This program is a web application written in Go that makes extensive use of the html/template package. Navigate to the C:\Users\your-name\MultilayerPerceptron\src\backprop\ directory and issue "go run ann.go" to start the Multilayer Perceptron (MLP) Neural Network server. In a web browser enter “http://127.0.0.1:8080/mlpbackprop” in the address bar. There are two phases of operation: the training phase and the testing phase. During the training phase, examples consisting of x-y coordinates in the Cartesian plane and the desired class are supplied to the network. The network itself, is a directed graph consisting of an input layer of nodes, one or more hidden layers of nodes, and an output layer of nodes. Each layer of nodes can be arbitrarily deep. The nodes of the network are connected by weighted links. The weights are trained by first propagating the inputs forward, layer by layer, to the output layer of nodes. The output layer of nodes finds the difference between the desired and its output and back propagates the errors to the input layer. The hidden and input layer weights are assigned “credit” for the errors by using the chain rule of differential calculus. Each neuron consists of a linear combiner and an activation function. This program uses the hyperbolic tangent function to serve as the activation function. The goal of this program is to classify an x-y coordinate to a class (a number). The boundary of each class can be in the shape of a square or a circle.

The user selects the MLP training parameters:

* Hidden Layers
* Layer Depth
* Training Examples
* Classes
* Learning Rate
* Momentum
* Epochs
* Separation
* Ensembles
* Class Shape

The *Learning Rate* and *Momentum* must be less than one. Each Epoch consists of the number of *Training Examples*. One training example is the x-y coordinate of the point in the Cartesian plane and the desired class (0, 1, …). The *Separation* specifies how far apart the classes are. Entering zero means there is no distance between the classes. Each class can therefore be regarded as a cluster of points; the hidden layers determine where those clusters are. The *Ensembles* are used to average the *Epochs* and to minimize the variance of the mean-square-error (MSE). It can be just set to one. The *Class Shape* is the shape of the boundary of each cluster of points.

When the *Submit* on the MLP Training Parameters form is clicked, the weights in the network are trained and the mean-square-error (MSE) is graphed versus Epoch. As can be seen in the screen shots below, there is significant variance over the ensemble. One way to reduce the variance is to select more than one ensemble; the ensemble average of the MSE will then be plotted. However, the weights are not averaged over the ensembles. The final weights are the result of one ensemble.

When the Test link is clicked, 10,000 examples are supplied to the MLP. It classifies the x-y coordinates. The test results are tabulated and graphed. Each class has a different color to make it easier to understand the results. As can be seen in the screen shots below, the farther apart the classes are, the better the results.