Post-stacking Acoustic Seismic Inversion using Recurrent Neural Network

Caio Graco Pereira Santos (PETROBRAS S/A); Alexandre Gonçalves Evsukoff, Webe João Mansur (COPPE/Universidade Federal do Rio de Janeiro)

Abstract

The inverted seismic data supports the geological model building. The acoustic impedance volume correlates with pretrophysical proprieties. Using data and inverse data relationships, the hard data from wells can be extended for the entire seismic survey area. The Recurrent Neural Network (RNN) was successful at the 1D reflectivity inversion process. The Recurrent Fuzzy System achieved better results, inverting the less frequent synthetic models. The RNN and RFS was applied to synthetic data seismic inversion, obtaining an equivalent result of the band limited impedance inversion.

Introduction

Inversion is not a simple task, even for synthetic data considering a normal incident wave; isotropic, horizontal and parallels layers; and the convolutional model for seismic trace. When dealing with real data, additional difficulties occur concerning, for instance, lost information related to high and low frequencies content, noise presence and wavelet shape modification due to dispersion. Artifical Neural Network is considered as an interpretation assistant, instead of an isolated tool (Latimer et al. (2000) van der Baan and Jutten (2000), Poulton (2002) Santos (2008)).

System identification by ANN requires a data set to adjust the model. In the seismic case, data set comes from field acquisition and processing. Therefore, is necessary to create models to subsidy the inversion. A large amount of seismic data, covering a representative number of geophysical profiles is required in order to establish a robust seismic identification net.

In the convolutional model, considered here, the Earth is represented by horizontal layers with constant properties, in the form of a pseudo 1D model. The wavelet is a continuous function, while the reflectivity (computed from the impedance log) is a series of short duration impulses, located at each reflector.

Recurrent Neural Networks Models

ANN is a computational model composed of simple processing elements, called nodes, connected through weights. These weights are adjusted during training to increase the net performance in the problem solution. The node is a mathematical model of a linear combination of N weights $w_1, \cdots, w_N$ and N inputs $x_1, \cdots, x_N$ and the result is passed through a non-linear function Φ.

The neuron is a operator of kind $\mathbb{R}^{N+1} \rightarrow \mathbb{R}$, where the input vector is given by $\boldsymbol{x} = [1, x_1, \cdots, x_N]^T$, and $\boldsymbol{w} = [w_0, w_1, \cdots, w_N]^T$ is the neuron weight vector, given by the equation:

$$y = \Phi\left(\sum_{i=1}^{N} w_i x_i + w_o\right) = \Phi(\boldsymbol{W}\boldsymbol{X}), \tag{1}$$

RNN architecture is the key to represent dynamic properties of a discrete temporal input signal and its ability to preview a future value Mandic and Chambers (2001). A discrete signal is usually obtained by sampling analogous measures.

Neural networks can use two kinds of topologies: feed-forward and recurrent. At feed-forward topology network is ordered into layers with no feedback paths; the lowest layer is the input layer, the highest is the output layer. Given layer generate outputs only to higher layers and its inputs come from lower layers Mandic and Chambers (2001). In recurrent topologies, the units in a layer can be connected to the units in a preceding layer. The feed-forward topologies can approximate non-linear stationary systems, while recurrent topologies can fit dynamic non-linear systems.

A signal $y(k)$ is predicted based on past values p, i.e., $y(k-1), y(k-2), \ldots, y(k-p)$, weighted by the a_i, $i = 1, 2, \ldots, p$ coefficients to form a prediction $\hat{y}(k)$. The prediction error, $e(k)$, to ANN is:

$$e(k) = y(k) - \hat{y}(k) = y(k) - \sum_{i=1}^{p} a_i y(k-i). \tag{2}$$

The feedback inside the RNN can be obtained from a local or global form. The local feedback is obtained by the introduction of a feedback inside a hidden layer and global feedback is produced by the connection of output with the net input (see figura 1).

Recurrent fuzzy systems

Consider the general state space representation of discrete time, non-linear, multi input multi output dynamic systems:

$$\boldsymbol{x}(t+1) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t)) \tag{3}$$

$$\boldsymbol{y}(t) = \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t)), \tag{4}$$

where $\boldsymbol{x}(t)$ is the state variable vector, $\boldsymbol{u}(t)$ is the input variable vector, $\boldsymbol{f}$ and $\boldsymbol{g}$ are respectively the transition and output funcitons and $\boldsymbol{y}(t)$ is the system output.

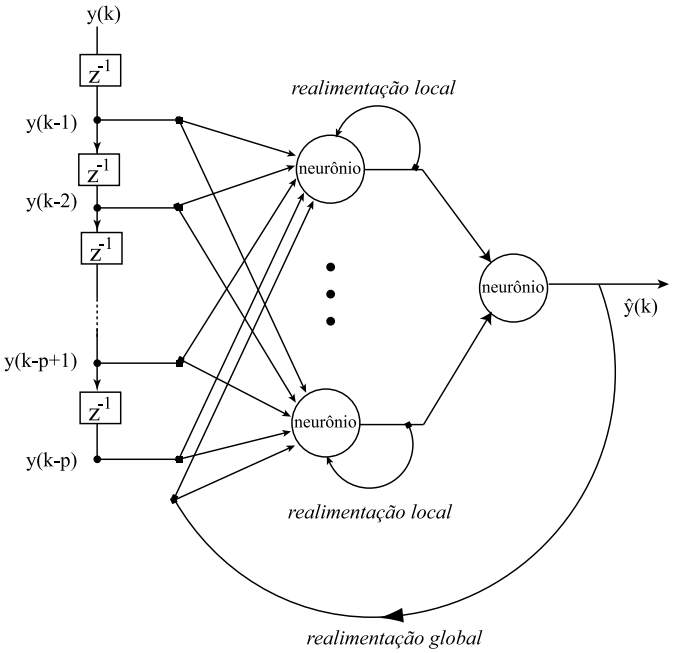


Figure 1: RNN structure with local and global feedback (extracted of Mandic and Chambers (2001)). Although the delays are not explicit in the feedback connections, they are presents in the neurons.

That state space representation can represent several types of dynamic systems according to input ranges and state variables and the choice of transition and output functions. Discrete valued input, state variables and discrete functions, $\boldsymbol{f}$ and $\boldsymbol{g}$ are used to model discrete systems, represented by discrete finite-state automata.

At TSK-type RFS model, the variable in the conclusion of the rules is a state variable, which also appears in the premise, providing recursion. The first order, single input, TSK-type RFS rules are written as:

$$\text{If } x(t) \text{is } A_i \text{ and } u(t) \text{ is } B_j \text{ then } x(t+1) = \theta_r, \qquad (5)$$

where $x(t)$ is the state variable and $u(t)$ is input variable (Gama et al. (2008)) . The fuzzy sets A_i and B_j are defined, respectively, on the state and input variables domain and θ_r is the output parameter for each one of $r = 1 \ldots M$ rules. The RFS model output is the state variable:

$$\boldsymbol{x}(t+1) = \boldsymbol{w}(t)\theta \qquad (6)$$

where θ is the output parameters vector, and $\boldsymbol{w}$ is the vector whose components are the activation values of the rules premises. The product operator is used as t-norm such that the vector $\boldsymbol{w}(t)$ can be computed by the Kronecker tensor product as:

$$\boldsymbol{w}(t) = \mu_{\boldsymbol{A}}(x(t)) \otimes \mu_{\boldsymbol{B}}(u(t)) \qquad (7)$$

Post-stack seismic inversion

The seismic trace inversion process by ANN is made in two stages: (i) reflectivity realization for ANN from seismic data and (ii) integration and trace exponentiation.

The inversion algorithm employed here considers 1-D stacked data. The earth is considered isotropic; thus the model parameters to be used/determined are compressional velocity (V_P), density (ρ_i), and thickness (h_i) of all layers.

The data set consists of a series of convolutionals horizontal layers models. Input data is the seismic trace and the output data (target) is the reflectivity. The RNN approach considers the input and output as non-stationary. The net topology involves the number of neurons determination in the hidden layer, its activation function, the delays and the places of feedback (recurrence).

Synthetic model inversion

To test seismic inversion accuracy using ANN, several synthetic models composed of horizontal layers were created 8 horizontal parallel 200m thick layers. The first layer velocity is $1\,500\text{m/s}$. After the velocity v_l is calculated for the layer l; the velocity for the layer $l+1$ is generated in a pseudo-random way as indicated below:

$$v_{l+1} = (v_l + 190\text{m/s} \pm 380\text{m/s}) \qquad (8)$$

The wavelet for this experiment is a Ricker one with dominant frequency of 8Hz and 300ms. The density is constant, 1.00g/cm^3. Figure 2 illustrates diagrams of acoustic impedance and reflectivity obtained from the velocity profile computed from equation 8 and the resulting seismic trace.

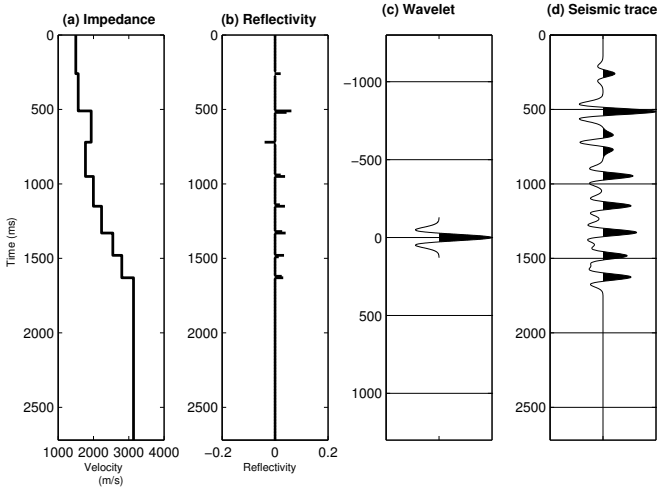


Figure 2: Acoustic impedance, reflectivity, Ricker wavelet of 8Hz and seismic trace for a synthetic model calculated using equation 8.

Recurrent Neural Net topology

The net was trained with the number of neurons increasing from 5 to 40. Figure 3 shows that for a net with nine neurons the mean square error (MSE) of validation is around 0.0065. A net with 15 neurons in the hidden layer revealed unstable, with higher errors.

The net topology that yielded optimum performance (convergence time associated with result quality), was that

of a RNN with one layer of nine neurons, with global recurrence.

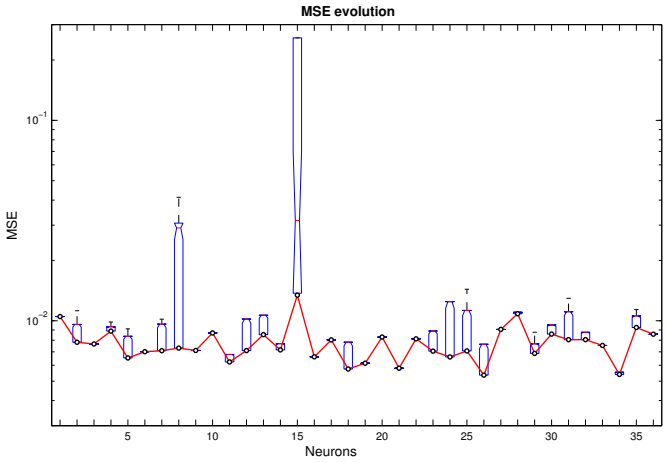


Figure 3: MSE *versus* number of neurons in the hidden layer. The bar size indicates the error amplitude. The continuous line represents the MSE of the net training with 98 examples. A net with nine neurons presents a reasonable performance.

The net validation was made with a data set not included in training set, resulting in fast training and noise-free outputs.

RFS Topology

The best joined structure for RFS inversion uses 4 fuzzy sets for the state variables (reflectivity) and 5 for the input (seismic trace). Those four sets for the reflectivity variable are a consequence of the way the model was created, with the velocity of layer $n+1$, $v_{n+1} = v_n + 190 \pm 380$. The groups correspond to the velocity intervals $[-190, 0]$; $[0, 190]$; $[190, 380]$ and $[380, 530]$ (in fuzzy language: negative, low positive, positive medium and high positive).

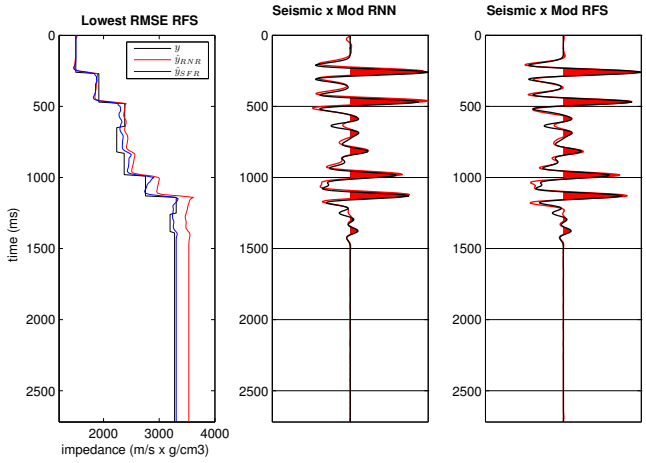
Concerning the seismic signal, one set with five different velocities elements, $\{n-2n, n-1n, n, n+1n, n+2n\}$, is necessary to originate four interfaces; where $n = 190\mathrm{m/s}$.

Comparing RNN and RFS

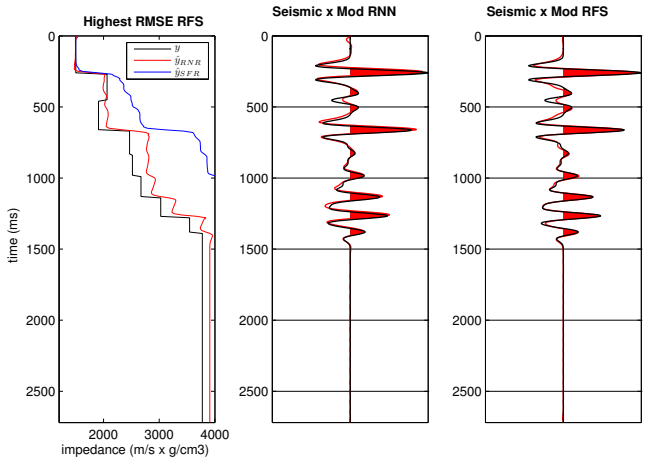
Figures 4 and 5 show RNN and RFS performances, for two cases in each method: minor and greater RMSE. The cases selected corresponds to best and worst case of each net in order to detect ANN application limitations.

Figure 4 shows the models that correspond to highest and lowest RMSE of inversion using RFS. In both cases (highest and lowest RMSE), it is plotted together: (i) syhthetic impedance model, its inverted profile using RFS and its inverted profile employing RNN; (ii) synthetic seismic trace and seismic trace calculated by the forward model using the acoustic impedance profile obtained employing RNN; (iii) synthetic seismic trace and seismic trace calculated by the forward model using the acoustic impedance profile obtained employing RFS. Figure 5 plots follow the same pattern as those presented in figure 4, the difference being that it display results that corresponds to highest and lowest RMSE of inversion using RNN.

Seismic traces shown in figure 5 indicates that the results corresponding to highest RNN, RMSE, indicate this net to diverge. It is important to notice that the divergence results ocurred to a model quite distant from the most frequents ones.



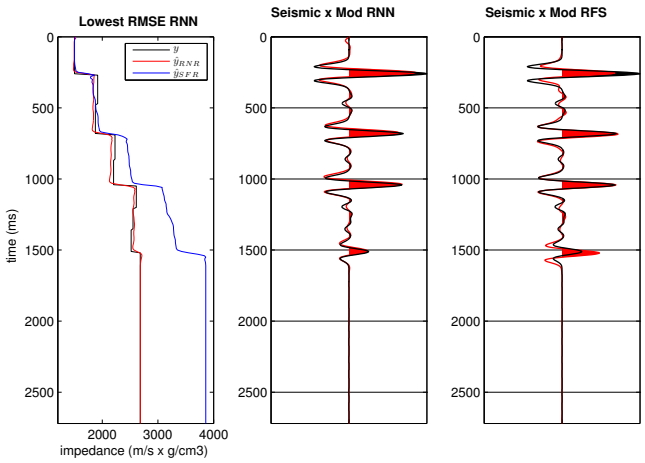
(a) Velocity profile correspondent to lowest RMSE using RFS.



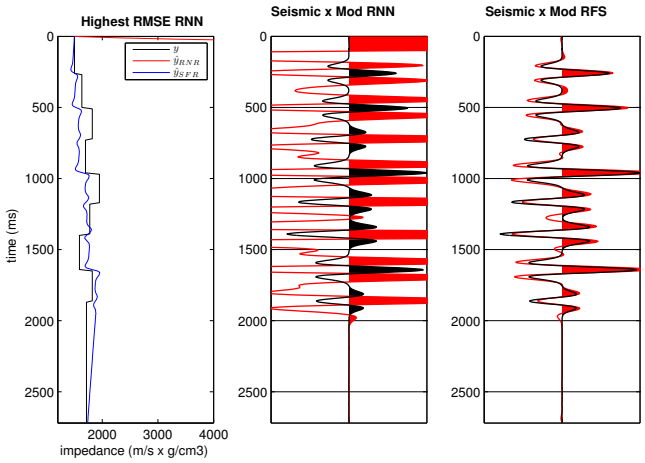
(b) Velocity profile correspondent to highest RMSE using RFS.

Figure 4: Velocity that correspond to highest and lowest RMSE to inversion using RFS. In both case, is plotted together: (i) superposition of the model, the inverted profile using RFS and the inverted profile using RNN; (ii) superposition of the synthetic trace with the model seismic trace calculated by forward model using the acoustic impedance profile obtained using RNN; (iii) superposition of the synthetic trace with the model seismic trace calculated by forward model using the acoustic impedance profile obtained using RFS.

Both nets have presented similar performances in the



(a) Velocity profile correspondent to lowest RMSE using RNN.



(b) Velocity profile correspondent to highest RMSE using RNN.

Figure 5: Velocity that correspond to highest and lowest RMSE to inversion using RNN. In both case, is plotted together: (i) superposition of the model, the inverted profile using RFS and the inverted profile using RNN; (ii) superposition of the synthetic trace with the model seismic trace calculated by forward model using the acoustic impedance profile obtained using RNN; (iii) superposition of the synthetic trace with the model seismic trace calculated by forward model using the acoustic impedance profile obtained using RFS.

majority of cases, but the RFS revealed more efficiency for two reasons: in the worst case (figure 4(b)), the inverted curve keeps the real log trend and the seismic trace calculated by the forward modeling using inverted RFS data as input are very similar to the seismic input whereas none of these two favorable features can be observe in the worst RNN case (see figure 5(b)).

Moreover, in figure 5(b), where the velocities do not increase with the depth, the RFS revealed efficiency in the data recovery, indicating a higher net versatility. Considering the characteristic of synthetic models sets, this log type is less frequent, therefore predominates logs with increasing velocities.

Another RFS advantage is the possibility to interpret the data, relating it with geologic criteria.

Despite the better inverted data, given by RFS, RNN presents lower MSE distribuition; this better RNN performance occurs in the majority of cases as shown in the figure 6.

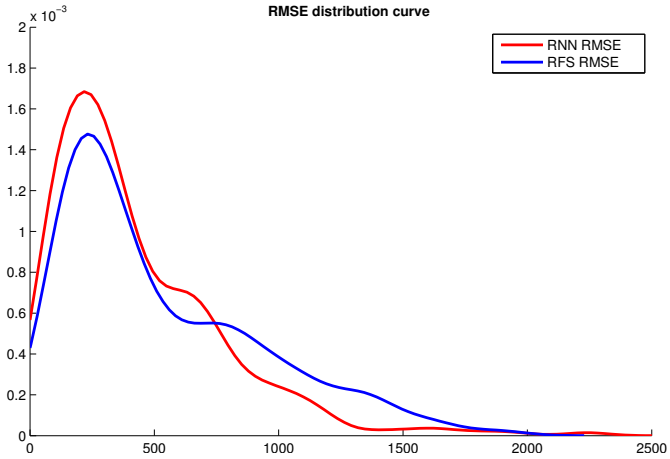


Figure 6: RMSE distribution of the inversions for the RNN and the RFS, excluded the higher values than $2500\mathrm{m/s} \times \mathrm{g/cm^3}$ (6 values).

Conclusions

Considering the simple reflectivity integration, the inverted and real data have qualitatively similar acoustic impedance data (figures 4 and 5), except in the cases where does not predominate the velocity increase trend (figure 5(b));Figures 4 and 5 shows multiplicity of the solution problem, although the inverted data differ from the synthetic input data, the seismic data calculated by the forward modeling present good correlation (with exception for the case shown in the figure 5(b));

The Recurrent Neural Net revealed useful in the 1-D model reflectivity inversion. Using 300 examples and one hidden layer of nine neurons, the net was capable to invert the data. The RNN performance is lower in velocity-depth decrease trend, because the most over of training data set has the opposite behavior.

Recurrent Fuzzy System had a better performance than simple RNN. It approximated in a more efficient way the less frequent training data set. The method showed promising; the results obtained are compatible with those of other methods of current application in the industry

References

Gama, C. A., A. G. Evsukoff, P. Weber, and N. F. F. Ebecken, 2008, Parameter identification of recurrent fuzzy systems with fuzzy finite-state automata representation: IEEE Transactions on fuzzy systems, **16**, 213–227.

Latimer, R. B., R. Davidson, and P. van Riel, 2000, An

interpreter's guide to understanding and working with seismic-derived acoustic impedance data: The Leading Edge, **19**, 242–256.

Mandic, D. P. and J. A. Chambers, 2001, Recurrent neural networks for prediciton. Wiley series in adaptative and learning systems for signal processing, communication, and control: John Wiley.

Poulton, M. M., 2002, Neural networks as an intelligence amplification tool: A review of applications: Geophysics, **67**, 979–993.

Santos, C. G. P., 2008, Inversão acústica de dados sísmicos pós-empilhamento através de redes neurais recorrentes: Master's thesis, COPPE/UFRJ, Rio de Janeiro.

van der Baan, M. and C. Jutten, 2000, Neural networks in geophysical applications: Geophysics, **65**, 1032–1047.