

**Department of Computer Science and Engineering**

**(UG Studies)**

**PES University, Bangalore-560085**

|  |  |
| --- | --- |
| Session : Aug - Dec 2017  Credits : 0-0-2-0-1 | UE14CS405 : Machine Learning Lab |
| Lab # : 05 | Implement a 3 layer neural network ( 32 input, 8 hidden and 4 output neurons) to train and classify alphabets/digits |

**Theory:**

Artificial neural networks (ANNs) or [connectionist](https://en.wikipedia.org/wiki/Connectionism)systems are computing systems inspired by the biological neural networks that constitute animal brains. Such systems learn (progressively improve performance) to do tasks by considering examples, generally without task-specific programming. For example, in image recognition, they might learn to identify images that contain cats by analyzing example images that have been manually labeled as "cat" or "no cat" and using the analytic results to identify cats in other images. They have found most use in applications difficult to express in a traditional computer algorithm using[rule-based programming](https://en.wikipedia.org/wiki/Rule-based_programming).

An ANN is based on a collection of connected units called [artificial neurons](https://en.wikipedia.org/wiki/Artificial_neuron), (analogous to axons in a [biological brain](https://en.wikipedia.org/wiki/Brain)). Each connection ([synapse](https://en.wikipedia.org/wiki/Synapse)) between neurons can transmit a signal to another neuron. The receiving (postsynaptic) neuron can process the signal(s) and then signal downstream neurons connected to it. Neurons may have state, generally represented by [real numbers](https://en.wikipedia.org/wiki/Real_number), typically between 0 and 1. Neurons and synapses may also have a weight that varies as learning proceeds, which can increase or decrease the strength of the signal that it sends downstream. Further, they may have a threshold such that only if the aggregate signal is below (or above) that level is the downstream signal sent.

Typically, neurons are organized in layers. Different layers may perform different kinds of transformations on their inputs. Signals travel from the first (input), to the last (output) layer, possibly after traversing the layers multiple times.

The original goal of the neural network approach was to solve problems in the same way that a human brain would. Over time, attention focused on matching specific mental abilities, leading to deviations from biology such as [backpropagation](https://en.wikipedia.org/wiki/Backpropagation), or passing information in the reverse direction and adjusting the network to reflect that information.

Neural networks have been used on a variety of tasks, including [computer vision](https://en.wikipedia.org/wiki/Computer_vision), [speech recognition](https://en.wikipedia.org/wiki/Speech_recognition), [machine translation](https://en.wikipedia.org/wiki/Machine_translation), social network filtering, playing board and video games, medical diagnosis and in many other domains.

**Data set:**

<https://github.com/selva86/datasets/blob/master/seeds.csv>

**Code:**

Please find the attached code in mail.

**TO Do for Students:**

1. Run the code and Identify the bug

2. Modify the code and print the accuracy (Understand Importance of different Activation functions in Artificial Neural Networks ex. TanH function,logistic,ReLU etc.)

3. With different epoches find the accuracy change. Do a comparitive study

**Outcome:**

Artificial neural network can be used for both classification and prediction problems. Given a new test tuple the code will be able to classify the tuple and label it with good accuracy.

**Applications:**

[**Character Recognition**](https://cs.stanford.edu/people/eroberts/courses/soco/projects/neural-networks/Applications/character.html)- The idea of character recognition has become very important as handheld devices like the Palm Pilot are becoming increasingly popular. Neural networks can be used to recognize handwritten characters.[**Image Compression**](https://cs.stanford.edu/people/eroberts/courses/soco/projects/neural-networks/Applications/imagecompression.html)- Neural networks can receive and process vast amounts of information at once, making them useful in image compression. With the Internet explosion and more sites using more images on their sites, using neural networks for image compression is worth a look.[**Stock Market Prediction**](https://cs.stanford.edu/people/eroberts/courses/soco/projects/neural-networks/Applications/stocks.html)- The day-to-day business of the stock market is extremely complicated. Many factors weigh in whether a given stock will go up or down on any given day. Since neural networks can examine a lot of information quickly and sort it all out, they can be used to predict stock prices.[**Traveling Saleman's Problem**](https://cs.stanford.edu/people/eroberts/courses/soco/projects/neural-networks/Applications/TSP.html)- Interestingly enough, neural networks can solve the traveling salesman problem, but only to a certain degree of approximation.[**Medicine, Electronic Nose, Security, and Loan Applications**](https://cs.stanford.edu/people/eroberts/courses/soco/projects/neural-networks/Applications/daveapps.html)- These are some applications that are in their proof-of-concept stage, with the acception of a neural network that will decide whether or not to grant a loan, something that has already been used more successfully than many humans.[**Miscellaneous Applications**](https://cs.stanford.edu/people/eroberts/courses/soco/projects/neural-networks/Applications/miscellaneous.html)- These are some very interesting (albeit at times a little absurd) applications of neural networks.