Seismic inversion with Gradient Boosting

Marcelo Guarido, Luping Qu, Zhan Niu, Kai Zhuang, David J. Emery, Daniel Trad, and Kristopher Innanen

ABSTRACT

Geophysics in the Cloud promoted a competition with the proposal to use machine learning algorithms to perform seismic inversion. The provided data showed to be challenging, due to the small number of well logs to train the model. During the competition, most of the competitors based their strategies using deep learning models (CNN, RNN, LSTM, etc). Our proposal was to evaluate the possibility of seismic inversion using more classical and simple approaches, hence the choice of the Gradient Boosting algorithm. Predictions of P and S impedances and density using an XGBoost model are stable and strongly rely on the trend features and use seismic traces to include low to mid-frequency content. Testing neural networks structures from Scikit-Learn and Tensorflow resulted in noisy and spiky inversions, pointing to overfitting and their instability drew attention to the issue of the small number of logs for training.

INTRODUCTION

Seismic inversion is the field that consists on retrieving rock properties from seismic gathers. Different methods can be used, and some of the classical ones are the trace integration (Lindseth, 1979), AVO (Hampson, 1991), *Full-Waveform Inversion*, or FWI (Tarantola, 1984; Pratt et al., 1998; Virieux and Operto, 2009), hybrid AVO-FWI (Chen et al., 2020), and *Full-Traveltime Inversion* (Luo et al., 2016). Each of these methods contains their pros and cons, and are strongly related to the physics behind them. Therefore, introducing machine learning methods to seismic inversion opens the possibility to data-driven outcomes, although a deep understanding among seismic features and rock properties is still necessary (Emery and Guarido, 2021).

Machine learning algorithms have been used for different approaches is geophysics and energy industry, and has being of great assistance for interpretation (Caté et al., 2017). One common application is to predict rock types from well logs (Hall, 2016; Guarido, 2019; Guarido et al., 2021a). Applications on fault detection in seismic sessions using *Convolutional Neural Networks* (Araya-Polo et al., 2017; Xiong et al., 2018; Guarido et al., 2021b), automatic-velocity picking from unsupervised learning (Smith, 2017), and neural networks assisted AVO (Russell et al., 2002) are also present in the literature. Combining machine learning with seismic inversion methods are getting more robust over the years. Lewis and Vigh (2017) use deep learning to generate more complex and reliable initial models for FWI, while Chen et al. (2017) use deep learning algorithms for automated time-window selection in FWI. Chen and Schuster (2020) use a wave-equation controlled auto-encoder to perform acoustic inversion with promising results on synthetic data. Roy et al. (2020) show the possibility on using tree-based methods to predict reservoir properties from seismic features. Sun et al. (2020) introduce a theory-designed recurrent neural network (RNN) algorithm that successfully performs acoustic inversion on synthetic data.

Alvarez et al. (2021) organized the *Geophysics in the Cloud* competition, which the goal was to perform elastic seismic inversion using machine learning algorithms. While most of the solutions were based on deep learning methods, our focus is to show the use of *extreme gradient boosting*, or XGBoost (Chen and Guestrin, 2016) a more classical and simple method. We compare the results with fully-connected neural networks, also called *multi-layer perceptron*, from the *Scikit-Learn* (Pedregosa et al., 2011) and *Tensorflow* (Abadi et al., 2015) libraries, show similar results for the impedances, but the XGBoost is more robust for the density inversion.

GEOPHYSICS IN THE CLOUD

Geophysics in the Cloud was a competition in 2021 organized by Alvarez et al. (2021) to perform seismic inversion of P and S impedances and density using machine learning. All solutions should be connected to the *Amazon Web Services* (AWS). A broad selection of data was offered to the competitors, such as seismic volumes from the Poseidon 3D survey in Australia (Figure 1), well logs, and horizons. Each competitor (or groups) needed to come with their own solution and provide inversions for P-Impedance, S-Impedance, and density.

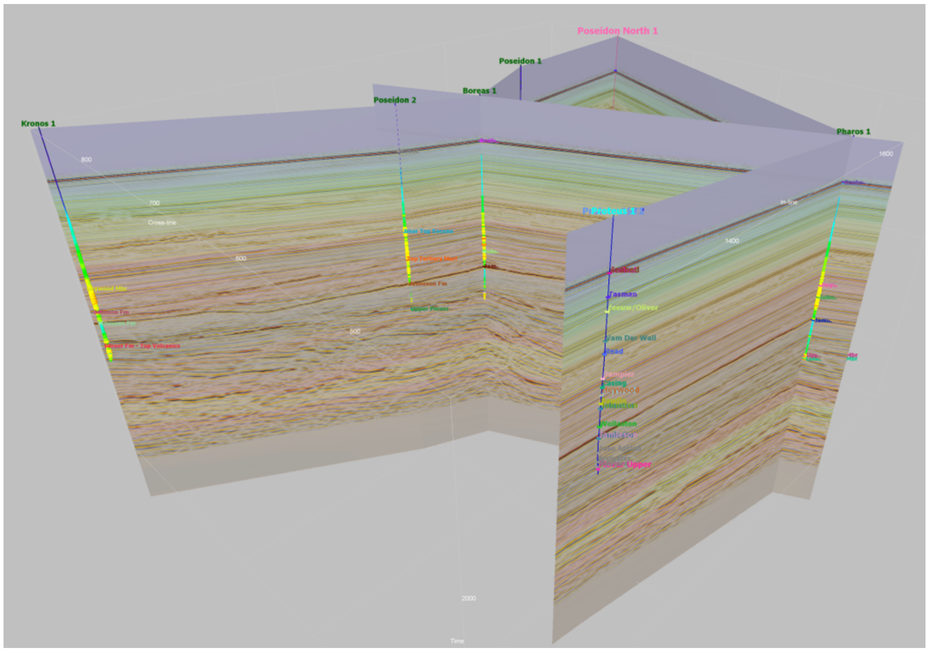

FIG. 1. Seismic sessions from Poseidon 3d

Well logs are tied up with the seismic, and seismic and logs are resampled to match the sample rate, and that's a step less for the competitors. The well logs are from 6 well locations among the survey area, and contain the targets (inversion goals). Density should be inverted using RHOB as target, and P-impedance and S-impedance can be calculated from DTC and DTS, respectively, and RHOB. The list complete list of provided data is in the Table 1.

For this project, the strategy was to use the near, mid, and far seismic stacks, in addition to the background velocity, to train a gradient boosting model (Hastie et al., 2001) at the 6 wells locations, as logs and seismic are tied Figure 2), and calculated the predictions at

Table 1. GITC data list

Provided Data		
Data Type	Contents	Comment
Seismic	Near, mid, far, and full stack	3D and Full stack with AGC
Velocity Model	Back ground velocity	Picked velocity
Well Logs	DTC, DTS, RHOB, GR, RSHAL, RDEEP, NPHI, CAL	6 wells tied to seismic
Horizons	WB + 4 tops	-

the testing location (areas of the seismic volume that contain wells logs not provided to the competitors). Predictions over the whole seismic volume were not required, and all the presented predictions are at the well locations.

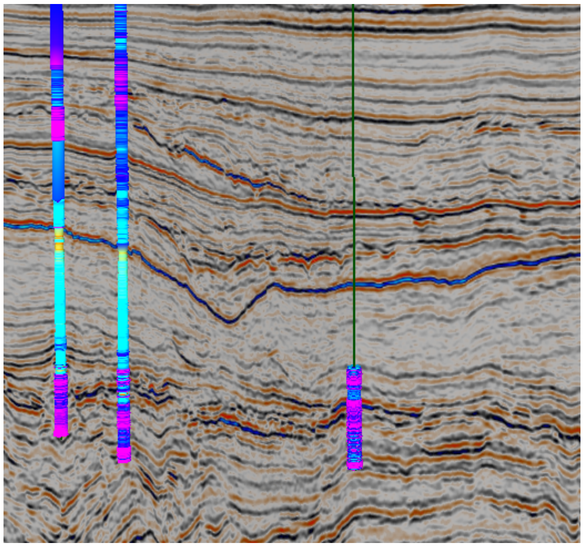

FIG. 2. Wells matching the seismic data

DATA PREPARATION

Guarido et al. (2021a) show the pitfalls working with competitions datasets and how missing data interferes for a reliable prediction, as well the amount of data to train a machine learning model. GITC competition provided well logs from 6 locations inside the survey area, which is not much, and it gets even more challenging when observing the large amount of missing samples in the target logs. Guarido et al. (2021a) also present an iterative data imputation method to recover missing samples in different well logs by completing the one with the less number of missing data, using the complete wells as features of a regression model, and then including the new complete well as features to predict for the missing data of the next log. This process is done until all desired logs are complete. Although, the scope of this report is to show the machine learning inversion methodology, assuming the data is already pre-processed. The data imputation step is presented by Huang et al. (2021), and a step-by-step hands-on demonstration can be found at Qu et al. (2021). Figure 3 shows the DTC log converted to P-velocity for the 6 provided wells (blue) and the background velocity (red). Note the "blocky" regions resulted from the data imputation (Qu et al., 2021). The same is observed on the other target logs. This effect will be neglected in the current workflow, but it will be tackled back in the future work.

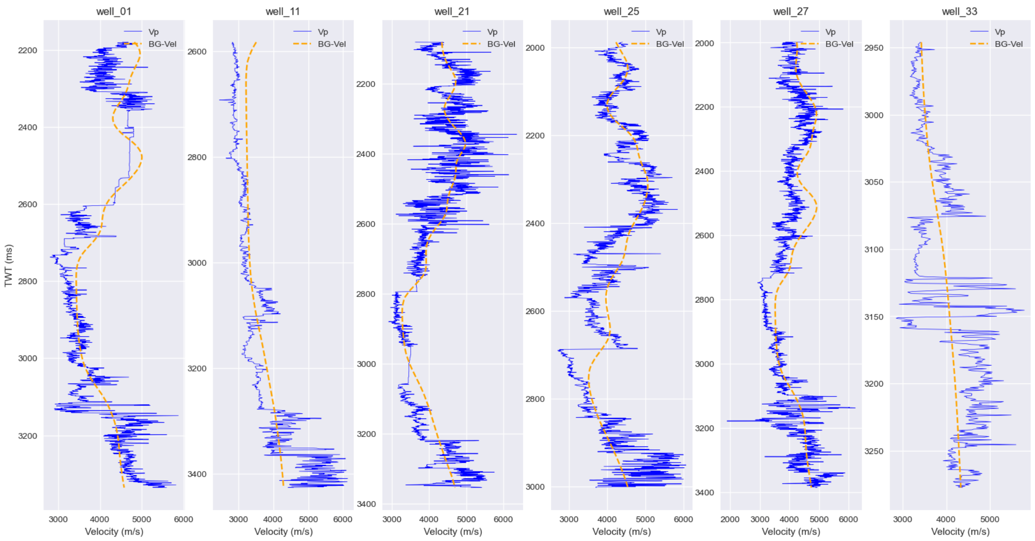

FIG. 3. DTC log converted to P-velocity for the 6 provided wells (blue) and the background velocity (red)

GRADIENT BOOSTING

A decision tree alone is considered a weak classifier or regressor and ensembles trees is a proposal to combine weak models into a strong one. Gradient Boosting (Hastie et al., 2001) ensemble decision trees in sequence, where the next tree fits over the residuals of the previous iteration (Figure 4). Each tree is created and fitted following the gradient descent method. It is usually a powerful method, but it can get computationally expensive, as it is difficult to do parallel processing, and it is highly sensitive to outliers, due to the residual fitting algorithm. However, it is less susceptible to overfit, requires less data pre-processing, as it handles both numerical and categorical features, and no standardization or normalization is required.

For this project, we are using the XGBoost (Chen and Guestrin, 2016) library for the modeling. It is an implementation of the gradient boosting that can be trained on GPUs.

MODELING WITH GRADIENT BOOSTING

As part of this project's proposal, gradient boosting models are trained for each one of the targets, using as input the seismic traces from the near, mid, and far stack, background velocity, and the two-way traveltime (TWT) that match the wells locations. P-impedance, S-impedance, and RHOB are the targets. As we are dealing with continuous numbers, this is a regression problem.

Observing Figure 3, the background velocity tends to follow the trends of the well log velocities, which can compensate the usual lack of low frequency content of the seismic traces. Our expectation is that our predictions for P-Impedance will follow closely the background velocity and will be refined by the seismic information. The *Well 27* will be our blind well for model evaluation, meaning we will train the models on the other 5 wells.

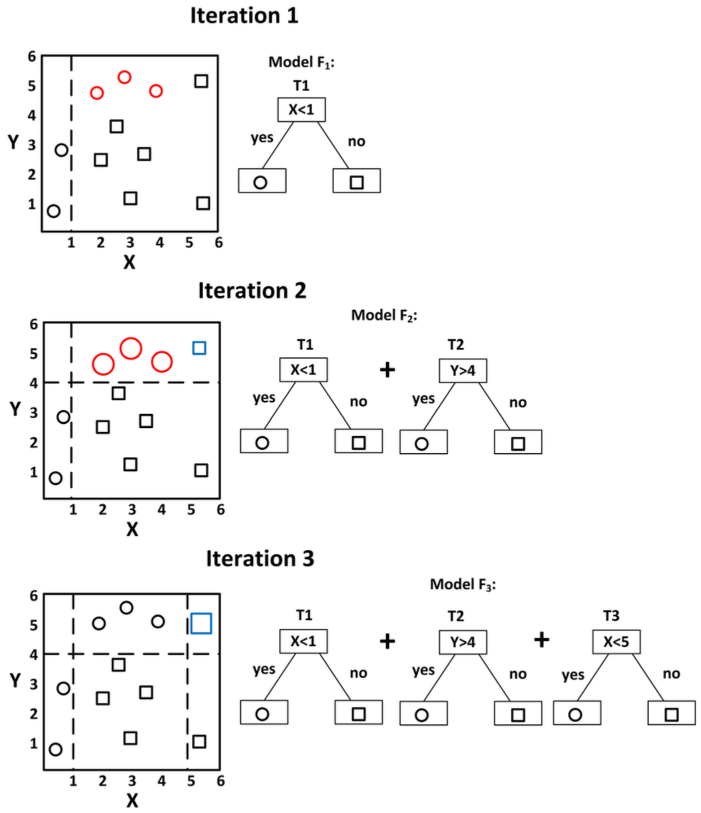

FIG. 4. Gradient Boosting diagram showing how new decision tress are used to fit on the residuals of the previous iteration. Figure from Zhang et al. (2018)

Figure 5 shows the P-impedance (I_P) prediction in orange, true P-impedance in blue, and background velocity multiplied by RHOB (to simulate a background P-impedance) in red. Note that from 2s to 2.8s, the background impedance in overestimated when compared with the true one, but the gradient boosting inversion is significantly correcting it, but still overestimating. In the 2.2s to 2.3s interval, the prediction stops following the trend and underestimates the true P-Impedance. Also, mid-frequency content is included to the predictions, information that came from the seismic traces, but could not follow the high-amplitude-high-frequency events starting at 3.1s.

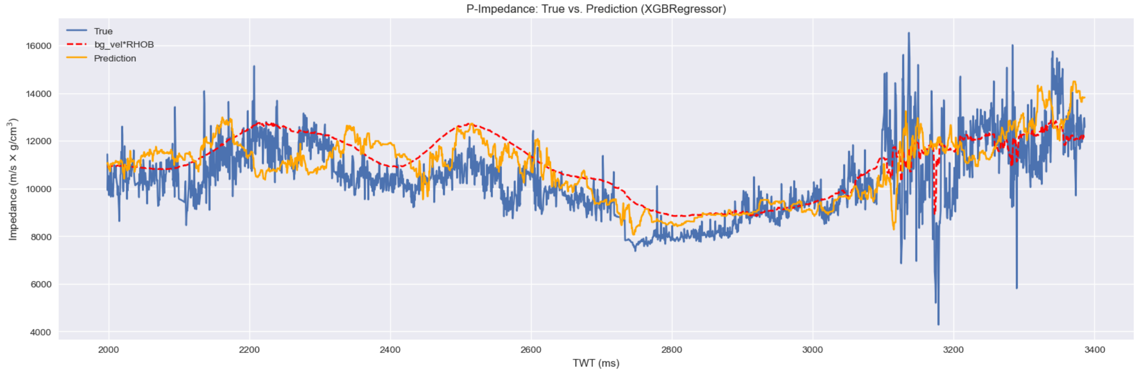

FIG. 5. P-impedance (I_P) prediction (orange), true P-impedance (blue), and background P-impedance (red).

One advantage on using gradient boosting is that it is relatively easy to interpret, and we can extract the feature importance, shown in Figure 6. Surprisingly, two-way traveltime is the most important feature for the prediction, followed by the background velocity, and the seismic traces a relatively unimportant. TWT may be roughly providing a linear trend, as density and velocities tend to increase with depth. This happened when we selected small trees (with max depth of 2). For larger trees, the seismic traces gained importance, but predictions started to show spikes, probably because the model was trying to fit on the high frequency content of the true impedance, which does not existing in the seismic, causing overfitting.

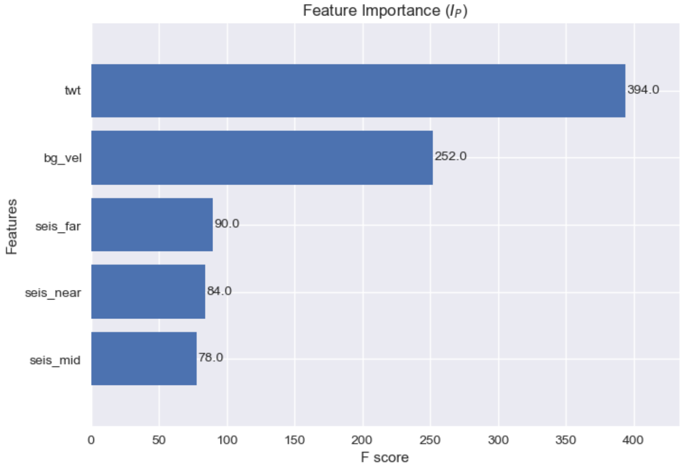

FIG. 6. Feature importance of the P-impedance modeling.

For the S-impedance (I_S) modeling, the predicted P-impedance is included to the feature list, and the result is shown in Figure 7. Predictions (orange) are following closely the true S-impedance (blue) trend, but we can still see the inversion in trend in the 2.2s to 2.3s interval, and this same behavior happens in the 3.1s to 3.2s and 3.3s to 3.4s intervals. Also, starting at 3s, the predictions start to oscillate more.

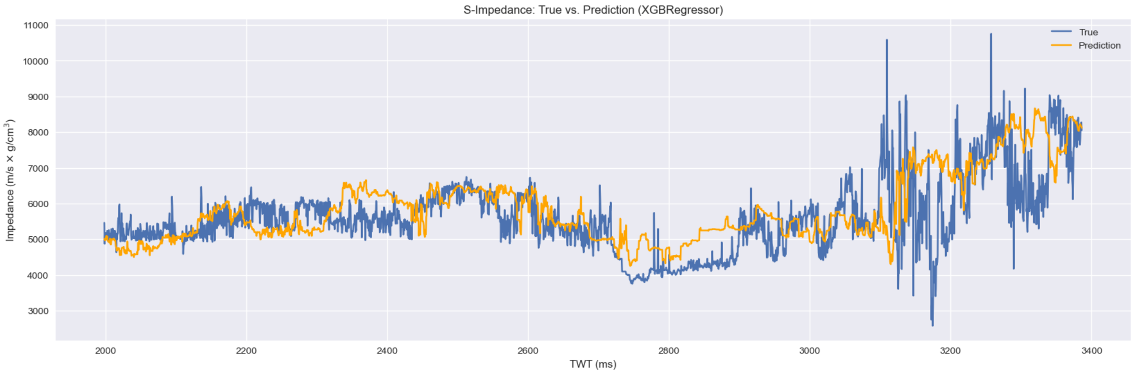

FIG. 7. S-impedance (I_S) prediction (orange) and true S-impedance (blue).

As for the P-impedance modeling, Figure 8 shows the feature importance for the S-impedance modeling. The model still relies more on the trend givers (TWT and background velocity), followed by the predicted P-impedance, and the seismic traces still show low importance.

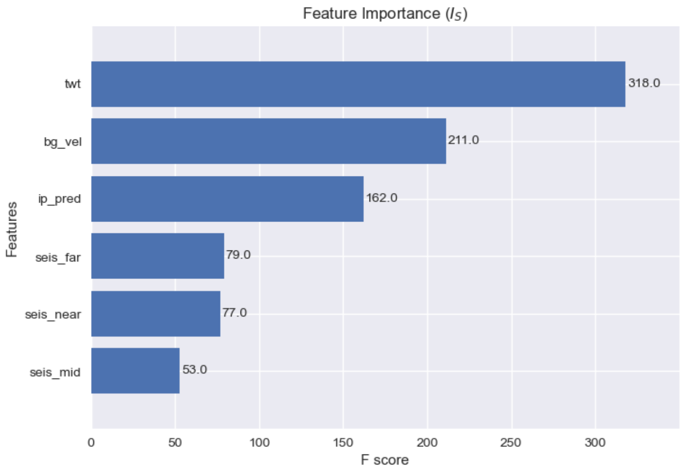

FIG. 8. Feature importance of the S-impedance modeling.

Density predictions (orange) are shown in Figure 9, and the predicted P and S impedances are included as features. It is clear a change in behavior of the true density (blue) starting at around 2.7s. It could be caused by two different wireline logging surveys. Or some effect from the data imputation process. But the predictions are following these changes in behavior, and follows the true values trends. At 3.2s, predictions strongly differ from the true density, but, in the overall, it is a reliable prediction.

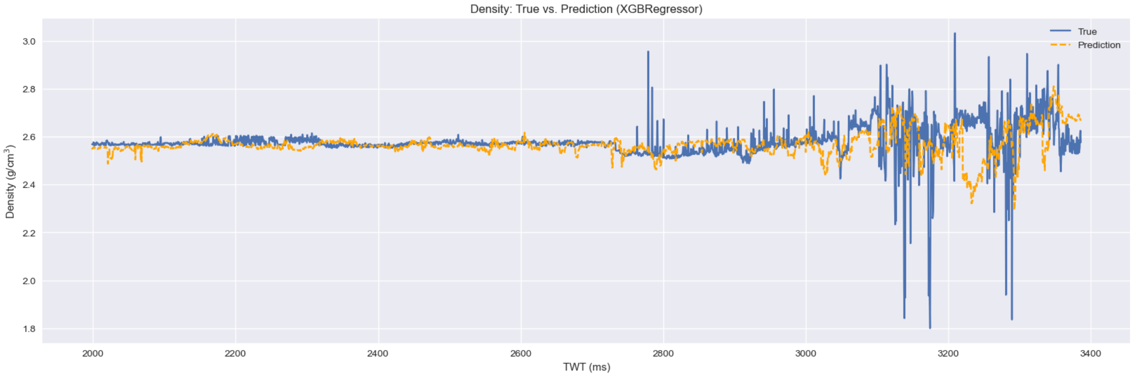

FIG. 9. Density (RHOB) prediction (orange) and true density (blue).

Analyzing the feature importance for the density modeling (Figure 10), the model still gives preference to the trend features, but now the predicted impedances and the far stack trace have increased their value to the model. Rough estimations of density could be extract by combining the background velocity with the P-impedance or the S-impedance, and that's probably what the model is trying to do.

In the next section, the gradient boosting predictions are compared with neural networks solutions.

NEURAL NETWORKS MODELING

Seismic inversion with gradient boosting is showing promising results without any heavy pre-processing of the provided data and results were stable when testing different

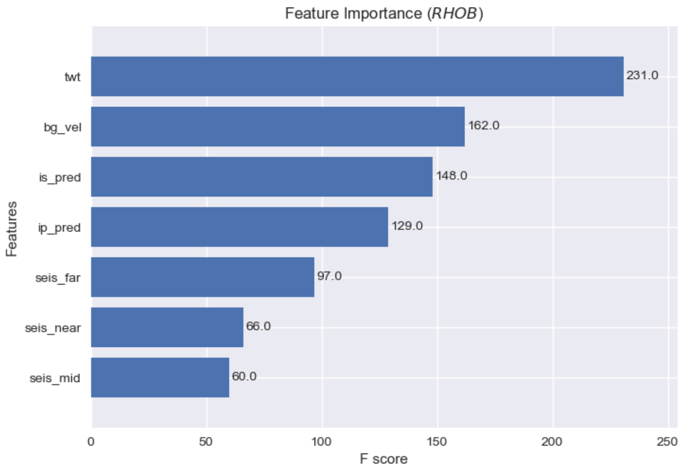

FIG. 10. Feature importance of the density modeling.

model parameters. But how does it compare with neural networks models? A comparison was done by testing fully-connected neural networks algorithms (Hastie et al., 2001), also known as Muilt-layer Perceptron (MLP).

MLP from Scikit-Learn

We trained a MLPRegressor algorithm from Scikit-Learn (Pedregosa et al., 2011) using 3 hidden layers with 20 neurons each, and ReLU as activation function inside the neurons. Figure 11 compares the new prediction (orange) against the XGBoost one (green). The MLP model follows the background impedance (red) closer, not having the flip effect at 2.2s. However, the predictions contain spikes and high frequency content that does not exists in the features (background velocity, seismic traces, and TWT), indicating overfitting.

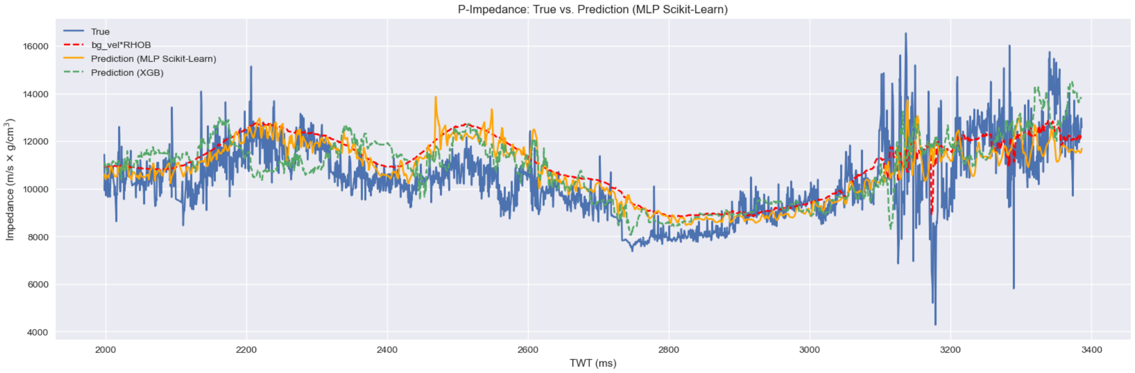

FIG. 11. P-impedance (I_P) prediction using the MLPRegressor from Scikit-Learn (orange), P-impedance prediction from XGBoost (green), true P-impedance (blue), and background P-impedance (red).

Predicting S-impedance (Figure 12, orange) resulted on a noisy trace. It follows close the true S-impedance (blue) trend, but the model gets confused starting at 3.1s. From this sample, the XGBoost prediction (green) has a higher slope than the MLP one. An interesting observation is that starting from 3.1s, the true S-impedance has a low frequency seasonality, and neither of the models could capture it. Rather, they decided to follow the upper (XGBoost) or lower (MLP) trends.

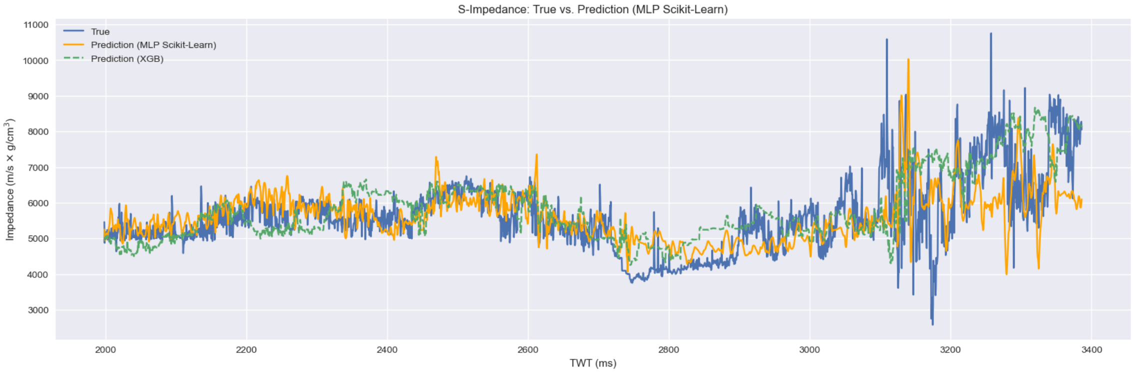

FIG. 12. S-impedance (I_S) prediction using the MLPRegressor from Scikit-Learn (orange), S-impedance prediction from XGBoost (green), and true S-impedance (blue).

We were unsuccessful predicting density using the MLPRegressor. Resulted predictions were dominated by large spikes, and even negative densities were observed.

The MLPRegressor from Scikit-Learn presented promising results for the P and S impedances, but the models were hard to parameterize, and smalls changes caused on highly different outcomes. We believe this is not an algorithm issue, but the amount of data available to train the model may not be the ideal. To optimize the modeling step, we would require more logs from different well locations.

MLP from Tensorflow

We created MLP models in Tensorflow (Abadi et al., 2015) using the same structure as we used for the MLPRegressor in Scikit-Learn. Figure 13 shows the P-impedance (I_P) prediction using Tensorflow (orange), P-impedance prediction from XGBoost (green), true P-impedance (blue), and background P-impedance (red). As for the MLPRegressor, the Tensorflow MLP is noisy and contains even higher amplitudes around 3.1s. It follows the background impedance closer and tends to overestimate when compared with the XGBoost (even with the MLPRegressor).

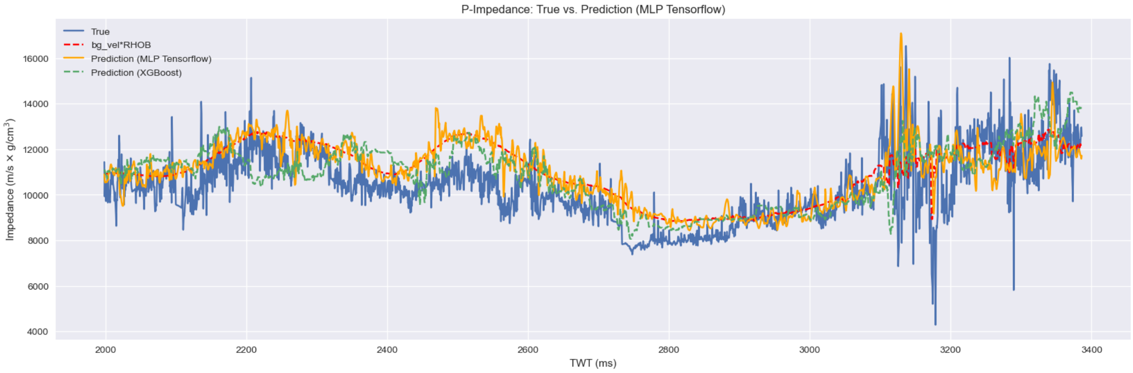

FIG. 13. P-impedance (I_P) prediction using Tensorflow (orange), P-impedance prediction from XGBoost (green), true P-impedance (blue), and background P-impedance (red).

Similar comments can be made when analysing the S-impedance predictions (Figure

14). Predictions (orange) are noisy (indicating overfitting) with high amplitudes around 3.1s.

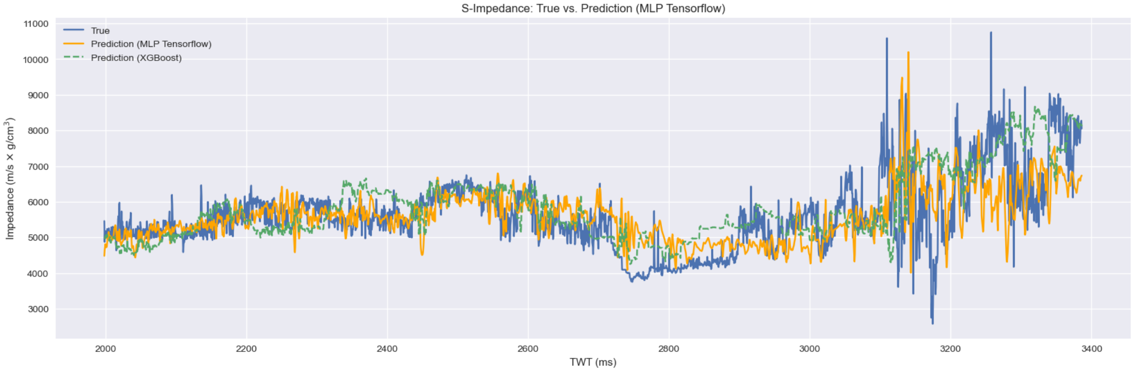

FIG. 14. S-impedance (I_S) prediction using Tensorflow (orange), S-impedance prediction from XGBoost (green), and true S-impedance (blue).

Density predictions (Figure 15) are interesting. The model tends to underestimate the values, but showed large spikes at 3.1s (probably heritage from the P and S impedances predictions).

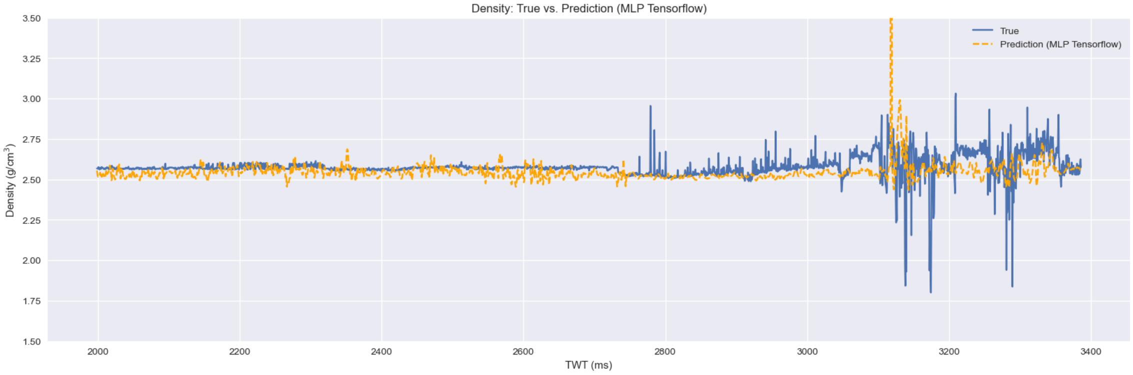

FIG. 15. Density prediction using Tensorflow (orange), using XGBoost (green), and true density (blue).

As for the MLPRegressor from Scikit-Learn, the MLP Tensorflow showed to be difficult to parameterize, with unstable predictions, which we believe is caused by the low amount of training data.

CONCLUSIONS

Geophysics in the Cloud promoted a competition with the proposal to use machine learning algorithms to performs seismic inversion. The provided data showed to be challenging, due the small number of well logs to train the model. During the competition, most of the competitors based their strategies using deep learning models (CNN, RNN, LSTM, etc). Our proposal was to evaluate the possibility of seismic inversion using more classical and simple approaches, hence the choice of the Gradient Boosting algorithm.

Predicting P and S impedances and density showed that the XGBoost model relies on the trend features (TWT and background velocity) to create an overall estimation, and then uses the seismic traces (containers of the lo to mid frequency information) to try to refine the predictions. The model also showed to be stable when changing the parameters, but tends to overfit when choosing trees with larger depths.

By modeling the data using MLP algorithms from Scikit-Learn and Tensorflow, it became clear that the short amount of training data was a challenging issue, as the models were unstable and hard to parameterize. The resulted predictions were noisy, indicating overfitting, and spiky.

In an overall scenario, the XGBoost showed promising results, that could be improved by larger amount of training data. As future proposal, generating synthetic logs can fill this gap.

ACKNOWLEDGEMENTS

We thank the sponsors of CREWES for continued support. This work was funded by CREWES industrial sponsors and NSERC (Natural Science and Engineering Research Council of Canada) through the grant CRDPJ 543578-19. The 1^{th} author was supported by the Canada First Research Excellence Fund, through the Global Research Initiative at the University of Calgary. We thank Soane Mota dos Santos for the suggestions, tips and productive discussions.

REFERENCES

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X., 2015, TensorFlow: Large-scale machine learning on heterogeneous systems, software available from tensorflow.org.
URL https://www.tensorflow.org/

Alvarez, E., Sansal, A., Lasscock, B., Babalola, A., Dennet, L. P., Jones, K., and Kahn, D., 2021, Geophysics in the cloud.
URL https://www.geoscienceml.org/

Araya-Polo, M., Dahlke, T., Frogner, C., Zhang, C., Poggio, T., and Hohl, D., 2017, Automated fault detection without seismic processing: The Leading Edge, **36**, No. 3, 208–214.

Caté, A., Perozzi, L., Gloaguen, E., and Blouin, M., 2017, Machine learning as a tool for geologists: The Leading Edge, **36**, No. 3, 215–219.

Chen, H., Li, J., and Innanen, K. A., 2020, Nonlinear inversion of seismic amplitude variation with offset for an effective stress parameter: GEOPHYSICS, **85**, No. 4, R299–R311.

Chen, T., and Guestrin, C., 2016, Xgboost: a scalable tree boosting system: 22nd SIGKDD Conference on Knowledge Discovery and Data Mining.

Chen, Y., Hill, J., Lei, W., Lefebvre, M., Tromp, J., Bozdag, E., and Komatitsch, D., 2017, Automated time-window selection based on machine learning for full-waveform inversion, SEG Technical Program Expanded Abstracts 2017, 1604–1609.

Chen, Y., and Schuster, G. T., 2020, Seismic inversion by newtonian machine learning: GEOPHYSICS, **85**, No. 4, WA185–WA200.

Emery, D. J., and Guarido, M., 2021, Lessons learned, pitfalls and feature engineering for force 2020: Log facies classification using machine learning: GeoConvention.

Guarido, M., 2019, Machine learning strategies to perform facies classification: GeoConvention Expanded Abstract 2019.

Guarido, M., Emery, D. J., Macquet, M., Trad, D. O., and Innanen, K. A., 2021a, Pitfalls and insights from a machine learning contest on log facies classification, 1671–1675.

Guarido, M., Wozniakowska, P., Emery, D. J., Lume, M., Trad, D. O., and Innanen, K. A., 2021b, Fault detection in seismic volumes using a 2.5D residual neural networks approach, 1626–1629.

Hall, B., 2016, Facies classification using machine learning: The Leading Edge, **35**, No. 10, 906–909.

Hampson, D., 1991, Avo inversion, theory and practice: The Leading Edge, **10**, No. 6, 39–42.

Hastie, T., Tibshirani, R., and Friedman, J., 2001, The elements of statistical learning - data mining, inference, and prediction: Springer, second edn.

Huang, S., Wozniakowska, P., Guarido, M., Emery, D. J., and Trad, D. O., 2021, Long short-term memory application for seismic inversion: CREWES, **33**.

Lewis, W., and Vigh, D., 2017, Deep learning prior models from seismic images for full-waveform inversion, SEG Technical Program Expanded Abstracts 2017, 1512–1517.

Lindseth, R. O., 1979, Synthetic sonic logs-a process for stratigraphic interpretation: Geophysics, **44**, No. 1, 3–26.

Luo, Y., Ma, Y., Wu, Y., Liu, H., and Cao, L., 2016, Full-traveltime inversion: GEOPHYSICS, **81**, No. 5, R261–R274.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E., 2011, Scikit-learn: Machine learning in Python: Journal of Machine Learning Research, **12**, 2825–2830.

Pratt, R. G., Shin, C., and Hick, G. J., 1998, Gauss-newton and full newton methods in frequency-space seismic waveform inversion: Geophysical Journal International, **133**, No. 2, 341–362.

Qu, L., Guarido, M., Trad, D. O., and Emery, D. J., 2021, Data Imputation of Well Logs using Regression Models - Geophysics in the Cloud Competition.
URL `https://www.crewes.org/ResearchLinks/DataScience/summary.php?docid=CDS2020015`

Roy, P., Zhu, X., and Fei, W., 2020, Machine learning assisted seismic inversion, 1705–1709, `https://library.seg.org/doi/pdf/10.1190/segam2020-3425889.1`.
URL `https://library.seg.org/doi/abs/10.1190/segam2020-3425889.1`

Russell, B., Ross, C., and Lines, L., 2002, Neural networks and avo: The Leading Edge, **21**, No. 3, 268–314.

Smith, K., 2017, Machine learning assisted velocity autopicking, SEG Technical Program Expanded Abstracts 2017, 5686–5690.

Sun, J., Niu, Z., Innanen, K. A., Li, J., and Trad, D. O., 2020, A theory-guided deep-learning formulation and optimization of seismic waveform inversion: GEOPHYSICS, **85**, No. 2, R87–R99.

Tarantola, A., 1984, Inversion of seismic reflection data in the acoustic approximation: Geophysics, **49**, No. 8, 1259–1266.

Virieux, J., and Operto, S., 2009, An overview of full-waveform inversion in exploration geophysics: Geophysics, **74**, No. 6, WCC1–WCC26.

Xiong, W., Ji, X., Ma, Y., Wang, Y., BenHassan, N. M., Ali, M. N., and Luo, Y., 2018, Seismic fault detection with convolutional neural network: GEOPHYSICS, **0**, No. ja, 1–28.

Zhang, Z., Mayer, G., Dauvilliers, Y., Plazzi, G., Pizza, F., Fronczek, R., Santamaria, J., Partinen, M., Overeem, S., Peraita-Adrados, M., Silva, A., Sonka, K., Río, R., Heinzer, R., Wierzbicka, A., Young, P., Högl, B., Bassetti, C., Manconi, M., and Khatami, R., 2018, Exploring the clinical features of narcolepsy type 1 versus narcolepsy type 2 from european narcolepsy network database with machine learning: Scientific Reports, **8**.