Application of the radial basis function neural network to the prediction of log properties from seismic attributes – A channel sand case study

Brian H. Russell*, and Daniel P. Hampson, Hampson-Russell Software Services Ltd., Laurence R. Lines, Department of Geology and Geophysics, University of Calgary.

SUMMARY

In this paper, we use the radial basis function neural network, or RBFN, to predict reservoir log properties from seismic attributes. We also compare the results of this approach with the use of the generalized regression neural network, GRNN, for the same problem, as proposed by Hampson et al (2001). We discuss both the theory behind these methods and the methodology involved in applying neural networks to seismic attributes. We then illustrate the method using the Blackfoot 3D seismic volume, a channel sand example from Alberta.

INTRODUCTION

The prediction of reservoir parameters from seismic data has traditionally been done using deterministic methods, in which we assume a mathematical model that relates our observed seismic data to the reservoir parameter of interest, and apply an inverse transform to our seismic data based on this model. Such methods include deconvolution, post-stack trace inversion, and AVO.

More recently, a statistical approach is used to look for a relationship between the seismic data and the reservoir parameters, using neural networks. For example, Ronen et al. (1994) used the radial basis function neural network (RBFN) for the seismic-guided estimation of log properties using averaged map slices. In more recent work (Hampson et al, 2001), the generalized regression neural network (GRNN) was used to predict reservoir log properties from the complete 3D seismic volume. In this paper, we extend RBFN to the computation of log properties from the full seismic volume, and show its relationship to the GRNN.

THEORY

In this paper we will be using neural networks to perform the supervised prediction of reservoir parameters. Our training dataset will consist of a set of \( N \) known training samples \( t_i \), which in our case will be some well log derived reservoir parameter. Each training sample is in turn dependant on \( L \) seismic attribute values, correlated in time with the training samples. These seismic attribute vectors can be written as \( s_i = (s_{i1}, s_{i2}, \ldots, s_{iL})^T \), \( i = 1, 2, \ldots, N \). The objective of our neural network is to find a function \( y \) such that:

\[
y(s_i) = t_i, \quad i = 1, 2, \ldots, N
\]

Once this function has been found, it can be applied to an arbitrary set of \( M \) seismic attribute vectors \( x_k \), \( k = 1, 2, \ldots, M \), where the attributes in the \( x_k \) vectors are identical to those in the \( s_i \) vectors. This is illustrated in Figure 1 for two arbitrary training samples and a single application sample.

Figure 1: A schematic illustration of the differences between the training vectors, \( s_i \) and \( s_j \), in which the output samples \( t_i \) and \( t_j \) are known, and the application vector \( x_k \), in which the output sample \( y_k \) is not known.

Figure 2: A schematic graph of the vectors, \( s_i, s_j, \) and \( x_k \), from Figure 1, where the coordinate axes represent attribute amplitude rather than Cartesian distance.

Both the RBFN and GRNN algorithms are best understood by first looking at the PNN algorithm. This algorithm is based on the concept of “distance” in attribute space, as shown in Figure 2. Note that “distance” on these graphs is attribute amplitude rather than Cartesian distance, and referring to Figure 1, we can define the three possible distances between these vectors as...
Application of the radial basis function

\[ d_{ij} = |s_i - s_j|, \quad d_{ik} = |s_i - x_k|, \quad d_{jk} = |s_j - x_k|. \]

In our neural networks, we will use a function of the distances, \( \phi(d) \), called a basis function. We will use the Gaussian, written

\[ \phi(d) = \exp\left(-\frac{d^2}{\sigma^2}\right), \quad (2) \]

where \( \sigma \) is a smoothness parameter. The PNN is then defined for each of the \( x_k \) points as the sum over all the possible \( \phi(d_{ij}) \) functions, or

\[ p(x_k) = \sum_{j=1}^{N} \exp\left(-\frac{|x_k - s_j|^2}{\sigma^2}\right) = \sum_{j=1}^{N} \phi_{kj}, \quad k = 1, M. \quad (3) \]

PNN can be thought of as an implementation of Bayes' Theorem, and works well as a classifier. Masters (1995) shows that the GRNN method can be seen as an extension of PNN, as seen below:

\[ \sum_{j=1}^{N} \frac{t_j \exp\left(-\frac{|x_k - s_j|^2}{\sigma^2}\right)}{p(x_k)}, \quad k = 1, \ldots, M. \quad (4) \]

Equation (4) is known in statistical estimation theory as the Nadaraya-Watson estimator, and was re-discovered in the context of neural networks by Specht (1990) and named the generalized regression neural network, or GRNN.

Equation (4) also gives us a starting point for understanding the RBFN approach, which was originally developed as a method for performing exact interpolation of a set of data points in multi-dimensional space (Bishop, 1995). In this method, we will also use Gaussian basis functions. But instead of calculating the weighting values "on the fly" using the training samples, we compute the weights initially from the training data using the equation

\[ t(s_i) = \sum_{j=1}^{N} w_j \exp\left(-\frac{|s_i - s_j|^2}{\sigma^2}\right) = \sum_{j=1}^{N} w_j \phi_{ij}, \]

\[ i = 1, 2, \ldots, N. \quad (5) \]

Equation (5) can be solved using standard matrix inversion techniques, noting the resulting matrix is symmetric, leading to an efficient solution algorithm.

Once the weights have been computed, they are applied to the application dataset using

\[ y(x_k) = \sum_{j=1}^{N} w_j \exp\left(-\frac{|x_k - s_j|^2}{\sigma^2}\right), \quad k = 1, \ldots, M. \quad (6) \]

Note that equation (6) can be thought of as the general form of both RBFN and GRNN if we rewrite the weights in the GRNN case as

\[ w_j = \frac{t_j}{p(x_k)} \quad (7) \]

This observation allows us to show that GRNN is a subset of RBFN, since the GRNN method does not contain off-diagonal elements in the weight calculation, as does RBFN. We would thus expect RBFN to produce higher frequency results than GRNN as the training dataset gets smaller, which is indeed the case.

**METHODOLOGY**

The methodology for reservoir prediction using multiple seismic attributes has been extensively covered by Hampson et al (2001) and will only be briefly reviewed here. The two key problems in the analysis can be summarized as follows: which attributes should we use, and which of these attributes are statistically significant?

To start the process, we compute as many attributes as possible and then find their order using a technique called stepwise regression. This consists of first finding the attribute with the lowest prediction error, then finding the pair of attributes with the lowest prediction error from all combinations of the first attribute and one other, and continuing this process as long as desired.

This will give us a set of attributes that is guaranteed to reduce the total error as the number of attributes goes up. So, when do we stop? This is done using a technique called cross-validation, in which we leave out a training sample and then predict it from the other samples. We then re-compute the error, but this time from the training sample that was left out. We repeat this procedure for all the training samples, and average the error, giving us a total validation error. We generally find that the validation error decreases for the first three or four attributes, but then increases, telling us when to stop adding attributes. Improved resolution can also be obtained using a convolutional operator (Hampson et al, 2001).

**A CHANNEL SAND CASE STUDY**

We will now illustrate our methodology using a channel sand case study from Alberta. This study involved the
Application of the radial basis function

prediction of porosity in the Blackfoot field of central Alberta. A 3C-3D seismic survey was recorded in October 1995, with the primary target being the Glauconitic member of the Mannville group. The reservoir occurs at a depth of around 1550 m, where Glauconitic sand and shale fill valleys incised into the regional Mannville stratigraphy. The well log input consisted of twelve wells, each with sonic, density, and calculated porosity logs, shown in Figure 3. One of the lines from the 3D survey, inline 95, is indicated in red on Figure 3, and the seismic and inverted sections from this line are shown in Figure 4, with the P-wave sonic log from well 08-08 inserted at cross-line 25. We will use the P-wave sonic as our target on which to train the seismic attributes.

Using multi-linear regression to find the optimum attributes with an operator length of 7 points, we found that only four attributes were statistically significant. These attributes consisted of the impedance section, and three filtered versions of the seismic volume. These attributes were then used in the neural network training. The validation of these results is shown in Figures 5 and 6.

We then applied the results of the training to the 3D seismic dataset itself. Figure 7 shows the application of the...
Application of the radial basis function

GRNN method to the seismic data, and Figure 8 shows the result of applying the RBFN method. In Figure 8, notice that there appears to be slightly more frequency content and lateral continuity. As we reduced the number of wells in the training, and the number of points in the convolutional operator, we found that this trend continued, and we were able to see more frequency content in RBFN over GRNN. And the error validation of the well logs, although similar, became slightly better for RBFN over GRNN.

Figure 7: Application of the GRNN algorithm to line 95 of the 3D volume, using 4 attributes and a 7-point operator.

Figure 8: Application of the RBFN algorithm to line 95 of the 3D volume, using 4 attributes and a 7-point operator.

CONCLUSIONS

In this paper, we have compared several neural network approaches for the prediction of log properties using multiple seismic attributes. These methods consisted of the generalized regression neural network, or GRNN, and the radial basis function neural network, or RBFN. After a discussion of the theory and our methodology, we applied the two methods to a channel sand case study from Alberta.

When we applied the two methods to the complete twelve well dataset, as shown here, the validation at the wells show a slightly smaller error for GRNN than for RBFN. However, the results of applying the techniques to the seismic data showed improved frequency content in the RBFN results. As we decreased the number of wells, this effect became more pronounced, although we did not show the results in this short abstract. And we also found that, as the number of wells decreases, the error in the RBFN method is slightly lower than for GRNN. However, we also found that this conclusion is highly dependent on the types of attributes used and the length of the convolutional operator. More work also needs to be done on the optimization of the sigma scaling values in the RBFN method.

ACKNOWLEDGEMENTS

We wish to thank our colleagues at the CREWES Project and at Hampson-Russell Software for their support and ideas, as well as the sponsors of the CREWES Project.

REFERENCES