Name: Elgene Menon Leo Anthony

Username: leoanelge

1. What a Neural Network is and what it can do

The neural network is type of software artificial intelligence (“AI”) program that mirrors the basic patterns of brain activity. The neural network is actually a collection of neurons that are situated in layers and communicate between themselves.

The neural network found in this exercise has two layers – one of which is hidden and another which is an output layer. The output layer of this neural network is comprised of 8 neurons and the hidden layer has 16 neurons. The amount of possible outputs the network has is the same as the number of neurons in the output layer.

Every piece of a normal computer program is “written” by a human software developer. However, a neural network is very different. It is given “training data” by a software developer and then the neural network can change itself until it provides the correct outputs required by this training data. Essentially in a basic manner the neural network is “writing” itself without human involvement.

Neural Networks can create sophisticated outcomes in areas such identification of images, pictures, letters and patterns.

2. What your program can do

This exercise contains a basic neural network. The neural network automatically scans 8x8 patterns of boxes and compares them to training material. The training material has 8 different combinations of box patterns. The neural network scans new box patterns and compares them to the training material to discover the box patterns it is looking for.

3. Which input data your program can take

The neural network has 64 inputs of integers. These form 8 categories. The inputs can be changed if required. The categories are determined by the index of the input data output target. This is -10 value is black and +10 value is white. The default is the input data from every single cell in an 8*8 grid shown below in Figure 1.

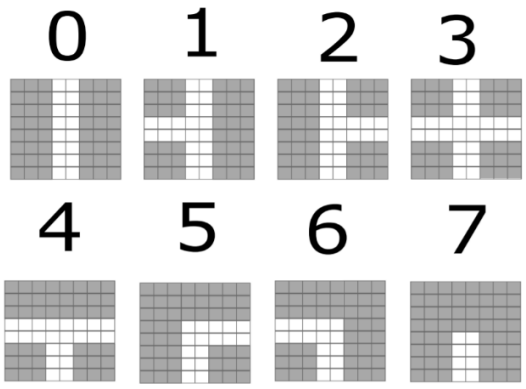

Figure 1: Images and their classifications

4. What should be changed in your code if input data are different from what

you used (number of inputs, number of outputs changed)

For example, suppose an input pattern base number of 4 was selected. The target array would then be {0, 0, 0, 0, 1, 0, 0, 0}. In this target array we can see that index 4 is the only number one evident. The working dataset must be like this but without the target output array. This is because it is the default configuration. To increase the amount of inputs in the neural network or the number of outputs that could be generated would mean altering the variables known as “nInputs” or “nOutputs”. Then the neural network will create the desired outcome without further human interference.

If the input dataset needed to be changed from the default of 750 then the variable “nPatterns” would need to be altered. Most importantly to consistently receive the desired outcome this neural network must be supplied with high quality training data in the right quantity so that the network can continue to learn.

However, if this training data has already been provided but the neural network is not producing the desired results this means that values within the program such as the hidden neurons “nHidden”, learning rate, or “InitNet” values must be altered. This task can be complex and time consuming.

5. What the customer should do to achieve good results from the Neural

network (all the things you changed when trying to achieve convergence)

The neural network does very well with the training data set we were given. It has a detection success rate of 100%. It functions by loading the training dataset as well as the working dataset. Then it initializes the

beginning biases and neuron weights and assigns them a random valuation somewhere between -0.10 and 0.10. There are 64 inputs meaning one per tile of the input patterns. There are 16 hidden neurons and finally there are 8 output neurons meaning one per possible tile. After training initialization, the program chooses a random row out of the training dataset and tunes the weighted values and biases present in our neural network based upon the way total network error value is moving. A backpropagation algorithm is used to do this. This algorithm determines the derivative of each weighted value and bias. Then it passes this information to a separate function which alters the values based upon the derivative. This alters the learning rate of the neural network as well delivering the smallest error value.

The neural network will then determine the total error average from all the training data. The network does this by scanning each pattern in the training data and determining the output/error for each neuron contained in the output layer. The results are then added together and divided by the number of patterns which results in an average error figure for all of the training inputs. This is done 5000 times. However, if the total average error is less than 10.0 the loop is broken.

Next the training data set is scanned again by row and a calculation is done to determine if the neural network gave the right output. This allows a detection rate to be formed based upon how many correct outputs results the neural network achieved.

To check the working data set each row is loaded and a calculation is done to show the output value the user must have to decide if the neural network is working. Alternatively, it can determine the initial values that must be implemented before training if this neural network is used in a different system. These values are discovered by testing the neural networking and making small adjustments. In particular, the random values used for minimum and maximum and the learning rate were discovered by testing the network until a good detection rate occurred.

6. You can make program interface more user-friendly

There are a few small tweaks that would make the system much easier to use. First the training data set should be 750 patterns formulated from the provided “base patterns” and this should give the user a detection rate automatically when the network finishes training. Secondly, similar to the first point, the work dataset should also be 750 patterns formulated from the provided “base patterns” and it should have the same automatic reporting function that will tell the outputs to the user. Finally, the user should be able to upload a training dataset and working dataset of their own and automatically receive back the detection rate from the training and the output rate from working dataset.