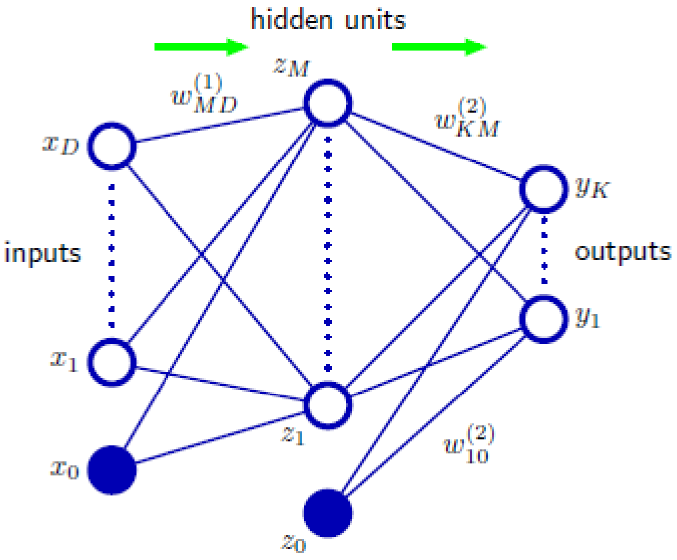
Note: **You have to set a stopping criterion** $||\nabla E(\mathbf{w})||^2 < \epsilon$.

(1) Set the initial **w** to be zero, and plot $||\nabla E(\mathbf{w})||^2$ versus number of epochs until convergence.

(2) Show the classification result.

No. sample	1	2	…	29	30
$p(C_1|\phi)$					

2. Neural Network



In this exercise, you will implement a feed-forward neural network for 10-class classification problem on MNIST handwriting digit data set. You can obtain the data in the following links

1. http://yann.lecun.com/exdb/mnist/train-images-idx3-ubyte.gz
2. http://yann.lecun.com/exdb/mnist/train-labels-idx1-ubyte.gz
3. http://yann.lecun.com/exdb/mnist/t10k-images-idx3-ubyte.gz
4. http://yann.lecun.com/exdb/mnist/t10k-labels-idx1-ubyte.gz

where 1 and 2 are training data, 3 and 4 are test data. To save your work, you may call `DataPrep` to read the data set. This will help you to convert the images to black and white color with 400 dimensional column vectors. Totally there are 50,000 instances for training and 10,000 for testing. Please implement the error back-propagation algorithm and construct a neural network by minimizing the cross-entropy error function as follows:

$$E(\mathbf{w}) = -\sum_{n=1}^{N}\sum_{k=1}^{K} t_{nk} \ln y_k(\mathbf{x}_n, \mathbf{w})$$

It is convenient to use a 1-of-K coding scheme to indicate the target class. **Please show the figures** including the cross-entropy versus number of epochs and the misclassification rate versus number of epochs. An example is shown in Figure 1. An epoch means running over all training data points. You have to decide the number of hidden layers, the number of hidden units, the learning rate, and the type of activation function in the hidden layer in advance. **Also show the misclassification rate of test data.**

(1) Build a neural network with two hidden layer.

(2) Set the activation function in hidden layer as a rectifier linear unit (ReLU), which is $z_j = \max(a_j, 0)$.

(3) In practice, minibatch stochastic gradient descent is very popular technique to train the neural network, which means we sequentially feed in a small size of data and calculate gradient based on the small size of data. Please set the batch size as 50.

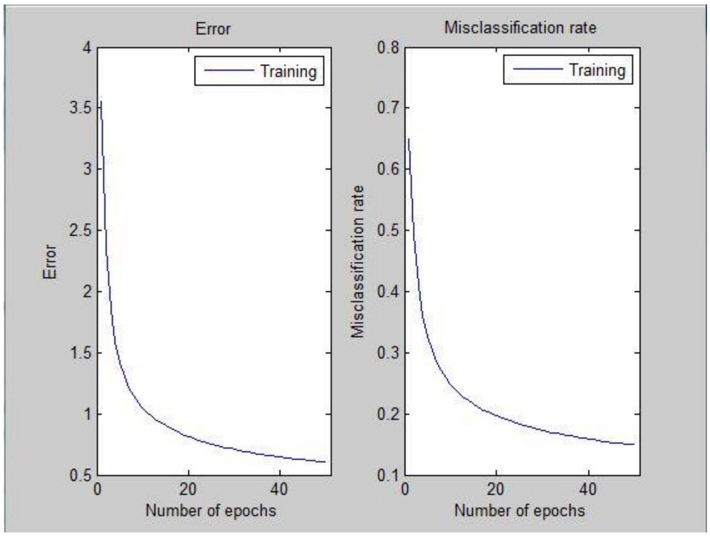

Error
Training
4
3.5
3
2.5
2
1.5
1
0.5
Error
0
20
40
Number of epochs
Misclassification rate
Training
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
Misclassification rate
0
20
40
Number of epochs

Figure 1