Quick and Dirty Manual to Thesis 1.0a1

This manual is just a quick guide to my program, it doesn't go into a lot of detail simply because this is only the first pass at this program, and I still have some work to do on it. Namely, the senior thesis part. It also **only** runs on a **Mac II**, and doesn't handle out of memory or file errors well.

Thesis is a portion of a program I wrote to do my senior thesis. The program, in its current form, allows you to create a simple three layer neural network by specifying the number of input, hidden and output units. You can then add and edit patterns into a pattern list and train the network to recognize said patterns. (You can't delete a pattern once you've entered it though. sorry, it was an oversight. You can change it into a different pattern, though.). You can then save the resulting weights and biases to a file, along with the patterns. You can even watch the network activations train!

Menu Guide:

File Menu

New: Doesn't do anything yet.

Open: Shows any hidden windows.

Close: Hides front window.

Save,Save As: These both do the same thing. They save the output window to a file.

Print: Prints the "output" window.

Edit Menu

Nothing on this menu does anything.

Neural Menu

New Net: Brings up a dialog allowing you to create a net.

Add Pats: Brings up the add/edit pattern dialog.

Train It: Trains the network until the sum of the errors squared is less then the maximum. (5% by default)

Train 100: Trains for 100 iterations. Use this if you're not sure that a net will converge so you don't get into an endless loop.

Train 500: As above only for 500 iterations.

Load Wts: Reads in a network from a file.

Save Wts: Saves a network to a file.

Load Pats: Reads in a set of patterns.

Save Pats: Saves a set of patterns.

Output Menu

Train:Never: Only print out the current summed squared error every 100,000 times. (Never is a long time.)

Train:10: Print out the CSSE every 10 iterations.

Train:100: As above except every 100 times.

Train:1000: As above except every 1000 times.

Dump Wts: Print the weights in the "output" window.

Draw Acts: Check/Unchecked determines whether or not to draw the activations each iteration.

Format Menus

The next three menus determine the way the output window is formatted.

Included Files:

Xor.wts,Xor.pats: Weights and patterns for standard xor example.

Count.wts,Count.pats: Weights and patterns for decoder.

Parity et. al. are files for doing the parity function.

Netlib.c and Netlib.h are the core routines for the neural network.

BrainFile.c is the code that reads in and writes the weight and pattern files. This is provided so that you can reverse engineer the file format.

Have Fun. You can give this program to your friends as long as you don't charge for it, period. Not even a duplication fee with out my written permission. You can send me suggestions and comments at

Wetter@csvax.caltech.edu or pwetter@caltech.bitnet

or

Pierce Wetter III
45 Vista Lago Dr.
Simi Valley, CA 93065