Neural Networks and H_2

January 15, 2021

1 Introduction

In this assignment we will be looking at a machine learning technique called a neural network. We will use this technique to generate the potential energy surfaces for the ground state and the first two excited states of H2 . The assignment contains two parts. We will first construct the Neural Network and validate its implementation on a small training set. We will then apply the Neural network to the data set of H2.

2 Constructing the neural network

We will build a neural network and test this network on a starting training set. In our starting training set, we have 4 sets containing a variety of true (1) and false (0) values. Each of these sets has 4 input values and 2 corresponding output values. The goal of the network is to be able to predict the 2 output values based on the 4 input values.

2.1 Construction the feed forward network

To construct the network we will start by building a feed forward network. The feed forward network consists of multiple layers. Our first layer is the input layer. We will pass the input data to this layer. The input is then connected to the hidden layer, by a number of weights. These weights are initialized randomly at the start. Each input node is connected to each hidden layer node with such a weighted connection. Make sure that each of the outputs of a node is squashed by the Sigmoid function.

The same process applies to each of the hidden nodes in the hidden layer, each of these are connected to the nodes in the next layer with a weighted connection. In our example model we only use one hidden layer. This hidden layer connects to the output layer. The output layer represents the output of our model.

When you have implemented the network, verify that it works by feeding it a set of inputs. Verify that it can process these.

2.2 Training the network

After a successful feed through the network, we will have to calculate the error on training. We start by calculating the error on the training set. It is possible to write this using a for-loop or to use a more mathematical approach and use matrices.

Algorithm 1 Calculate the Training error

```
Calculate the training error
Calculate the layer 1 error
for Each Weighted Connection do
  Calculate the individual error
  Update the weight
end for
Calculate the layer 2 error
for Each Weighted Connection do
  Calculate the individual error based on the L1 errors
  Update the weight
end for
```

To test if the algorithm works, run the feed forward network and save the output of both the hidden layer and output layer of each node. Use these to run the training code and check if the weights are being updated. You can verify if the weights are update successfully by running the network with the new weights. If the error has decreased (even if it is slightly), the training is working.

To fully train the network, we will write a loop that repeats this cycle until the error has become sufficiently small. Cycle each of the training sets through the loop and save the combined absolute error. Repeat the training cycle until the error has reached > 0.001.

3 Modelling H_2

Import the dataset use the loader. The dataset will be divided into a training and test set. We will be using the training set to train the model while using the test set to verify that it can make predictions based on data the model has never seen.

As we will now be looking at values beyond 1, we cannot use the sigmoid for the weighted connection to the output layer. Therefor we will remove it. This means that our derivative and therefor the error function must also be changed to the derivative of wx. After you have made these changes, verify the model still works by running the first dataset again.

Next load the training data and set 10.000 training iterations. See how your model is performing. To improve perfomance, change the hyperparameters your model uses. Look at the effects of increasing and decreasing the amount of hidden nodes, training time and learning rate. Report the effects of each of these on your performace on the test data set. Plot these results.