# Seismic Inversion for Porosity using a Backpropagation Neural Network

*Philippe Ecoublet* [*], *William W. Symes* [†] *and Stewart Levin* [‡]

**ABSTRACT**

Fluid properties in seismic reservoirs are commonly assessed from the computation of seismic attributes. We investigate both qualitatively and quantitatively how seismic inversion results can improve fluid property predictions in reservoir characterization by implementing a neural network as an inversion tool.

This study is first conducted locally by comparing log-to-log elastic prediction of porosity with seismic inversion prediction and with direct seismic attributes prediction, before tackling multi-offset seismic data.

We also investigate the performance achieved from using different classes of neural networks for prediction and pattern classification, including the popular backpropagation network and the radial-basis function networks.

## Introduction

The feasibility of predicting reservoir fluid properties directly from seismic inversion results, as an alternative to seismic attributes prediction, is investigated in this report by carrying out a number of log-to-log prediction experiments, using a neural network. Our underlying aim is not to replace seismic attributes, but to study whether and how much the information gleaned from seismic inversion, i.e. impedance, velocity and density, might be expected to improve fluid property predictions from seismic data.

First, a backpropagation neural network is trained to predict porosity, volume fraction

[*]The Rice Inversion Project, Department of Computational and Applied Mathematics, Rice University, Houston TX 77005-1892 USA, email `ecoublet@caam.rice.edu`

[†]The Rice Inversion Project, Department of Computational and Applied Mathematics, Rice University, Houston TX 77005-1892 USA, email `symes@caam.rice.edu`

[‡]Mobil Technology Company, Strategic Research Center, Dallas, TX 75265 USA, email `salevin@dal.mobil.com`

of shale, and water saturation in a reservoir interval from sonic log, density log, and synthetic impedance computed out of the recorded logs. The experiment is then repeated after having upscaled the log data to seismic scale.

Second, synthetic 1.5D offset seismograms are generated using a visco-elastic modeling code developed by The Rice Inversion Project, with the model parameters compiled from sonic, $V_p/V_s$ ratios and checkshot information. A 1.5D seismic inversion is applied to the synthetic seismograms to produce seismic-scale elastic logs which are used to train the neural network to predict fluid properties.

Finally, the neural network is trained to predict fluid properties from a set of single trace seismic attributes calculated from the synthetic seismograms previously generated plane-layered modeling.

Based on the results obtained from these experiments, we aim to establish the utility of predicting fluid properties directly from the inversion of the seismic data. A correlation coefficient between predicted and actual fluid properties is compiled to assess and quantify the recovery of fluid properties from well logs both at log scale and at seismic scale, from seismic inversion, and from seismic attributes.

## The experiment

### The log data

For this analysis, carried out on an offshore, clastic reservoir , we selected wells for which a sonic log has been recorded and which are proximate to a single seismic sail line. A handful of wells met both conditions and among those we retained the wells 58c and its side track 58cst as training wells, i.e. used to train the neural network. The well 56c and the well 55a from a different platform were chosen as test wells to predict fluid properties from the trained network. Available at either 0.5 ft or 1 ft depth sampling, the well logs were resampled every 0.5 ft. Table 1 summarizes, for each selected well, the number of samples composing the logs, the measured depth interval, as well as the corresponding true vertical depth (TVD) interval below sea surface.

| Well | Number of Samples | Measured Depth | True Vertical Depth |
|---|---|---|---|
| 58c | 1830 | 6599.5 - 7514.0 | 5050.6 - 5698.5 |
| 58cst | 1720 | 6200.0 - 7059.5 | 5052.4 - 5594.1 |
| 56c | 2574 | 5400.0 - 6686.5 | 4913.3 - 6027.5 |
| 55a | 2088 | 5806.0 - 6849.5 | 4865.4 - 5653.0 |

Table 1. Selected wells for the experiment

The gas-oil contact in that section of the reservoir is located at approximatively 5350 ft TVD and the oil-water contact at 5550 ft.

## Preprocessing the data

The density ($\rho$), e.g. ($RHO$) in figures, and sonic ($DT_p$) logs were edited and a "synthetic" $P$-impedance ($Z_p$) log computed. Using the ($V_p/V_s$) ratios in sand and shale, supplied in Table 2, "synthetic" $S$-sonic ($DT_s$) and $S$-impedance ($Z_s$) logs were computed. These five logs constitute the input data for the neural network to predict porosity ($\phi$), e.g. ($PHI$) in figures, volume fraction of shale ($V_{sh}$), e.g. ($VSH$) in figures, and water saturation ($S_w$), e.g. ($SWT$) in figures, which constitute the three "targeted" outputs of the neural network.

| Formation | $V_p/V_s$ |
|---|---|
| wet sands | 2.12 |
| gas sands | 1.74 |
| shales | 1.87 |

Table 2. Assumed $V_p/V_s$ ratio in reservoir interval

The log data are not noise-free and the presence of data out of the range of the "physical model" for which the network is trained may dramatically affect the prediction performances. Preprocessing the data is therefore critical.

The logs are first cleaned up by removing outlier samples. This is achieved by calculating the histogram of each log and clipping the sample values to lie within the 2-98 percentiles of the histogram. The logs are then linearly rescaled between 0 and 1 using the transformation

$$\log_r = \frac{(\log - min)}{(max - min)}$$

where $min$ and $max$ are the extreme values of the log, and $\log_r$ the rescaled log.

Such a normalization is applied to reduce neural network "saturation" during the training process, and to seek superior values for the network weights close to their initial values. Initialization issues are discussed in further details later.

Figure 1 displays crossplots between input training logs and porosity, before normalization of the data. The corresponding correlation coefficients $\mathrm{Cor}_{xy}$ defined by

$$\mathrm{Cor}_{xy} = \frac{E((x-\overline{x})(y-\overline{y}))}{E((x-\overline{x})^2)^{\frac{1}{2}} E((y-\overline{y})^2)^{\frac{1}{2}}},$$

are given in Table 3. $E()$ denotes the expected value function, $\overline{x}$ and $\overline{y}$ the mean of $x$ and $y$ respectively.

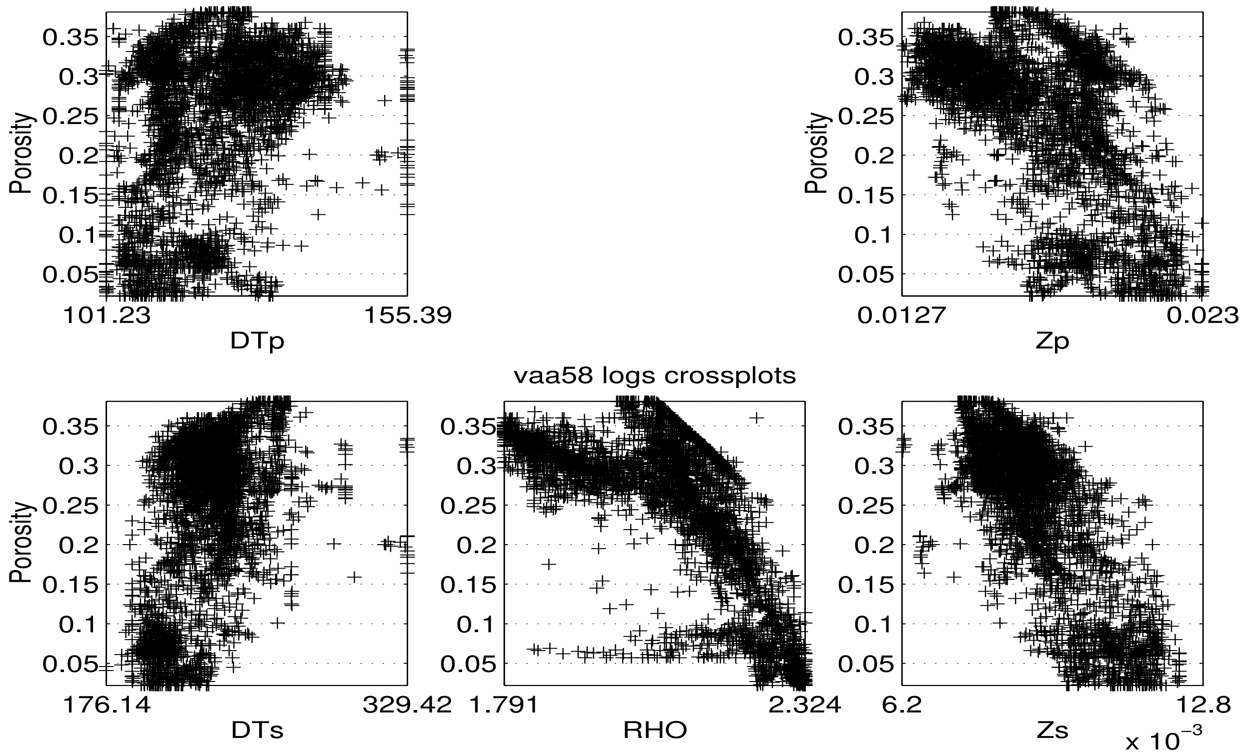


FIG. 1. Cross-plots between input training logs and porosity log from wells 58c and 58cst.

| correlation coefficients (%) | | | | | |
|---|---|---|---|---|---|
| | $\rho$ | $DT_p$ | $Z_p$ | $DT_s$ | $Z_s$ |
| $\phi$ | -70.28 | 34.50 | -58.30 | 46.77 | -77.56 |
| $V_{sh}$ | 68.89 | -29.82 | 54.18 | -36.12 | 69.37 |
| $S_w$ | 70.76 | -43.72 | 63.77 | -17.87 | 56.22 |

Table 3. Correlation between input training logs and fluid properties corresponding to Figure 1.

Note the higher correlation between $S$-wave parameters and porosity than between $P$-wave parameters and porosity. Also density and impedances are better correlated with fluid properties than sonic transit times.

## Backpropagation Neural Network

Neural networks have been widely used in many different fields for classification and prediction (Wasserman, 1993; Bishop, 1996), especially where no theoretical relationships are established from physics between inputs, i.e. the log data and outputs, i.e. the model parameters. Applications of backpropagation neural networks (BPNN) appeared in geophysics in recent years for log-to-log predictions (Rogers *et al.*, 1992). As a supervised training neural network, that is to say training based on input-output pairs, the BPNN is quite appropriate for prediction applications and has been chosen for this study.

A BPNN is defined by an input layer, an output layer and a variable number of so-called hidden layers. A schematic representation of the backpropagation neural network built to predict fluid properties from logs is given on Figure 2.

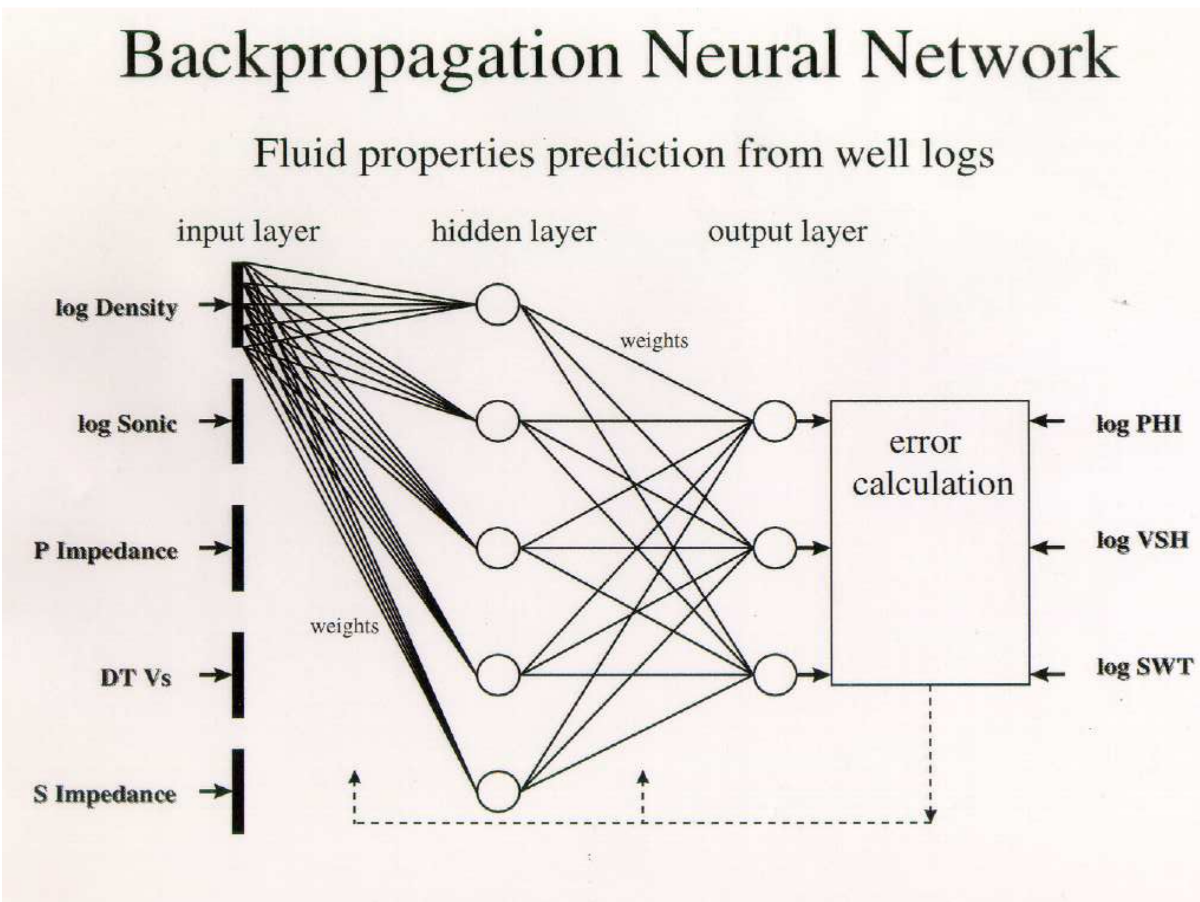


FIG. 2. Schematic representation of the backpropagation neural network used in this experiment

Each layer is composed of one or more "neurons" connected to each neuron of the prior and the next layers. The connection between two neurons is parameterized by a weight whose value is estimated during the training process so as to minimize the difference between the computed output of the network and the targeted output. Each individual neuron is connected to an additional weight independent of any inter-layer connection, which plays the role of an offset or *bias*.

The input layer is constructed with twenty five neurons. Each of the five input logs is windowed within a depth interval of five samples feeding five input neurons. This construction allows the network to incorporate local averages of the logs as well as individual samples. The network is trained to predict fluid properties at the depth of the central sample of the input window.

Windowing the input logs is justified by the fact that log measurements within 2.5ft depth interval are often correlated and also that porosity prediction based on consecutive measurements reduces the effect of noise in the data by mean of enhanced S/N ratio.

We built a network with one hidden layer composed of five neurons. Several configurations have been investigated with up to three hidden layers and a varying number of neurons per hidden layer. We have not observed a systematic decrease of the training error by increasing the number of neurons in the hidden layers. However, the optimization of the neural network in terms of the number of neurons, i.e. model parameters, has not been investigated yet.

The output layer consists of three neurons, each of which outputs one of the predicted fluid properties: $\phi$, $V_{sh}$ and $S_w$. Our backpropagation neural network previously described with three layers is constituted of 148 weights or model parameters.

The neurons in the input layer do not perform any operation, they are the access gate of the data. Each neuron in the hidden layer sums the values of the input neurons multiplied by the weight connecting the two neurons. This sum is then filtered by a transfer function $f$, the standard sigmoid (Figure 3) defined by

$$f(x) = \frac{1}{1+\exp(-x)},$$

which acts as a gain function and outputs a value between 0 and 1.

Note here the importance of data normalization and weight initialization. If the input lies outside approximatively [-5,5], the hidden neurons reach saturation by outputting either 0 or 1. The effect of saturation results in slow network training because at saturation the error function becomes invariant to small perturbations in the weight values.

The neurons in the output layer compute the sum over each neuron in the hidden layer, multiplied by the weights connecting the hidden layer to the output layer. The neurons in the output layer do not apply the transfer function.

In short, the neural network performs the following operations on the input $I_i$ at the

$i^{th}$ neuron of the input layer, to produce the output $O_k$ at the $k^{th}$ neuron of the output layer:

$$O_k = \sum_j f(\sum_i I_i w_{ij} + b_j) w_{jk} + b_k,$$

where $w_{ij}$ are the weights connecting the input layer to the hidden layer, $w_{jk}$ the weights connecting the hidden layer to the output layer, $b_j$ the bias of the $j^{th}$ neuron of the hidden layer, and $b_k$ the bias of the $k^{th}$ neuron of the output layer.

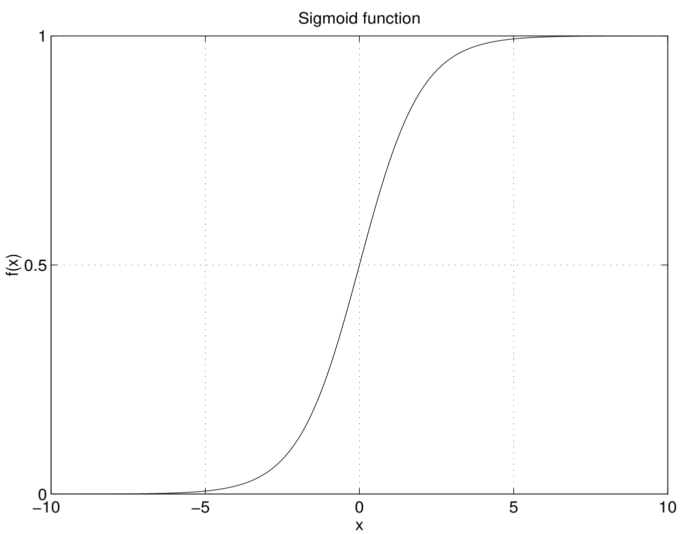


FIG. 3. Transfer function (sigmoid) applied by the neurons in the hidden layer.

The network is trained so as to produce a targeted output vector, denoted $T$, when fed with an input vector $I$. During the training phase, the weights of the network are estimated in such a way as to minimize the misfit $\mathcal{E}$ defined as the mean square error between the $n^{th}$ targeted output vector $\mathbf{T^n}$, of components $T_k^n$, and the $n^{th}$ calculated output vector $\mathbf{O^n}$, of components $O_k^n$,

$$\mathcal{E} = \frac{1}{N_{tr} N_{ne}} \sum_{n=1}^{N_{tr}} \sum_{k=1}^{N_{ne}} (O_k^n - T_k^n)^2.$$

In the expression of the misfit $\mathcal{E}$, $N_{ne}$ denotes the number of neurons in the output layer and $N_{tr}$ the number of training pairs. The $n^{th}$ training pair is composed of an input training vector $\mathbf{I^n}$ and a corresponding targeted output vector $\mathbf{T^n}$.

Each training pair is composed of input windowed logs and a corresponding targeted output; $\phi$, $V_{sh}$ and $S_w$. The training data set is formed by sliding the window down along the logs by one sample at a time, so that a total of 3550 training pairs are used to compute the error function defined as the mean square error between calculated and targeted outputs.

### Training strategy

The weights are initialized to random numbers ranging between -1 and 1. The BPNN is trained by iteratively adjusting the weights of the connections so as to minimize the misfit $\mathcal{E}$. The original training algorithm, often called "delta rule" in the literature, is a steepest descent algorithm which, at each iteration, updates the weights by propagating the error back into the network's weights. We first implemented a steepest descent method which requires thousands of iterations over the training process before reaching an acceptable solution. We significantly improved the convergence rate of the training process by coupling our BPNN with a BFGS (Broyden-Fletcher-Goldfarb-Shanno) (Polak, 1971) optimization program developed by The Rice Inversion Project as part of the Hilbert Class Library.

The transfer function causes nonlinearity in the optimization problem. This nonlinearity takes the form of local "plateaus" in the error function, where the error is quasi-invariant. This occurs when the sigmoid function is at saturation, that is to say when the output becomes invariant to a small change in the weights of the hidden layer. A consequence is an extremely slow convergence rate of the optimization algorithm.

If the initialization of the weights is close enough to the solution, the algorithm will eventually converge towards the global minimum of the error function.

Instead of implementing a global optimization algorithm, exploring the entire model space, we adopted a pragmatic optimization strategy which consists in looping over random initializations of the weights until the error function gets smaller than a threshold or the number of re-initialization reaches a maximum in which case the best solution is kept. Satisfactory results were obtained using that procedure.

## Fluid properties predictions at log scale

The data set used to train the network consists of the normalized log data from the wells 58c and 58cst, resulting in 3550 training vectors represented in Figure 4. A training vector is defined by five input; density $\rho$, $P$-wave sonic $DT_p$, $P$-wave impedance $Z_p$, $S$-wave sonic $DT_s$, $S$-wave impedance $Z_s$, and three output; porosity $\phi$, volume fraction of shale $V_{sh}$, and water saturation $S_w$.

None of the constraints related to limiting network saturation were applied to obtain the presented results. The optimization of the weights requires about a hundred iterations of the BFGS algorithm and the best result, corresponding to the minimum error was retained from fifty random re-initializations of the weights.

The result of the training is shown in Figure 5, with a root mean square (rms) error of 0.239 between predicted and targeted fluid properties. Rms errors are given for each output in Table 4 along with a prediction coefficient $C_p$ between predicted $P$ and targeted $T$ output defined as

$$C_p = \frac{T \cdot P}{||T||\ ||P||},$$

which measures how well the prediction matches the actual data. As one could expect from the correlation given in Table 3, porosity is the best prediction.

The logs from test wells 56c and 55a are rescaled according to the normalization applied to the training wells 58c and 58cst. Log data and prediction results on the test wells 56c and 55a are shown in Figures 6, 7, 8, and 9.

Note that the range of porosity in well 56c is much narrower than the range of porosity for which the neural network has been trained to predict. One can observe the resulting shift between predicted and actual porosity in Figure 7. Also, Figure 8 illustrates the normalization problem, where the $P$-sonic lies outside the range [0,1] at the bottom of the log. Although the prediction in Figure 9 does not seem to be affected, the test data are also partially outside the range of the training data.

| | | $\phi$ | | $V_{sh}$ | | $S_w$ | |
|---|---|---|---|---|---|---|---|
| Well | No. of samples | rms $\times 10^2$ | $C_p$ % | rms $\times 10^2$ | $C_p$ % | rms $\times 10^2$ | $C_p$ % |
| 58c+st | 3550 | 11.78 | 98.46 | 15.18 | 93.77 | 24.26 | 93.54 |
| 56c | 2574 | 20.03 | 95.51 | 20.60 | 92.02 | 29.91 | 93.29 |
| 55a | 2088 | 21.10 | 94.04 | 24.08 | 90.04 | 30.77 | 91.21 |

Table 4. Rms error and prediction coefficient between predicted and targeted fluid properties from log data at log scale.

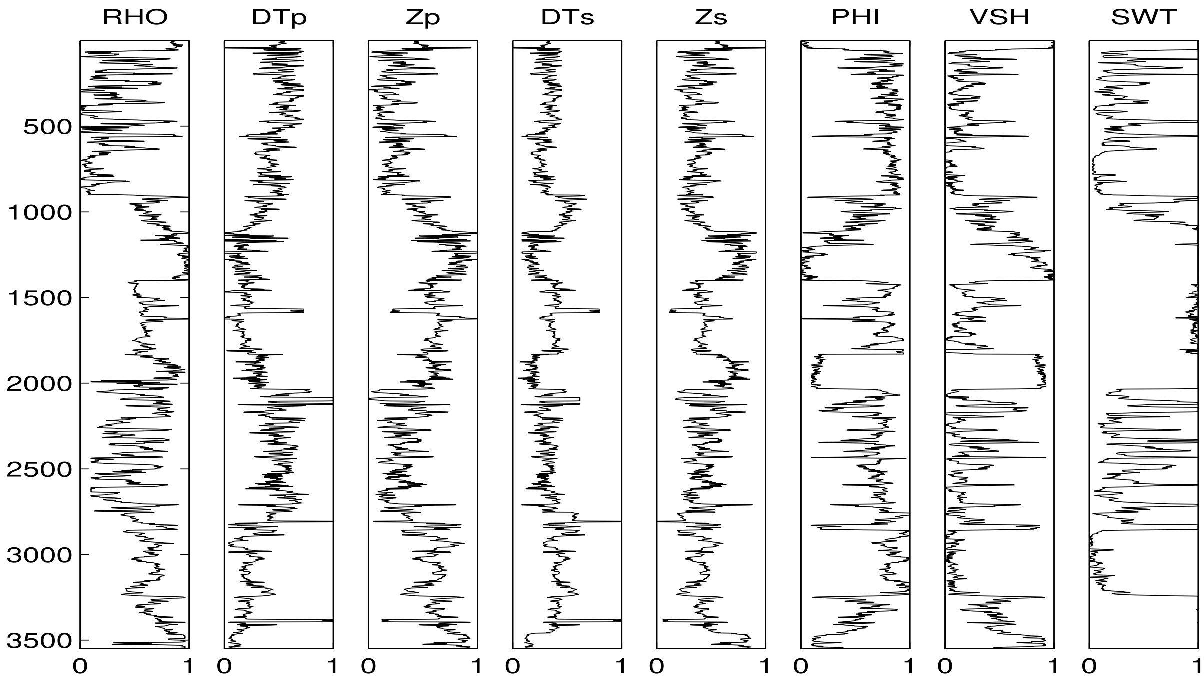


FIG. 4. Fluid properties prediction at log scale. Normalized training data set from wells 58c and 58cst. Input training vectors: ($\rho$, $DT_p$, $Z_p$, $DT_s$, $Z_s$). Output training vectors: ($\phi$, $V_{sh}$, $S_w$). Vertical coordinates are in samples.

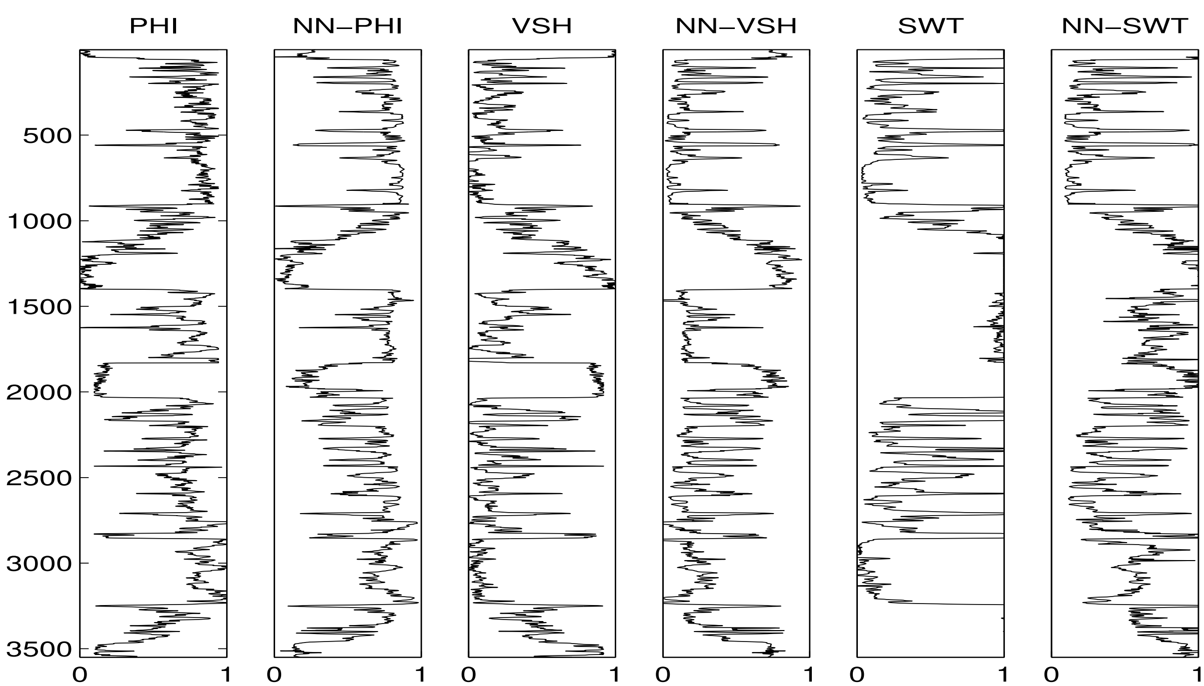


FIG. 5. Fluid properties prediction at log scale. Neural network prediction on the training data from wells 58c and 58cst. Targeted output logs ($\phi$, $V_{sh}$, $S_w$) are plotted against predicted output (*NN-PHI*, *NN-VSH*, *NN-SWT*).

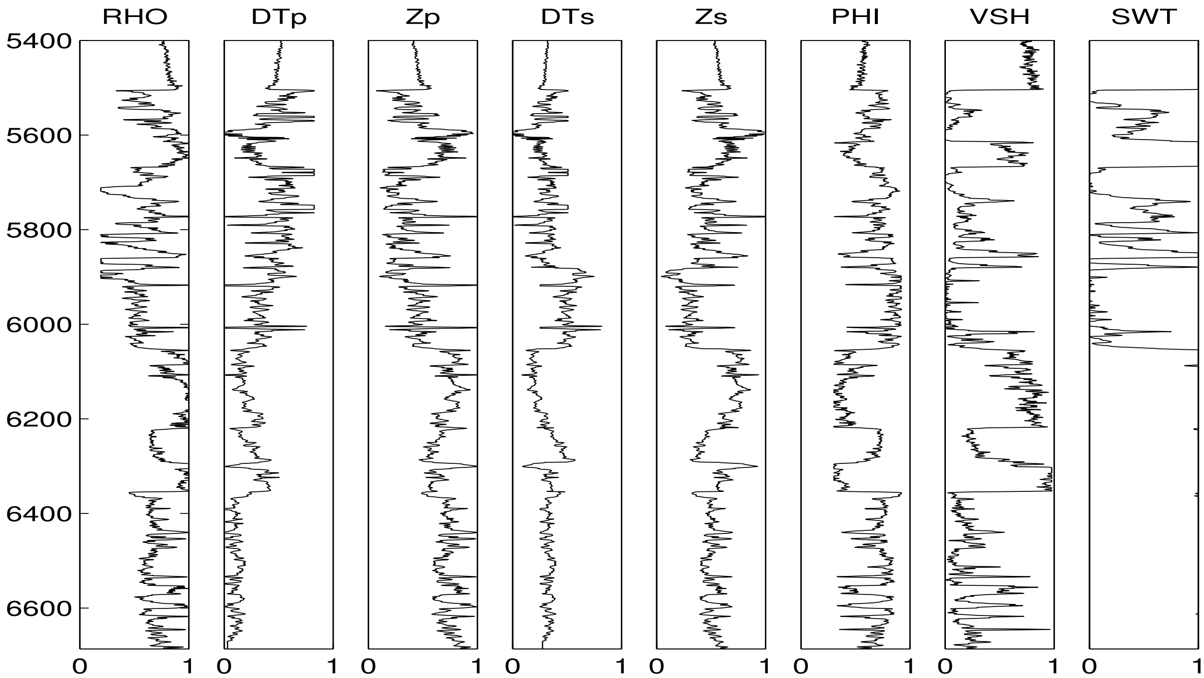


FIG. 6. Fluid properties prediction at log scale. Normalized test data set from well 56c. Vertical coordinates are in measured depth.

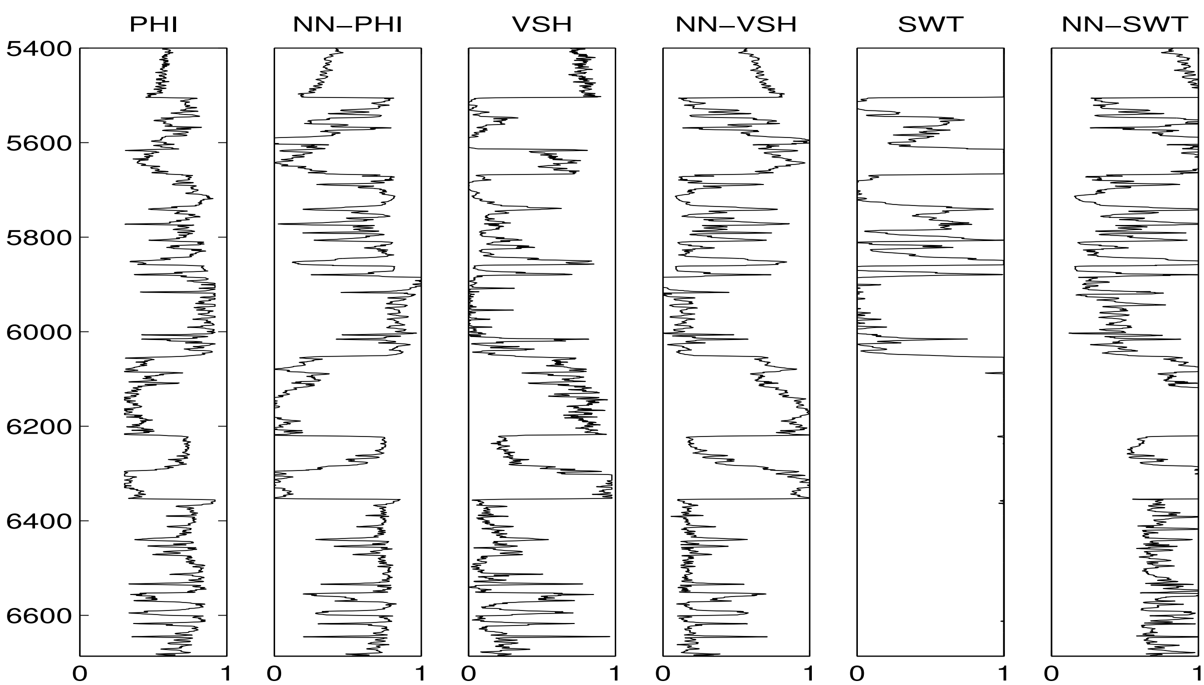


FIG. 7. Fluid properties prediction at log scale. Neural network prediction on test data from well 56c. Vertical coordinates are in measured depth.

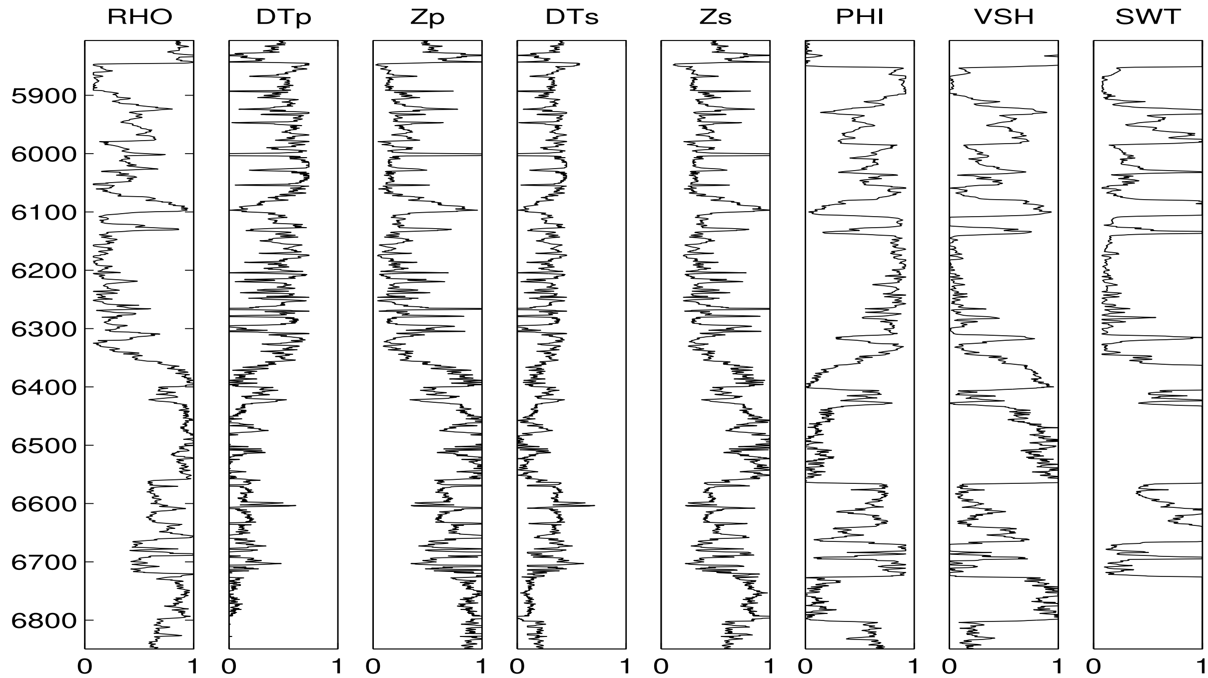


Fig. 8. Fluid properties prediction at log scale. Normalized test data set from well 55a. Vertical coordinates are in measured depth.

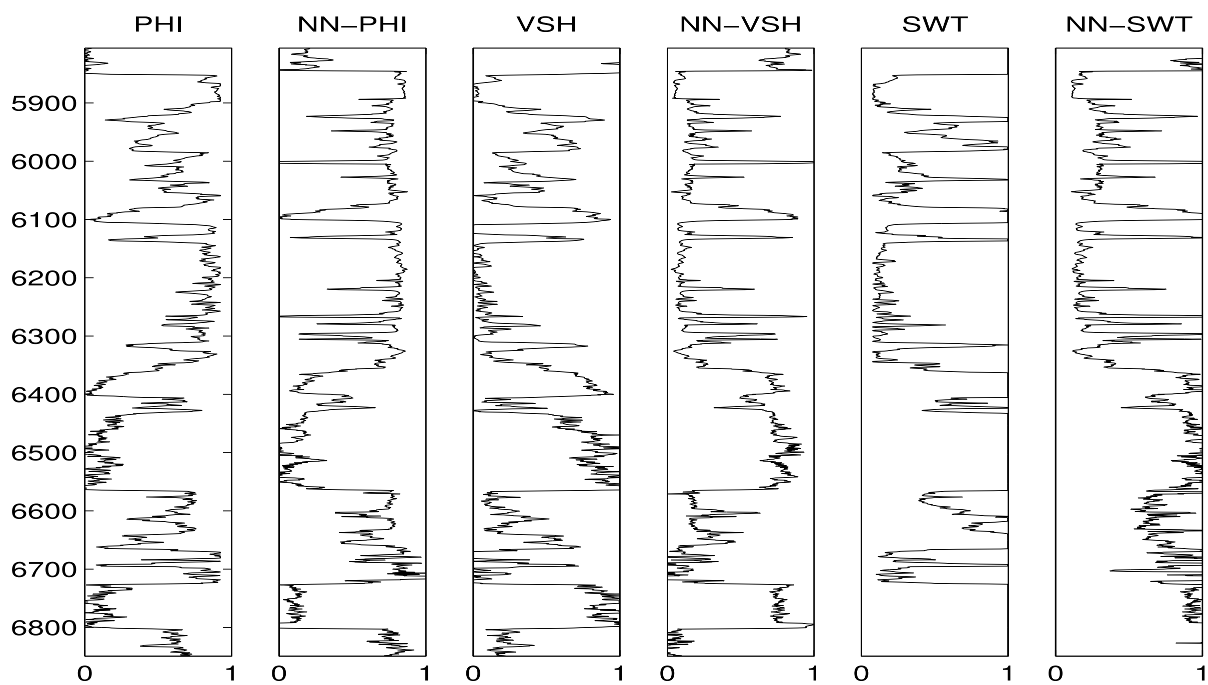


Fig. 9. Fluid properties prediction at log scale. Neural network prediction on test data from well 55a. Vertical coordinates are in measured depth.

### Fluid properties predictions at seismic scale

The preceding analysis was carried out again at a seismic scale in order to compare the log-to-log prediction with the seismic inversion and attributes prediction at similar resolution scales.

The log data were upscaled to a seismic scale by averaging the logs within a time window of 2 ms calculated from the sonic. This is shorter than the seismic wavelet length but it corresponds to the seismic time sampling. The time logs were then resampled in depth at a 3 ft interval and consequently the training wells consisted of 591 training vectors.

The results are displayed in Figures 10 to 15. Prediction errors and prediction coefficients are summarized in Table 5. The training error is reduced, which is not surprising since the effect of upscaling is to smooth the logs and so to remove the high frequency components. One can observe in Figures 13 and 15 that normalization is more critical.

| Well | No. of samples | $\phi$ | | $V_{sh}$ | | $S_w$ | |
|---|---|---|---|---|---|---|---|
| | | rms $\times 10^2$ | $C_p$ % | rms $\times 10^2$ | $C_p$ % | rms $\times 10^2$ | $C_p$ % |
| 58c+st | 591 | 09.95 | 98.91 | 12.48 | 95.68 | 22.18 | 94.49 |
| 56c | 426 | 23.87 | 93.58 | 21.43 | 92.99 | 36.66 | 89.32 |
| 55a | 347 | 16.40 | 96.28 | 20.87 | 92.62 | 31.42 | 89.56 |

Table 5. Rms error and prediction coefficient between predicted and targeted fluid properties from log data at seismic scale.

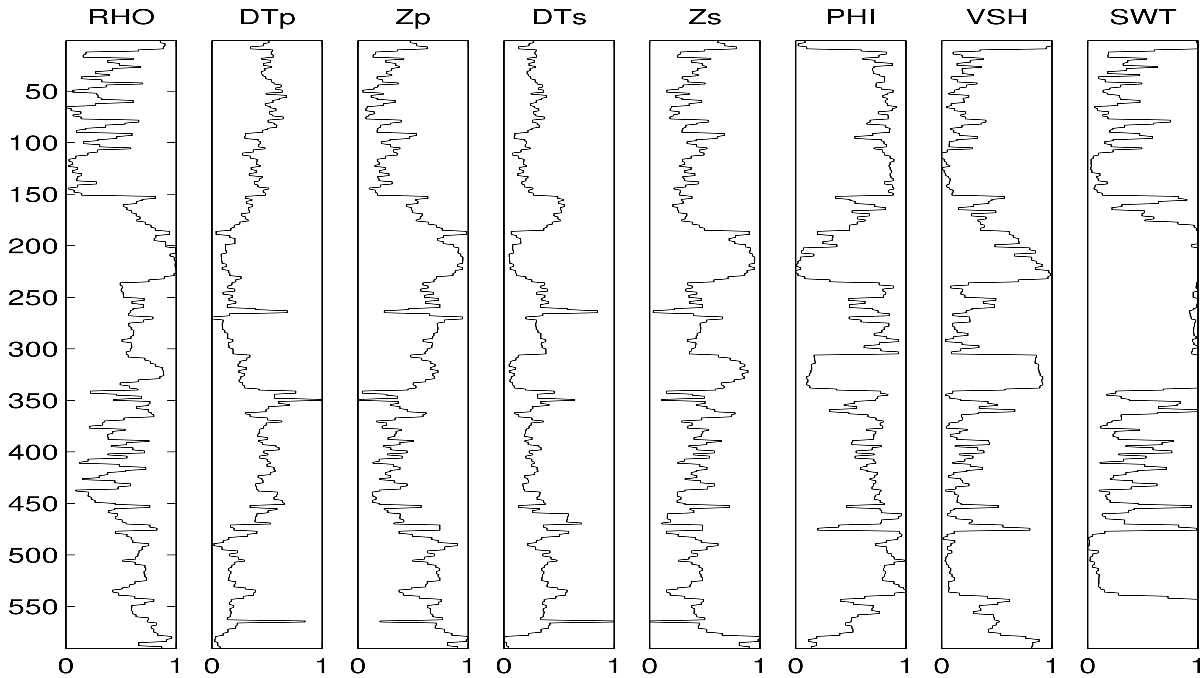


FIG. 10. Fluid properties prediction from log data at seismic scale. Normalized training data set from wells 58c and 58cst. Input training vectors: ($\rho$, $DT_p$, $Z_p$, $DT_s$, $Z_s$). Output training vectors: ($\phi$, $V_{sh}$, $S_w$). Vertical coordinates are in samples.

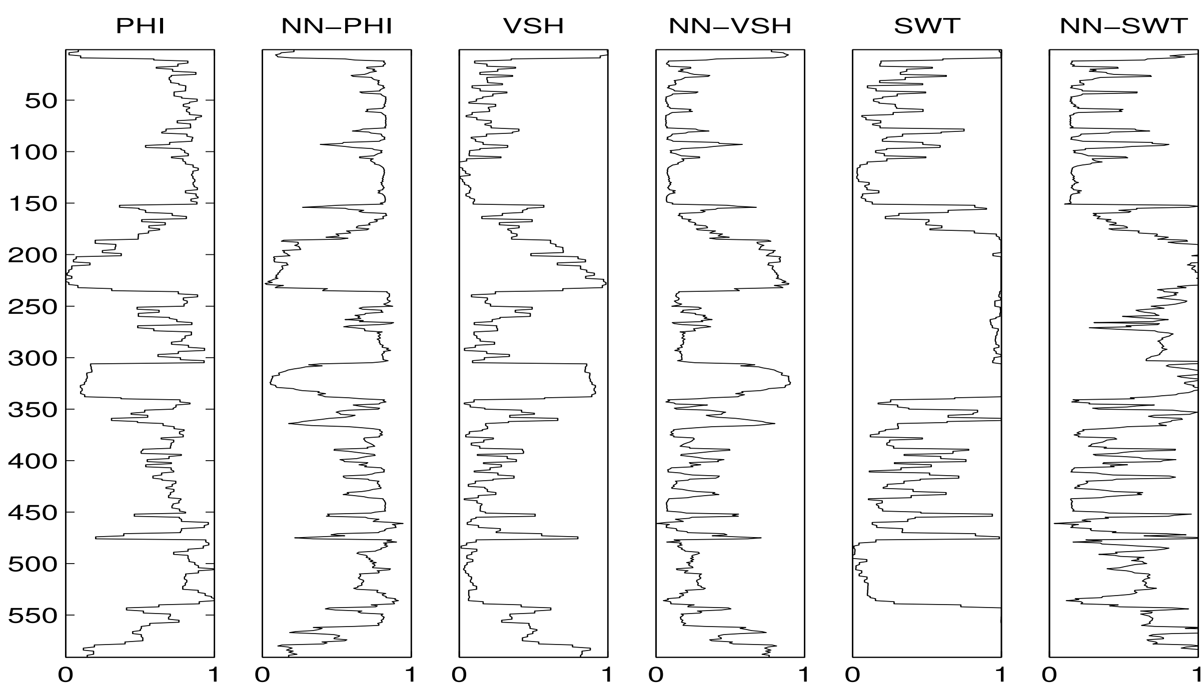


FIG. 11. Fluid properties prediction from log data at seismic scale. Neural network prediction on the training data from wells 58c and 58cst. Targeted output logs ($\phi$, $V_{sh}$, $S_w$) are plotted against predicted output (*NN-PHI*, *NN-VSH*, *NN-SWT*).

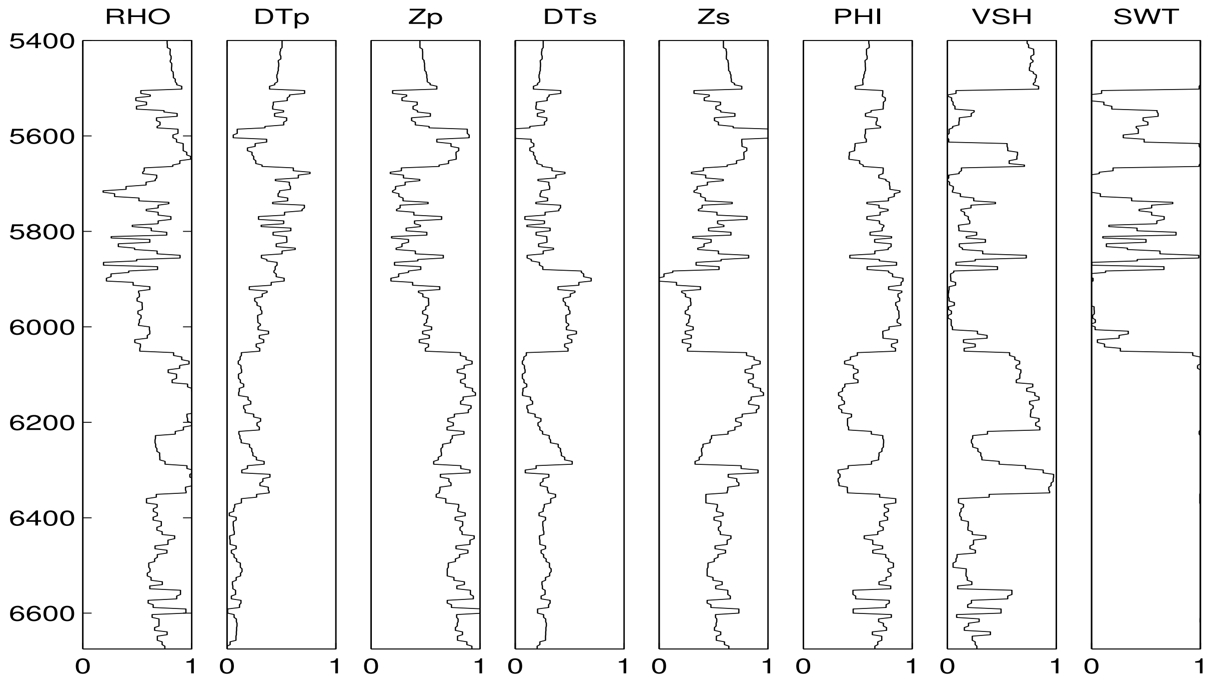


FIG. 12. Fluid properties prediction from log data at seismic scale. Normalized test data set from well 56c. Vertical coordinates are in measured depth.

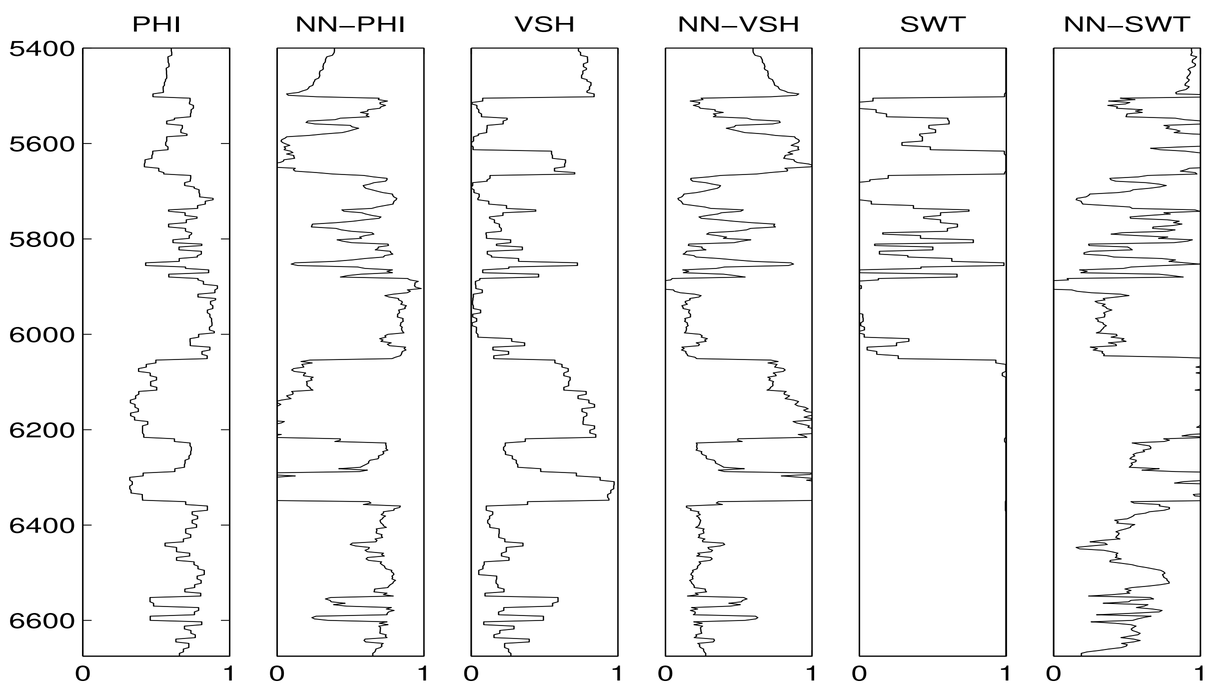


FIG. 13. Fluid properties prediction from log data at seismic scale. Neural network prediction on test data from well 56c. Vertical coordinates are in measured depth.

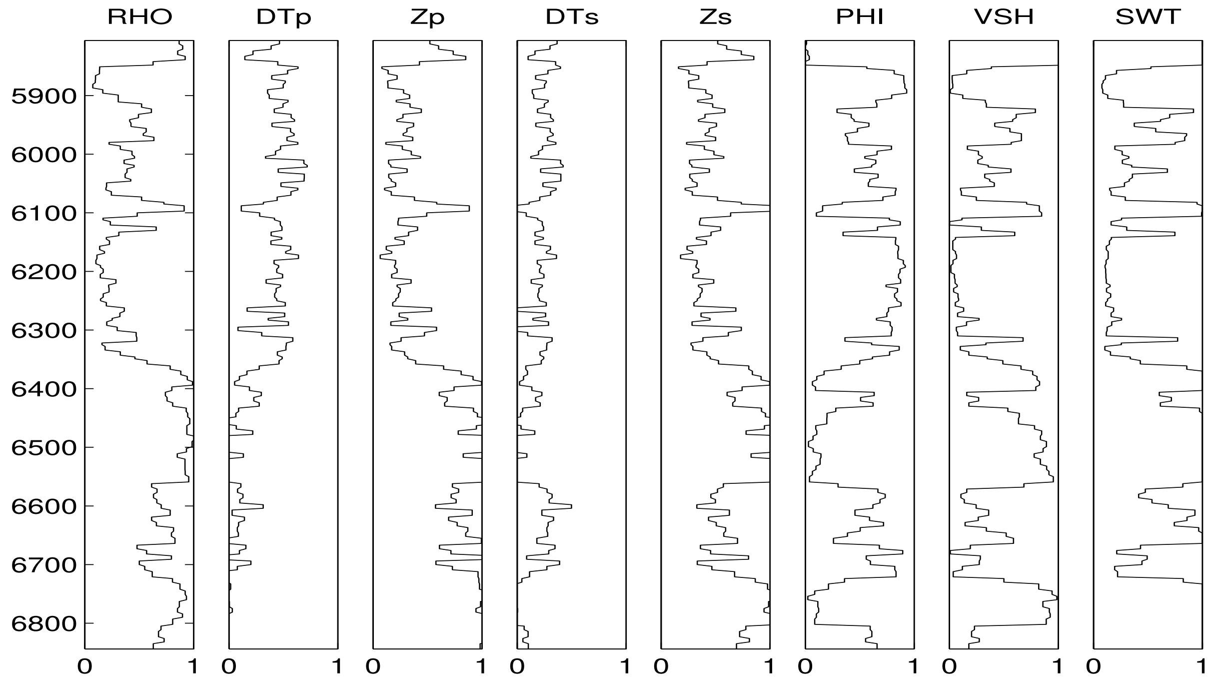


FIG. 14. Fluid properties prediction from log data at seismic scale. Normalized test data set from well 55a. Vertical coordinates are in measured depth.

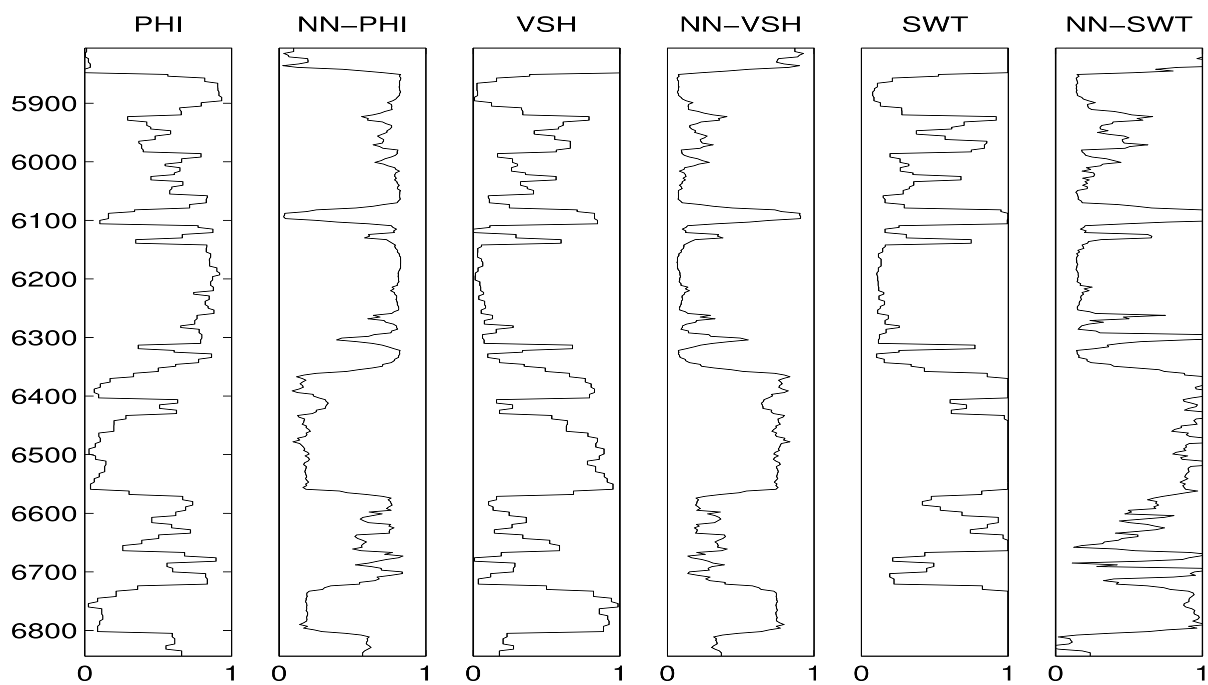


FIG. 15. Fluid properties prediction from log data at seismic scale. Neural network prediction on test data from well 55a. Vertical coordinates are in measured depth.

## Fluid properties predictions from seismic inversion

As described above, the log data were upscaled to seismic scale by averaging the log within a 2 ms time window estimated from the sonic and then resampled in depth at a 3 ft sampling.

A 1D elastic model was built combining sonic, $V_p/V_s$ ratios from Table 2, and checkshot information and used to generate 1.5D synthetic offset seismograms using a visco-elastic modeling code developed by The Rice Inversion Project (Gockenbach and Symes, 1996a; Gockenbach and Symes, 1996b). A linear regression fit provides a smooth background elastic model. The reflection coefficients are computed from the relative fluctuations of the elastic parameters using a high frequency asymptotic approximation. 61 CMP traces were generated at 1ms time sampling, and 75 ft offset spacing to a maximum offset of 4500 ft.

The synthetic seismic traces were then inverted for the relative fluctuations of density $\Delta\rho/\rho$, $P$-wave velocity $\Delta V_p/V_p$, and shear wave velocity $\Delta V_s/V_s$ using an inversion code also developed at TRIP. Theoretical developments of the seismic modeling and inversion methods can be found in (Stolt and Weglein, 1985).

40 iterations of a conjugate gradient method were required to achieve an excellent fit with a rms error of 0.02. The density, $V_p$, $V_s$, and impedances logs, reconstructed from the inverted relative fluctuations and background model, are shown for the different wells in Figures 16, 18, and 20.

The neural network is trained on the reconstructed logs to predict fluid properties at seismic scale. Training logs are represented in Figure 16. The training rms error is 0.144 and is detailed for $\phi$, $V_{sh}$ and $S_w$ in Table 6. Prediction of the training data is shown in Figure 17, where most of the features of the porosity log are accurately predicted. For the test wells 56c and 55a (Figures 18 and 20), the normalized sonic logs swing outside the range of the training data, resulting in unreliable prediction below 6400 ft depth for well 56c and 55a, which is almost half the depth range of the 55a logs. However, prediction coefficients for well 56c are significantly higher than those for well 55a.

| | | $\phi$ | | $V_{sh}$ | | $S_w$ | |
|---|---|---|---|---|---|---|---|
| Well | No. of samples | rms $\times 10^2$ | $C_p$ % | rms $\times 10^2$ | $C_p$ % | rms $\times 10^2$ | $C_p$ % |
| 58c+st | 591 | 11.43 | 98.56 | 12.77 | 95.48 | 18.22 | 96.31 |
| 56c | 426 | 32.22 | 89.03 | 34.89 | 83.24 | 38.08 | 92.55 |
| 55a | 347 | 57.61 | 67.74 | 64.69 | 63.30 | 80.70 | 71.66 |

Table 6. Rms error and prediction coefficient between predicted and targeted fluid properties from seismic inversion.

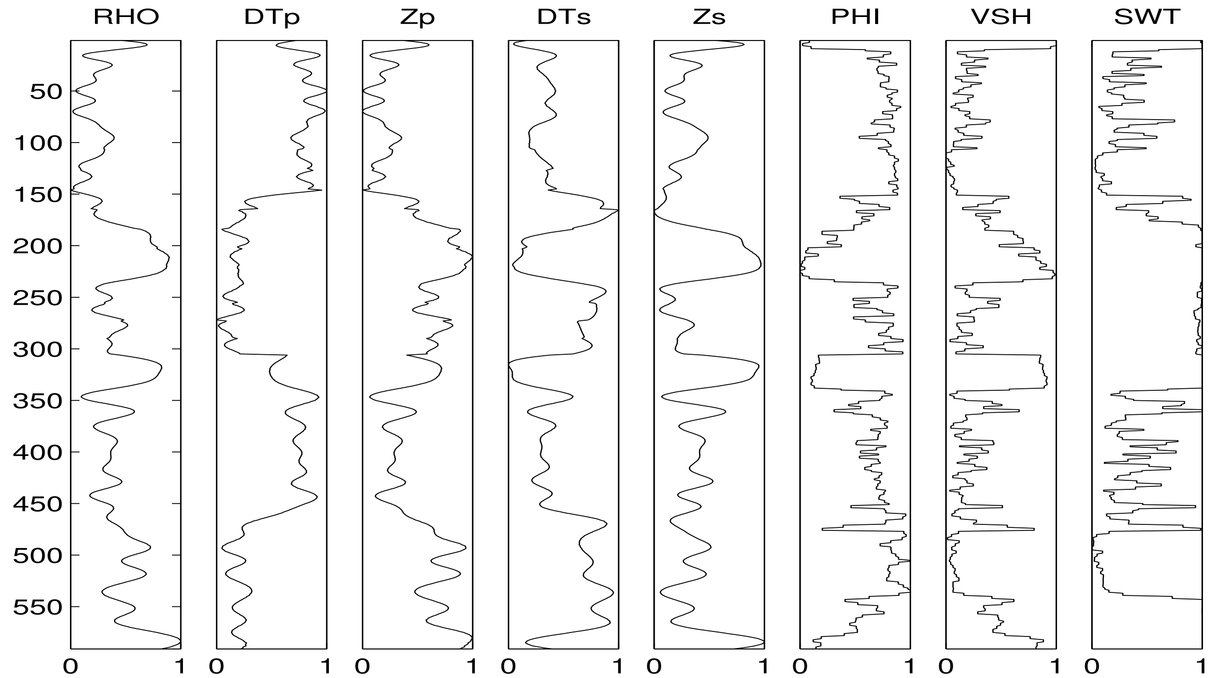


FIG. 16. Fluid properties prediction from seismic inversion. Normalized training data set from wells 58c and 58cst. Input training vectors: ($\rho$, $DT_p$, $Z_p$, $DT_s$, $Z_s$). Output training vectors: ($\phi$, $V_{sh}$, $S_w$). Vertical coordinates are in samples.

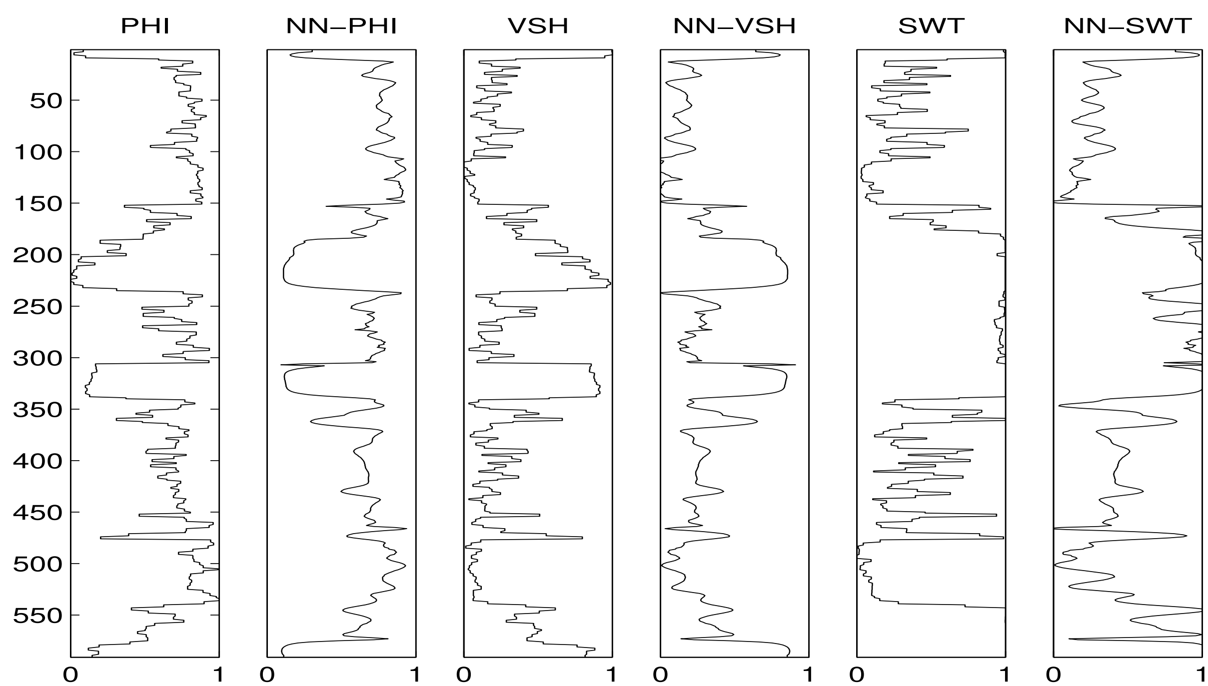


FIG. 17. Fluid properties prediction from seismic inversion. Neural network prediction on the training data from wells 58c and 58cst. Targeted output logs ($\phi$, $V_{sh}$, $S_w$) are plotted against predicted output (*NN-PHI*, *NN-VSH*, *NN-SWT*).

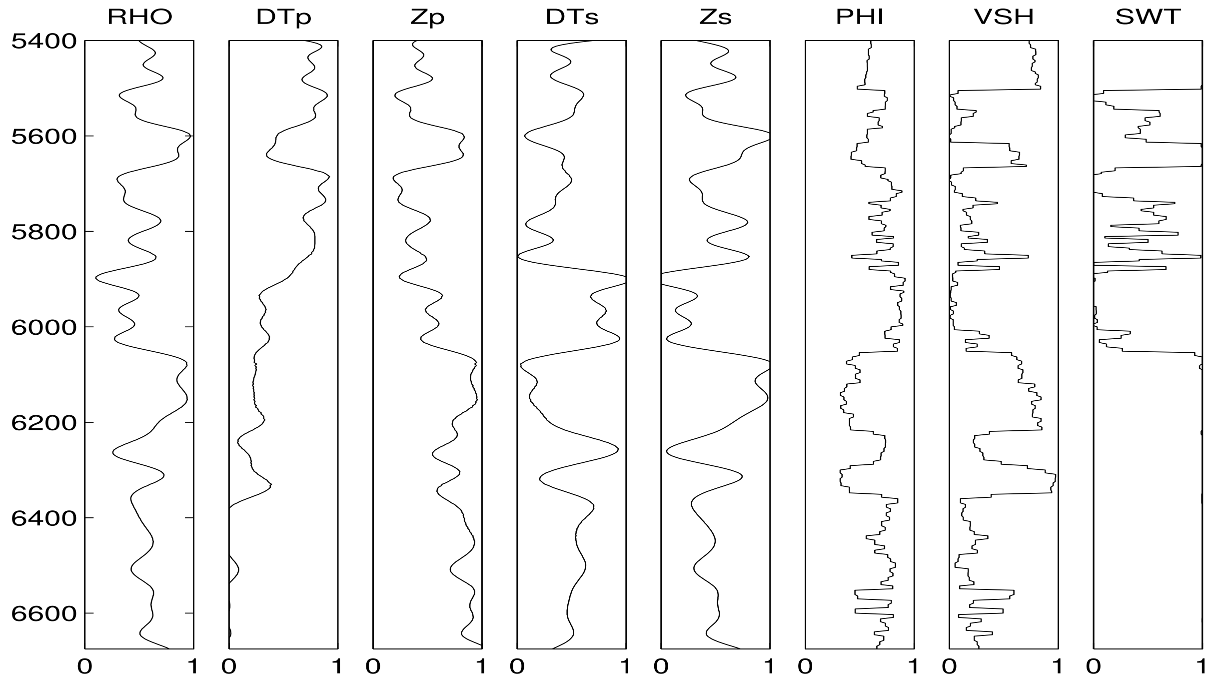


FIG. 18. Fluid properties prediction from seismic inversion. Normalized test data set from well 56c. Vertical coordinates are in measured depth.

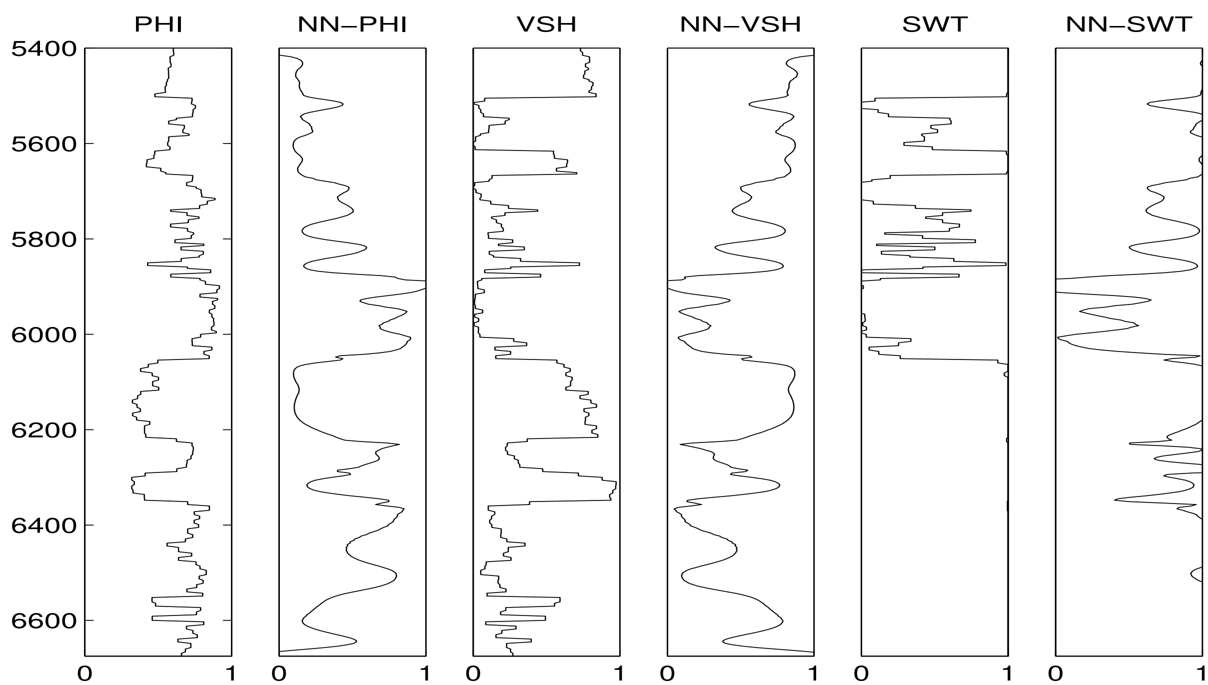


FIG. 19. Fluid properties prediction from seismic inversion. Neural network prediction on test data from well 56c. Vertical coordinates are in measured depth.

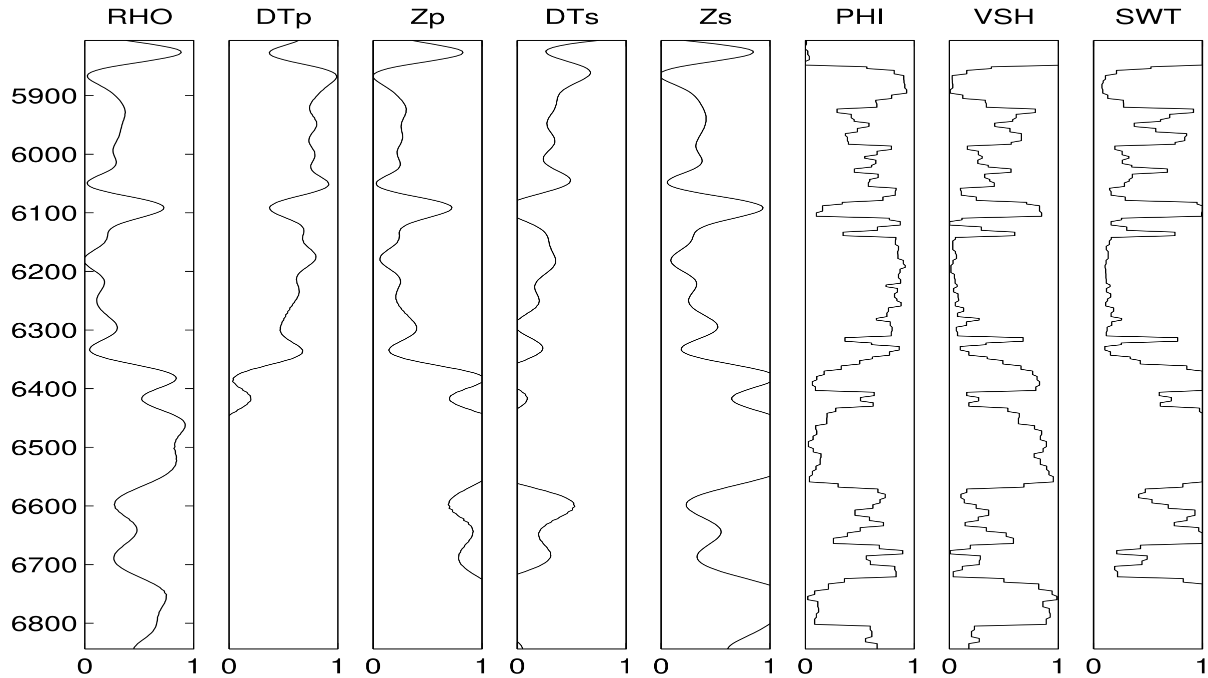


FIG. 20. Fluid properties prediction from seismic inversion. Normalized test data set from well 55a. Vertical coordinates are in measured depth.

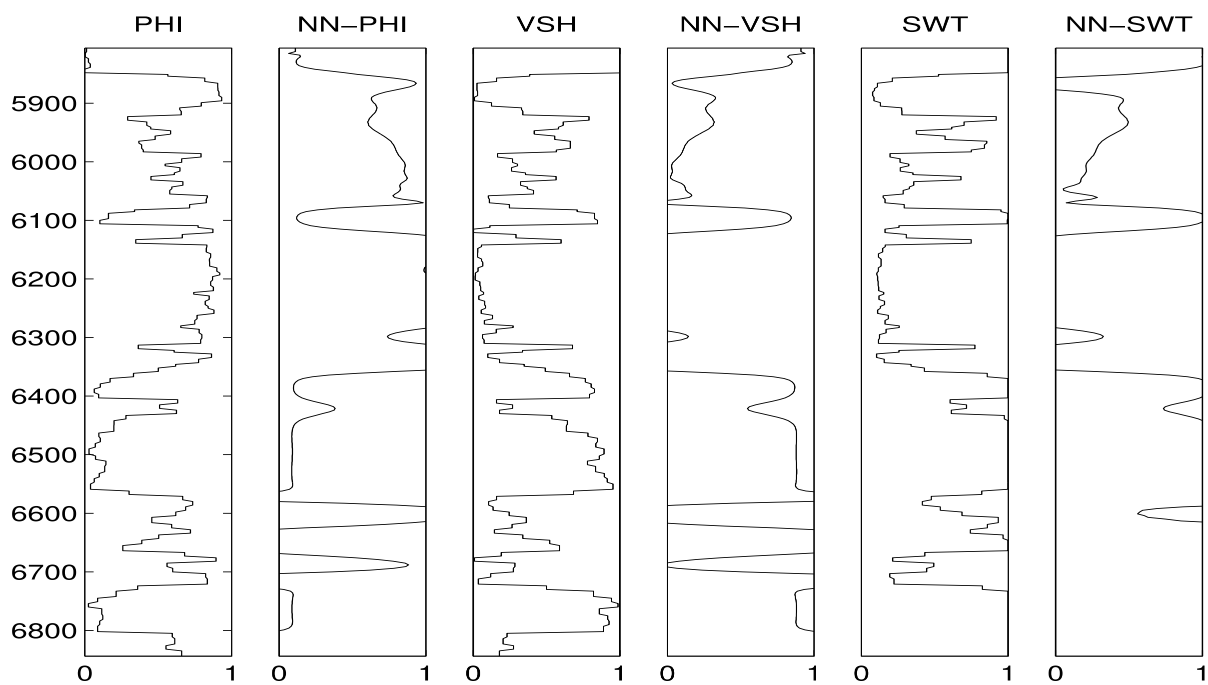


FIG. 21. Fluid properties prediction from seismic inversion. Neural network prediction on test data from well 55a. Vertical coordinates are in measured depth.

## Fluid properties predictions from seismic attributes

A commercial program was used to generate single trace seismic attributes from the pre-stack synthetic seismic traces. A total of 27 attributes, listed in Table 7 were generated from the synthetic CMPs.

| Class I attributes - correlation coefficients (%) | | | |
|---|---|---|---|
| | $\phi$ | $V_{sh}$ | $S_w$ |
| Real part of complex trace | 11.08 | -07.62 | -02.23 |
| Imaginary part of complex trace | -17.49 | 15.37 | 07.47 |
| Trace envelope | -24.05 | 25.91 | 11.48 |
| Time derivative of the envelope | 12.18 | -09.77 | -00.19 |
| Second derivative of the envelope | -00.76 | 01.30 | -02.30 |
| Instantaneous phase | -03.39 | 01.44 | -04.31 |
| Instantaneous frequency | 11.52 | -09.50 | -07.25 |
| Envelope weighted inst. freq. | 10.94 | -08.87 | -11.29 |
| Thin bed indicator | 08.61 | -07.18 | -03.03 |
| Acceleration of phase | 04.01 | -03.61 | -02.60 |
| Dominant frequency | -00.03 | 02.05 | -05.06 |
| Band width | 05.82 | -06.44 | -10.32 |
| Instantaneous Q factor | -02.22 | 03.66 | 01.57 |
| Normalized amplitude | 15.00 | -11.61 | -08.20 |
| Envelope amplitude modulated phase | -17.59 | 18.31 | 07.80 |
| Relative acoustic impedance | -33.57 | 30.04 | 14.30 |
| Class III attributes - correlation coefficients (%) | | | |
| | $\phi$ | $V_{sh}$ | $S_w$ |
| Trace envelope | -24.94 | 22.55 | 06.82 |
| Time derivative of the envelope | 19.83 | -17.23 | 17.17 |
| Second derivative of the envelope | -14.00 | 13.88 | 48.11 |
| Instantaneous phase | -23.28 | 15.01 | -01.94 |
| Instantaneous frequency | 13.59 | -10.65 | 11.76 |
| Envelope weighted inst. freq. | 15.50 | -11.86 | 04.91 |
| Acceleration of phase | -21.33 | 17.34 | 15.27 |
| Dominant frequency | 13.58 | -10.65 | 11.76 |
| Band width | 30.16 | -26.59 | 12.03 |
| Instantaneous Q factor | -21.81 | 19.93 | -03.03 |
| Apparent polarity | 08.02 | -00.34 | 28.01 |

Table 7. Correlation coefficients between seismic attributes and fluid properties

Class I attributes are computed trace by trace individually. Class III attributes are derived from Class I attributes and are the instantaneous attributes picked at the peak of the trace envelope (Taner *et al.*, 1979).

Using programs from the Center for Wave Phenomena, a normal moveout (NMO) correction is first applied to the CMPs and the depth-velocity model converted into time. The NMO traces are then converted into depth and attributes are computed within a depth interval encompassing the reservoir. The generated attributes are then stacked and resampled every 3 ft.

The neural network is trained on the 27 seismic attributes to predict fluid properties. Each input attribute is windowed by three samples, so that the network is constituted of 81 input neurons. The hidden layer is again defined with five neurons. Therefore a total of 428 weights is to be estimated.

Training and prediction results summarized in Table 8 show that the training error is the smallest of the four exercises. On the other hand, prediction error on the test wells is the largest. Predicted logs are displayed in Figures 22 to 24 against targeted logs.

The small error achieved during the learning phase may be explained by the numerous degrees of freedom introduced in the model by the large number of weights. Despite the small correlation between input and output training data, the large number of degrees of freedom enables the network to fit with a small residual error but the generalization capabilities of the network are limited. Attributes are not displayed in this report but the normalization of the test data, calculated on the training data, shows that the range of the attributes generated from the test data differs from the range of the attributes used to train the network.

It is interesting to see how individual attributes are correlated with fluid properties. The correlation, defined in page 4, is detailed in Table 7. The largest correlation, which occurs between relative acoustic impedance and porosity, does not exceed 33.6%, to be compared with the 70% correlation observed between training log data.

| Well | No. of samples | $\phi$ | | $V_{sh}$ | | $S_w$ | |
|---|---|---|---|---|---|---|---|
| | | rms $\times 10^2$ | $C_p$ % | rms $\times 10^2$ | $C_p$ % | rms $\times 10^2$ | $C_p$ % |
| 58c+st | 591 | 08.78 | 99.15 | 09.34 | 97.61 | 11.31 | 98.59 |
| 56c | 426 | 55.36 | 72.36 | 63.99 | 31.89 | 80.92 | 45.28 |
| 55a | 347 | 68.10 | 48.51 | 71.50 | 37.38 | 84.84 | 44.33 |

Table 8. Rms error and prediction coefficient between predicted and targeted fluid properties from seismic attributes.

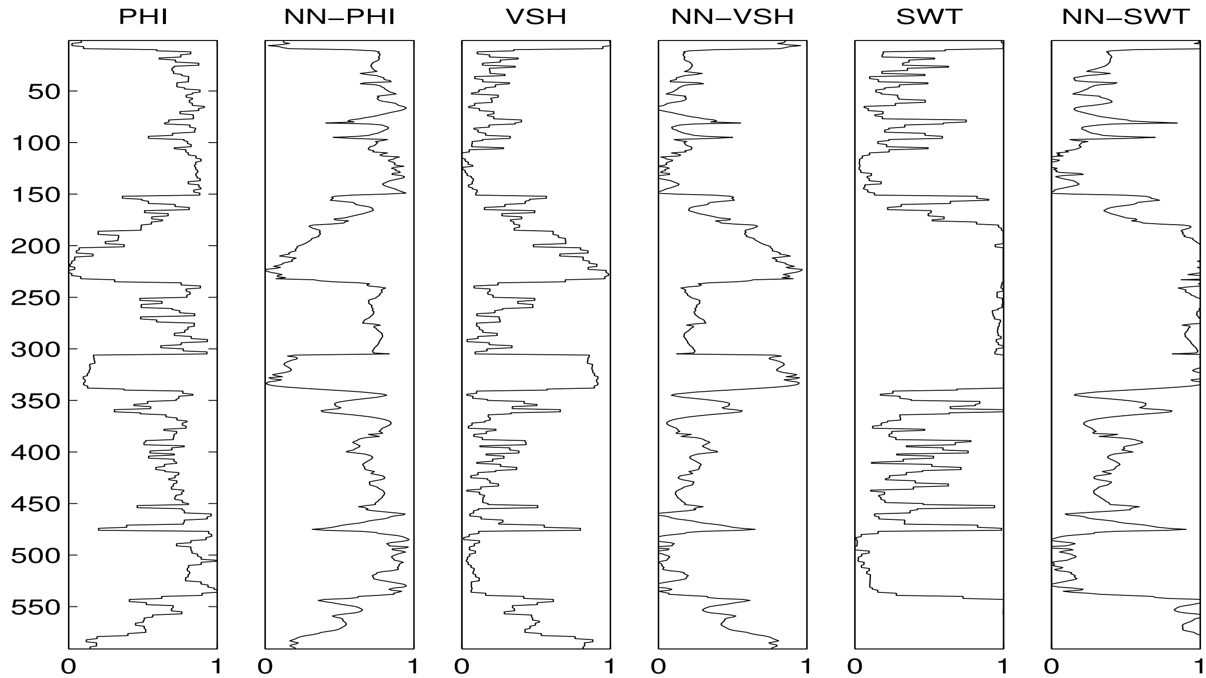


FIG. 22. Fluid properties prediction from seismic attributes. Neural network prediction on the training data from wells 58c and 58cst. Targeted outputs logs ($\phi$, $V_{sh}$, $S_w$) are plotted against predicted outputs (*NN-PHI*, *NN-VSH*, *NN-SWT*). Vertical coordinates are in samples.

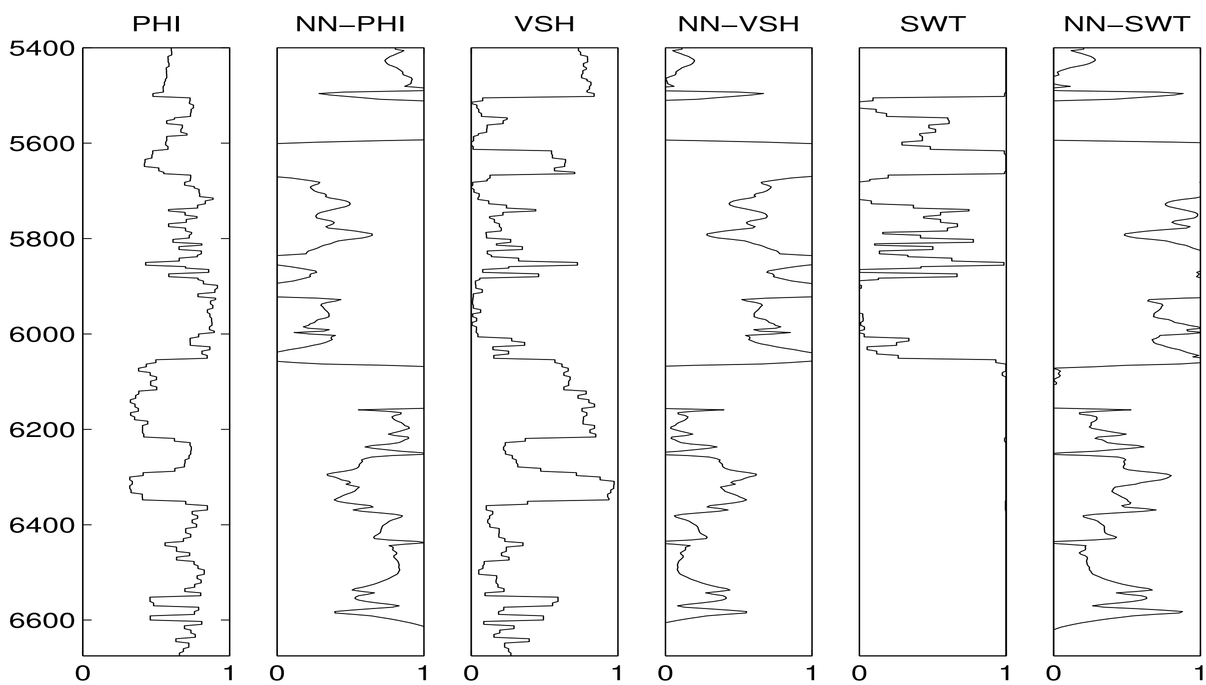


FIG. 23. Fluid properties prediction from seismic attributes. Neural network prediction on test data from well 56c. Vertical coordinates are in measured depth.

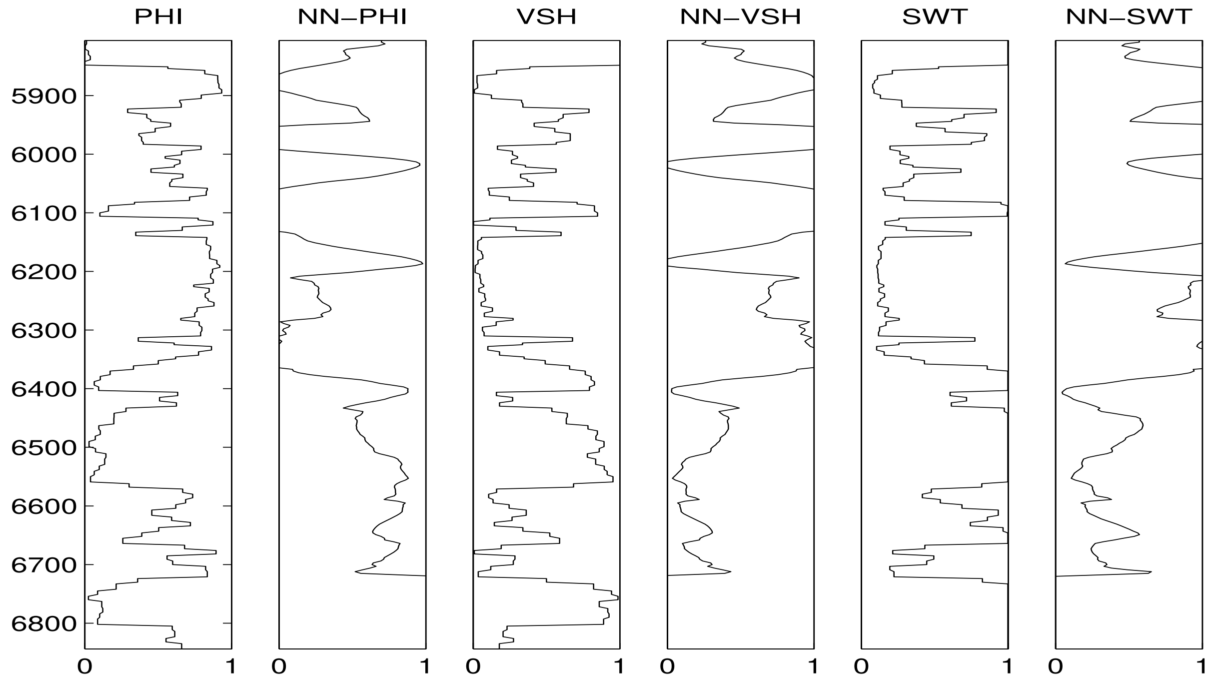


FIG. 24. Fluid properties prediction from seismic attributes. Neural network prediction on test data from well 55a. Vertical coordinates are in measured depth.

## Discussion

The previous results raise two important issues: data normalization and neural network optimization. Results obtained from training on seismic attributes, i.e. good learning but poor generalization, suggest to reduce the model space by training the network on density, $P$ and $S$-wave impedance only, which are best correlated with fluid properties, to predict either porosity, volume fraction of shale or water saturation independently.

### Predict on relative fluctuations

Normalization of the data may lead to unreliable predictions when a network is used to predict outside the domain the network has been trained.

Figure 25 shows crossplots between relative fluctuations of $\rho$, $V_p$, $V_s$, $Z_p$, $Z_s$ and $\phi$. Relative fluctuations have been computed by subtracting the linear fit of each log. The quasi-linear increasing trend of the $P$-impedance log, particularly significant for the wells 56c and 55a, is in this way removed.

The correlations coefficients calculated between relative fluctuations of input logs and fluid properties, summarized in Table 9, are similar or even better than correlations between "raw" logs given in Figure 1 and Table 3.

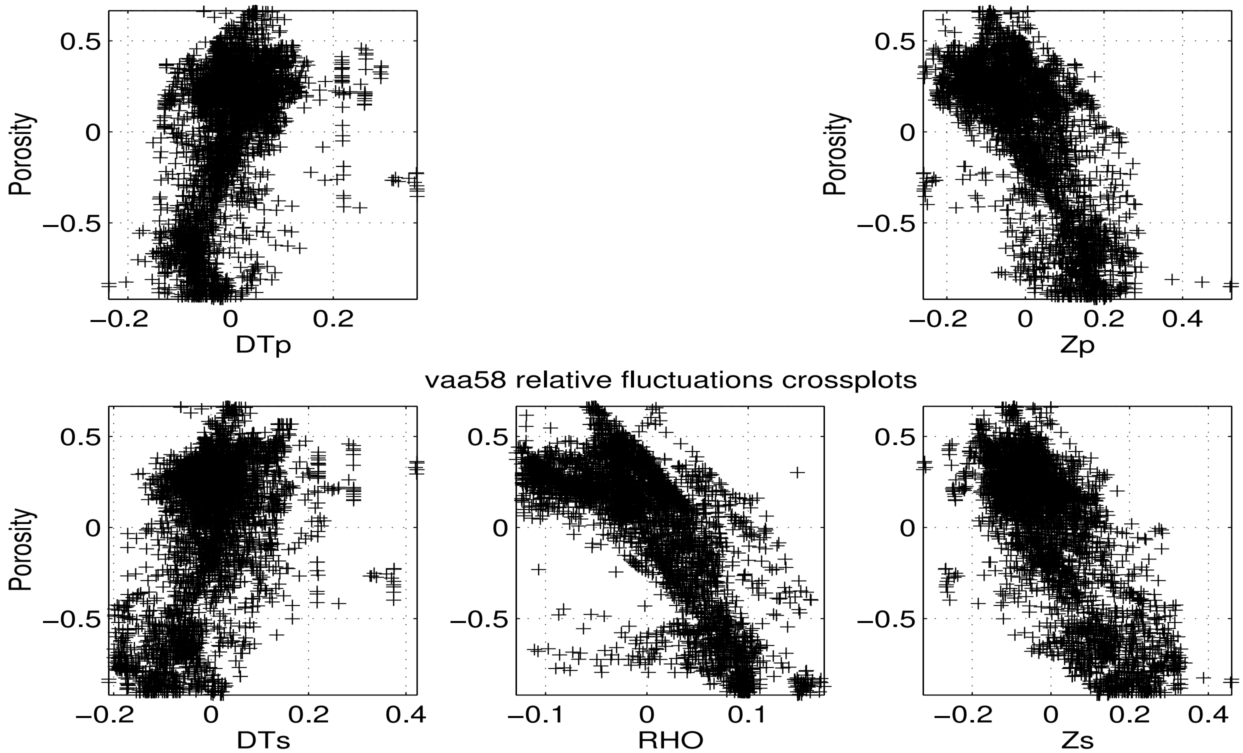


FIG. 25. Cross-plots between relative fluctuations of input logs and porosity log from training wells 58c and 58cst.

| correlation coefficients (%) | | | | | |
|---|---|---|---|---|---|
| | $\rho$ | $DT_p$ | $Z_p$ | $DT_s$ | $Z_s$ |
| $\phi$ | -73.35 | 45.03 | -70.87 | 46.82 | -76.18 |
| $V_{sh}$ | 72.28 | -34.23 | 63.96 | -35.29 | 68.05 |
| $S_w$ | 70.08 | -26.05 | 58.55 | -20.69 | 54.75 |

Table 9. Correlation coefficients between relative fluctuations of input logs and fluid properties.

The following results were obtained from log-scale training of the network, for porosity only, on relative fluctuations of density, $P$-wave and $S$-wave impedances. The network is therefore limited to three by five input neurons, i.e. three input logs and five samples window per log, and a single output neuron to represent porosity. The network is again built with one hidden layer of five neurons, so that there are a total of 86 network weights.

Results are summarized in Table 10. The rms error on porosity prediction is significantly reduced compared to the values obtained from training on the "raw" logs and given in Table 4. The logs are displayed in Figures 26, 27, and 28.

Resolution in terms of correlation between targeted and predicted porosity is very similar (compare prediction coefficients in Table 4 and Table 10), but the "amplitude" is better recovered by training on relative fluctuations, reflected in the rms error is smaller.

| Well | No. of samples | $\phi$ | |
|---|---|---|---|
| | | rms $\times 10^2$ | $C_p$ % |
| 58c+st | 3350 | 05.21 | 98.01 |
| 56c | 2574 | 03.98 | 98.83 |
| 55a | 2088 | 09.02 | 94.05 |

Table 10. Rms error and prediction coefficient between targeted and predicted porosity from training on relative fluctuations of $\rho$, $Z_p$ and $Z_s$.

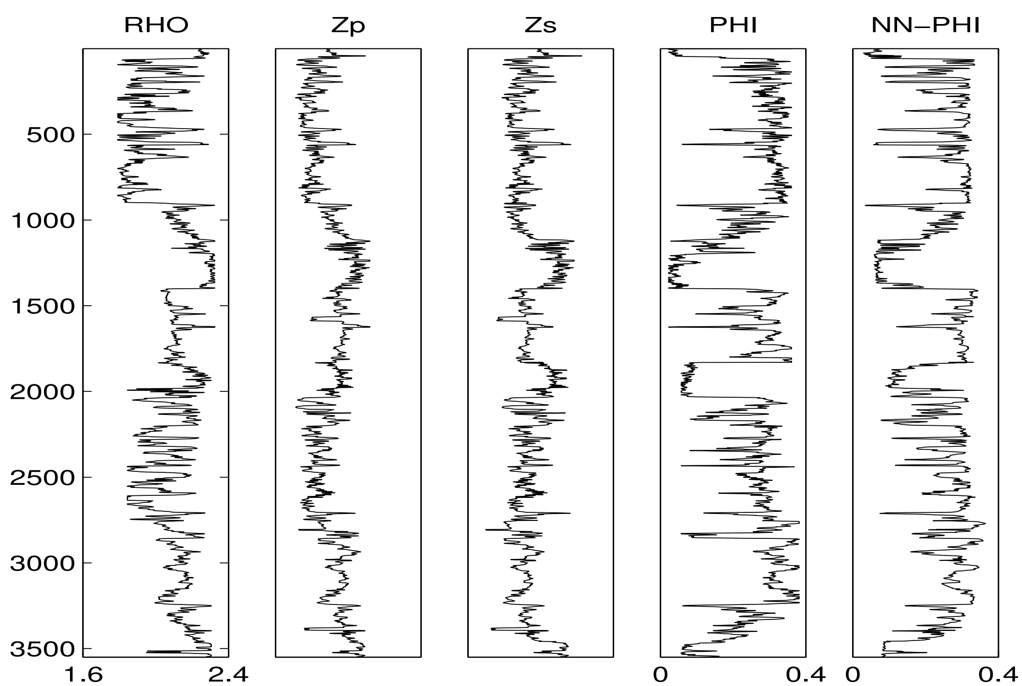


FIG. 26. Porosity prediction from relative fluctuations of density and impedances logs. Training data set from wells 58c and 58cst. Input training vectors: ($\rho$, $Z_p$, $Z_s$). Output training vector: $\phi$. Vertical coordinates are in samples. $P$-impedance ranges within [0.01,0.03] and $S$-impedance within [0.005,0.015].

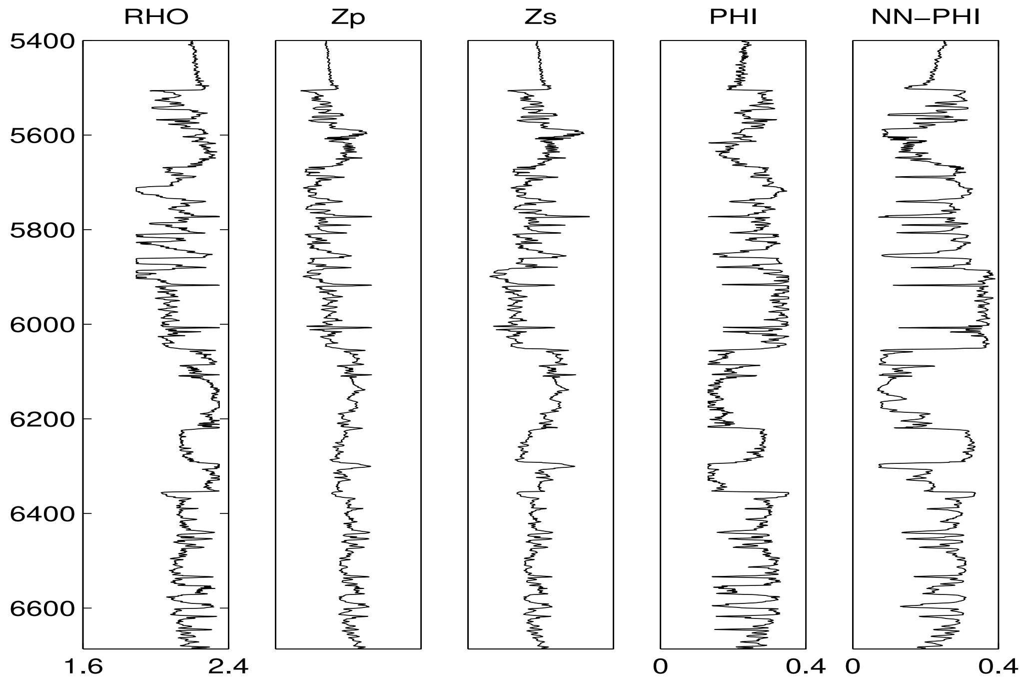


FIG. 27. Porosity prediction from relative fluctuations of density and impedances logs. Test data from well 56c. Vertical coordinates are in measured depth.

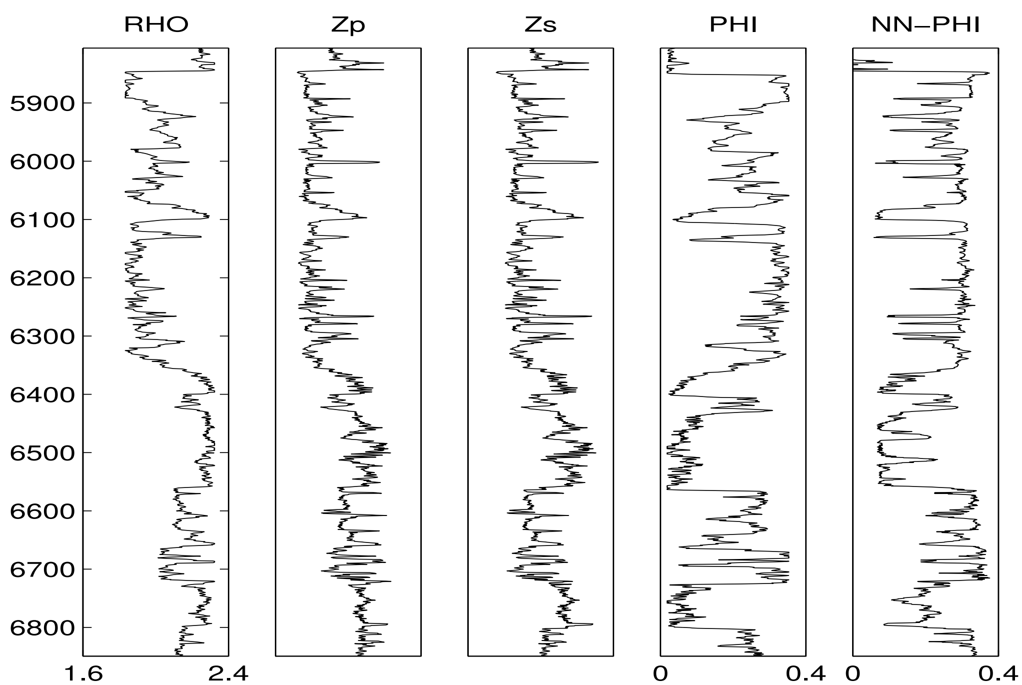


FIG. 28. Porosity prediction from relative fluctuations of density and impedances logs. Test data from well 55a. Vertical coordinates are in measured depth.

### Classification neural network

A different issue concerns the prediction tool itself. We observed that correlation between relative fluctuations of the logs and fluid properties is at least as good as between absolute values of the logs and fluid properties and we showed that predicting on relative fluctuations of the logs improves porosity prediction.

Futhermore, we are more concerned with assessing with what degree of confidence the porosity lies within a given interval than a definite value of the porosity. Following this idea, we used a "classification" version of the backpropagation neural network to predict interval porosity from relative fluctuations of density and impedances logs.

The "classification" network differs from its "prediction" version previously used, by the output layer. Each neuron in the output layer applies the sigmoid to its input value, forcing output in the range [0,1]. Instead of predicting the porosity value at a single output neuron, the output layer is constructed of as many neurons as porosity intervals one wants to distinguish. The network is train to output either one or zero at each neuron in the output layer. The output neuron associated to the interval porosity into which a training vector lies, is trained to produce one and all the other neurons zeros.

In the following experiment, we chose to divide the porosity range, roughly varying between 0 and 0.4, into 10 equally spaced intervals: [0,0.04], [0.04,0.08], [0.08,0.12], and so on up to [0.36,0.4].

The output layer has ten neurons, each of which is assigned to an interval porosity. No hidden layers were defined. The input layer consists, as previously, of three by five neurons. The network has a total of 160 weights.

The interval porosity prediction exercise is carried out on the logs at log scale. Rms errors and prediction coefficients, given in Table 11, are similar to those obtained for the previous experiment.

| Well | Nbr of samples | $\phi$ | |
|---|---|---|---|
| | | rms $\times 10^2$ | $C_p$ % |
| 58c+st | 3350 | 05.77 | 97.59 |
| 56c | 2574 | 05.98 | 97.43 |
| 55a | 2088 | 10.00 | 92.42 |

Table 11. Rms error and prediction coefficient between targeted and predicted interval porosity from training on relative fluctuations of $\rho$, $Z_p$ and $Z_s$.

The logs are shown in Figures 29, 30, and 31. The actual porosity log *PHI* is plotted against the targeted interval porosity, denoted *class-PHI*, and the predicted interval porosity *NN-PHI*. Interval porosity logs are graphically represented by their mean value in each interval.

A major improvement from using the classification network resides in its training rate. We observed a systematic drop of the misfit within less than ten iterations followed by a slow decrease of the misfit, whereas the prediction network converges slowly.

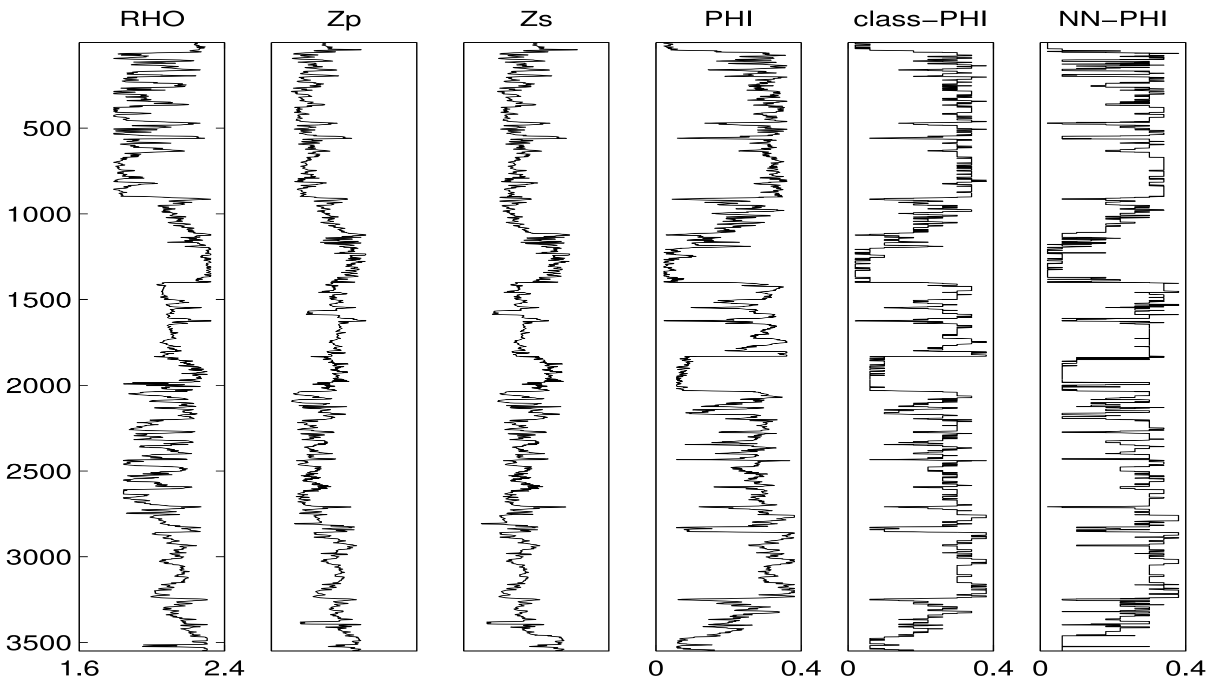


FIG. 29. Interval porosity prediction from relative fluctuations of density and impedances logs using a classification network. Training data set from wells 58c and 58cst. Input training vectors: ($\rho$, $Z_p$, $Z_s$). Output training vector: 10 intervals of $\phi$ between 0 and 0.4. *PHI* is the actual log, *class-PHI* the targeted interval porosity log represented by its mean value, and *NN-PHI* the predicted interval porosity. Vertical coordinates are in samples.

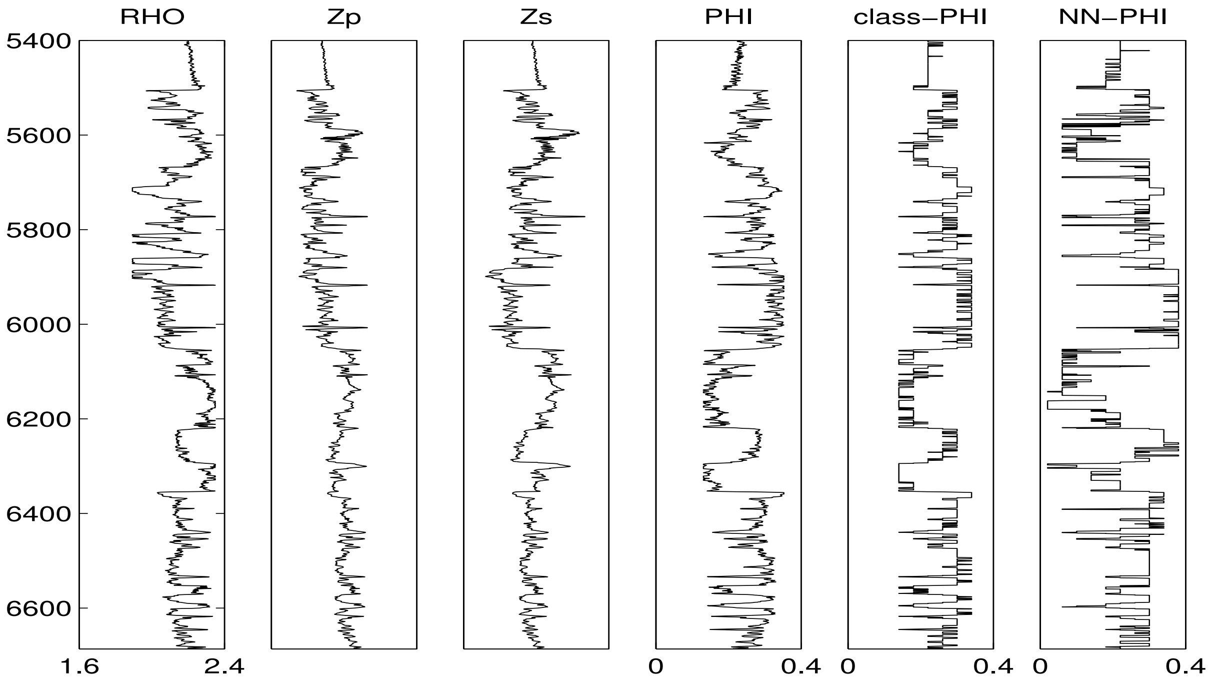


FIG. 30. Interval porosity prediction from relative fluctuations of density and impedances logs. Test data from well 56c. Vertical coordinates are in measured depth.

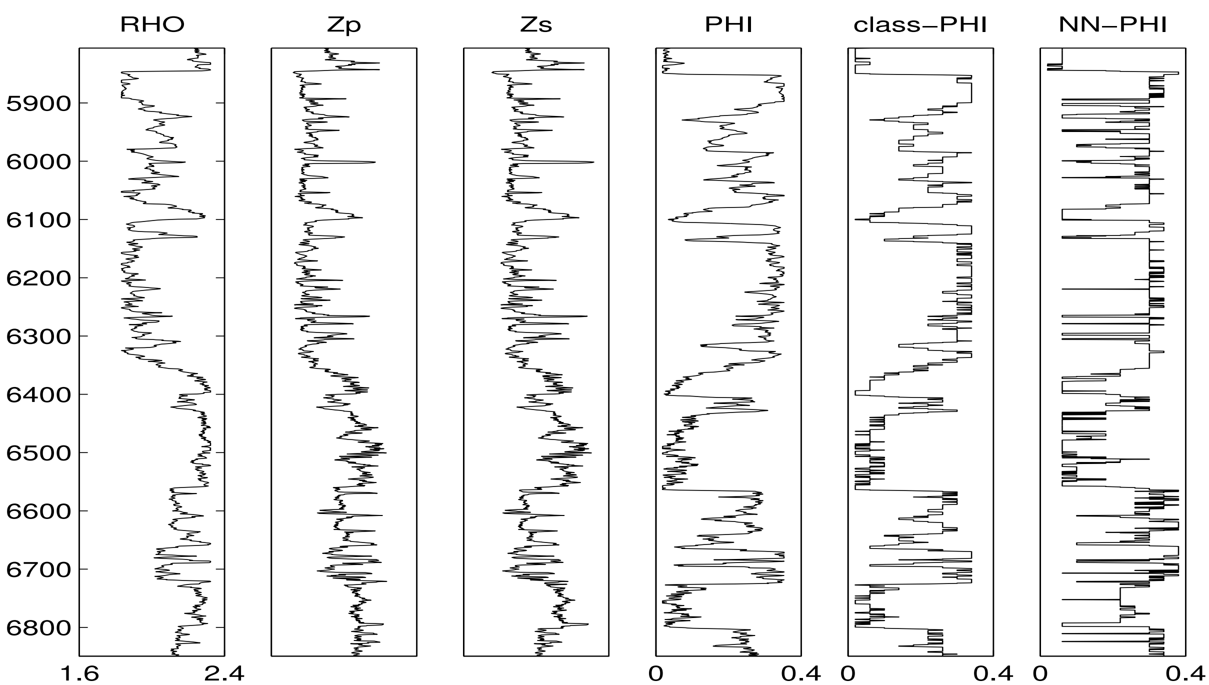


FIG. 31. Interval porosity prediction from relative fluctuations of density and impedances logs. Test data from well 55a. Vertical coordinates are in measured depth.

## Conclusion

The results presented in this report tend to demonstrate that seismic inversion can bring a significant improvement to the reservoir predictions, especially to the prediction of porosity. Petrophysical analysis of the logs is beyond the scope of this report which focuses on prediction performances. Correlation analysis shows a better match between seismic inversion results and fluid properties than between seismic attributes and fluid properties, particularly between density, seismic impedances and porosity. Futhermore, the large degree of freedom introduced by training on 27 attributes reduces the generalization capabilities of the network. Note that the very low correlation existing between some seismic attributes, such as dominant frequency or derivative of the trace envelope, and fluid properties does not exclude the existence of a more effective combination of attributes, eventually by coupling some seismic attributes with seismic inversion.

We observed that a network trained on density and impedances only performs as well, if not better than, a network trained on density, impedances and transit times.

The rms error on porosity prediction is significantly reduced by training on relative fluctuations of the logs, cleared from their "background" trend imprinted by a number of factors including pressure and temperature.

The use of a classification network appears to be particularly attractive for our application. The prime motivation is that rather than a definite value of the porosity, we would prefer to assess with what confidence porosity is between, for example, 20 and 30%. Also an uncertainty of the prediction can be compiled.

## Next steps

The logical step is to move into processing field seismic data. We also want to introduce rock physics in the training process in order to constrain the optimization. A previous exercise we carried out consisted in generating synthetic training vector satisfying the covariance matrix computed from the actual data, and pre-training the network with such "unlimited" training data set. The weights estimated from pre-training the network on synthetic data initialize the training phase on actual data. We propose to adopt a similar procedure to constrain the optimization by imposing physics in the training process of the neural network.

We propose to implement a probabilistic neural network (PNN) which, unlike the BPNN, estimates the probability density function for an input data vector to belong to a given interval. Clustering of the data can be investigated by analyzing the singular value decomposition of the cross-correlation matrix. The probabilistic neural network presents a number of advantages. PNN trains much faster that BPNN (Specht, 1991) with fewer model parameters, i.e. weights. The training process may be improved with additional new training vectors without having to re-train with the entire data set. Also, the caliper logs can be utilized as confidence factor on the data in the form of a weight applied to the probability density function.

**Acknowledgement and Disclaimer:** This work was partially supported by the National Science Foundation under grant numbers DMS 9404283 and DMS 9627355; the Office of Naval Research under grant number N00014-96-1-0156; the Air Force Office of Scientific Research, Air Force Materials Command, USAF, under grant number F49620-95-1-0339; and The Rice Inversion Project. TRIP Sponsors for 1997 are Advance Geophysical, Amoco Production Co., Conoco Inc., Cray Research Inc., Discovery Bay, Exxon Production Research Co., Interactive Network Technologies, Mobil Research and Development Corp., and Shell International Research. The US Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Office of Scientific Research, the Office of Naval Research, the National Science Foundation, or the US Government.

## REFERENCES

Bishop, C. M., 1996, *Neural Networks for Pattern Recognition*: Oxford, Clarendon Press.

Gockenbach, M. S., and Symes, W. W., 1996a, The Hilbert Class Library: a library of abstract C++ classes for optimization and inversion: *Computers and Mathematics with Applications*, 32:6.

Gockenbach, M. S., and Symes, W. W., 1996b, The Hilbert Class Library: a library of abstract C++ classes for optimization and inversion: Fourth Object Oriented Numerics Conference: OON-SCI'96 Proceedings.

Polak, E., 1971, *Computational Methods in Optimization.* New York: Academic Press.

Rogers S. J., Fang, J. H., Karr, C. L., Donald, D., and Stanley, F. A., 1992, Determination of lithology from well logs using a neural network: *AAPG Bulletin*, 76:731-739.

Specht, D. F., 1990, Probabilistic Neural Networks: *Neural Networks*, 3:109-118.

Specht D. F., and Shapiro, P. D., 1991, Generalization accuracy of probabilistic neural networks compared with back-propagation networks: *Proc. IEEE International Conference on Neural Networks*, I:525-532.

Stolt R. H., and Weglein, A. B., 1985, Migration and Inversion of Seismic Data: *Geophysics*, 50:2458-2472.

Taner M. T., Koehler, F., and Sheriff, R. E., 1979, Complex seismic trace analysis: *Geophysics*, 44:1041-1063.

Wasserman P. D., 1993, *Advanced methods in neural computing.* New York: Van Nostrand Reinhold.