Approximating and Interpolating Spatial Data Functions using a Neural-Fuzzy Technique

PM Wong[1], Y Huang[1,2] and TD Gedeon[2]

Technical Report RCTR97/01

[1]Centre for Petroleum Engineering
[2]School of Computer Science and Engineering

The University of New South Wales
Sydney NSW 2052

September 1997

ABSTRACT

In this paper, we propose a neural-fuzzy technique for function approximation and interpolation. We particularly apply the technique to spatial data analysis. The methodology involves the use of neural networks to approximate functions through existing data points. The trained networks are then used as fuzzy rules. The results of these rules are interpolated based on the relative location of the point-of-interest. We also demonstrate the use of this methodology in petroleum reservoir modelling where properties are estimated between two oil wells. The end result is a simple and computationally-cheap method in real-life engineering studies.

Keywords: Neural, Fuzzy, Spatial Interpolation, Petroleum Reservoir, Oil Wells.

INTRODUCTION

Function approximation and interpolation are important in most engineering disciplines. The former involves the fitting a curve or a surface through existing data points. These data points are usually clustered. One example of function approximation method is linear regression in which the total error of predictions is minimised. The later is similar to function approximation, however, the available data points are more scattered and separated in space (e.g. physical distance). One example is distance-weighted interpolation methods in which the weighting factors of samples are calculated based on the relative location of the point-of-interest.

Figure 1 shows an example for the need to approximate and interpolate spatial functions. In this figure, we have two sets of noisy data measured at two sample locations, **Z** and **Z+h**, where **Z** is the distance between the first sample location and a reference location, and **h** is the separation distance between the two sample locations. The notion of function approximation is to fit two noise-free curves through the data points. This is done to ensure that we can obtain function values, **Y** (the dependent variable), for any **X** values (the independent variable).

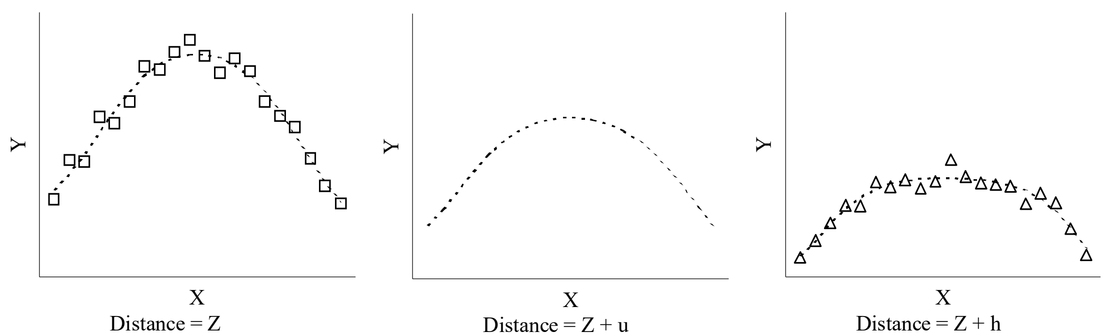


Figure 1. Example for function approximation and interpolation.

In spatial data analysis, there is also a need to predict function values at location **Z+u** where 0<**u**<**h** if the measurable **X** values are available at that location. This is the purpose of function interpolation. If, however, **u**<0 or **u**>**h**, this becomes an extrapolation problem. There is usually a larger uncertainty in extrapolating functions than interpolating functions, especially when **u**<<0 or **u**>>**h**.

In this paper, we propose a methodology to achieve these tasks by the use of neural-fuzzy technique. The next section will visit the residual problems of neural networks, followed by the introduction of our new methodology. In the later sections, we will demonstrate the use of this technique in petroleum reservoir modelling.

PROBLEMS OF NEURAL NETS

Artificial neural networks, or simply neural nets, are well-known and successful in approximating complex, non-linear functions. The use of supervised learning, such as back-propagation neural nets (BPNN), is most appropriate for this task as observable X-Y values (see data plots at locations **Z** and **Z+h** in Figure 1) can be simply used as input and target data respectively. When we use neural nets for function interpolation, the problem is not as straightforward. Two issues are discussed below:

1) Pattern Selection

Choosing proper training and validation patterns in neural learning is important. In most applications, random sampling is a popular method to select patterns for both training and validation sets. In function interpolation, however, it may not be appropriate, especially when there is a strong correlation between the functions approximated at locations **Z** and **Z**+**h**. Under this situation, if **u** is close to 0, one may expect that its function relation should be similar to that at location **Z**.

One solution to the above problem is to construct a training set based on the relative proportions of patterns at locations **Z** and **Z+h**. For example, we can construct the training set using random selection of 50% patterns from location **Z** and 50% patterns from location **Z+h**. We can then duplicate the patterns in such a way that, at location **u**, the training set contains [100(**h**-**u**)/**h**]% patterns from location **Z**, and (100**u**/**h**)% patterns from location **Z+h**. The same procedure applies to the validation set. The disadvantage of this method is that it is a computationally-expensive process as we need to run these analyses for a large number of times by drawing different random seeds for pattern selection.

2) Binary Classification

In the example shown in Figure 1, some may suggest to introduce an extra input neuron to represent the locations of these two functions. For example, we can use 0 for location **Z** and **h** for location **Z+h**. This approach, however, could be dangerous to apply in practice because we only have samples from two locations. Under these circumstances, neural nets may see 0 as APPLE and **h** as NOT-APPLE, and the solution implemented (learnt) by the neural net may be more appropriate for a binary classification problem.

NEURAL-FUZZY TECHNIQUE

Some related works have been done using neural nets to interpolate spatial data in both mining and petroleum industries (Wu and Zhu, 1993; Shibli et al., 1996), however only areal data maps are produced and there is no indications on how to extend the methodology to vertical data interpolation. The proposed methodology is appropriate for interpolating vertical measurements. It is developed based on the use of learning curve in neural networks and fuzzy rules interpolation (Kóczy and Hirota, 1993; Baranyi et al., 1995). It consists of three steps: 1) pattern selection for neural learning; 2) neural learning to extract fuzzy rules; 3) interpolating the results of fuzzy rules to obtain final estimate.

1) Pattern Selection

In the proposed technique, the selection of training and validation sets is straightforward. All the patterns at location **Z** are used for the training set, and all the patterns at location **Z+h** are used for the validation set. There is no random pattern selection. The statistics of these two sets may not necessarily the same, but they need to be correlated.

2) Neural Learning

The purpose of this step is to approximate functions at the sample locations. Standard BPNN procedure is utilised. Most neural nets avoid over-training by stopping iteration at minimum error on the validation set (see Figure 2a). The proposed technique uses minimum "interpolated error" or IErr to stop training. This error is defined as:

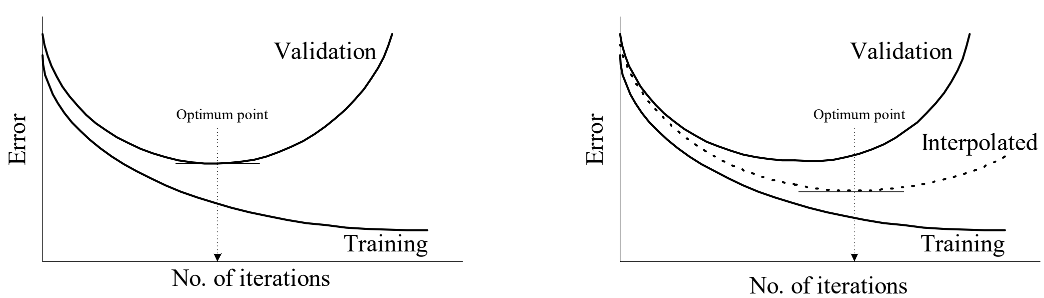
$$\mathrm{IErr} = \frac{\sum_{j}^{n} W_j E_j}{\sum_{j}^{n} W_j} \tag{1}$$

and,

$$W_j = \frac{1}{D_j} \tag{2}$$

where $\mathbf{W_j}$ is the weighting factor of the j^{th} data set, $\mathbf{D_j}$ is the separation distance between the point-of-interest and j^{th} sample location.

In the case shown in Figure 1 (n=2), $\mathbf{D_1}$ and $\mathbf{D_2}$ are simply **u** and (**h**-**u**) respectively. $\mathbf{E_j}$ represents the error on the learning curve using j^{th} data set for training (and the other for validation). Thus, when **u** equals 0, this is equivalent to over-learning of the training set, and when **u** equals **h**, this is equivalent to the use of the standard early-stopping method in BPNN (i.e. stop training when minimum validation error is reached). In cases where $0<\mathbf{u}<\mathbf{h}$, the interpolated error curve is between the validation and training errors as displayed in Figure 2b. When the minimum interpolated error is reached, training is stopped.



a. Optimum point based on validation error.

b. Optimum point based on interpolated error.

Figure 2. Stopping criteria for BPNN learning.

3) Rules Interpolation

Once each data set has been used as the training set, we will have **n** trained neural nets. These networks are used as fuzzy rules. These are disjoint rules in a large sparse rule-base. Each of these rules can be expressed as:

$$\text{Rule}_j\text{:}\ \textbf{IF}\ \ X_j = (x_1,\ldots,x_m)\ \text{is}\ A_j\ \ \textbf{THEN}\ \ Y_j = NN_j(x_1,\ldots,x_m) \tag{3}$$

where **m** is the dimension of the input vector **X**, $\mathbf{A_j}$ is the fuzzy set of the j^{th} partitioned rule space, and $\mathbf{Y_j}$ is the output from j^{th} trained neural net or $\mathbf{NN_j}$.

The conceptualisation of the use of neural nets as fuzzy rules allows us to use the developed formal machinery of fuzzy set theory such as fuzzy interpolation and defuzzification. We can then interpolate the relevant rules to location **u**, and only recombine the effects of the interpolated rules subsequently. The interpolated estimates can be expressed as:

$$Y_u = \frac{\sum_j^n W_j Y_j}{\sum_j^n W_j} \tag{4}$$

where $\mathbf{Y_u}$ is the final estimate at location **u**.

In the case displayed in Figure 1, if **u** equals 0, $\mathbf{Y_u}$ is the result of the fuzzy rule using data at location **Z** as the training patterns. Similarly, if **u** equals **h**, $\mathbf{Y_u}$ is the result of the fuzzy rule using data at location **Z+h** as the training patterns. Since data set is trained independently at each location and if more data become available at new sample locations, it is straightforward to incorporate the new information in the existing system and there is no need to re-learn all the data.

CASE EXAMPLE

1) Background

The proposed technique is applied to petroleum reservoir modelling. In reservoir modelling, we often drill only a few holes at different locations. Reservoir properties are measured at the laboratory from the rock samples retrieved at different depths. One of these properties is porosity which is a measure of void space (amount of fluids containment) within the bulk volume of the sample. There is, however, no such a measurement between holes. Thus, predictions at between-wells are obtained from the interpolation of known properties at well locations. This is an essential step for further performance evaluation studies.

In some cases, seismic survey is run across the field. Therefore, acoustic measurements, such as sonic travel times, are available at both the wells and between-wells. Correlation models can be built between porosity and sonic travel times at different well locations. In this case, porosity can be treated as dependent variable **Y** and sonic travel times as independent variable **X** as discussed previously. If we have only two wells, their locations can be represented by **Z** and **Z+h**.

2) Objective

In this example, we had data from two oil wells separated by 49 units of length (h=49). The data was obtained from the North West Shelf, offshore Australia. Each well had 50 measurements of porosity and sonic travel times. Therefore, the total number of grid blocks was 2,500 in which 100 of them were known. Figure 3 shows the sonic travel times across of the reservoir together with the porosity measurements at the two well locations, namely Well A and Well B. The objective was to estimate the porosity values at the remaining 2,400 blocks where sonic travel times were available.

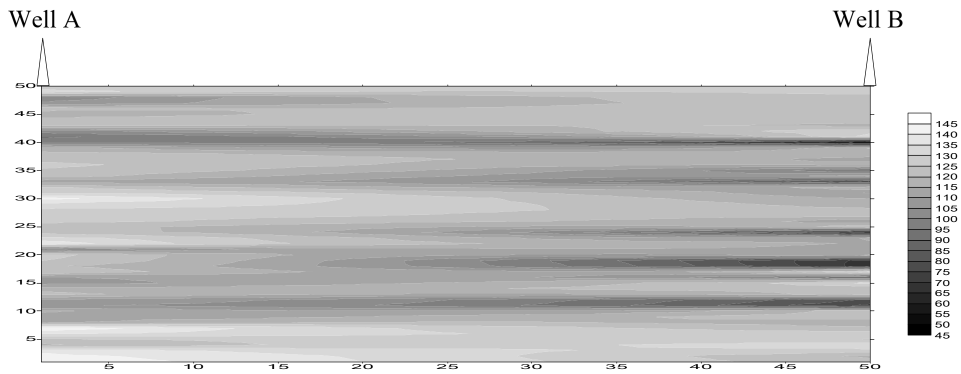


Figure 3. Sonic measurements across the reservoir (gray scale in msec per foot).

3) Neural modelling

All the sonic travel times were normalised in the range of 0 and 1. They were used as the input data. Standard back-propagation neural nets were employed in this study. Sigmoid function was used as the transfer function, and hence the porosity values (the target data) were normalised in the range of 0.1 and 0.9 for faster convergence. The data set in Well A was first used as the training set, and those in Well B was used for the validation set.

Single-input-single-output networks were used throughout the study. Three hidden neurons were found to produce the best and stable results. The learning and momentum terms were 0.1 and 0.5 respectively. For each unknown location, interpolated error was calculated. Connection weights were obtained based on the minimum interpolated error. The maximum allowable iteration was 15,000.

4) Interpolated results

The whole process repeated by swapping the training set as validation set and validation set as the training set. All the relevant rules were interpolated to obtain the final estimate as shown in Equation (4). The interpolated porosity image is displayed in Figure 4.

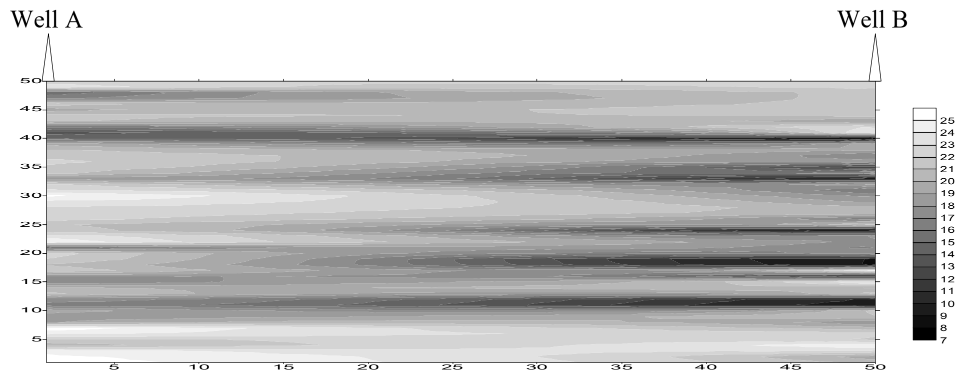


Figure 4. Interpolated porosity map (gray scale in percentage).

The quality of the interpolated porosity map is clear. It is nevertheless worth observing that while Figure 4 is similar to Figure 3 as expectable, the major difference in terms of the observable features is an increase in fine detail. This is in contrast to the averaging effects of most computational techniques for deriving such maps. That we have managed to increase the level of detail is due to the nature of our technique and its success in capturing the non-linear interaction of the seismic measurements at each grid location and the geology at the well locations as captured by our neural-fuzzy rules bases.

CONCLUSIONS

The paper uses a new technique in function approximation and interpolation based on neural networks and fuzzy rules interpolation. The technique is applied to petroleum reservoir modelling. Neural networks are first used to approximate the relations between laboratory (porosity) and seismic (sonic travel times) measurements at well locations. The trained networks are used as fuzzy rules and are then interpolated to between-well locations where only seismic measurements are available. The interpolated porosity map shows very good fine detail, and surpasses our expectations. The end result is a simple and computationally-cheap method, which can be applied to many other applications in real-life engineering studies.

The current methodology assumes that the spatial functions are fully correlated which is not always true in geological modelling (e.g. in the presence of faulting). Hence, further work is required to extend the methodology for interpolating partial correlated functions.

REFERENCES

Baranyi, P., Gedeon, T.D. and Kóczy, L.T., 1995. "A General Method for Fuzzy Interpolation: Specialised for Crisp Triangular and Trapezoidal Rules", *Proceedings of the 3rd European Congress on Intelligent Techniques and Soft Computing*, Aachen, 5 pages.

Kóczy, L.T. and Hirota, K., 1993. "Approximate Reasoning by Linear Rule Interpolation and General Approximation", *International Journal of Approximate Reasoning*, no. 9, pp. 197-225.

Shibli, S.A.R., Wong, P.M. and Maignan, M., 1996. "Interpolation Neural Networks: A New Methods for Interpolation", *5th International Workshop on New Computing Techniques in Physics Research*, September 2-6, Lausanne, Switzerland, 4 pages.

Wong, P.M., Gedeon, T.D. and Taggart, I.J., 1995. "An Improved Technique in Porosity Prediction: A Neural Network Approach", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 33, n. 4, pp. 971-980.

Wu, X. and Zhou, Y., 1993. "Reserve Estimation using Neural Network Techniques", *Computer and Geosciences*, vol. 19, no. 4, 567-575.