# SEISMIC APPLICATION USING FEEDFORWARD NEURAL NETWORK TECHNOLOGY: A REVIEW

LEONG LAP SAU[1] & NG TONG SAN[2]

[1]Geophysics Program, School of Physics, Universiti Sains Malaysia, 11800 USM Penang
[2]PETRONAS Carigali Sdn Bhd, Wisma Peladang, Jalan Bukit Bintang, 50776 Kuala Lumpur

Many simple tasks present themselves in the oil and gas industry that appear trival but give results that are dependent on the experience, and to a certain degree, subjective judgement of the geophysicist concerned. A case in point are labour intensive decisions like editing noisy traces, first-break picks and velocity analysis. At another level is the integration of petrophysical parameters with multiple seismic attributes. Here the interpreter is confronted with such a vast array of input variables that it becomes difficult to recognize and weigh the most pertinent combination of influencing factors. Lateral reservoir prediction poses a similar dilemma and so is lithologic identification from well-log values.

A new approach to modern day data analysis which requires a subjective decision is computation by neural network technology. Neural networks are a form of automated pattern recognition in which a set of input pattern is related to an output by a transformation encoded in the network weights. Whereas conventional processing algorithms are deterministic and inevitably sequential, neural networks are instead particularly appropriate for applications where the physics producing an output is not well understood and an exact algorithm cannot be formulated. The network is first 'trained' or calibrated to a certain preset group of patterns or features and when a predefined level of competence has been achieved it is then ready to 'decide' whether a new input sequence is congruent with its memory weights. The purpose of this paper is to review some seismic applications from feedforward neural networks using the backpropagation algorithm. Multi-layered feedforward network and the backpropagation learning algorithm are explained. Seismic applications and input attributes used by various authors are tabulated and discussed.

Neural networks are simple computer models that attempt to simulate the operations of neurons in the brain. The biological aspects of it involves neurophysiological and cognitive processes in the human brain. In neural computing, the computation and approximation power of neural architecture, learning algorithms and applications draws inspiration from physical models of the human brain instead. A very large class of network algorithms satisfying different paradigms and goals exist in neural computing.

Feedforward neural network as discussed here can be represented as a mapping function which is defined by a set of weights and connection parameters. The weights and parameters are determined by minimizing the error function with respect to the known input and output vector pairs. The backpropagation algorithm is the most widely used learning algorithm for feedforward neural networks. Figure 1 (a and b) illustrates the structure of an elemental computational unit with N inputs and one output, and a neural network architecture with two hidden units respectively. Input to the network is weighted, summed and squashed through a nonlinear transfer function (usually a sigmoidal function defined by $f(s) = 1/(1 + e^{-s})$). Comparing with neurons in the human brain, the input corresponds to the dendrites, the weights correspond to the synapses, the summation and transfer functions correspond to the cell body while the output corresponds to axons fanning out to other neurons. The sigmoidal function is the mathematical approximation to the firing rate in the neuron. Supervised training is accomplished by error minimization (usually a form of gradient descent optimization) in the following steps:

(1) Initialize weights and thresholds to small random numbers.
(2) Present input and desired output.
(3) Compute output, i.e. forward propagate the input.
(4) Adapt the weights, i.e. compute error for network output and update new weights to minimize the error.
(5) Use resulting output as new input, repeat steps 2 through 5 until error is sufficiently small. Repeat entire process for each input in the training set.

Knowledge in the feedforward neural network is now contained in the interconnecting weights.

Summaries some applications using the backpropagation learning algorithm. Relevant sources can be identified from the references for furthur reading. The paper concludes with a discussion of some strengths and weaknesses of neural networks.