Handwriting Analysis

Backpropagation Neural Network

Kyle Lavorato | 10141235

October 17, 2016

Network Architecture

To first begin construction of the backpropagation network, the input dataset needed to be chosen. The two options were the original or preprocessed dataset. The original set consisted of 32x32 grid, where each number was represented by a series of 0s and 1s, with the 1s being the 'on' pixels of the number. The preprocessed set is an 8x8 grid, represented as a series of 64 integer values. These values are a reduced version of the binary grid, with each 4x4 section of the original being represented as an integer in the range [0…16], where the value is the number of 1s in that density. The preprocessed dataset has been selected to be the inputs for the network on the basis that it will significantly reduce training time and network size. This is because only 64 input nodes and their accompanying weights are required instead of 1024, which is a large decrease in the number of weights needing to be adjusted each iteration. Additionally, one extra input node must be added for the bias, giving a total of 65 input nodes required for this ANN.

Next the number of hidden nodes needed to be chosen. Initially a value 15 hidden nodes was selected to be used, based on a visual inspection of the features of the numbers in the dataset. Once the network was completed and put into testing, the number of nodes was varied and the decrease in error was compared with the additional training time required. Through this trial-and-error analysis a final value of 20 hidden nodes was selected. Using less nodes caused error to be too high and using more nodes decreased error by a small fraction while increasing the training time by a noticeable amount.

Finally, the number of output nodes required was easily determined to be 10 nodes. This is simply because there are 10 possible output values, being the numeric values [0…9]. While 4 nodes could have been used with binary encoding to give 16 possible output classes, the reduction in training time was not worth the possibility of crosstalk interference. With the network architecture defined, a visual plan has been created in Figure 1 to display how the network should look.

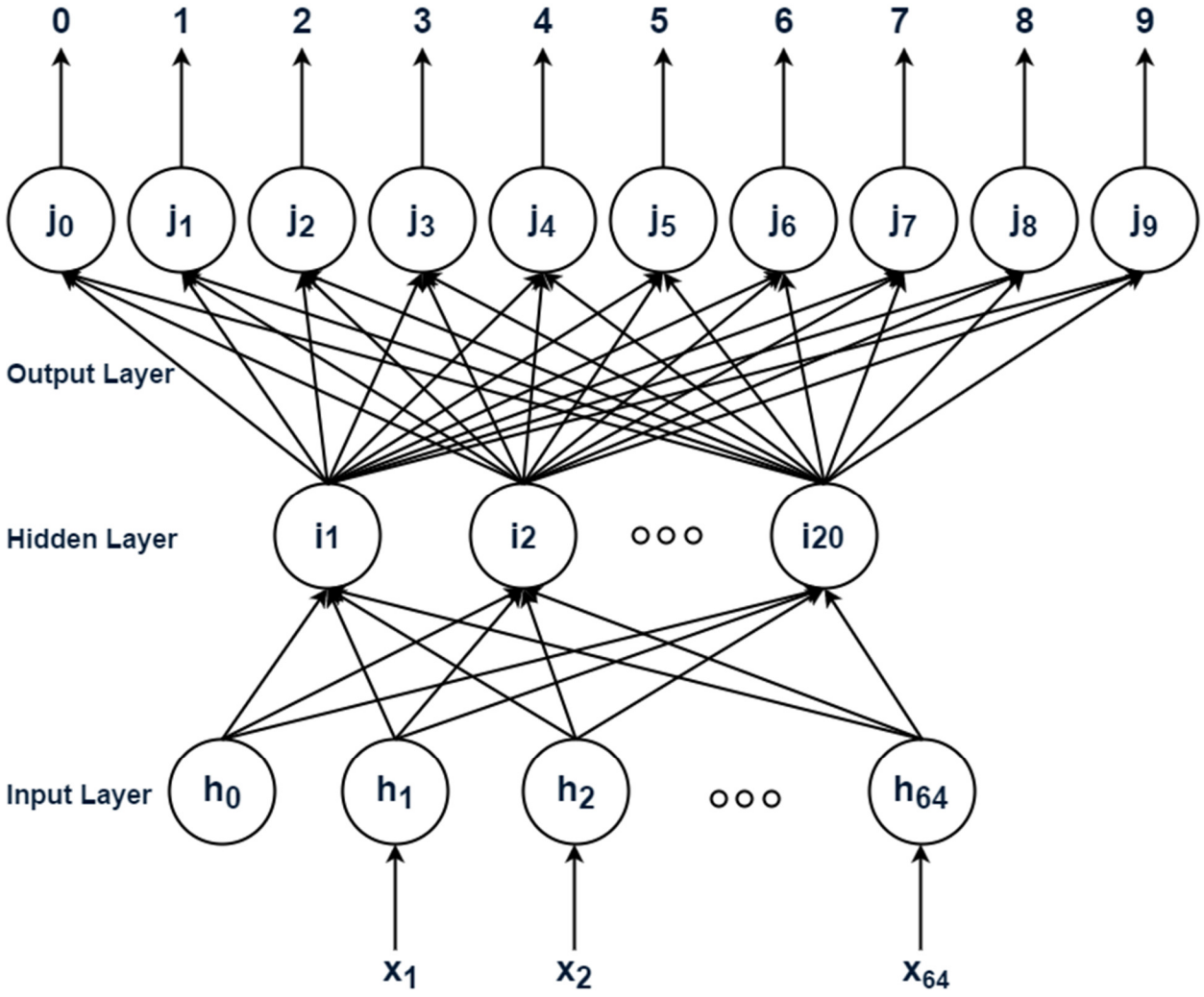

Figure 1 Architecture for the backpropagation network

Weight Values

The initial weight values were selected to be a random value between [-1…1] initially. This was chosen as it is the suggested starting point for the weight values. Through trial-and-error testing, error seemed to reach a minimum when setting the value between [-0.5…0.5]. The update frequency was chosen to use the per-iteration method instead of the per-epoch method. Since the network was not performing well, it was important to try to reduce error where possible and the time saving nature of the per-epoch method was not worth the noticeable error increase.

Learning Rate

The learning rate was initially set to a value of 0.5 as it is a good midrange starting point. Through testing of the network, it was found that 0.5 was too big of an adjustment, while a lower value of 0.05 was too small and caused extra training time. Therefore, learning rate reduction was employed to assist in training. The learning rate starts at 0.5 and is decreased by 50% every three iterations of the data set until it reaches a minimum of 0.15. This process allows for a large initial adjustment to get the weights

to allow them to quickly converge to the correct values. Then once the initial adjustment is made, the learning rate is reduced to allow for the fine tuning of the weight values to reduce error.

Momentum

The initial momentum value was suggested to be 0.4, which was integrated into the backpropagation network. This value produced fine results, but through trial-and-error, a final value of 0.1 was selected. Through multiple trials, 0.1 seemed to produce a lower error consistently more often then other values in the same value region (± 0.2).

Network Training and Performance

During the testing of the backpropagation network, reduced sample sets were used for both the training and testing data, using 100 elements for the training and 50 elements for testing. This is due to the long training time required when using the full set of 3824 training data points. The network is set to run for a total of 200 iterations or until the error reaches 0.08. With this in mind, the reduced training set takes in the range of 32 to 40 seconds and the full data set takes approximately 29.26 minutes to train. Additionally, the mean squared error from the small training set averages 0.09334, while the full set averages an error of 0.09519. Since the full set has a higher error, it is possible that overtraining is occurring, it is more probable that not a large enough sample size was used to calculate the averages of both errors. Therefore, since both are approximately equal, it is more beneficial to use a smaller sample size for the training set as the extra training did not make a noticeable difference. For convenience both sample sizes are enabled in the network, with a toggle controlling which will be used upon start-up of the network.