Brazilian Journal of Geophysics, **42** (2024), 2, 1–11
Brazilian Geophysical Society
ISSN 2764-8044
DOI 10.22564/brjg.v42i2.2313

ESTIMATION OF WATER VELOCITY AND RECEIVER POSITION CHANGES IN TIME-LAPSE SEISMIC USING MACHINE LEARNING

Ramon C.F. Araújo [1*], Gilberto Corso [2], Hugo A. D. do Nascimento [3], Samuel Xavier-de-Souza [4], João M. de Araujo [1], and Tiago Barros [4]

[1]Universidade Federal do Rio Grande do Norte - UFRN, Dep. Física Teórica e Experimental, Natal, RN, Brazil
[2]Universidade Federal do Rio Grande do Norte - UFRN, Dep. Biofísica e Farmacologia, Natal, RN, Brazil
[3]Universidade Federal de Goiás - UFG, Instituto de Informática, Goiânia, GO, Brazil
[4]Universidade Federal do Rio Grande do Norte - UFRN, Dep. Eng. Computação e Automação, Natal, RN, Brazil

*Corresponding author: ramon@fisica.ufrn.br

ABSTRACT. Non-repeatability is one of the main obstacles in time-lapse seismic, as it significantly degrades the interpretation of reservoir-related signals. Correcting the data variations caused by non-repeatability (4D noise) is of paramount importance, which usually requires the estimation of the changing parameter. In this paper, we propose a Machine Learning (ML) workflow for the quantitative estimation of two types of 4D noise: changes in the acoustic velocity in water and receiver lateral positions. A synthetic database, modeled from a velocity model estimated from the Brazilian pre-salt and containing many time-lapse seismic surveys, was generated for the supervised training of ML models. The input samples consisted of subsets of common-shot seismograms. We studied many combinations of ML regression algorithms and feature extraction techniques for scenarios where data are contaminated, or not, with Gaussian random noise. The regression algorithms considered were Fully-Connected Neural Network, Extreme Gradient Boosting and Bayesian Ridge. Four rectangular crops of the common-shot seismogram were tested as input features: full seismogram, first half of time samples, 11 smallest-offset traces and the time samples focusing on the region of first arrivals. The combination with the best trade-o between accuracy and model complexity is the Bayesian Ridge fed with the 11 smallest-offset traces, which estimated position and velocity time-lapse changes with median accuracy of 0.115 m and 0.017 m/s for the case with Gaussian noise. Besides the correction of repeatability-related variations, our results are useful in the 4D Full-Waveform Inversion which needs accurate parameters to produce good seismic images.

Keywords: 4D seismic; non-repeatability noise; neural network; XGBoost; Bayesian Ridge.

INTRODUCTION

Time-lapse seismic, the state-of-the-art technology for oil and gas reservoir monitoring, consists of taking successive seismic surveys in the same geographic region over time (Johnston, 2013). The first survey, called the baseline, is typically performed before exploration. After production starts, additional surveys (monitors) are taken every several months to a few years. By analyzing the differences in seismic data across surveys, it is possible to infer the changes in the reservoir over time due to fluid substitution (Nguyen et al., 2015).

In the ideal case, the differences in the seismic data from one survey to the other are related only to reservoir changes (production-related) during the corresponding time period. However, the seismic signature is also influenced by other factors, which in practice are not exactly replicated across the surveys. The mismatch of such factors is called non-repeatability, and the resulting data differences (not related to the reservoir) are termed 4D noise.

In marine seismic, the major non-repeatability factors are the changes of sound velocity in seawater, mispositioning of sources and receivers and tidal variations (Nguyen et al., 2015; Borges et al., 2022).

Seawater velocity varies continuously and is influenced primarily by temperature and water salinity (Carvill, 2009). Also, GPS inaccuracies, ocean currents and weather conditions hinder the replication of the positioning of sources and receivers across surveys (Bertrand and MacBeth, 2005).

Non-repeatability effects significantly degrade 4D seismic signals (Bertrand and MacBeth, 2005; Ritter, 2010). If not properly treated, those effects corrupt data and may lead to false indications of the sea bottom structure and reservoir properties. As a consequence, important tasks such as stacking, depth migration, imaging and inversion are also affected (MacKay et al., 2003; Ritter, 2010). To mitigate these issues, one must correct the variations introduced by the non-repeatability effects in the seismic data during the processing, which usually requires a direct or indirect measure of the causing variable (e.g., water velocity) (MacKay et al., 2003; Amini et al., 2016). An indirect measure of water velocity, for example, would be calculating the associated time shifts in the signal.

In the literature, there are a number of papers about estimating non-repeatability noise and/or correcting its effects. These works usually focus on water velocity and, to a lesser degree, tidal variations. Among the many estimation methods in the literature, we can mention those based on the direct measurement of velocity and height of the water layer with instruments (Wang et al., 2015; Bagaini et al., 2021), tomographic inversion (Ritter, 2010), matching of theoretical and observed direct arrival traveltimes (Amini et al., 2016), and time-differences minimization (MacKay et al., 2003) or time-lapse diffraction (Osdal and Landrø, 2011) analyses of sea bottom reflections. Many of the aforementioned methods rely on the manual picking of key seismic events from the data, which is a laborious and time-consuming task. Also, they are built on top of physical assumptions that may not be satisfied by the seismic data at hand, limiting their applicability.

Besides using physics-based methods, another option is to devise data-driven techniques using machine learning (ML). Machine learning has been increasingly used in geosciences, showing promising results in many applications (Dramsch, 2020; Yoon et al., 2021; Shen et al., 2022). Provided that enough representative data are available, ML-based regression workflows have the potential to be applicable to a broader set of seismic data due to fewer assumptions, while also dismissing manual pickings.

In this paper, we propose a ML-based method for the quantitative estimation of two types of 4D noise that are common in time-lapse marine seismic: the change in the lateral position of receivers and seawater velocity. By modeling a set of time-lapse ocean bottom node (OBN) surveys in deep water, we generated a synthetic database of common-shot seismograms. To generate these seismic data, we employed a velocity model estimated from the Brazilian pre-salt. From each seismogram, relative to a different monitor survey, a subset of data (features) is extracted for supervised training of the ML models. Many combinations of ML regression algorithms and feature extraction techniques were tested, considering the scenarios where data are contaminated or not with Gaussian noise. The motivation of this work is that accurate estimates of 4D noise can be used for correcting the associated variations in the data (Osdal and Landrø, 2011; Wang et al., 2015). Those estimates also have the potential to improve 4D seismic inversion workflows, such as obtaining a more accurate initial velocity model for the 4D Full-Waveform Inversion (FWI-4D) technique (Yuan et al., 2019).

METHODOLOGY

We employ ML for performing the regression of two specific 4D noise parameters: the time-lapse change in acoustic velocity of seawater, ΔV, and in receiver horizontal positions, ΔX. For simplicity, we assume that the velocity is homogeneous in the water layer and that the shift in position is the same for all receivers.

Seismic modeling

To test the method proposed in this work, we generated a synthetic database modeling a number of time-lapse OBN seismic acquisitions in deep water. The seismic data were generated by solving the following acoustic wave equation in 2D with a time-domain finite difference approximation:

$$\frac{1}{v^2(\mathbf{x})}\frac{\partial^2}{\partial t^2}p(\mathbf{x},t) - \nabla^2 p(\mathbf{x},t) = S(t)\,\delta(\mathbf{x}-\mathbf{x_s}), \quad (1)$$

where $\mathbf{x} = (x, z)$ is the position vector, $p(\mathbf{x}, t)$ is the pressure field and $v(\mathbf{x})$ is the acoustic velocity. The source term to the right-hand side describes a point source located at $\mathbf{x_s}$, with waveform $S(t)$.

The main parameters of the numerical simulations are: the time step of 2 ms, the simulated time span of 10 s, grid cell sizes of 8 m, the 4th order finite-difference approximation of the Laplacian operator and, to simulate propagation to infinity at the borders, 25 layers of absorbing CPML (Convolutional Perfectly Matched Layer) cells were added around the analysis domain.

Figure 1a illustrates the seismic surveys modeled in the database. The acquisition geometry consists of 1 source and 49 receivers. The source is fixed at the center of the domain in the horizontal direction and at 8-meter depth; the receivers are spaced 400 meters apart from each other along the sea bottom, reproducing an OBN acquisition. The source employed was a Ricker wavelet with a dominant frequency of 8 Hz. The velocity model used, shown in Figure 1b, is a simplification of a real-world 2D model estimated

from Gato do Mato oil field (Silva et al., 2021), located in Santos basin, offshore Brazil. The simplified model was obtained by taking the first column of the original velocity matrix and replicating it along the horizontal direction.

The data set is composed of 1 baseline and 1000 monitor shot-gather seismograms, each with 49 traces and 5000 time samples. The monitor gathers correspond to different possible realizations of the monitor survey that will be effectively taken in the same region after the baseline. The baseline seismogram was obtained by modeling a survey with reference values of receiver positions and water velocity (equal to 1500 m/s). Monitor seismograms were generated by adding different pairs of (ΔX, ΔV) perturbations to the baseline survey model.

The ΔX and ΔV values used to model the monitor surveys were obtained from an adaptation of the empirical distribution of typical 4D noise values reported in (Cypriano et al., 2019). Although the distribution originally refers to receiver position errors, some of its statistical properties, namely the range of variation and median, are compatible with typical velocity time-lapse noise values in the unit of meters per second. This distribution was truncated to the $[0, 5]$ range and mirrored around zero to cover the negative noise values. Then, 1000 error position values were randomly sampled from this modified distribution, half positive and half negative. The same was performed for ΔV independently. As a result, the values of each noise type vary within the $[-5, 5]$ range, with median ± 2 for each sign (unit in meters for ΔX and meters per second for ΔV). Figures 1c and 1d show the histograms of sampled values of each noise type.

Machine learning strategy

The proposed algorithm employs seismic data to quantify two types of 4D noise in monitor surveys. As previously mentioned, the method assumes that the 4D noise affecting the velocity is constant in the entire water layer and that all receivers are shifted by the same distance along the X direction. The ML model receives as input a subset (input type) extracted from seismic data and outputs the estimated ΔX and ΔV values. To perform this task, we tested many combinations of ML regression algorithms and types of input extracted from seismic data. Those combinations are henceforth called treatments in this paper.

The considered regression techniques are Bayesian Ridge (Tipping, 2001), Fully-Connected Neural Network (FCNN) (Bishop, 1995) and Extreme Gradient Boosting (XGBoost) (Chen and Guestrin, 2016). Bayesian Ridge is a kernel method similar to the Gaussian Process regression (Rasmussen and Williams, 2005); neural networks are universal function approximators inspired by neural interconnections in biological brains; and XGBoost combines the outputs of many decision trees (boosting) to achieve better estimation performance.

The standard Bayesian Ridge and XGBoost are univariate regression techniques. To predict the two noise types, two independent models were trained for each Bayesian Ridge and XGBoost treatment, one for estimating ΔX and the other for ΔV. For FCNN treatments, on the other hand, a single neural network was considered for estimating both noise types.

The FCNN considered is a feedforward network formed of three hidden layers of 128-64-16 neurons, all using the ReLU activation function, and 2 neurons with no activation function in the output layer. The network was trained for 300 epochs using the RMSProp optimizer, with 10^{-3} learning rate and MSE loss function. The XGBoost model is an ensemble of 50 boosted trees with maximum depth equal to 6, learning rate equal to 0.3 and the L2 regularization term on weights equal to 1. As for the main parameters of the Bayesian Ridge model, the maximum number of iterations is equal to 200 and the priors over the target noise variance and over the regularization parameters of the algorithm (Tipping, 2001) were chosen as the Gamma distribution with shape and rate parameters equal to 10^{-6}. We used the implementations of neural networks, Bayesian Ridge and XGBoost algorithms from the libraries TensorFlow/Keras 2.4.1, scikit-learn 1.0.2 and dmlc/xgboost 1.5.1, respectively.

The tested input types are the full shot-gather seismogram and also three rectangular crops extracted from it: the first half of time samples (called "Half seismogram"), the 11 smallest-offset traces and the "Focus arrivals" input, which is formed by all traces within the 1.3–1.8 s time range. This last input type focuses on the region of first arrivals of the smallest-offset traces. Figure 2 illustrates all the input types tested in this work.

We also experimented with contaminating seismic data with random noise, in order to assess the robustness of the methodology against a kind of disturbance commonly found in real data. All the combinations of ML algorithm and input type (treatments) are tested for two scenarios: seismic signals free of random noise and seismic data contaminated with additive white Gaussian noise (AWGN) of 10 dB signal-to-noise ratio (S/N). An example of a single trace polluted by 10 dB AWGN is illustrated in Figure 3. Table 1 lists all the 12 treatments tested in this work. Considering the scenarios in which random noise is present or absent, we performed 24 simulations in total.

The ML 4D noise regression is carried out according to the workflow shown in Figure 4. From the database described in the previous section, we calculate the difference between each monitor and the baseline seismogram.

These difference gathers may or may not be contaminated with Gaussian noise, depending on the scenario being considered. Next, one of the input types

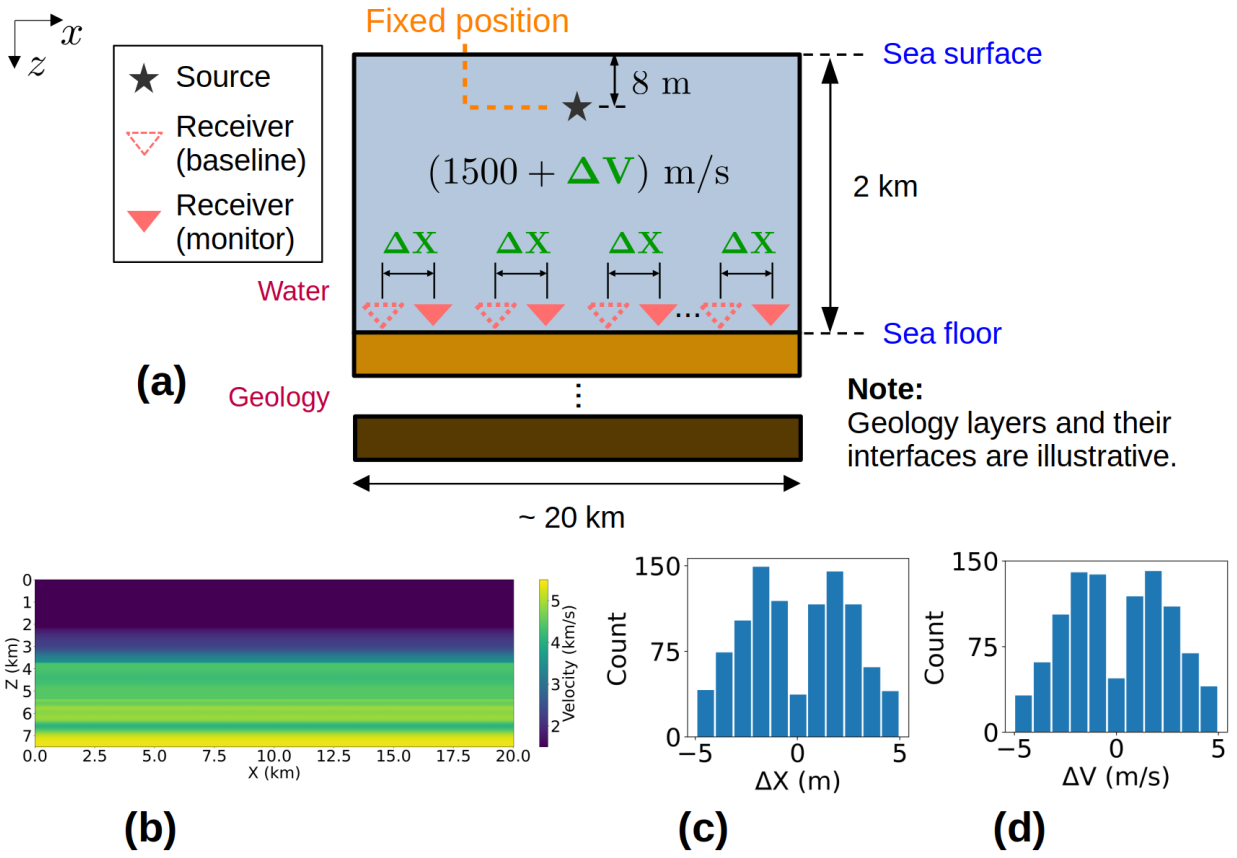

Figure 1: Seismic surveys modeled in the synthetic database. (a) Acquisition geometry (off-scale); baseline survey corresponds to $\Delta X = \Delta V = 0$. (b) Baseline velocity model. (c) Histogram of position 4D noises. (d) Histogram of velocity 4D noises.

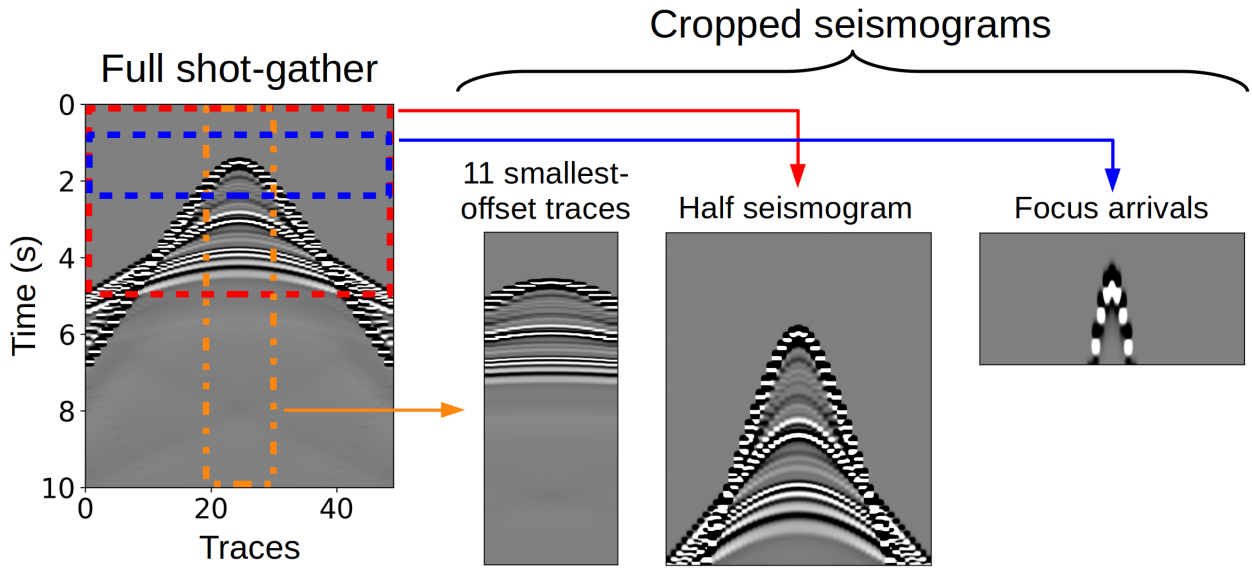

Figure 2: Considered types of input extracted from shot-gather seismograms.

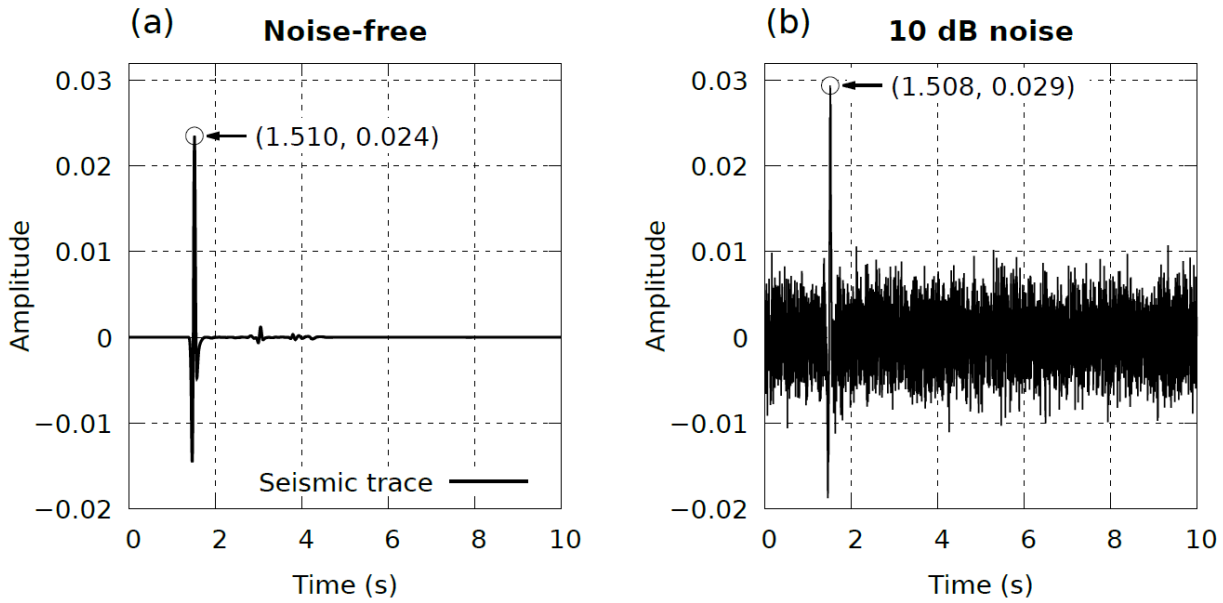

Figure 3: Illustration of seismic trace (a) free of random noise and (b) contaminated with 10 dB S/N Gaussian noise. Noise causes a strong disturbance in the data. The trace peak, for example, which can be used to calculate the first arrival, undergoes amplitude changes and time shifts.

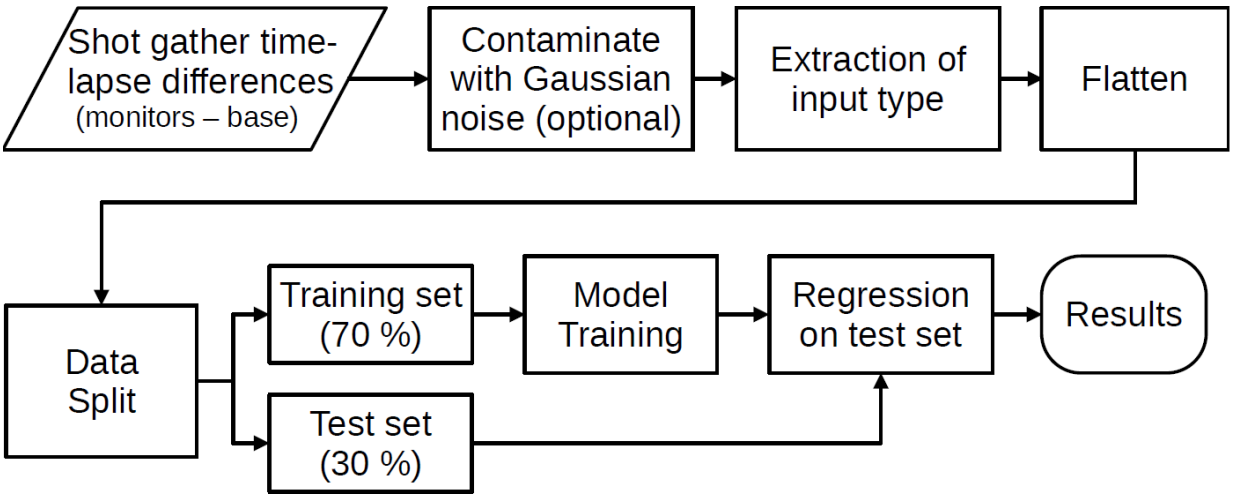

Figure 4: Block diagram of the proposed 4D noise regression methodology.

Table 1: Treatments according to the ML method and input type.

Treatment Label	Regression Algorithm	Input Type
S1	FCNN	Full seismogram
S2	FCNN	Half seismogram
S3	FCNN	11 smallest-offset traces
S4	FCNN	Focus arrivals
S5	XGBoost	Full seismogram
S6	XGBoost	Half seismogram
S7	XGBoost	11 smallest-offset traces
S8	XGBoost	Focus arrivals
S9	Bayesian Ridge	Full seismogram
S10	Bayesian Ridge	Half seismogram
S11	Bayesian Ridge	11 smallest-offset traces
S12	Bayesian Ridge	Focus arrivals

shown in Figure 2 is extracted from the difference seismograms. The extracted crop is then rearranged as a 1D vector (flattened) so it can be passed as input to the ML techniques. The data are randomly split into 70% and 30% disjoint subsets. The first 70% set is used for supervised training of the ML regression model. After training, the 30% set is used for assessing the model accuracy in estimating ΔX and ΔV. In all simulations, the same monitor shot gathers were used to generate the training and test data in order to facilitate the comparison.

RESULTS AND DISCUSSION

In this section, we present the results of the proposed methodology for all treatments, considering the scenarios with and without Gaussian noise. The treatments are labeled according to Table 1. All simulations were performed on a computer with Intel® Xeon® E5-2698 v3 CPU and 512 GB RAM. The predictive performances obtained by the treatments are assessed by computing the estimation error magnitudes (EEM) on the test data. The EEM is calculated as the magnitude of the difference between the true value of 4D noise (ΔX or ΔV) and the corresponding value estimated by the ML regression model. For treatments using Bayesian Ridge, the estimated $\Delta X/\Delta V$ is taken as the mean of the predictive distribution of the model.

Most results are shown as boxplots describing the distribution of the test EEMs obtained by each treatment for position and velocity noise. The fundamental element of a boxplot is the box with whiskers (Walpole et al., 2016). The box marks the distribution quartiles: its lower and upper edges coincide respectively with the 25^{th} and 75^{th} percentiles, and the horizontal line inside marks the median. The vertical lines extending above and below the box (whiskers) span the range containing most of the data points below the first quartile and above the third quartile. The circles beyond the whiskers represent individual data points of extreme values (outliers).

Figure 5 shows the boxplot of EEMs obtained for AWGN-free seismic signals. The position errors can be compared with velocity errors after dividing all values by 10, which is the amplitude of the range of variation of each noise type (both vary within the $[-5, 5]$ range, as described in the seismic modeling section). After doing that, we notice that the normalized ΔX errors are higher than the normalized ΔV errors, suggesting that receiver position 4D noise is harder to estimate. It can also be observed that the XGBoost algorithm (treatments S5 to S8) tends to estimate velocity noise accurately on average (despite the outliers), but, on the other hand, it presents the worst results when estimating position noise.

The same general observations can be made for the case where seismic data is contaminated with 10 dB Gaussian noise, shown in Figure 6. The difference is that EEMs are larger in this case, which is expected because the physical attributes correlated with the 4D noise types, such as times of first arrival (Osdal and Landrø, 2011), are harder to extract accurately from seismic data when random noise is present (see Figure 3). A simple comparison between the vertical axes of Figures 5 and 6 shows that EEMs are five to ten times larger when AWGN is present.

For each studied 4D noise type, water velocity and receiver position, we employ a three-way ANOVA sta-

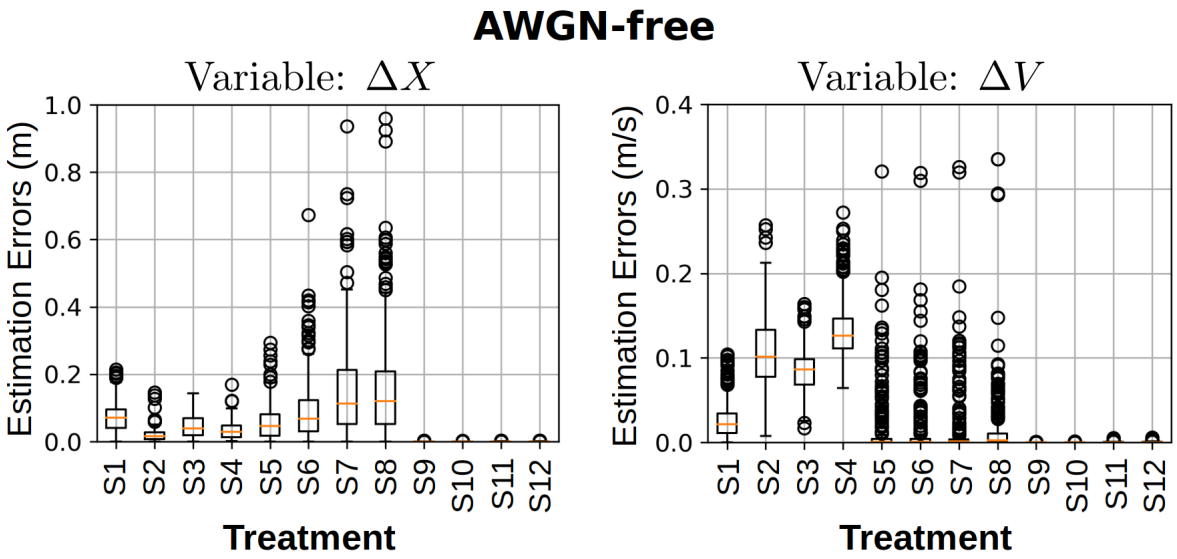

Figure 5: Block diagram of the proposed 4D noise regression methodology.

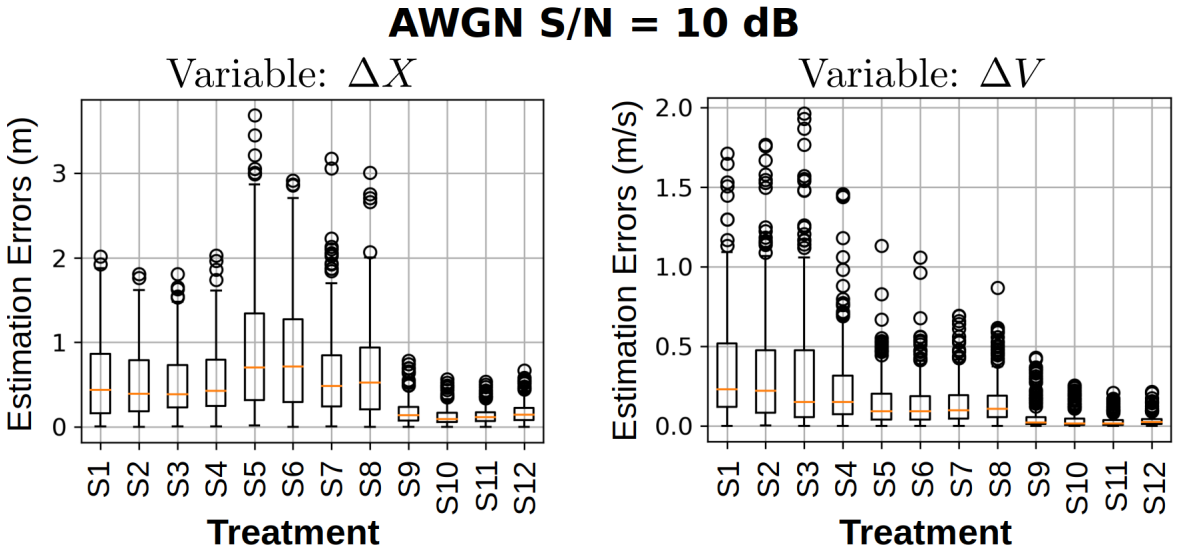

Figure 6: Estimation error magnitudes for the 10 dB-AWGN case.

tistical test (Walpole et al., 2016) with the following factors: Gaussian noise, ML technique and seismogram crop. The first factor has two classes: AWGN-free and 10 dB Gaussian noise. The second factor has three classes: Bayesian Ridge, FCNN and XGBoost. The third factor has four classes, one for each input type. The total number of classes in the two statistical experiments is $2 \times 3 \times 4 = 24$. The ANOVA test is calculated from the 300 EEM values (test set) associated to each class. The results are shown in Table 2.

The performed ANOVA test allows us to answer the questions: "is our methodology affected by random noise?", "what is the best ML technique?" and, finally, "is there an input crop that performs better than others?". The first two questions can easily be answered by visual inspection of Figures 5 and 6, and the statistical test confirms: noise affects estimation accuracy significantly, and Bayesian Ridge is the best regression algorithm. The answer to the third question is inconclusive, as shown in Table 2; the difference among the crops is statistically significant for estimating ΔX, but no crop performs better than others for the regression of ΔV.

The Bayesian Ridge algorithm greatly outperforms all other techniques, for both the cases with and without Gaussian noise. The success of the Bayesian strategy is evident for all crops; indeed, the crop choice is not as important as the employed ML regressor (Figures 5 and 6). Among the reasons for its success, the Bayesian Ridge uses one optimized prior for each basis function weight, resulting in fewer basis functions (sparsity) and better generalization capabilities (Tipping, 2001). However, the high accuracy comes at a cost: it has the largest computational complexity (resource usage increases the fastest with input size) among the studied algorithms. Using the full seismogram input, for example, takes about 500 GB RAM to train one Bayesian model for one noise type. Note that the memory consumption is much lower for the other treatments: 122 GB for S10, 26 GB for S11 and less than 4 GB for the others.

Figure 7 shows estimation errors only for the treatments using the Bayesian Ridge. In the AWGN-free case, estimation errors are so low that the input types can be considered equally good from a practical point of view. However, when Gaussian noise is present, a more realistic scenario, estimation errors are large enough for the differences to matter, especially outliers. In qualitative terms, we argue that the treatment with the best balance between accuracy and ML model complexity is the Bayesian Ridge fed with the 11 smallest-offset traces (S11), because in the noisy case its median accuracies are among the lowest, with the smallest outliers. This treatment is also favored by the smaller input size, as per the principle of Occam's razor. In treatment S11, ΔX and ΔV are estimated with median accuracy of 2.9×10^{-4} m and 2.5×10^{-4} m/s for the AWGN-free scenario, and 0.115 m and 0.017 m/s for the noisy case.

Finally, we note that the proposed methodology is supervised and depends on information not available in the real world for training (true 4D noise values). To apply the method in practice, it is necessary to synthetically model a number of monitor surveys from the baseline, as performed in this study. If the modeling is realistic, covering many plausible 4D noise settings, there is a high confidence that the ML model can generalize to data not seen in training, so that its estimations can be used for correcting the 4D noise effects on real monitor data, for example. In many areas of geosciences, such as in hydrocarbon reservoir characterization, it has been demonstrated that ML algorithms trained on realistically modeled synthetic data present good performance when applied to field data (Côrte et al., 2020). In the case of our method, realistic modeling of monitor surveys is possible by using some results of full-track baseline processing, especially the inverted velocity model. Although some simplifications were adopted in the modeling of this study, the results herein presented are still useful because the very good performances demonstrate the potential of using ML-based algorithms for estimating 4D noise on field data.

CONCLUSION

In this study, we proposed a ML methodology for quantifying the time-lapse changes, relative to the baseline survey, in the acoustic velocity in seawater and receiver lateral positions. A synthetic database, modeling deep water OBN time-lapse surveys using a velocity model estimated from the Brazilian pre-salt, was built for supervised training of ML models.

We tested many combinations (treatments) of regression algorithms and types of input extracted from common-shot seismograms, for the scenarios where seismic data are free of random noise and contaminated with 10 dB Gaussian noise. Our results highlight the importance of an adequate selection of the regression algorithm and, to a lesser degree, of the input type. Also, the presence of Gaussian noise significantly affects accuracies, which is expected because it is harder to extract from noisy data the physical attributes (e.g., time of first arrival) correlated with 4D noise. The treatment with the best balance between accuracy and model complexity estimated position and velocity time-lapse changes with a median accuracy of 0.115 m and 0.017 m/s for the 10 dB Gaussian noise case.

This study demonstrates the feasibility of using supervised ML-based techniques for estimating 4D noise values. Accurate estimates of such parameters can be used for correcting their effect on seismic data during processing, or to improve time-lapse inversion workflows by, for example, obtaining better initial approximations of the time-lapse velocity model for the FWI-4D technique.

Table 2: Summary of three-way ANOVA results for each 4D noise type, discriminating among the factors varying in ML treatments.

4D Noise Type	Factor	F	p-value
Receiver position	Gaussian Noise	3221	$< 10^{-16}$
	ML technique	783	$< 10^{-16}$
	Seismogram crop	10.01	1.4×10^{-6}
Water velocity	Gaussian Noise	1294	2×10^{-16}
	ML technique	884	$< 10^{-16}$
	Seismogram crop	2.09	0.099

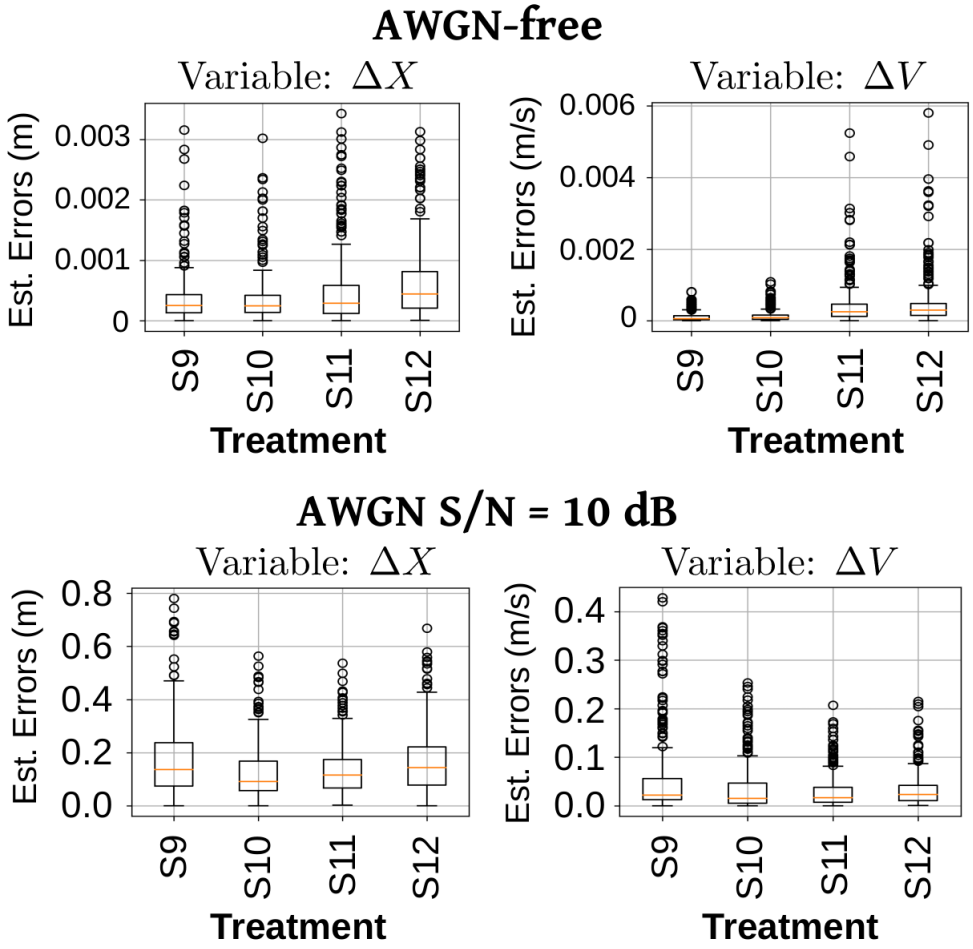

Figure 7: Estimation error magnitudes of treatments using Bayesian Ridge regression algorithm.

ACKNOWLEDGMENTS

The authors gratefully acknowledge support from Shell Brasil through the "New computationally scalable methodologies for target-oriented 4D seismic in pre-salt reservoirs" project and the strategic importance of the support given by ANP through the R&D levy regulation. We acknowledge NPAD/UFRN for the computational support and the National Council for Scientific and Technological Development (CNPq) for funding (grant number 307907/2019-8).

DATA AND MATERIALS AVAILABILITY

The data used in this research are private.

REFERENCES

Amini, A., H. Peng, Z. Zhang, R. Huang, and J. Yang, 2016, Joint inversion of water velocity and node position for ocean-bottom node data: SEG Technical Program Expanded Abstracts 2016, 5490–5494. doi: 10.1190/segam2016-13964730.1.

Bagaini, C., E. Muyzert, K. Kapllani, M. Salgadoe, and N. Seymour, 2021, On the usage of direct measurements of water velocity in deep water OBN surveys: 21–25, doi: 10.1190/segam2021-3583937.1.

Bertrand, A., and C. MacBeth, 2005, Repeatability enhancement in deep-water permanent seismic installations: a dynamic correction for seawater velocity variations: Geophysical Prospecting, **53**, 229–242, doi: 10.1111/j.1365-2478.2004.00465.x.

Bishop, C. M., 1995, Neural networks for pattern recognition: Oxford University Press.

Borges, F., M. Muzzette, L. E. Queiroz, B. Pereira-Dias, R. Dias, and A. Bulcão, 2022, Analysis of water velocity changes in time-lapse ocean bottom acquisitions – A synthetic 2D study in Santos Basin, offshore Brazil: Journal of Applied Geophysics, **197**, doi: 10.1016/j.jappgeo.2021.104521.

Carvill, C. V., 2009, A new approach to water velocity estimation and correction: Presented at the 71st EAGE Conference & Exhibition. Amsterdam, Netherlands. doi: 10.3997/2214-4609.201400384.

Chen, T., and C. Guestrin, 2016, XGBoost: A scalable tree boosting system: 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16). San Francisco, CA, USA, 785–794. doi: 10.1145/2939672.2939785.

Côrte, G., J. Dramsch, H. Amini, and C. MacBeth, 2020, Deep neural network application for 4D seismic inversion to changes in pressure and saturation: Optimizing the use of synthetic training datasets: Geophysical Prospecting, **68**, 2164–2185, doi: 10.1111/1365-2478.12982.

Cypriano, L., Z. Yu, D. Ferreira, B. Huard, R. Pereira, F. Jouno, A. Khalil, E. Urasaki, N. Cruz, A. Yin, D. Clarke, and C. C. Jesus, 2019, OBN for pre-salt imaging and reservoir monitoring – Potential and road ahead: Presented at the 16th International Congress of the Brazilian Geophysical Society. Rio de Janeiro, Brazil. doi: 10.22564/16cisbgf2019.318.

Dramsch, J. S., 2020, Chapter one – 70 years of machine learning in geoscience in review, *in* Machine Learning in Geosciences: Elsevier, volume **61** *of* Advances in Geophysics, 1–55. doi: 10.1016/bs.agph.2020.08.002.

Johnston, D. H., 2013, Practical applications of time-lapse seismic data: Society of Exploration Geophysicists.

MacKay, S., J. Fried, and C. Carvill, 2003, The impact of water-velocity variations on deepwater seismic data: The Leading Edge, **22**, 344–350, doi: 10.1190/1.1572088.

Nguyen, P. K. T., M. J. Nam, and C. Park, 2015, A review on time-lapse seismic data processing and interpretation: Geosciences Journal, **19**, 375–392, doi: 10.1007/s12303-014-0054-2.

Osdal, B., and M. Landrø, 2011, Estimation of changes in water column velocities and thicknesses from time lapse seismic data: Geophysical Prospecting, **59**, 295–309, doi: 10.1111/j.1365-2478.2010.00923.x.

Rasmussen, C. E., and C. K. I. Williams, 2005, Gaussian processes for machine learning: The MIT Press.

Ritter, G., 2010, Water velocity estimation using inversion methods: Geophysics, **75**, U1–U8, doi: 10.1190/1.3280232.

Shen, S., H. Li, W. Chen, X. Wang, and B. Huang, 2022, Seismic fault interpretation using 3-D scattering wavelet transform CNN: IEEE Geoscience and Remote Sensing Letters, **19**, 1–5, doi: 10.1109/LGRS.2022.3183495.

Silva, D. d., E. F. Duarte, W. Almeida, M. Ferreira, F. A. Moura, and J. a. M. d. Araújo, 2021, Target-oriented inversion using the patched Green's function method: Geophysics, **86**, R811–R823, doi: 10.1190/geo2020-0640.1.

Tipping, M. E., 2001, Sparse Bayesian learning and the relevance vector machine: Journal of Machine Learning Research, **1**, 211–244.

Walpole, R., R. Myers, S. Myers, and K. Ye, 2016, Probability & Statistics for Engineers & Scientists, 9 ed.: Pearson.

Wang, K., S. Dunn, J. Ward, and P. Hatchell, 2015, Direct measurement of water velocity and tidal variations for improved 4D repeatability in marine seismic acquisition: First Break, **33**, 73–79, doi: 10.3997/1365-2397.33.5.80180.

Yoon, D., Z. Yeeh, and J. Byun, 2021, Seismic data reconstruction using deep bidirectional long short-term memory with skip connections: IEEE Geoscience and Remote Sensing Letters, **18**, 1298–1302, doi: 10.1109/LGRS.2020.2993847.

Yuan, C., X. Zhang, X. Jia, and J. Zhang, 2019, Time-lapse velocity imaging via deep learning: Geophysical Journal International, **220**, 1228–1241, doi: 10.1093/gji/ggz511.

Araujo, R.C.F.: writing original draft, state-of-the-art literature review, methodology design, synthetic database development, result development and analysis; **Corso, G.:** conceptualization, manuscript review, hypothesis test result development, result discussion, research funding acquisition; **do Nascimento, H.A.D.:**manuscript review, improvements to result analysis and presentation; **Xavier-de-Souza, S.:** manuscript review, result discussion, research funding acquisition, supercomputing resource access; **de Araujo, J.M.:** manuscript review, result discussion, research funding acquisition, supercomputing resource access; **Barros, T.:** manuscript review, research scope, result discussion.

Received on November 24, 2023 / Accepted on June 20, 2024