AASPI

Multiattribute Linear Regression vs Neural Networks

David Lubo, and Kurt J. Marfurt*

The UNIVERSITY of OKLAHOMA
Mewbourne College of Earth and Energy
ConocoPhillips School of Geology and Geophysics
ConocoPhillips

Motivation

Neural Networks is one of the best-established classification algorithms used in seismic interpretation with several excellent commercial implementations available. Our goal is to provide our own implementation to allow interpreters to easily compare the results of PNN and MLFN to multiattribute linear regression, PCA, k-means, SOM, GTM, and PSVM classifiers within the AASPI framework.

Multiattribute Linear Regression

Multiattribute Linear Regression is an extension of linear regression to "N" Variables. In other words, "N" attributes $\{A_1, A_2, \ldots, A_N\}$ are used to predict the log property. The goal is to predict N+1 weights $\{w_0, w_1, \ldots, w_N\}$ when multiplied by the attributes, predict the log property. (Hampson, 2001)

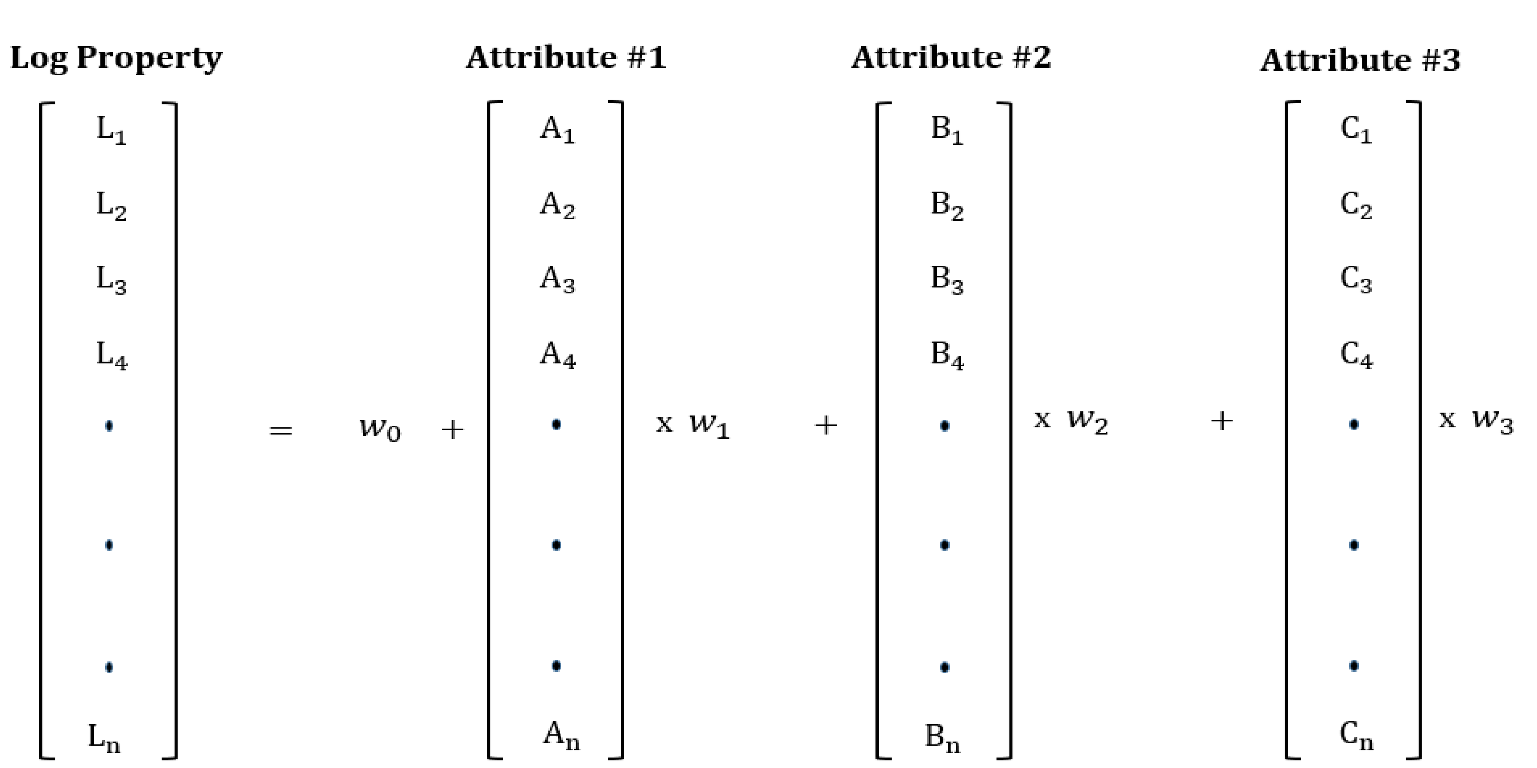

Figure 1: Multiattribute Linear Regression (Hampson, 2001)

We write,

$$L = w\,A \qquad (1)$$

which can be solved using least-squared minimization

$$w = [A^T A]^{-1} A^T L \qquad (2)$$

Multilayer Feedforward Networks (MLFN)

MLFN consists of a set of neurons that are logically arranged in two or more layers. The Neuron is characterized by n+1 weights which multiply each input, and an "activation function" which is applied to the weighted sum of inputs in order to produce the neuron's output. (Masters,1993)

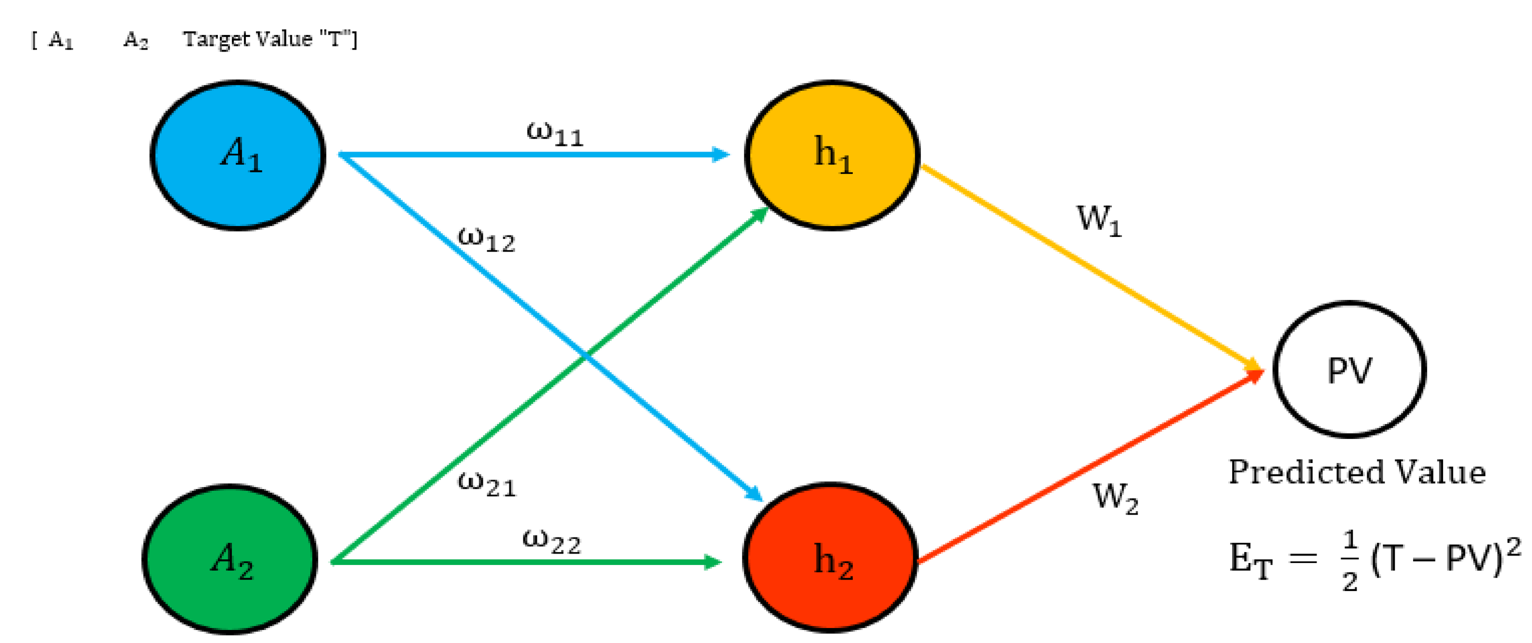

Figure 2: Design of a Multilayer Feedforward Network (Mazur, 2015)

Calculating the value for each neuron "h_j",

$$h_j = \left[1 + \exp\left\{-\left[\sum_{k=1}^{K} \omega_{kj}\, A_k\right]\right\}\right]^{-1} \qquad (3)$$

and the predicted value "PV",

$$PV = \left[1 + \exp\left\{-\left[\sum_{j=1}^{N} W_j\, h_j\right]\right\}\right]^{-1} \qquad (4)$$

Optimizing the weights The "Backpropagation Algorithm" updates the weights in the network, minimizing the error between the output and the target value (Mazur, 2015)

Applying the Chain Rule,

$$\frac{\delta E_T}{\delta W_1} = \frac{\delta E_T}{\delta PV} \times \frac{\delta PV}{\delta W_1} \qquad (5)$$

Updating W_1,

$$W_1^{+} = W_1 - \eta \frac{\delta E_T}{\delta W_1} \qquad (6)$$

where η is the learning rate.

Probabilistic Neural Networks (PNN)

PNN is a mathematical interpolation scheme which uses a neural network architecture for its implementation. (Hampson, 2001)

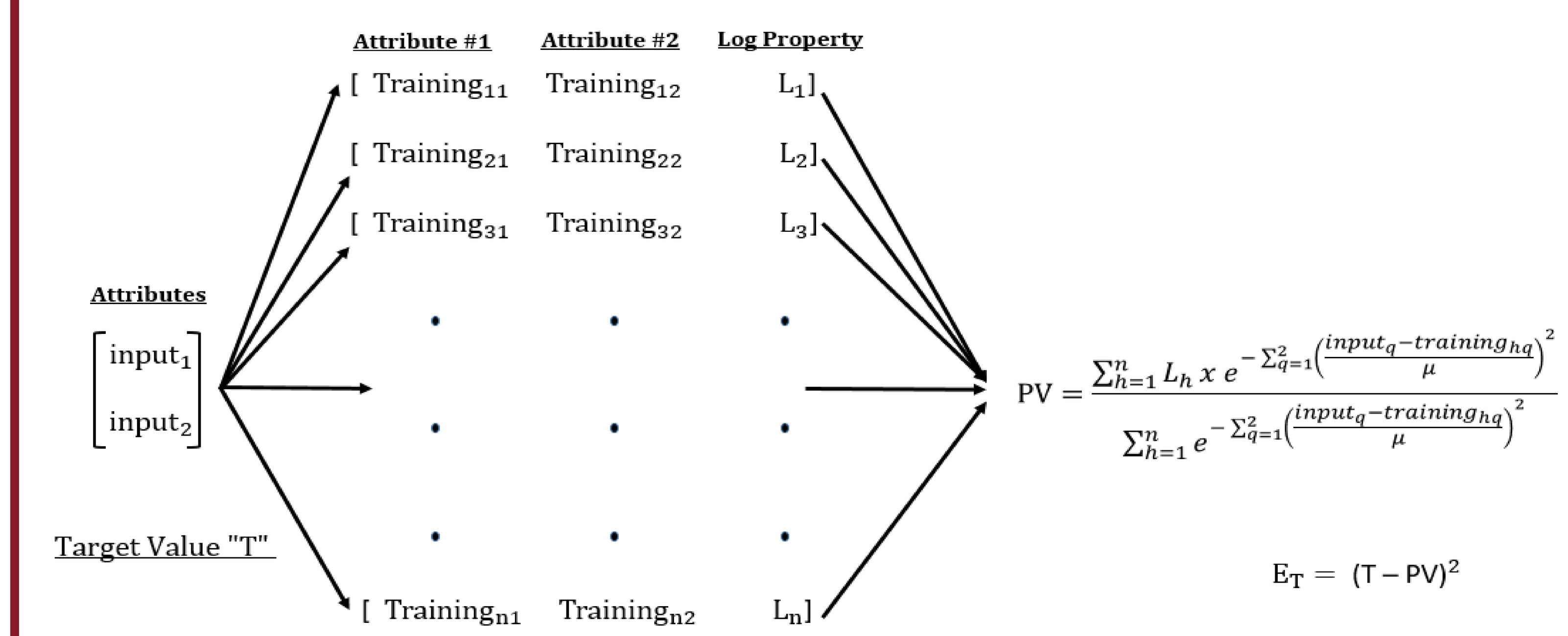

Figure 3: Basic design of a Probabilistic Neural Network (Hampson, 2001)

Optimizing μ Using the "Jackknifing Method" (Masters, 1993) or "Convex Combination of L_n"

Summary

PNN	
Strengths	**Weaknesses**
• Learning speed for this method is fast to instantaneous • Can be more accurate than MLFN • PNN are insensitive to outliers	• Classification time may be slow • Memory requirements are large

Table 1: PNN strengths and weaknesses vs MLFN and Multiattribute Linear Regression. (Masters, 1993)

References

Hampson, D.P., Schuelke, J.S., Quirein, J.A., 2001, Use of multiattribute transforms to predict log properties from seismic data. Geophysics, **66**, 220-236.

Masters, T., 1993, Practical neural network recipes in C++.

Mazur, M., 2015, A step by step backpropagation example. https://mattmazur.com/2015/03/17/a-step-by-step-backpropagation-example/ November 17^{th},2016