Robert Herriott
Undergraduate
Machine Leaning
Spring 2014
Neural Network

As usual, just run the JAR file and follow the instructions. There are no special requirements, but be sure to give this one time. It can take a good chunk of time, on my computer it takes about 35 seconds from the time I double click the data file to the time the results GUI pops up for the 3,3,3 configuration. I have it set to run 10,000 epochs for each of the 10 folds.

Pretty much just followed the algorithm exactly. I wrote FeedForward, CalculateDeltas, and UpdateWeights methods. When training I use a FeedIris method that calls the three previous methods in turn. Then I have a GetClass method that returns the classified class of the last Iris fed through the network. As far as structure, a Network has Layers, and Layers have Nodes.

Now for results:(All results are with 10,000 epochs run for each of the 10 folds, with a learning rate of .001)

(3,3,3):

Average Accuracy: 96%

Fold 1: 15/15 classified correctly
Fold 2: 15/15 classified correctly
Fold 3: 15/15 classified correctly
Fold 4: 11/15 classified correctly
Fold 5: 14/15 classified correctly
Fold 6: 14/15 classified correctly
Fold 7: 15/15 classified correctly
Fold 8: 15/15 classified correctly
Fold 9: 15/15 classified correctly
Fold 10: 15/15 classified correctly

Confusion Matrix:
55,0,0
0,36,0
0,6,53

(4,4,3):

Average Accuracy: 96%

Fold 1: 13/15 classified correctly
Fold 2: 15/15 classified correctly
Fold 3: 15/15 classified correctly
Fold 4: 14/15 classified correctly
Fold 5: 14/15 classified correctly
Fold 6: 15/15 classified correctly

Fold 7: 14/15 classified correctly
Fold 8: 14/15 classified correctly
Fold 9: 15/15 classified correctly
Fold 10: 15/15 classified correctly

Confusion Matrix:
49,0,0
0,37,1
0,5,58

(3,3,3,3):

Average Accuracy: 38.6%  (You were right, it takes a LOT longer to train a 4-layer network)

Fold 1: 8/15 classified correctly
Fold 2: 13/15 classified correctly
Fold 3: 3/15 classified correctly
Fold 4: 3/15 classified correctly
Fold 5: 1/15 classified correctly
Fold 6: 2/15 classified correctly
Fold 7: 7/15 classified correctly
Fold 8: 4/15 classified correctly
Fold 9: 6/15 classified correctly
Fold 10: 11/15 classified correctly

Confusion Matrix:
18,6,0
18,17,28
13,27,23

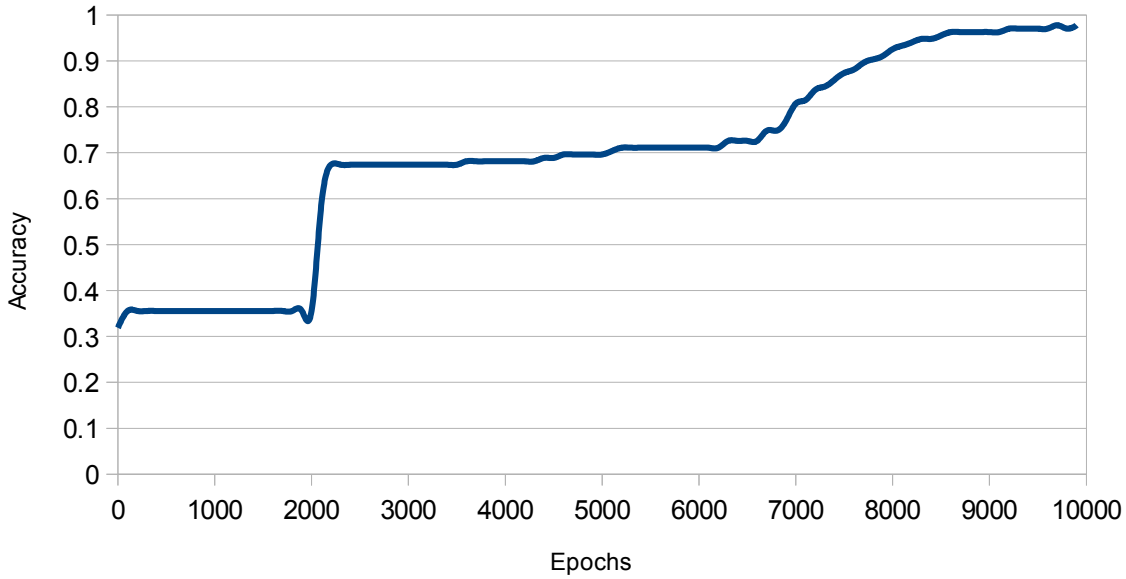
(4,12,3):

Average Accuracy: 96%

Fold 1: 13/15 classified correctly
Fold 2: 15/15 classified correctly
Fold 3: 15/15 classified correctly
Fold 4: 14/15 classified correctly
Fold 5: 14/15 classified correctly
Fold 6: 15/15 classified correctly
Fold 7: 14/15 classified correctly
Fold 8: 14/15 classified correctly
Fold 9: 15/15 classified correctly
Fold 10: 15/15 classified correctly

Confusion Matrix:
49,0,0
0,45,0
0,6,50

Since three of the networks performed with the same average accuracy on these particular runs, I will just use the (3,3,3) network to generate the scatter plot since that is the network that will be in the JAR I submit to you(picking that one because it's the fastest). Also, this is the scatter plot from 1 of the folds of the (3,3,3) network rather than the full data set.




And that's it, I didn't have any unresolved issues and I am very pleased with the performance.