geoconvention
Virtual Event
September 13 - 15
2021

Convolutional Neural Networks for Density Log Prediction

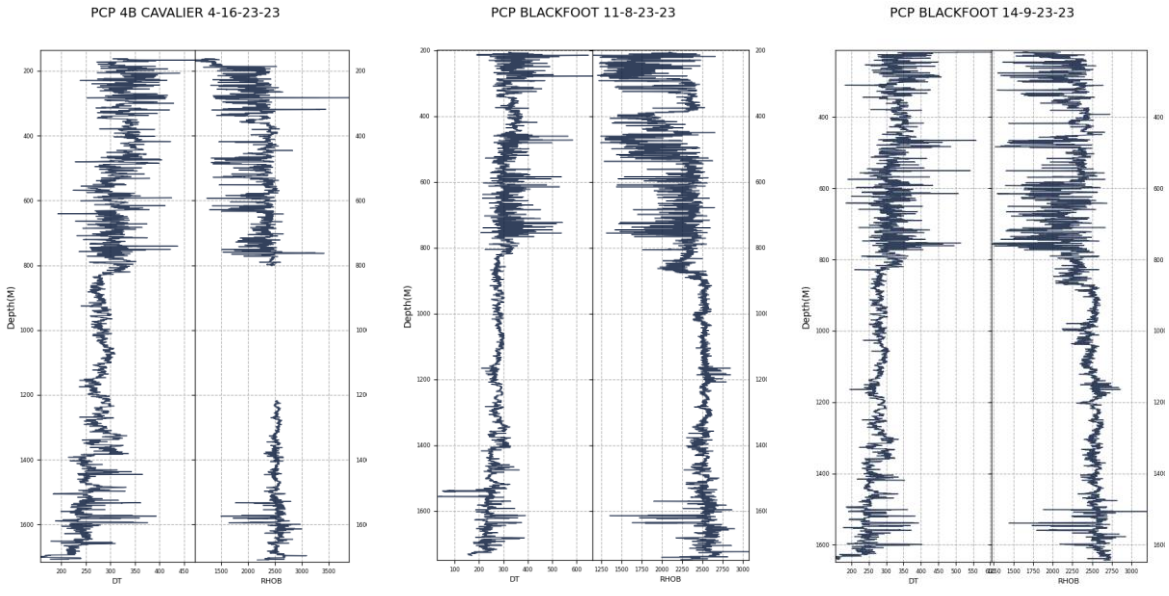
Alex Falkovskiy
RM Seismic Software

Summary

This work investigates the possible application of Convolutional Neural Networks (CNN) for prediction of missing logs that may be advantageous compared to the other methods. In this research a CNN with added attributes/features including depth was used to predict missing density log from known sonic log. Optimal CNN dimensions were selected using an objective function.

Motivation

One example of why prediction of missing logs is necessary comes from seismic modeling. When a synthetic seismic trace is computed from density and sonic logs, it is a common situation that one of these logs in missing. In most cases the missing log is a density log or some part of it. Figure 1 shows an example of such situation. It is possible to replace missing density values with some constant value (like $\rho = 1$) when computing impedance as $Z = \rho V$, but it will affect a synthetic trace. A better solution would be to be able to restore missing density log using available log data.



a. Missing density log, well 4-16

b. Training well 11-08

c. Validation well 14-09

Figure 1. An example of well logs from Blackfoot dataset including with missing density log.

Theory

Machine learning approach is becoming popular for solving geophysical problems over recent years. Some examples of successful applications are presented in Russell (2020), Mardani (2020), Rops and Lines (2017). There are numerous publications on using neural networks for well log prediction like Pham et all. (2019), which showed good results. This research is focused on prediction of density log from sonic log.

Since missing log values would depend on the values of known logs only in vicinity from the point where we need to predict the unknown value, it seems that convolutional neural network will be suitable for such predictions because it works with short windows. However, in addition to the convolution window, we introduced a depth parameter to the neural network to take into account possible local changes in dependencies.

Figure 2 shows the architecture of the convolutional neural network. The following equation shows a 3-point example, but in the calculations different window sizes were used

$$\rho_i = \theta_0 + \theta_1 d_i + \theta_2 d_i^2 + \theta_3 V_{i-1} + \theta_4 V_{i-1}^2 + \theta_5 V_i + \theta_6 V_i^2 + \theta_7 V_{i+1} + \theta_8 V_{i+1}^2 \quad (1)$$

where ρ_i is density at the i-th point, V_i is velocity , d_i is the depth of the i-th point, and θ_0- θ_8 are computed weights.

It can be rewritten in a matrix form as

$$P = A * \Theta \quad (2)$$

Equation (2) was solved using normal equation: $\Theta = (A^T A)^{-1} A^T \mathrm{P}$ to find a vector of weights Θ in (1).

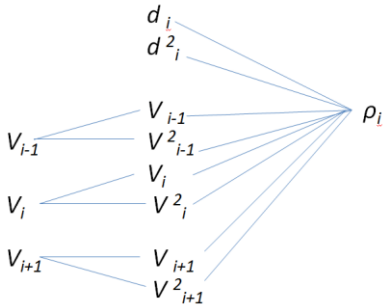


Figure 2. Used CNN architecture example.

Results

The following figures show comparison results of Gardner equation, linear interpolation and CNN that were uses for prediction of density log.

For prediction of density, we used standard Gardner et al. (1974) equation

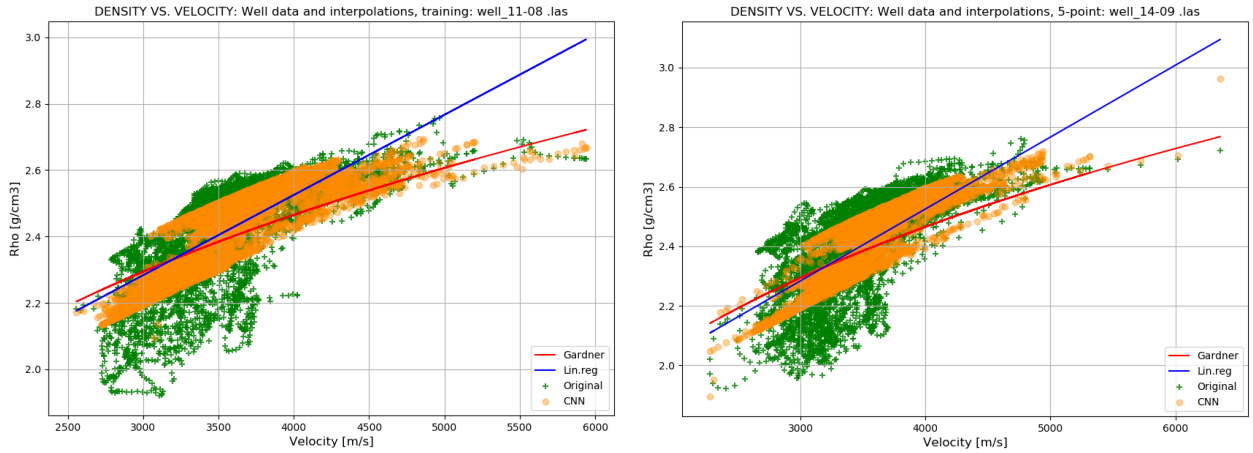
$$\rho = a\,V^{m} \tag{3}$$

where ρ is density in g/cm^3, and V is P-wave velocity in m/s, with default values for a is 0.31, and for m is 0.25 as used in Quijada and Stewart (2007).

A simple best fit linear regression $\rho = aV + b$ was also used for prediction for comparison.

These results were compared with prediction by CNN using 5-point window with added quadratic and cubic depth features.

Figure 3 is a cross-plot of density versus velocity for comparison of actual measurements with density predicted by Gardner relationship, linear regression and CNN.



a. Training well

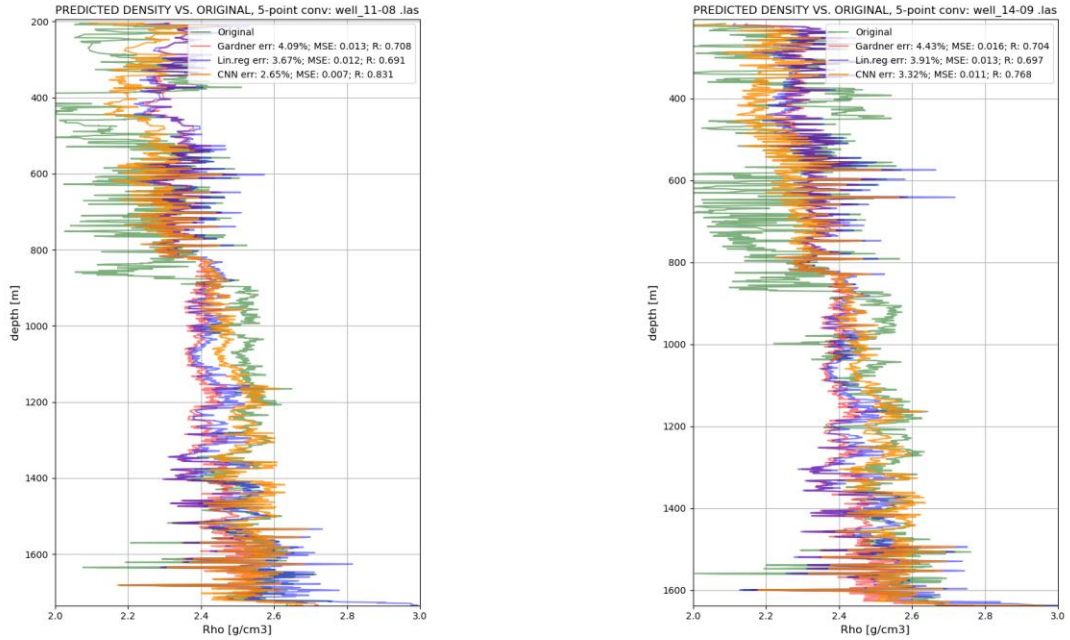
b. Validation well

Figure 3. Actual density vs. velocity (green) with interpolations: Predicted Gardner (red), Linear Regression (blue) and 5-point CNN (orange dots)

It is obvious that no trend line can perfectly fit the sparseness of green marks since there are several possible values of density for each velocity value. That is because density depends not only on velocity but on some other factors that are not presented in these plots. To take into account the other factors, we decided to add depth and its derived attributes to CNN. It is not a perfect solution, but at least allows corrections for different depths in calculation of convolution where the same values of parameter Θ are used for convolution windows. It may be considered as an equivalent to using different a and m in Gardner equation for different rock types as in Quijada and Stewart (2007).

As a result, the orange dots of CNN prediction seem feet better to measured data (shown green).

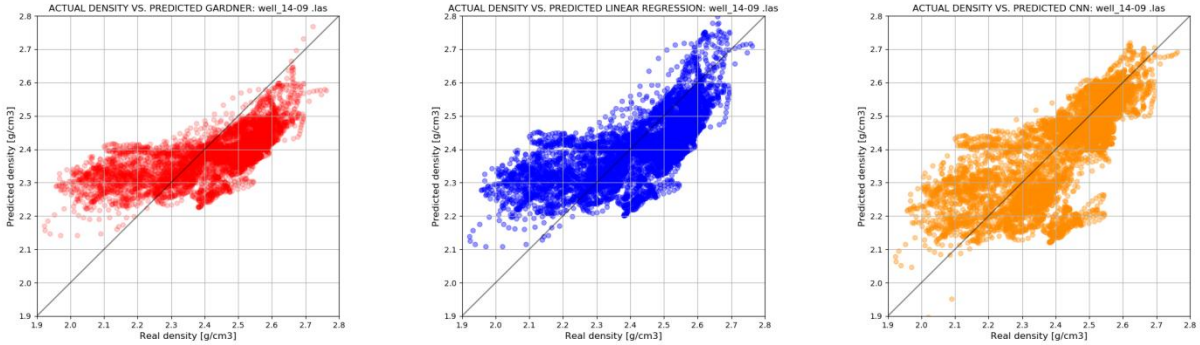
Figure 4 shows original and predicted density logs on training and validation wells calculated with Gardner equation, linear regression and CNN. In both cases, CNN prediction errors were less than errors of Gardner linear regression. On a training well, CNN mean prediction error reduced to 2.65% compared to Gardner 4.08% and linear regression error of 3.67%. Accordingly, MSE reduced to 0.007 from 0.013 and 0.012. Correlation coefficient increased from 0.71 for Gardner equation and 0.69 for linear regression to 0.93 for CNN.



a. Training well b. Validation well

Figure 4. Predicted vs. original density calculated with Gardner equation, linear regression and CNN.

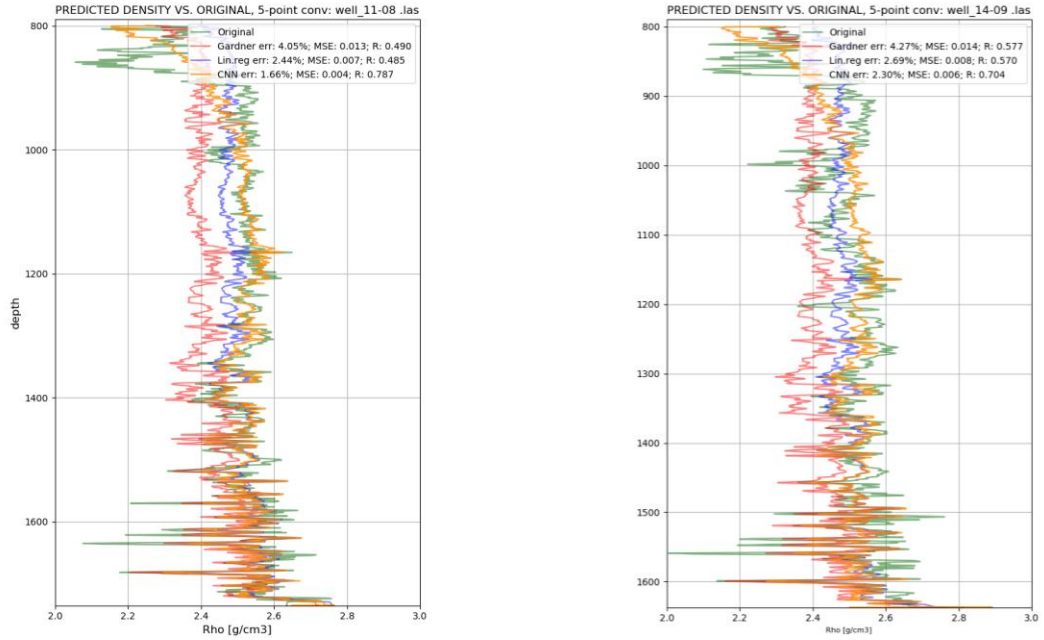
Figure 5 shows predicted versus actual density that is an alternative way to assess the results. A perfect prediction would be a straight line with a slope of 1. Neither of these predictions is perfect, but it is visible that CNN prediction lines up better along the diagonal line of perfect fit.



a. Gardner equation b. Linear regression c. CNN

Figure 5. Predicted density versus actual calculated by three different methods.

For some reason, all these well logs have one common feature – in a shallow part (depth range from 0 to 800 m) there is a high variation of logs amplitude, which becomes much lower at depth deeper 800m. It is not clear if it is the result of higher measurement errors (caliper logs also show high variations in this zone), or physical properties, but the prediction results are worse in this shallow zone for all three methods. However, if we try to predict porosity for depth more than 800m meters, the results become much better. Figure 4 shows density prediction results for this deeper zone. In this case the CNN mean prediction error is even better – it becomes 1.7% for the training well and 2.3% for the validation well.



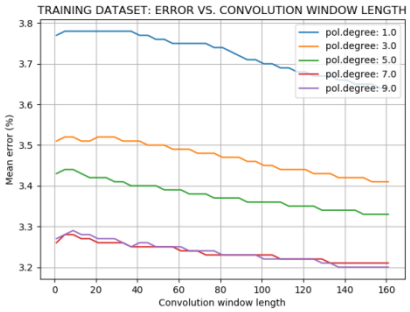
a. Training well b. Validation well

Figure 4. Deeper 800m zone. Predicted vs. original density calculated with Gardner equation, linear regression and CNN.

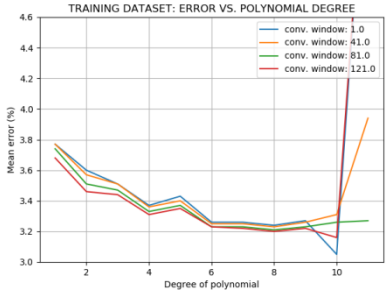
Finding optimal network size

Neural Network architecture selected for this log prediction depends on two parameters: a length of convolution window and a degree of polynomial as a function of depth. It is not clear if the parameters we used so far are the best. Using mean error as a function of these two parameters, it is possible to find their optimal values for prediction of density log.

In this case three wells 11-08, 01-08, 13-16 were used to create a training dataset, and well 14-09 used for validation. Figure 4 shows how mean error depends on a length of convolution window and a degree of polynomial for a training dataset. The error slowly decreases as the convolution window and decree of polynomial increase. All error curves are getting flatter for larger parameter values, but for polynomials at the edge of the function domain there is a high error raise for high degree polynomials probably caused by overfitting.



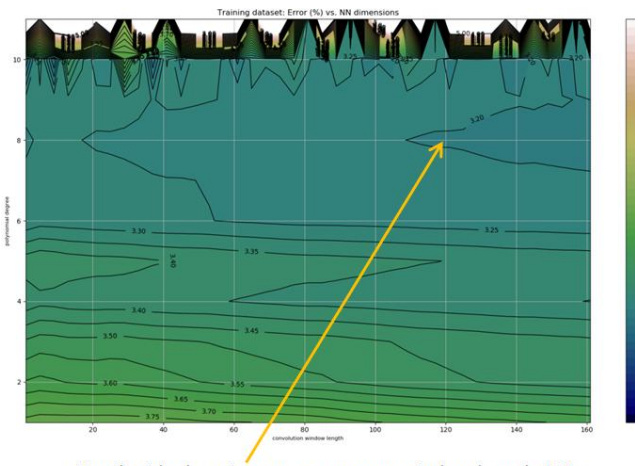
a. Mean error (%) vs. convolution window length



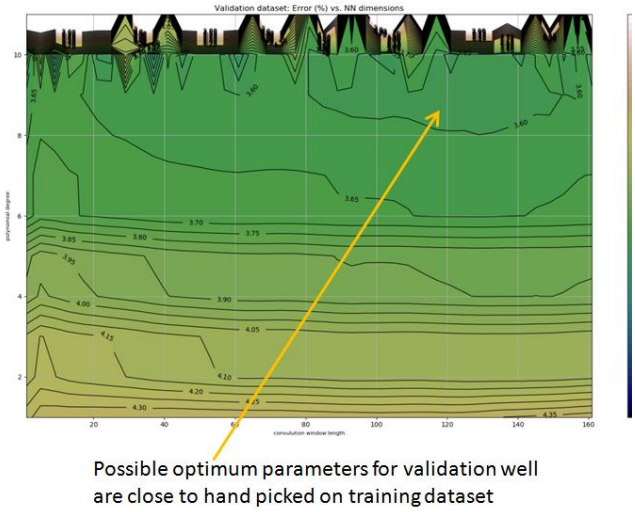
b. Mean error (%) vs. degree of polynomial

Figure 5. Training dataset: Picking optimal convolution window length and polynomial degree.

Figure 6 shows the same mean error functions as 2-d maps for a wider domain of parameter values. It was used for picking optimum parameters. Optimal parameters values picked on training dataset were close to that for a validation well.



a. Training dataset: horizontal axis - convolution window, vertical – polynomial degree



b. Validation well

Figure 6. Finding optimum NN parameters: Hand picking convolution window length and degree of polynomial.

Being obvious in hand picking, implementation of an automatic selection of these optimum parameters is a challenging problem. We cannot use the mean error as a function of window size and polynomial degree as an objective function due to the following two reasons:

1. Global minimum of such function does not necessary match to the best NN parameters due to instability at the edges of the function domain for higher degree polynomials (more than 9 in this case) and may not give good results on validation well.

2. Starting from some higher parameter values, the mean error decreases very slow. In a reality we do not need to find a global minimum, we need to find some minimum of a given accuracy (e.g. 0.01 %) but which also minimizes the size of neural network. We do not want our windows or polynomial to be too big.

To address the instability of the mean error function, a running average filter was applied.
To address the trade off problem of finding a minimum for a given accuracy error and minimum neural network dimensions, the following objective function was selected:

$$F(i,j) = round(Err(i,j), 2) + 0.001 * \frac{i+j}{\max(i) + \max(j)} \quad (4)$$

where i,j are the neural network dimension parameters (like a convolution window length), $round(x,2)$ is a Python rounding function. Since the second term is much less than the first one, this objective function first gives a priority to finding i,j to minimize the error with a given accuracy, and then it is looking for the smallest i,j values among those providing this minimum.

Figure 7 shows optimum parameters that were selected using this objective function: a convolution window length of 145 and a degree of polynomial 8. Note that this point does not lie on the edge of the function domain.

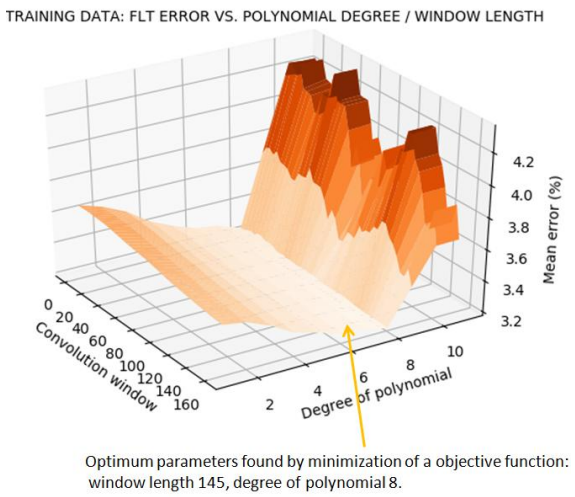


Figure 7. Finding minimum of objective function.

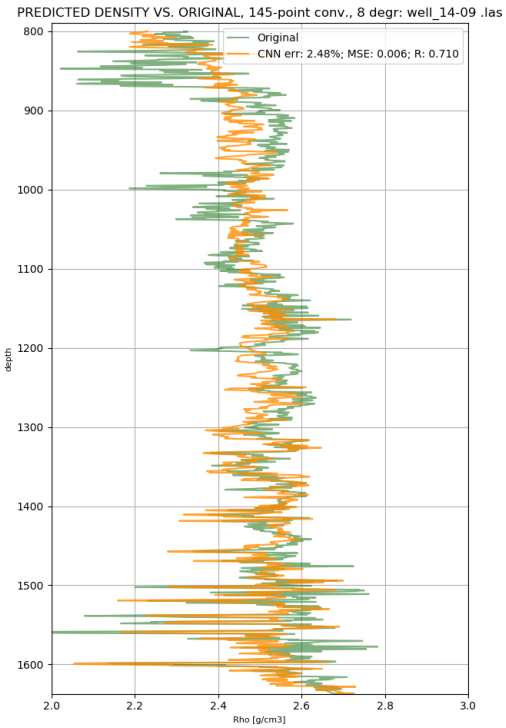


Figure 8. Predicted density vs. original using 145 point convolution window and 8 degree polynomial.

Figure 8 shows predicted density log calculated with 145 point convolution window and 8 degree polynomial NN compared to original density log. Density and sonic logs on Blackfoot dataset have smoother logs deeper 800m were application of CNN provided better results.

Conclusions

Using CNN instead of Gardner equation may improve mean error from 4-5% to 1.5-3.5%.

For some reason, density and sonic logs on Blackfoot dataset have high dispersion in shallow zone, and smoother logs deeper 800m. Application of CNN on logs deeper 800m provided better results.

All predictions work better on smooth data (filtered, removed outliers). Apparently that is because that it is possible to predict trend, but not random noise.

It is possible to automate picking optimal NN parameter using an objective function.

Acknowledgements

Well logs used in this research are from Blackfoot dataset courtesy of CREWES.

References

Gardner, G., L. Gardner, and A. Gregory, 1974. Formation velocity and density—The diagnostic basics for stratigraphic traps: Geophysics, 39,770–780.

Pham, N., Wu, X. and Naeini, E.,Z. 2019. Missing well log prediction using convolutional long short-term memory network. GEOPHYSICS, VOL. 85, NO. 4 (JULY-AUGUST 2020); P.WA159–WA171.

Quijada, M.F. and Stewart, R.R. 2007. Density estimations using density-velocity relations and seismic inversion. CREWES Research Report — Volume 19 (2007).

Mardani, R.A. 2020. Estimation of Acoustic Impedance from Seismic Data in Well-log Resolution Using Machine Learning, Neural Network, and Comparison with Band-limited Seismic Inversion. 2020 CSEG GeoConvention.

Margrave, G.F., Lawton, D.C., Stewart, R.R., Miller, S., Yang, G., Simin, V., Potter, C., Zhang, Q., and Todorov, T. 1997. The Blackfoot 3C-3D seismic survey: A case study. CREWES Research Report — Volume 9 (1997).

Rops, E.A. and Lines, L.R. Lines, 2017. Predicting oil sands viscosity from well logs, NMR logs, and calculated seismic properties. 2017 CSEG GeoConvention.

Russell, B.H., 2020. Demystifying Machine Learning. 2020 CSEG GeoConvention.