In this homework, you will implement an artificial neural network (ANN) classifier. You should learn the weights of this classifier implementing the backpropagation algorithm. In your implementation, you SHOULD NOT use pretraining, which means you should select the initial weights randomly. You will use pretraining in your third homework.

You will run your ANN classifier on two datasets: the “Thyroid data set”, which you also used for the first homework, and the MNIST dataset of handwritten digits, whose details are given in http://yann.lecun.com/exdb/mnist/. This homework has five parts. Complete the first four parts (from Part 1 to Part 4) for the Thyroid data set and the last one (Part 5) for the MNIST dataset.

Part 1:
Implement a neural network classifier with a single hidden layer. In this part,

(a) Write down the update rules that your implementation uses. Specify the loss function.

(b) List all the parameters (except the hidden unit number) that your implementation uses. Explain how you select the parameter values. Here you are not required to conduct cross-validation to select the values (of course, if you’d like, you can). Just explain why you select these values.

(c) Explain how you manage the situation of having unbalanced class distributions. Also explain if you use any normalization on the features.

(d) For different values of the hidden unit number (at least five different values), obtain training and test set accuracies. Report the overall accuracy as well as the class-based accuracies. You may report these accuracies in a “nice” table or you may draw them in a plot.

(e) Fix the hidden unit number to a value that gives good class-based test set accuracies. Now conduct the following experiments with your selected hidden unit number.
   - Obtain the results with and without normalization. Compare the overall test set accuracy and the class-based test set accuracies. Discuss how the normalization affects the results.
   - Obtain the results when you use the data as it is and when you handle the situation of having unbalanced class distributions (for example, by duplicating the samples from the minority classes). Compare the overall test set accuracy and the class-based test set accuracies. Discuss how the unbalanced class distributions affect the results.
   - Use the stochastic, batch, and mini-batch approaches for the weight updates. Compare their results and discuss the difference in between the use of these approaches, if any.

Part 2:
Implement a neural network classifier with TWO hidden layers. In this part,

(a) Write down the update rules that your implementation uses. Specify the loss function.

(b) List all the parameters (except the hidden unit numbers) that your implementation uses. Just give your selected values of these parameters. In this part, you do not need to explain how or why you select these values.

(c) Select at least two different hidden unit numbers for the first hidden layer, and at least two different hidden unit numbers for the second hidden layer. Run your ANN classifier for each of these combinations (i.e., at least for four different architectures). Obtain training and test set accuracies. Report the overall accuracy as well as the class-based accuracies.
Part 3:
Implement a neural network classifier with **THREE hidden layers**. In this part,

(a) Write down the update rules that your implementation uses. Specify the loss function.

(b) List all the parameters (except the hidden unit numbers) that your implementation uses. Just give your selected values of these parameters. In this part, you do not need to explain how or why you select these values.

(c) Select at least two different hidden unit numbers for the first hidden layer, at least two different hidden unit numbers for the second hidden layer, and at least two different hidden unit numbers for the third hidden layer. Run your ANN classifier for each of these combinations (i.e., at least for eight different architectures). Obtain training and test set accuracies. Report the overall accuracy as well as the class-based accuracies.

Part 4:
Compare the results that you will obtain in Part 1, Part 2, and Part 3. Discuss how additional hidden layers affect the results. Are these results consistent with your expectations? Discuss it very briefly.

Part 5:
Now run your ANN classifier for the MNIST data set. In this part,

(a) Run the ANN classifier with a single hidden layer. Use at least three different hidden unit numbers. Report training and test set accuracies for each selected hidden unit number. (You do not need to specify the values of the other model parameters or you do not need to explain how or why you select these values. However, to get acceptable results, you may need to try different values of the other parameters.)

(b) Run the ANN classifier with two hidden layers. Use at least three different architectures. Report training and test set accuracies for each selected architecture.

(c) Run the ANN classifier with three hidden layers. Use at least three different architectures. Report training and test set accuracies for each selected architecture.

(d) Compare the results obtained for different number of hidden layers. Discuss them very briefly.

(e) Compare the results obtained for the Thyroid and MNIST data sets. Discuss them very briefly.

You will have three weeks to complete your implementation and your runs. However, the runs of this homework may take a considerable amount of time, especially if you use Matlab in your implementation. Therefore, do not leave this homework to the last minute; make sure to give yourselves enough time to finish your runs before the deadline.

Similar to the first homework, you are expected to write your report neatly and properly. The format, structure, and writing style of your report as well as the quality of the tables and figures will be a part of your grade. Use reasonable font sizes, spacing, margin sizes, etc. You may submit either a one-column or a double-column document. In your report, do not give any screen shots or outputs of your program but summarize what you have found at the end of your runs. Do not forget to address the questions specifically asked to you.

Please submit the hardcopy of your report before the deadline. DO NOT submit the printout of your source code. However, you need to email the source code of your implementation before the deadline. The subject line of your email should CS 550: HW2.