Jason Argo & Seth Lugibihl

SYS 411

Dr. White

03/07/19

In-Class Exercise: Iris Data

The goal of this assignment is to create the back propagation Neural Network algorithm that creates the best classifier for the Iris data. We have three input nodes for the three independent variables to try to classify between four types of flowers through the four output nodes. The goal of the experiment is to figure out the true error rate of the system with the test data that a botanist would have the best system for classifying the flowers.

We experiment with different configuration to try and obtain the best possible Network through a variety of modifications to the data to see if we can get better classification results.

For complexity and times sake, we are going to constrain ourselves to a max of 100,000 iterations with early exit based on low root means squared error or perfect classification of the test data for modifying different variables to run the system.

Some of the modifications we look into to modify the test error are:

* Number of Hidden Nodes
  + How complex should our system be with hidden layers?
  + We will look into the range of 2 – 9 test nodes for testing our system
* Learning Rate
  + By how much should our system change with each iteration in order to reach an optimal solution?
  + We will look at different learning rates we store in a list to test [.005, .009,.01, etc.]
* Number of Epochs
  + How many iterations do we want the system to run on our test data?
  + We will run between 100 to 1000 epochs on the system
* Normalize the Data
  + We will normalize our data and see if this provides superior results for the system to reach an optimal value for efficiently?
  + After preliminary investigation, we found our system was performing very well without the need to normalize our data, so we decided to not implement this as part of our solution.
    - For other experiments, this may be a way to easily improve the efficiency of the system
* Cross Validation
  + This is an option to cross-check the system’s results by grabing part of the test data as an intermediate cross validation set
  + Because of the systems effectiveness without cross validation (we are reaching over 95% accuracy on the test data without cross validation) we did not think this step would be necessary to get excellent results from our system
* Random samples of data with
  + We begin by shuffling our data list to ensure the data will be randomly sampled
  + Our experiment allows the user to specify a percentage of the data that will be randomly sampled without replacement to be the test data. The remaining data from our input list will serve as the training data.
  + For example, for 100 units of data specified at 20%, 20 units will be randomly popped from the list to serve as the test data while the remaining 80 will serve as the training data

Control the number of iterations of randomization we want to use

* Elapsed Time
  + Our system will track a worst case time for running iterations
  + This will help us understand how quickly our model was able to converge to a good solution.

**Baseline Information:**

Dr. White Success Measures for .0164 error rate

MaxEphoch = 20000

learnRate = .1

numInput = 4

numHidden = 5

numOutput = 3