Computational Intelligence – Assignment 2 – 20% 2018

Description

For your second assignment, you are required to build a multilayer neural network using Google's TensorFlow library. You will be supplied with training and evaluation data for a classification problem and you will have to design and build a neural network that adjusts its weights to solve the given problem.

Training data: 500 examples, 2 inputs and 1 classification label
Evaluation data: 100 examples, 2 inputs and 1 classification label

This assignment should be broken down into the following stages:
Installation of TensorFlow (See documentation on MOODLE and read the assignment MOODLE forum for advice.) **[10 marks]**

Design of network. You should decide on how many inputs, outputs and hidden layer nodes are required along with what learning rate to use. It will be helpful to draw a diagram of the network layout here. **[10 marks]**

Network training. During this phase, you should experiment with a number of variables to see what effect they have on the training accuracy (reducing the overall error) and the time it takes to train the network. The variables you should experiment with are the **learning rate**, **number of hidden nodes** and the **number of training epochs**. These variables will have an impact on how effectively the network trains, but do remember that the network will start training with random weights, so when testing the effect of changing any of the above variables you should an average measurement over a about ten runs. Your goal should be to get the network to train such that each predication/ classification on the training data is accurate to within **0.01** of the target value. Once you are happy that you have come up with an optimum network model you can train it and save the network model to disk. **[30 marks]**

Network evaluation. During this phase, you will load the saved network model from disk and evaluate it using the evaluation dataset. This is data that the network has not seen before so it should provide a good way to measure the accuracy of the network classification. In order to measure the networks accuracy, you should assume that a correct predication/classification is one where the error or difference between that actual output and the target output is no more than **0.01**. In this sense, **0.01** is an acceptable error tolerance. **[20 marks]**

Deliverables

- Demo in lab. **[30 marks]**
- Network code and saved network model files.
- Report on training and evaluation process. To include tests with different network parameters.

Deadline

Friday 13th of April at 6pm

Plagiarism

This is an individual assignment and any evidence of plagiarism will result in a mark of zero for all parties involved.