Modeling Sonic Logs in Oil Wells: A Comparison of Neural Networks Ensembles and Kernel Methods

P.M. Granitto, H.D. Navone, P.F. Verdes & H.A. Ceccatto
Instituto de Física Rosario (CONICET-UNR)
Bv. 27 de Febrero 210 Bis, 2000 Rosario, Argentina.

ABSTRACT: Oil well logs are frequently used to determine the mineralogy and physical properties of potential reservoir rocks, and the nature of the fluids they contain. Recently we reported an exploratory use of neural network ensembles for modeling these records. We showed that ensembles are clearly superior to linear multivariate regression as modeling technique, revealing an underlying nonlinear functional dependency between the correlated variables. In this work we use kernel methods to develop nonlinear local models relating Sonic logs (transit time of compressional waves) with other commonly measured properties (Resistivity and Natural Formation Radioactivity Level or Gamma Ray log). The kernel considered is conceptually simple and numerically robust, and allows to obtain the same performance as neural networks ensembles on this task.

KEYWORDS: Neural Networks, Ensemble Methods, Kernel Methods, Petroleum Industry

E-MAIL: granitto@ifir.edu.ar
navone@ifir.edu.ar
verdes@ifir.edu.ar
ceccatto@ifir.edu.ar

1. INTRODUCTION

Well logs are records related to properties of the well bore or the formations surrounding it, measured using special equipment. These logs are used to identify and correlate underground rocks, and to determine the mineralogy and physical properties of potential reservoir rocks and the nature of the fluids they contain. A well log is recorded during a survey operation in which a sonde is lowered into the well bore. The measurements made by the downhole instrument are of a physical nature (*i.e.*, electrical, acoustical, nuclear, thermal, dimensional, etc.), pertaining to some part of the well bore environment or the well bore itself. After a well has been investigated and completation is started it has to be cased with steel pipes, which prevents from new log operations. In spite of this, the "open hole" well-logging operations are usually not performed with all the measurement instruments available due to budget limitations.

Formation and fluid properties measured at reservoirs can be divided into four classes:

1. *Resistivity*. Measured indirectly using electromagnetic induction, which works well provided the formations are not too resistive (*i.e.*, mud not too saline).
2. *Natural Formation Radioactivity Level*. In sediments this log mainly reflects clay content because clay contains the radioisotopes of potassium, uranium, and thorium. Potassium feldspars, volcanic ash, granite wash, and some salt deposits containing potassium (potash, for example) may also give significant gamma-ray readings. Limestone, sandstone and other reservoir rocks usually have low readings.
3. *Radioactive Bombardment Response*. Density log (measured indirectly using a gamma-ray source) and Neutron log (which measures the slowdown of high-energy neutrons colliding with atomic nuclei from the rocks) are the most frequent.
4. *Sonic Conductivity*. Measured using the interval transit time (usually called Sonic), which is the travel time of a compressional wave over a given distance (proportional to the reciprocal of the compression wave velocity).

The last two properties are indirect measures of porosity, which is related to the organic content of the reservoir. That is why a porosity indicator is always included in any log well operation, although more than one related log is hardly measured (Density is the quantity most usually determined). The less-frequently-recorded Sonic log is of utmost utility because, in addition to the porosity levels, it relates well logs to seismic information (bi or tridimensional). It is then of great importance to estimate Sonic logs from other more usual records.

Empirical relations among Sonic and other logs can be found in the literature. For example, Gardner[1] developed a relation between Sonic and Density, and Faust[2] obtained a nonlinear approximation between Sonic, Depth and Resistivity. These methods attempt to find models of general validity, not dependent on the local properties of the well. However, the rough approximations involved usually lead to linear correlation coefficients between measured and predicted Sonic values as low as 0.2.

Recently[17] we explored the possibility of developing nonlinear local models relating Sonic logs with other commonly measured quantities, such as Resistivity and Natural Formation Radioactivity Level (Gamma Ray log). We used data from six wells belonging to a common basin, thus sharing geological properties. We later included the depth of each register as a new input variable. This inclusion improved performance by incorporating dependencies on other variables not directly measured, although probably limiting the spatial validity of the developed models. In particular, to model the Sonic log we used a recently developed algorithm for building neural networks ensembles[3]. Here we approach the same problem using the Kernel-Adaline algorithm[11]. Both methods have shown excellent performance when tested on other real and artificial regression problems. We also develop linear regression models to be used as a basis for comparison.

The work is organized as follows: In Section 2 we describe the available data and the applied preprocessing. In Section 3 we outline the ensemble method used in [17]. In Section 4 we describe the Kernel-Adaline. Then, in Section 5 we show the results obtained and compare both nonlinear methods against the linear model. Finally, in Section 6 we draw some conclusions and discuss future lines of work.

2. DATA DESCRIPTION AND PREPROCESSING

We work with data from six wells belonging to a common basin located at the mid-west of Argentina. Available logs are Resistivity, Gamma Ray and Sonic. We take the logarithm of the resistivity, instead of the raw variable, because this enhances resolution of the readings in the low-resistivity range (as is a common practice in Geophysics). Measures were recorded from the top of the wells to some variable final depth, always near 2000 meters, at intervals of approximately 0.15 meters. All wells have certain zones where records are known to be wrong, due to mud properties that alter readings, or to saturation of the measuring instrument (for example, as mentioned above, mud too saline). Once these wrong records have been eliminated, we perform a smoothing of the remaining data using a sliding window of length 3 in order to filter high frequency measurement noise. As a final step, we rescale all input variables to a uniform range. Figure 1 shows the final available data for one representative well[17].

3. NEURAL NETWORK ENSEMBLES

Ensemble techniques have been used recently in regression/classification tasks with considerable success. They are theoretically motivated on the bias/variance decomposition of the generalization error[4]. This procedure is based on the intuitive idea that by combining the outputs of several individual predictors one might improve on the performance of a single generic one[5]. However, this idea has been proved to be true only when the combined predictors are simultaneously accurate and diverse enough, which requires an adequate trade-off between these two conflicting conditions. Some attempts[6] to achieve a good compromise between these properties include elaboration of *bagging*[7], *boosting*[8] and *stacking*[9] techniques. Feedforward artificial neural networks (ANNs) provide a natural framework for ensemble techniques. This is so because ANN is a very unstable learning algorithm, *i.e.*, small changes in training set and/or parameter selection can produce large changes in prediction outputs. This diversity of ANNs comes naturally from the inherent data and training process randomness, and also from the intrinsic non-identifiability of the model (many different but *a priori* equally good local minima of the error surface). On the other hand, the above mentioned trade-off between the ensemble diversity and the accuracy of the individual members posses the problem of generating a set of ANNs with both reasonably good (individual) generalization capabilities and distributed predictions for the test points. This problem has been considered in several recent works in the literature[6,10].

In [17] we used an alternative way of generating an ANN ensemble, proposed in [3], which leads to ensemble members that are both accurate and diverse. The method essentially amounts to the sequential aggregation of individual predictors where, unlike in standard aggregation techniques which combine individually optimized ANNs[10], the learning process of a (potential) new member is validated by the *overall* aggregate prediction performance. That is, an ANN is incorporated to the ensemble only when this improves the generalization capabilities of the previous-stage aggregate predictor. This is accomplished by the following procedure:

> **Step 1:** Split the data set D at random in a training set T_1 and a validation set V_1; generate a model f_1 by training a network on T_1 until a minimum e_1 in the generalization error on V_1 is reached.
>
> **Step 2:** Split the data set D at random in new training and validation sets T_2 and V_2 respectively. Produce a model f_2 by training a network until the generalization error on V_2 of the *aggregate* predictor $\Phi_2 = ½ (f_1+f_2)$ reaches a minimum $e_{\Phi_2}(V_2) < e_1$. In this step the parameters in model f_1 remain constant and the model f_2 is trained with the usual (quadratic) cost function on T_2.
>
> **Step 3:** If the previous step cannot be accomplished after a maximum N_E of training epochs, disregard the current training and validation sets and start again with step 2 using new random sets T_2 and V_2. Repeat this step until either a second model f_2 which satisfies the required conditions is found or the number of attempts to find it reaches a maximum N_A. In this last case the algorithm terminates.
>
> **Step 4:** If a model f_2 is found, incorporate it to the ensemble and proceed again with steps 2 and 3 seeking for a model f_3 such that $e_{\Phi_3}(V_3)$, the minimum generalization error on V_3 of the aggregate $\Phi_3=(f_1+f_2+f_3)/3$, becomes smaller than $e_{\Phi_2}(V_2)$.
>
> **Step 5:** Iterate the process until N_A unsuccessful attempts to find a new model are performed. The individual networks collected up to this point constitute the final ensemble.

Notice that the algorithm incorporates a new member to the ensemble only when it helps in improving the *aggregate* generalization performance on a particular validation set; in practice it was found useful to check that the performance on the *whole* data set D also improves or otherwise to stop the algorithm. This helps in reducing the computational time since for large values of N_A the procedure will first terminate by this second condition. On the other hand, this condition seems to lead to final ensembles with nicer generalization capabilities on the test set.

4. THE KERNEL-ADALINE ALGORITHM

The use of potential functions[13] allows the generalization of linear-nature learning algorithms to non-linear problems. Such functions, called ‘kernels’, map the original data to a high dimensional space (called linearisation space), in which it is possible to apply linear techniques (either regression or classification). The idea has been exploited in the potential function algorithm[13] and also in the

Support Vector Machine[14]. We use in this work the kernel-based generalized version of the Widrow and Hoff's Adaline algorithm[11,12].

The Adaline is given by

$$f(\vec{x}) = \vec{w} \cdot \vec{x} + b \qquad \vec{w}, \vec{x} \in \Re^d, b \in \Re$$

where $\vec{x}$ is a vector of input data, $\vec{w}$ is a set of *d* weights and *b* is a bias term. The Adaline algorithm is a member of a general class known as Perceptrons. Its objective is to adjust the weight vector $\vec{w}$, so that the mean squared error between the model output $f(\vec{x}_i)$ and the target value t_i is minimized on the training set $\{\vec{x}_i, t_i\}$, $i = 1...L$. The gradient descent solution produces the following adaptation rule:

$$\vec{w} \leftarrow \vec{w} + \eta\,(t - f(\vec{x}))\,\vec{x}$$
$$b \leftarrow b + \eta\,(t - f(\vec{x}))$$

where η is a positive constant.

The weights $\vec{w}$ can also be represented as an expansion on the data samples[12],

$$\vec{w} = \sum_{i=1}^{L} \alpha_i\, \vec{x}_i$$

allowing the re-writing of the Adaline in its data dependent form[15,16]:

$$f(\vec{x}_j) = \sum_{i=1}^{L} \alpha_i\, (\vec{x}_i \cdot \vec{x}_j) + b\,.$$

Now, the (equivalent) update rule for the scalar multipliers α and the bias term *b* is given by

$$\alpha \leftarrow \alpha + \eta\,(t - f(\vec{x}))$$
$$b \leftarrow b + \eta\,(t - f(\vec{x}))$$

The data dependent form of the Adaline allows us to introduce the non-linear extension of the model. If we find a fixed mapping of the input data $\vec{z}_j = \varphi(\vec{x}_j)$ which is rich enough to capture the underlying functional form of the signal we want to learn, then the data could be transformed and fitted by a linear combination of the *z*'s. Through the use of Mercer kernels it is possible to perform the mappings implicitly because they represent inner products in some Hilbert space[13]. Therefore, the data dependent non-linear Adaline (K-Adaline) is given by

$$f(\vec{x}_j) = \sum_{i=1}^{L} \alpha_i\, K(\vec{x}_i, \vec{x}_j) + b\,,$$

where

$$K(\vec{x}_i,\vec{x}_j) = \varphi(\vec{x}_i)\cdot\varphi(\vec{x}_j)$$

is the inner product defined in a high dimensional linearisation space. Any potential function satisfying Mercer's conditions may be used. In this work we use radial basis functions, given by

$$K(\vec{x}_i,\vec{x}_j) = \exp\left(\frac{-\left|\vec{x}_i-\vec{x}_j\right|^2}{2\sigma^2}\right),$$

where σ is an associated parameter providing a degree of freedom.

The pseudo-code of the Kernel-Adaline algorithm is given below:

```
Step 1: Split the data set $D$ in a training set $T$ and a validation set $V$
Step 2: Assign a initial value to the internal parameter $\sigma$
Step 3: Choose $\alpha_j = 0 \;\; j = 1\ldots L, b = 0$ and $\eta$
Step 4: WHILE (stopping criterion not met)

            FOR $j = 1...L$
               Calculate output of the K-Adaline: $f(\vec{x}_j) = \sum_{i=1}^{L}\alpha_i K(\vec{x}_i,\vec{x}_j) + b$
               Update corresponding multipliers and bias by the rule:

                       $\alpha \leftarrow \alpha + \eta\,(t - f(\vec{x}))$
                       $b \leftarrow b + \eta\,(t - f(\vec{x}))$
            END FOR

          END WHILE

Step 5: Compute MSE on the Validation Set
Step 6: Assign a new value to $\sigma$ (for example, from a given list) and GOTO 3
```

We keep the value of the internal parameter σ that minimizes the error on the validation set.

5. RESULTS

Like in [17], we first developed models for all six wells using only (the logarithm of) Resistivity and Gamma Ray as input regression variables. For each well we fit a Kernel-Adaline model by randomly splitting the corrresponding records in two datasets: one of them, containing 67% of the data, is used to fit the model and the remaining 33% is used to test its performance. We considered a maximum of 30 iterations as the stopping criterion, and the parameter η was set to 0.1. Within these restrictions, we found $\sigma = 0.07$ to be the optimum value.

For the sake of completeness we remind the reader that in [17] we considered feedforward ANNs with a single hidden layer as the ensemble members. After some preliminary exploration, we set the ANN learning parameters and architecture as follows: a learning rate of 0.1, a momentum of 0.9, and 10 units in the hidden layer. The parameters for building the ensemble were set to N_A=10 attempts and N_E=10000 epochs.

To compare the results we computed the Normalized Mean Square Error (*NMSE*), defined as the *MSE* divided the data variance, and also the linear Correlation Coefficient (*CC*) between predicted and observed values. Results for the test sets are shown in Table 1. In all 6 cases, both ANN ensembles and K-Adaline perform better than multivariate linear regression in prediction error (*NMSE*) as well as in the linear correlation (*CC*). These results suggest the presence of nonlinear dependencies in the data, which can be modeled by the non-linear methods. When comparing the two non-linear methods no statistically significant differences can be found, although K-Adaline shows a (very small) improvement over the ANN ensembles.

We also developed new models including the depth at which each register has been taken as a third input variable. We performed the same splitting of the data in training and test sets, and used the same values for all parameters involved to allow a direct comparison between models ($\sigma = 0.04$ was the optimum value in this case). The results obtained for the test sets are shown in Table 2. Again, in all 6 cases both non-linear methods are better than linear regression. Furthermore, when comparing the 2- and 3-variables settings it becomes apparent that, irrespective of the method used, the inclusion of Depth as an input variable improves performance. This is a clear indication that Resistivity and Gamma Ray cannot fully explain Sonic variability by themselves. Like in the 2-variables case, when comparing the non-linear methods we can see that K-Adaline is slightly better than ANN ensembles.

5. CONCLUSIONS

In this work we applied the K-Adaline algorithm to develop local empirical models for Sonic logs in oil wells, using different input variables as regressors. We compared the results of linear multivariate regressions and ANN ensembles (obtained in [17]) with those of K-Adaline models for the prediction of Sonic logs from two commonly measured properties, namely Resistivity and Gamma Ray. The non-linear methods proved to be clearly superior, showing an underlying nonlinear functional dependency between these variables. This result remained true after the inclusion of Depth as a new input variable. In this case both linear and non-linear methods improved their performances, which is a clear indication that Resistivity and Gamma Ray cannot fully explain Sonic variability by themselves.

We stress that, in spite of its simplicity, the K-Adaline algorithm showed a small improvement in performance over ANN ensembles in both the 2- and 3-regression variables problems. This is very encouraging and in future works we plan to use other potential functions that might improve the algorithm performance. Furthermore, we will explore the spatial extent of the generalization capabilities of the developed models. In particular, we would like to establish whether the inclusion of Depth as an input regression variable severely limits the applicability of the correspopnding models to other wells, as suggested in the Introduction.

REFERENCES

[1] G.H.F. Gardner and M.H. Harris, "Velocity and attenuation of elastic waves in sands", Society of Professional Well Log Analysts, *9th Annual Logging Symposium Transactions,* 1-19 (1968).

[2] L.Y. Faust, , "A velocity function including lithologic validation", *Geophysics* **18**, 271-297 (1953).

[3] H.D. Navone, P.F. Verdes, P.M. Granitto and H.A. Ceccatto, "A Learning Algorithm for Neural Networks Ensembles", in G. P. Henning, ed., *ASAI'2000: Proceedings of the Argentine Symposium on Artificial Intelligence*, 199-207 (2000).

[4] S. Geman, E. Bienenstock and R. Doursat, "Neural Networks and the Bias/Variance Dilemma", *Neural Computation* **4**, 1-58 (1992).

[5] Krogh and J. Vedelsby, "Neural network ensembles, cross-validation and active learning", in G. Tesauro, D. Touretzky and T. Lean, eds., *Advances in Neural Information Processing Systems* 7, 231-238 (MIT Press, 1995).

[6] H. Drucker, R. Schapire and P. Simard, "Improving performance in neural networks using a boosting algorithm", in S. J. Hanson, J. D. Cowen and C. L. Giles, eds., Advances in Neural Information Processing Systems 5, 42-49 (Morgan Kaufman, 1993).

[7] L. Breiman, "Bagging predictors", *Machine Learning* **24**, 123-140 (1996).

[8] Y. Freund and R. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting", in *Proceedings of the Second European Conference on Computational Learning Theory*, 23-37 (Springer Verlag, 1995).

[9] D. Wolpert, "Stacked generalization", *Neural Networks* 5, 241-259 (1992).

[10] U. Naftaly, N. Intrator and D. Horn, "Optimal ensemble averaging of neural networks", *Network: Comput. Neural Syst.* **8**, 283-296 (1997).

[11] Widrow, B. & Hoff, M. *Adaptive Switching Circuits*, 96-104 (1960).

[12] Frieb T-T & Harrison, R.F., *Intelligent Data Analysis* **3**, 307-313 (1999).

[13] Aizerman, M., Braverman, E. & Rozonoer, L. *Automations and Remote Control* **25**, 821-837 (1964).

[14] Cortes, C., Vapnik V., *Machine Learning* **20**, 273-297 (1995).

[15] Courant A., Hilbert D., *Methods of Mathematical Physics* (Wiley, New York, 1951).

[16] Frieb, T-T, Harrison R.F., *Research Report No. 728*, The University of Sheffield, Sheffield, (1998).

[17] P.M. Granitto, P.F. Verdes, H.D. Navone, H.A. Ceccatto, D. Curia and C. Larriestra, "Modeling of Sonic Logs in Oil Wells with Neural Network Ensembles", *Argentine Symposium on Artificial Intelligence (ASAI'01)* (Bs. As., September 12-13, 2001).

Oil Well	**Linear Regression**		**ANN Ensemble**		**KA (RBF)**	
	NMSE	*CC*	*NMSE*	*CC*	*NMSE*	*CC*
1	0,62	0,62	0,51	0,70	0,50	0,71
2	0,56	0,66	0,55	0,70	0,49	0,72
3	0,68	0,57	0,37	0,79	0,37	0,80
4	0,43	0,76	0,26	0,86	0,28	0,86
5	0,42	0,77	0,24	0,87	0,25	0,87
6	0,56	0,66	0,25	0,87	0,23	0,88
Average	**0,54**	**0,67**	**0,36**	**0,80**	**0,35**	**0,81**

Table 1: Normalized Mean Square Error (*NMSE*), and linear Correlation Coefficient (*CC*) in the 2- input variables regression setting.

Oil Well	**Linear Regression**		**ANN Ensemble**		**KA (RBF)**	
	NMSE	*CC*	*NMSE*	*CC*	*NMSE*	*CC*
1	0,51	0,70	0,19	0,90	0,20	0,89
2	0,31	0,83	0,17	0,91	0,16	0,92
3	0,41	0,77	0,16	0,92	0,15	0,92
4	0,22	0,88	0,11	0,94	0,09	0,95
5	0,27	0,85	0,14	0,93	0,17	0,93
6	0,52	0,70	0,18	0,91	0,14	0,93
Average	**0,37**	**0,79**	**0,16**	**0,92**	**0,15**	**0,92**

Table 2: Same as Table 1 for the 3-input variables regression setting.

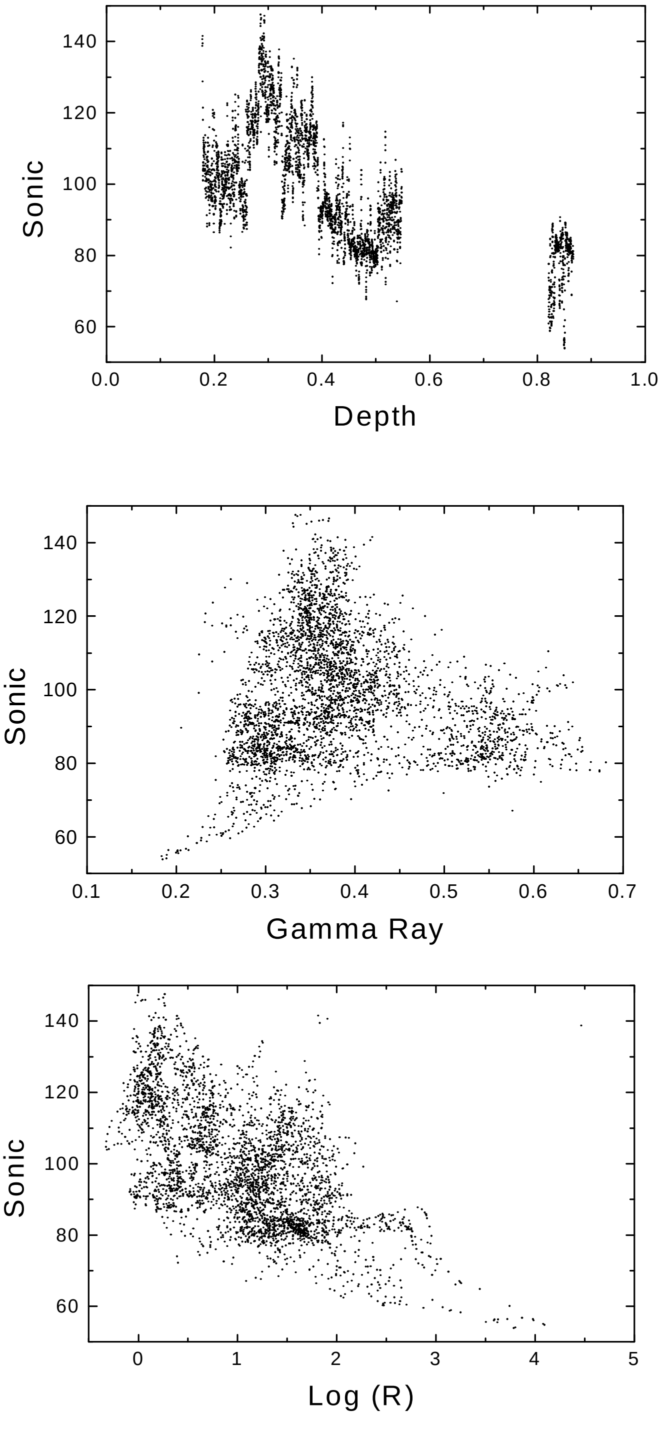

Figure 1: From top to bottom: Sonic dependence with Depth, Gamma Ray and Logarithm of Resistivity, in arbitrary units.