GEOPHYSICS, VOL. 85, NO. 4 (JULY-AUGUST 2020); P. WA41–WA52, 13 FIGS., 6 TABLES.
10.1190/GEO2019-0405.1

A comparison of deep machine learning and Monte Carlo methods for facies classification from seismic data

Dario Grana[1], Leonardo Azevedo[2], and Mingliang Liu[1]

ABSTRACT

Among the large variety of mathematical and computational methods for estimating reservoir properties such as facies and petrophysical variables from geophysical data, deep machine-learning algorithms have gained significant popularity for their ability to obtain accurate solutions for geophysical inverse problems in which the physical models are partially unknown. Solutions of classification and inversion problems are generally not unique, and uncertainty quantification studies are required to quantify the uncertainty in the model predictions and determine the precision of the results. Probabilistic methods, such as Monte Carlo approaches, provide a reliable approach for capturing the variability of the set of possible models that match the measured data. Here, we focused on the classification of facies from seismic data and benchmarked the performance of three different algorithms: recurrent neural network, Monte Carlo acceptance/rejection sampling, and Markov chain Monte Carlo. We tested and validated these approaches at the well locations by comparing classification predictions to the reference facies profile. The accuracy of the classification results is defined as the mismatch between the predictions and the log facies profile. Our study found that when the training data set of the neural network is large enough and the prior information about the transition probabilities of the facies in the Monte Carlo approach is not informative, machine-learning methods lead to more accurate solutions; however, the uncertainty of the solution might be underestimated. When some prior knowledge of the facies model is available, for example, from nearby wells, Monte Carlo methods provide solutions with similar accuracy to the neural network and allow a more robust quantification of the uncertainty, of the solution.

INTRODUCTION

Facies prediction from geophysical data, such as well logs or seismic surveys, has been approached using several mathematical methods. Here, we assume that a facies definition is provided by geologists based on sedimentology, stratigraphy, and core analysis (Boggs, 2001), and the aim of the application of a classification method is to classify facies from well-log data and seismic data (or inverted seismic properties) within the reservoir model. Standard classification methods classify facies independently at each location. In other words, the facies value predicted at one location is independent of the facies values estimated at adjacent locations. The continuity of the predicted model only depends on the continuity of the measured data. However, facies models must be spatially correlated to mimic the stratigraphy and the depositional processes (Lindberg and Grana, 2015). Similarly, the fluid effect within the facies is controlled by the gravity effect. Spatially correlated models must be imposed to preserve the geologic ordering in facies sequences (Lindberg and Grana, 2015). We compare probabilistic methods with prior spatial correlation models with deep machine-learning methods with spatially correlated training data sets.

We first review the main works in the field of facies classification from seismic data. The most common methods applied for facies classification are pattern recognition and clustering algorithms (Hastie et al., 2002; Martinez and Martinez, 2007). Within these methods, we classify a set of measurements in several clusters that represent different facies. The goal is to classify the measurements such that the similarity between two measurements within each facies is higher than the similarity between two measurements from two different facies. The set of measurements generally consists of elastic and/or petro-

Manuscript received by the Editor 25 June 2019; revised manuscript received 7 October 2019; published ahead of production 28 October 2019; published online 16 January 2020.
[1]University of Wyoming, Laramie, Wyoming, USA. E-mail: dgrana@uwyo.edu (corresponding author); mliu4@uwyo.edu.
[2]CERENA, DECivil, Instituto Superior Técnico, Lisbon, Portugal. E-mail: leonardo.azevedo@tecnico.ulisboa.pt.

physical properties, as well as computed properties estimated from the data. Examples of these methods include, among others, Bayesian classification as in Doyen (2007), statistical facies classifications based on maximum-likelihood criteria as in Da Veiga and Le Ravalec (2012), Monte Carlo classification methods as in Grana et al. (2012), support vector machine algorithms as in Wang et al. (2014), and discriminant analysis approaches as in Negahdari et al. (2014). Markov chain Monte Carlo (MCMC) methods have also been proposed to integrate spatial statistics models, such as Markov models in the inversion (Lindberg and Grana, 2015). Several comparisons between classification methods have also been proposed. Other classification methods such as fuzzy logic, k-nearest neighbor, and artificial neural network have been proposed and compared (Dubois et al., 2007). Wong et al. (1995) compare the performance of neural networks and discriminant analysis, whereas Li and Anderson-Sprecher (2006) compare discriminant analysis and Bayesian classification. Such methods have been applied in different depositional environments such as the marine turbiditic system (Chen and Hiscott, 1999) or the Marcellus gas shale reservoir (Wang and Carr, 2012). Some of these methods have been initially developed for well-log classification and have been extended to inverted seismic properties or related seismic attributes (Mukerji et al., 2001; Avseth et al., 2005; Doyen, 2007).

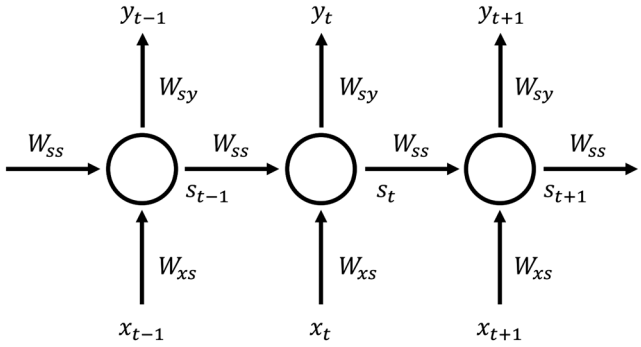

Figure 1. Recurrent neural network unit.

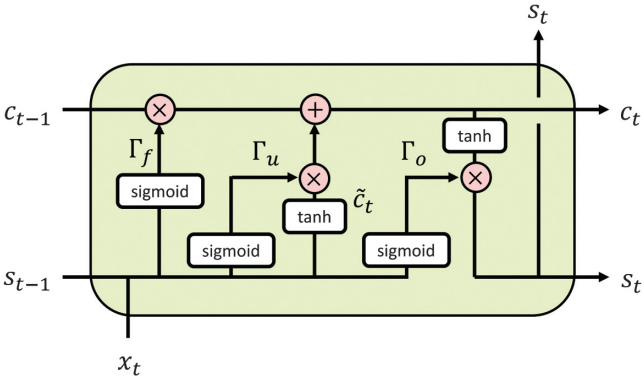

Figure 2. Long short-term memory unit.

The development of machine-learning methods, especially those based on deep learning techniques, constitutes a major advancement in facies classification studies. The applications of machine learning in geophysics also include seismic processing and inversion studies. Preliminary works in the field include those by Strecker and Uden (2002), Caers and Ma (2002), West et al. (2002), Saggaf et al. (2003), Coléou et al. (2003), Tartakovsky and Wohlberg (2004), and de Matos et al. (2006). Recent developments include, among others, the findings of Roy et al. (2013), Guillen et al. (2015), Zhao et al. (2015), Hall (2016), Zhang and Zhan (2017), Bestagini et al. (2017), Huang et al. (2017), Hall and Hall (2017), Maniar et al. (2018), Wu et al. (2018), Li (2018), and Jin et al. (2018). Uncertainty assessment in machine-learning predictions using recurrent neural networks have been studied in recent publications especially in computer science and engineering applications (Lin et al., 2002; Park et al., 2008; Cong and Liang, 2009; Graf et al., 2010), but applications to facies classification and geologic modeling is still ongoing research.

Despite recent advances, some challenges in facies classification remain open. In particular, the assessment of the uncertainty of the estimated model is challenging due to the nonuniqueness of the solution of the classification and inversion problem. Furthermore, preserving the geologic meaning of the spatial distribution of the facies, such as the vertical transitions mimicking the geologic deposition, is difficult due to the lack of explicit spatial statistics models. Examples of statistical models include variogram models (Doyen, 2007) and the Markov chain (Lindberg and Grana, 2015). Facies classification with underlying spatial statistics models leads to larger computational costs compared to standard pointwise classification and also requires assumptions on the parameters of the statistical models, such as the variogram length or the prior transition probabilities. The assessment of the uncertainty under these statistical assumptions then requires a large number of statistical realizations. Deep machine-learning methods aim to "learn" the intrinsic spatial relations within the facies distribution from a training data set, either a set of measurements or a data set predicted with physical models.

Here, we propose a comparison of facies classification algorithms using machine-learning approaches, based on a long short-term recurrent neural network (Mikolov et al., 2010), and Monte Carlo methods, including acceptance-rejection sampling and MCMC methods (Gilks et al., 1995). We evaluate the performance of the methods in terms of computational efficiency and the quality of the results based on accuracy and precision. Accuracy is a measure of statistical bias between measured data and prediction. Precision is a

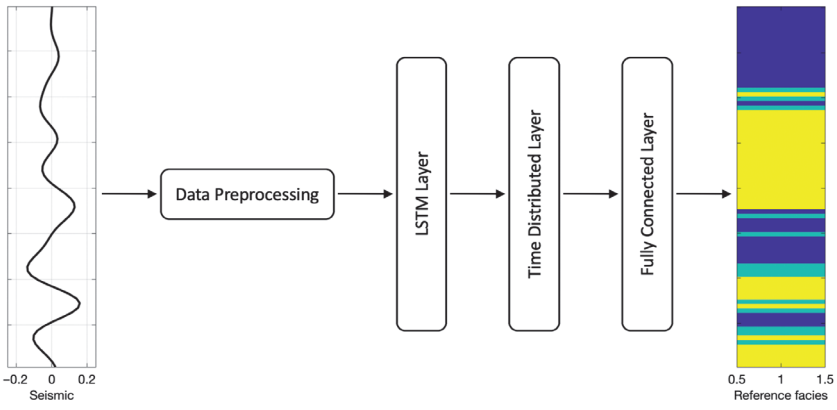

Figure 3. Schematic workflow of the proposed RNN model for facies classification from seismic data.

measure of statistical variability among the predictions. High accuracy means that the prediction is close to the true value, in our case, the reference facies profile, whereas high precision means that the variability between multiple predictions is small. In some cases, predictions might show high accuracy but low precision (i.e., the mode of the predictions is close to the true value, but the variability is large) or low accuracy and high precision (i.e., the variability is small, but the predictions are far from the true value). We point out that high precision might also mean that the uncertainty in the predictions is underestimated and a potential solution might not be captured by the algorithms, which may occur when the predictor after training is overfitted. The methods are tested and validated using a benchmark data set. We show that machine learning and Monte Carlo methods provide accurate and geologically consistent results. In terms of accuracy, machine-learning algorithm provides better results if the training data set is large enough. Monte Carlo methods based on Markov models provide accurate results if prior information is available, such as the transition probabilities of the facies. In terms of uncertainty quantification, machine-learning methods can underestimate the variability of the solution if the training data set is not large enough.

METHODOLOGY

Recurrent neural networks

Recurrent neural networks (RNNs) are a specialized class of deep neural networks for processing sequential data such as audio signal and text data. Unlike feed-forward neural networks for example, the multiple-layer perceptron and convolutional neural network (CNN), RNNs allow information cycles through a feedback loop that takes into consideration the current input and also what was learned from previous time steps. Such feedback loops enable them to capture long temporal dependencies in sequential data and allow an exhaustive analysis of time series data such as seismic reflection data. Here, we propose an RNN-based method for facies prediction from seismic data.

RNNs use a hidden state serving as memory to allow information to be passed between time steps (Figure 1). We denote the input and output sequence with length T using the variables $x = [x_1, x_2, \ldots, x_t, \ldots, x_T]$ and $y = [y_1, y_2, \ldots, y_t, \ldots, y_T]$, respectively. At the specific time step t, the hidden state and output of RNN are

$$s_t = g(W_{ss}s_{t-1} + W_{xs}x_t + b_s), \quad (1)$$

$$y_t = g(W_{sy}s_t + b_y), \quad (2)$$

where s_t is the hidden state; W_{xs}, W_{ss}, and W_{sy} are the trainable weights for the input-to-hidden, hidden-to-hidden, and hidden-to-output connections, respectively; b_s and b_y are the bias vectors for hidden state and output, respectively; and g is the activation function.

RNNs are theoretically capable of learning long-term dependencies from the hidden states, but in practical applications, they might fail when training with long sequences due to numerical issues with the gradient calculation (Pascanu et al., 2013). To solve this limitation, improved RNNs using gated units such as long short-term memory (LSTM) (Sak et al., 2014) and gated recurrent units

Table 1. True transition matrix.

	Facies 1	Facies 2	Facies 3
Facies 1	0.81	0.19	0
Facies 2	0.31	0.23	0.46
Facies 3	0.03	0.12	0.85

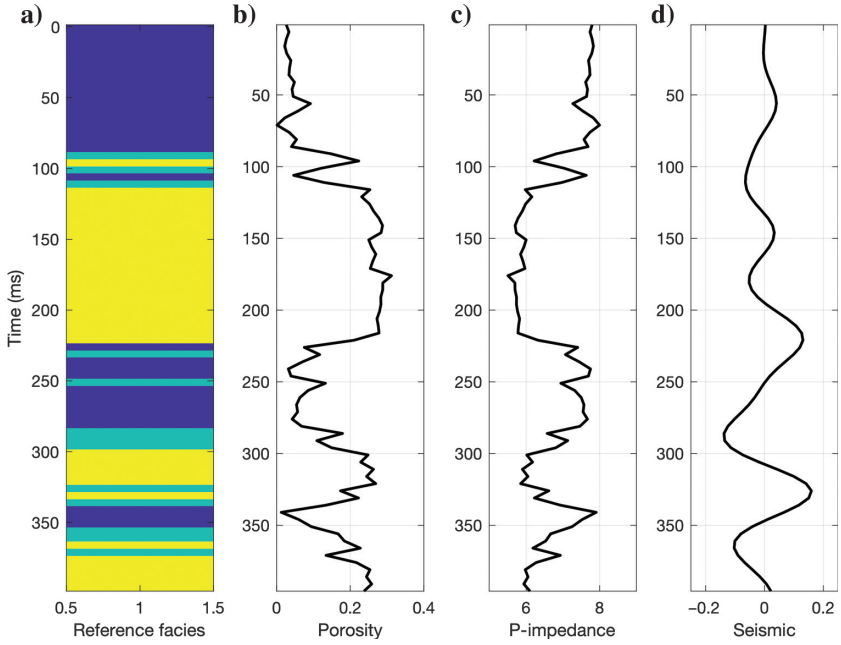

Figure 4. Synthetic well-log data: (a) facies classification, (b) porosity, (c) P-impedance, and (d) seismic amplitudes.

Table 2. Simulated transition matrices (realizations 1, 5000, and 10,000).

	Facies 1	Facies 2	Facies 3
	Realization 1		
Facies 1	0.60	0.16	0.24
Facies 2	0.41	0.29	0.30
Facies 3	0.53	0.11	0.36
	Realization 5000		
Facies 1	0.36	0.17	0.47
Facies 2	0.08	0.26	0.66
Facies 3	0.18	0.28	0.54
	Realization 10000		
Facies 1	0.14	0.84	0.02
Facies 2	0.21	0.59	0.20
Facies 3	0.16	0.29	0.55

(Chung et al., 2014, 2015) have been developed with satisfactory results in the field of speech recognition. Here, we adopt LSTM because it is suitable for data sequences.

LSTM introduces a mechanism known as cell states with the update gate, forget gate, and output gate (Figure 2). The value of the cell state at each time step (c_t) is determined by the candidate value at the current time step $(\tilde{c}_t)$ and the value at the previous time step (c_{t-1}) with the update and forget gate , and the activation output is then regulated by the output gate :

$$\tilde{c}_t = g(W_{cs}s_{t-1} + W_{cx}x_t + b_c), \tag{3}$$

$$\Gamma_u = \sigma(W_{us}s_{t-1} + W_{ux}x_t + b_u), \tag{4}$$

$$\Gamma_u = \sigma(W_{us}s_{t-1} + W_{ux}x_t + b_u), \tag{5}$$

$$\Gamma_f = \sigma(W_{fs}s_{t-1} + W_{fx}x_t + b_f), \tag{6}$$

$$\Gamma_f = \sigma(W_{fs}s_{t-1} + W_{fx}x_t + b_f), \tag{7}$$

$$s_t = \Gamma_o c_t, \tag{8}$$

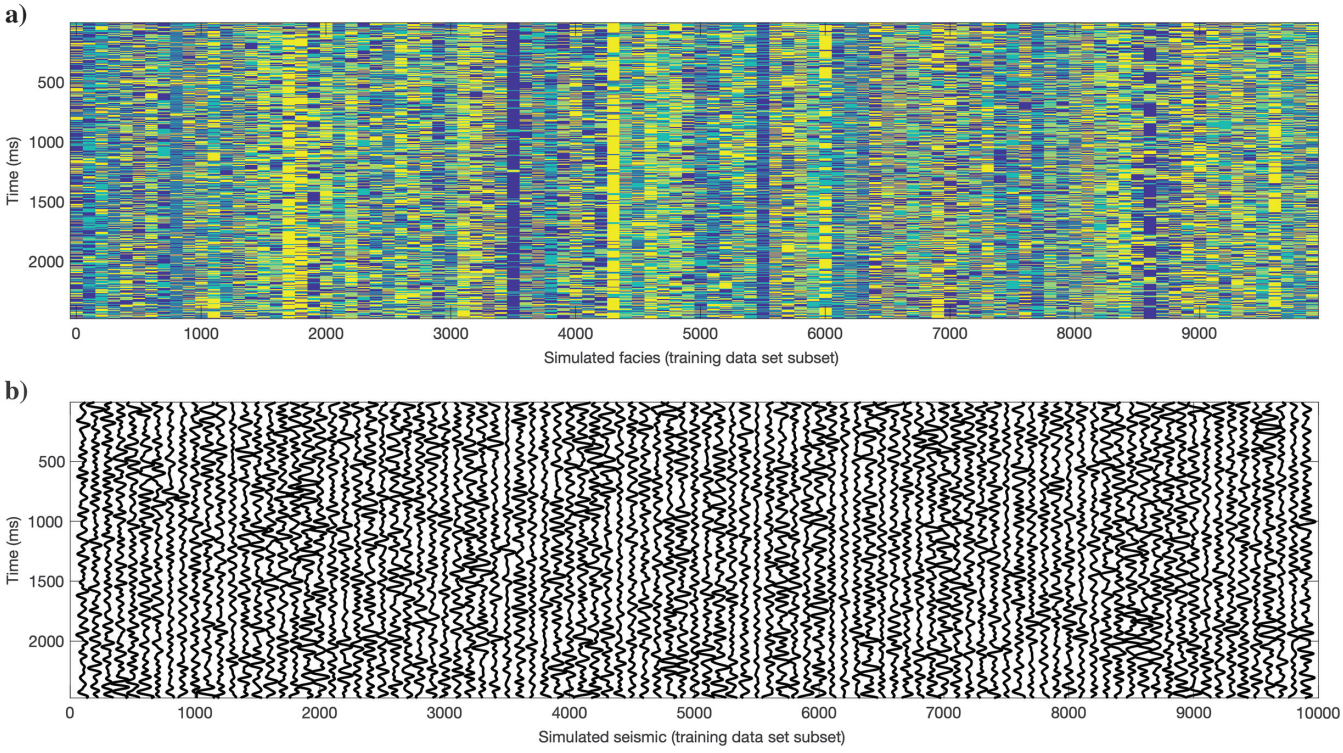

Figure 5. Training data set: subset of 100 simulations (extracted from the full data set by selecting 1 realization of every 100) of (a) facies realizations and (b) the corresponding seismic response.

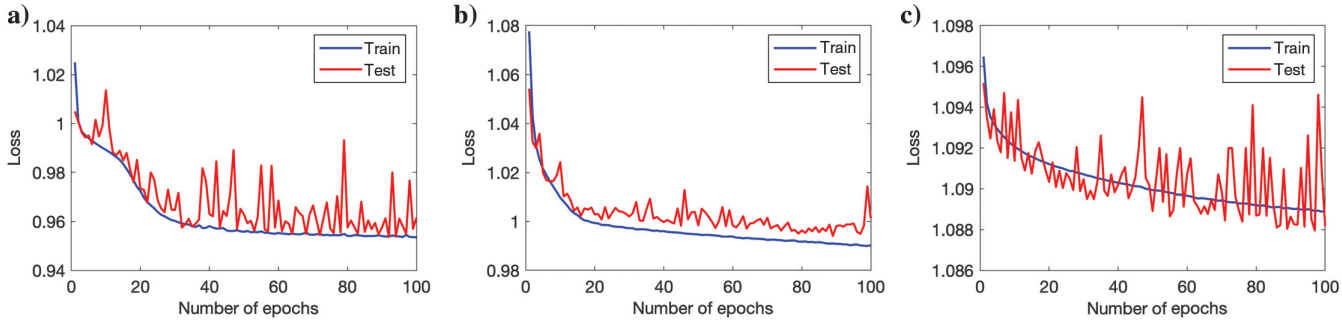

Figure 6. Training and test loss: (a) case ML1, (b) case ML2, and (c) case ML3.

where **W** and **b** are the trainable weight matrices and bias vectors, and σ is the sigmoid function with output between 0 and 1.

The proposed model consists of an LSTM layer, a time distributed layer, and a fully connected layer with SoftMax as the activation function (Figure 3). The SoftMax function turns logits into probabilities that sum to one and outputs a vector that represents the probability distributions of a list of potential outcomes (Jang et al., 2016). The LSTM layer is used to capture the long temporal dependencies in seismic traces. With the time distributed wrapper, we can apply the fully connected layer to every temporal slice of an input independently rather than being entangled with each other.

Monte Carlo methods

Monte Carlo methods include a broad set of algorithms that aim to numerically approximate a probability density function using repeated random sampling. In inverse theory, Monte Carlo methods are generally used to approximate the posterior probability of the model variables conditioned by the data by iteratively sampling from a proposal distribution and evaluating the likelihood of observing the data given the drawn model values (Mosegaard and Tarantola, 1995; Sen and Stoffa, 1996; Mosegaard, 1998; Sambridge and Mosegaard, 2002). Here, we adopt two different approaches: Monte Carlo acceptance/rejection sampling and an MCMC algorithm.

The Monte Carlo acceptance/rejection sampling consists of repeated random sampling from a prior distribution and accepting or rejecting the drawn model by evaluating the similarity between the model predictions and observed data. In the proposed implementation, the sampling of the vertical profile of the facies is based on stationary first-order Markov chains (Krumbein and Dacey, 1969; Elfeki and Dekking, 2001; Lindberg and Grana, 2015; Fjeldstad and Omre, 2017). A Markov chain is a stochastic process in which the probability distribution of the next state of the process is conditioned by the previous states of the process, rather than the entire sequence of previous states. The order of the Markov chain indicates the number of previous states in the conditional distribution. Categorical Markov chains represent a special case defined in a discrete-valued state space; hence, each state belongs to a finite number of n classes. In the facies simulation, the state is the facies value at a given location; therefore, at each location, the probability of a given facies depends only on the facies at the previous location. The probabilities of transitioning from a given state to another one, that is, the transition probabilities, are generally represented in a square matrix, that is, the transition matrix, where the rows represent the previous states and the columns represent the current states. In the facies simulation, the element (i, j) of the transition matrix represents the probability of a transition from facies i located above the interface to facies j located below. The elements of the transition matrix are commonly assumed to be known because they can be estimated from available data (i.e., facies profiles) and most of the sampling algorithms assume that they are constant in the entire reservoir. However, these parameters could vary in the reservoir and the estimates at the well locations might be biased by subjective data interpretations. Hence, we propose a Monte Carlo algorithm based on two sequential sampling components: (1) We sample the transition matrix from a uniform distribution of facies proportions, and (2) we sample a facies sequence assuming the transition matrix sampled in the first step. The description of the sampling algorithm is provided in Appendix A. Alternatively, we propose an MCMC method based on a Metropolis algorithm. In this approach,

Table 3. Predicted transition matrices (ML1, ML2, and ML3).

	Facies 1	Facies 2	Facies 3
		ML1	
Facies 1	0.81	0.15	0.04
Facies 2	0.12	0.76	0.12
Facies 3	0.06	0	0.94
		ML2	
Facies 1	0.84	0.08	0.08
Facies 2	0.05	0.90	0.05
Facies 3	0.06	0	0.94
		ML3	
Facies 1	0.90	0	0.10
Facies 2	0.05	0.96	0.04
Facies 3	0.06	0	0.94

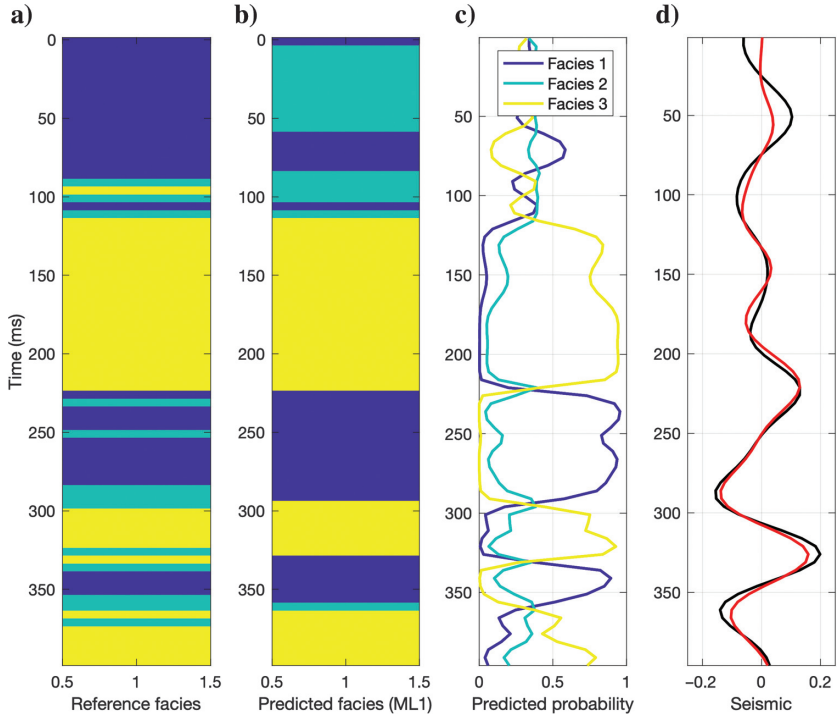

Figure 7. Inversion results using RNN with the full training data set (case ML1): (a) reference facies classification, (b) predicted facies, (c) predicted facies probability, and (d) seismic response (the predicted data in black versus the reference data in red).

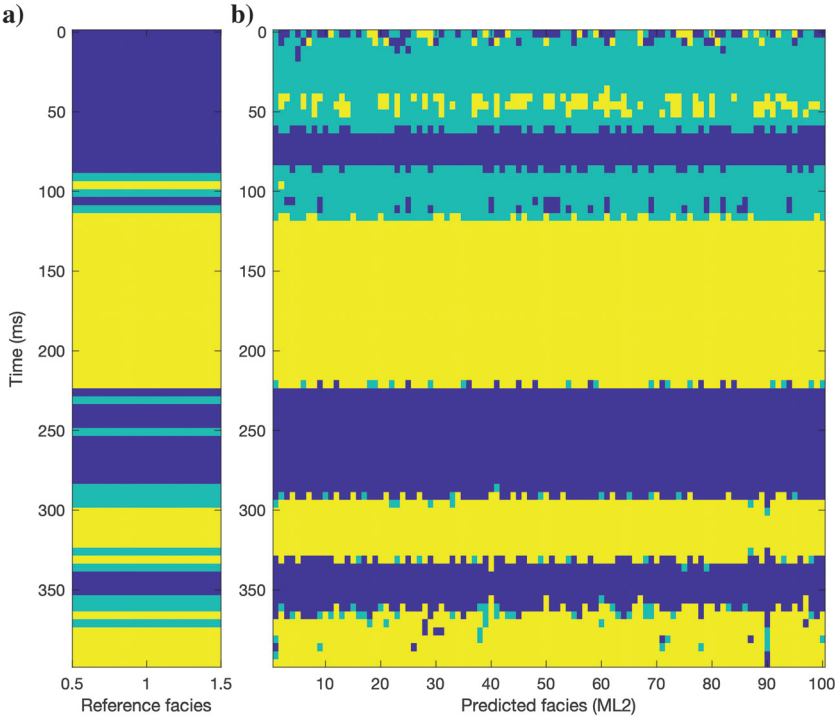

Figure 8. Inversion results (100 runs) using RNN with subsets of the full training data set (case ML2): (a) reference facies classification and (b) 100 profiles of the predicted facies.

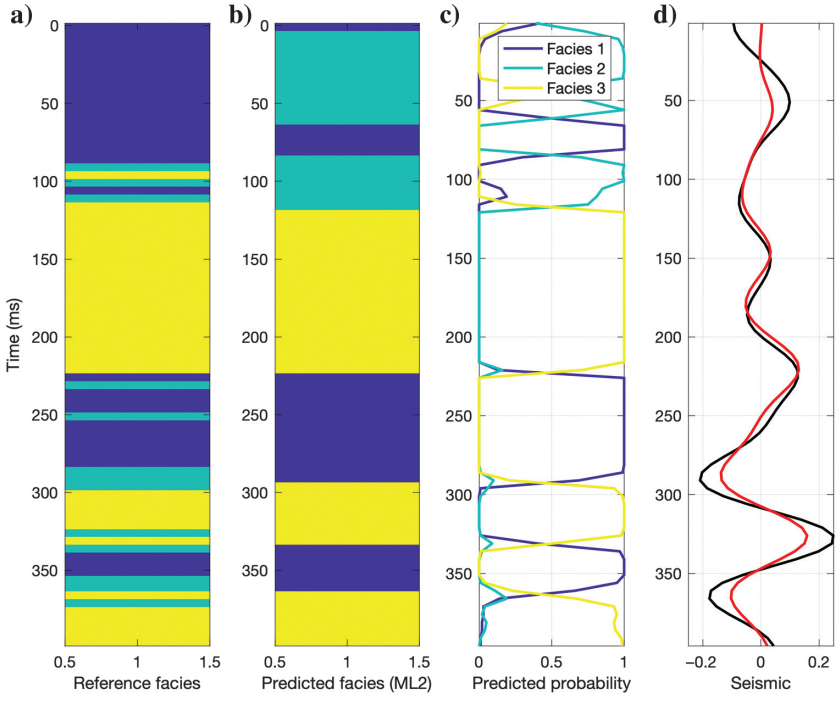

Figure 9. Inversion results (100 runs) using RNN with subsets of the full training data set (case ML2): (a) reference facies classification, (b) predicted facies, (c) predicted facies probability, and (d) seismic response (the predicted data in black versus the reference data in red).

we assume a given transition matrix, then, at each iteration, we sample a facies realization from the given transition matrix and compute the likelihood of the predicted data given the observed data based on the correlation coefficient between the two seismic traces. We then propose a new realization by sampling from a proposal distribution and iterate until convergence. The details of the algorithms are given in Appendix A.

APPLICATION

The methods presented in the "Methodology" section are tested on a 1D synthetic data set. The data set includes a reference facies model with $N = 80$ samples and a seismogram corresponding to a zero-offset trace (Figure 4). The facies model represents three different rock types (e.g., stiff-sand, soft-sand, and shale). For simplicity, we refer to these as facies 1 (blue), facies 2 (green), and facies 3 (yellow). The true transition matrix is estimated from the data by counting the number of transitions and normalizing by the total number of transitions ($N - 1$), and it is shown in Table 1. The true seismogram is generated using a convolutional model with a known Ricker wavelet, with a dominant frequency of 30 Hz, and the series of reflection coefficients. To compute the reflection coefficients, we first sampled a porosity value from a univariate Gaussian mixture model with three facies-dependent components and computed the P-impedance and the corresponding reflectivity using a linearized rock physics model. The Gaussian distributions of porosity in each facies are assumed to be very narrow to avoid overlaps of the values between the different facies: Facies 1 is a low-porosity rock (between 0 and 0.1), Facies 2 is a mid-porosity rock (from 0.1 to 0.2), and Facies 3 is a high-porosity rock (from 0.2 to 0.3). Figure 4 shows the facies profile and the corresponding porosity, impedances, and seismic amplitudes plots.

We first apply the RNN method. We build a training data set of $s = 10{,}000$ realizations of facies using the abovementioned sampling method: We sample s transition matrices from independent uniform distributions with the constraints that the sum of each row is equal to 1, and we generate one realization for each transition matrix. Table 2 shows three different examples of transition matrices, corresponding to the realizations 1 (with prior proportions [0.55, 0.17, 0.28]), 5000 (with proportions [0.19, 0.25, 0.56]), and 10,000 (with proportions [0.18, 0.56, 0.26]). Figure 5 shows a subset of 100 realizations extracted from the full data set by selecting one realization from every 100. As discussed in the "Methodology" section, in each facies, we sample a porosity value, compute the P-impedance and zero-offset reflectivity,

and estimate the corresponding seismograms using a convolutional model. The 100 seismograms are also shown in Figure 5. The so-obtained facies profiles and seismic traces constitute the training data set of the neural network.

We then apply the RNN method to the reference seismogram in Figure 4, according to three different cases: ML1, we use the entire training data set of 10,000 realizations to estimate the most-likely facies profile; ML2, we run the algorithm 100 times using 100 subsets of 1000 realizations randomly selected in the full training data set to estimate 100 most-likely facies profiles; and ML3, we use the probability distributions used to generate the full probability distributions (rather than the realizations) and estimate the most likely facies. All three cases use the same RNN model with only one LSTM layer of 50 units. The optimizer adopted is the RMSprop (Tieleman and Hinton, 2012) with a learning rate of 0.001 and a moving average parameter of 0.9. In each case, we randomly split the data set into two subsets: 90% as the training set and 10% as the validation set. The training and test loss of the three cases are shown in Figure 6. To minimize the probability of overfitting, we chose the models around epoch 30 as the ideal model for prediction.

The results of case ML1 are shown in Figure 7. The predicted facies profile matches the reference facies accurately well, except for the top layer where we observe a mismatch due to the proximity of the top boundary where there are no data above the interface. Some of the thin layers in the lower interval are also not predicted due to the limited resolution of the data. The algorithm also returns the probability of the facies and the predicted response. The results show a good agreement with the reference model and data in terms of the mismatch between predictions and measurements. The transition matrix estimated from the predicted data is shown in Table 3: The rows corresponding to facies 1 and 3 are close to the true ones, whereas the transition from facies 2 to facies 2 is overestimated. Indeed, the predicted realization shows thicker layers of facies 2 than the true facies profiles with sub-seismic resolution layers.

We then randomly select 100 subsets of 1000 facies realizations and the corresponding seismograms from the training data set and run the same algorithms 100 times (case ML2). Each run assumes a training data set of 1000 samples, which is more realistic than the large training data set used in case ML1. The results of case ML2, that is, the set of 100 most likely facies, one per run, is shown in Figure 8. The mode of the predicted realizations is shown in Figure 9 together with the predicted probability and the model response. All runs converge to a similar solution that resembles the most likely model obtained in case ML1 (Figure 7). This suggests that a smaller training data set (1000 realizations instead of 10,000) might be enough to retrieve accurate results with a smaller computational cost associated with the training of the RNN. However, the predicted probabilities show values closer to the bounds of the interval, 0 or 1, and are less smooth than the case ML2, which means that the uncertainty in the prediction is underestimated compared to case ML1 with the full training data set. The number of runs does not affect the results because all of the realizations converge to similar solutions. Furthermore, the estimated transition matrix (Table 3) is less accurate due to the overestimation of the probabilities on the diagonal.

Finally, we run the RNN method using the facies probabilities as input rather than the facies realizations using the SoftMax function to convert the facies values into probabilities. The results are shown in Figure 10 and Table 3. Among the three cases using RNNs, these results are the lowest in terms of accuracy and precision. Indeed, the number of samples predicted in the correct facies decreases (Figure 10) the probabilities on the diagonal of the transition matrix increase and overestimate the true ones (Table 3). In this example,

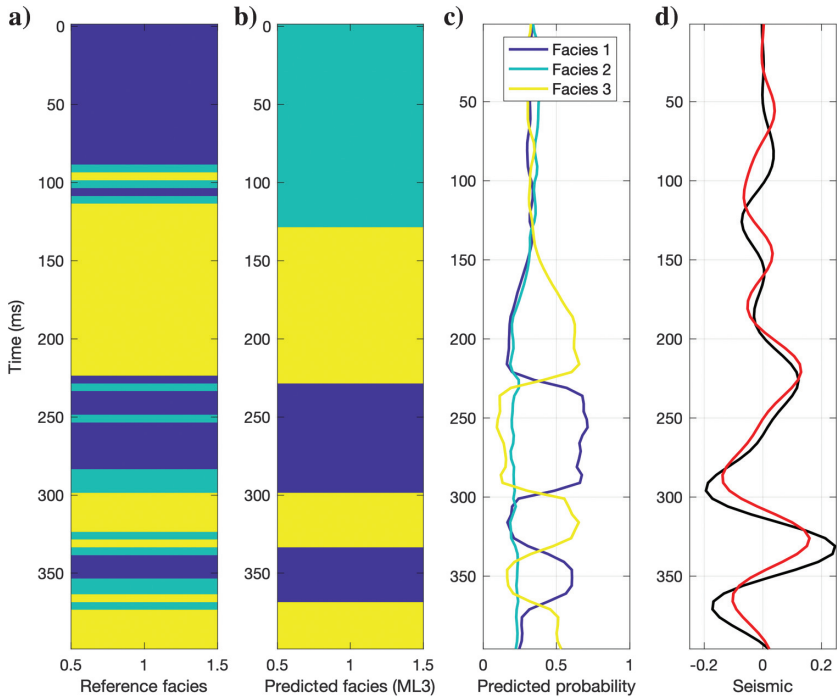

Figure 10. Inversion results using RNN using facies probability distributions (case ML3): (a) reference facies classification, (b) predicted facies, (c) predicted facies probability, and (d) seismic response (the predicted data in black versus the reference data in red).

Table 4. Predicted transition matrices (MC and MCMC).

	Facies 1	Facies 2	Facies 3
	MC		
Facies 1	0.71	0.23	0.06
Facies 2	0.44	0.28	0.28
Facies 3	0.08	0.15	0.77
	MCMC		
Facies 1	0.64	0.20	0.16
Facies 2	0.19	0.57	0.24
Facies 3	0.15	0.15	0.70

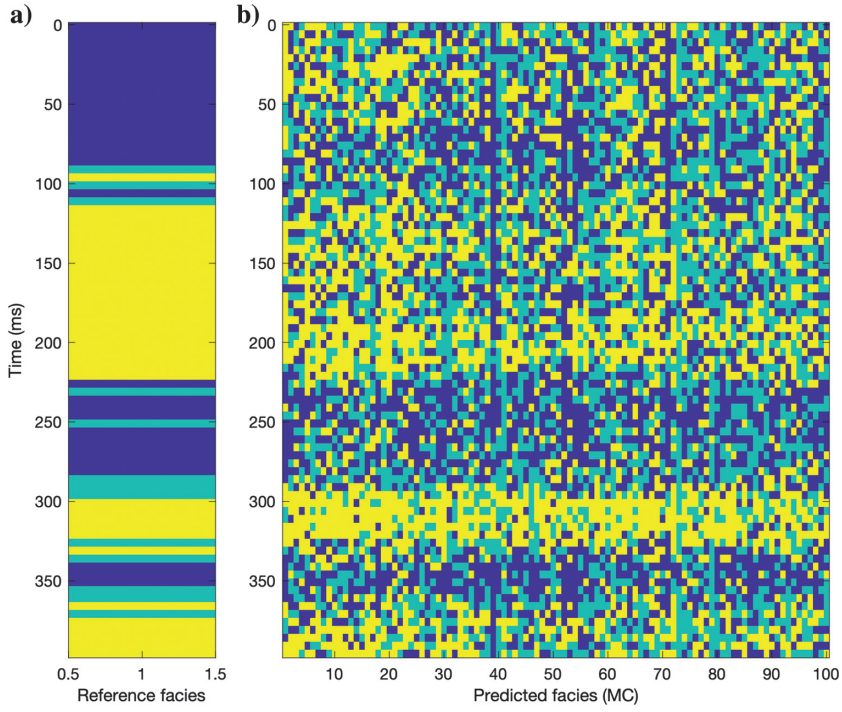

Figure 11. Inversion results using Monte Carlo acceptance/rejection sampling with 10,000 simulations (case MC): (a) reference facies classification and (b) 100 profiles of the predicted facies.

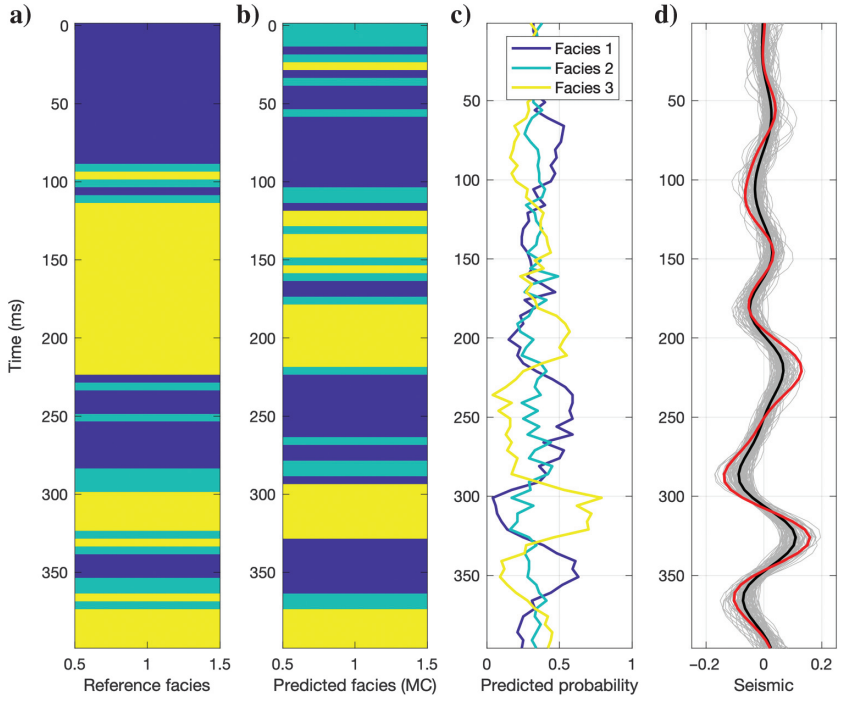

Figure 12. Inversion results using Monte Carlo acceptance/rejection sampling with 10,000 simulations (case MC): (a) reference facies classification, (b) predicted facies, (c) predicted facies probability, and (d) seismic response (the predicted data in black versus the reference data in red).

the RNN method works better with categorical variables rather than continuous properties.

We then apply Monte Carlo methods and compare the results. We first apply the acceptance/rejection sampling. We generate a set of facies realizations based on first-order Markov chains with a randomly generated transition matrix and compute the seismic response according to the same approach used to generate the training data set. To guarantee that the facies profile is representative of the transition matrix, we generate profiles with 500 samples (i.e., more than six times longer than the actual data). After generating the seismogram corresponding to the seismic response, we then define a moving window of the size of the wavelet and select the interval in the proposed seismic trace with the highest correlation coefficient with the measured data. If the correlation coefficient is higher than 0.9, we accept the proposed facies profile; otherwise, we reject it. We can also rank all of the realizations based on their correlation. In Figure 11, we show the 100 best realizations with a correlation coefficient between 0.92 and 0.95. Higher correlations can be achieved using a larger number of initial realizations: 100,000 realizations to achieve correlations greater than 0.95 and 1,000,000 realizations to achieve correlations greater than 0.96. We compute the mode of the accepted realizations and display the so-obtained profile in Figure 12. Different from RNN methods, thin layers are predicted by this approach, but the locations of the thin layers are sometimes misclassified. Facies profiles with thin layers interbedded in thick layers generally produce a seismic response similar to profiles with thick homogeneous layers, when the interbedded layers are below the seismic resolution. However, both models are solutions of the inverse problem, and this variety of solution is captured by Monte Carlo acceptance/rejection sampling. The probabilities show the larger uncertainty in the Monte Carlo solution compared to the RNN method; that is, the predicted probabilities are not close to the bounds of the interval [0,1]. The transition matrix estimated using our proposed approach is close to the true value. In particular, despite the misclassification of the positions of the facies 2 layers, the probability of transitioning in facies 2 is more accurate than the previous cases. Table 4 shows the transition probability of the realization with the highest correlation with measured seismic.

We finally run the MCMC methods with the prior transition matrix equal to the transition matrices of the best realization obtained in the Monte Carlo acceptance/rejection sampling. The proposed distribution is based on equations A-1 and A-2. In Figure 13, we show the most likely model obtained from the MCMC samples. The acceptance rate of the algorithm is relatively low, approximately 0.09,

probably due to the constraints included in equations A-1 and A-2, to preserve the spatial feature of the prior model. Table 4 shows the posterior transition matrix. Overall, the results of MCMC are comparable to the Monte Carlo approach, but to capture the uncertainty in the transition matrix, it is necessary to run several independent chains. The contingency matrices for the different cases are shown in Table 5. A contingency matrix is a table that quantifies the accuracy of a classification method on a set of test data for which the true values are known, by counting the number of samples belonging to a given group classified in each of the possible groups. The perfect classifications are represented as "target" in Table 5. The contingency matrices support the above-discussed results; however, we point out that it only provides statistics about the pointwise mismatch and does not account for the spatial distribution of the errors. In other words, the misclassification rates in the matrix do not provide information about the correct estimation of spatial patterns such as thickness and interbedded layers.

Table 5. Contingency matrix for all the inversion cases.

	Predicted Facies 1	Predicted Facies 2	Predicted Facies 3
True Facies 1	Target — 32	Target — 0	Target — 0
	ML1 — 20	ML1 — 12	ML1 — 0
	ML2 — 19	ML2 — 13	ML2 — 0
	ML3 — 12	ML3 — 19	ML3 — 1
	MC — 22	MC — 9	MC — 1
	MCMC — 29	MCMC — 3	MCMC — 0
True Facies 2	Target — 0	Target — 13	Target — 0
	ML1 — 6	ML1 — 4	ML1 — 3
	ML2 — 7	ML2 — 3	ML2 — 3
	ML3 — 8	ML3 — 3	ML3 — 2
	MC — 8	MC — 3	MC — 2
	MCMC — 8	MCMC — 2	MCMC — 3
True Facies 3	Target — 0	Target — 0	Target — 35
	ML1 — 1	ML1 — 1	ML1 — 33
	ML2 — 1	ML2 — 4	ML2 — 30
	ML3 — 1	ML3 — 4	ML3 — 30
	MC — 5	MC — 6	MC — 24
	MCMC — 3	MCMC — 7	MCMC — 25

DISCUSSION

Table 6 shows the comparison of the computational costs (CPU: Intel Core i5-4570 @ 3.20 GHz; RAM: 16 GB) of the different methods applied in this study. The cost of the RNN method with the full training data set of 10,000 realizations is approximately 4 h. As observed in the "Application" section, for an accurate solution, a smaller training data set than the one adopted in this study could be used. Indeed, all 100 runs in case ML2 use a training data set of 1000 realizations and converge to similar solutions, close to the true model. However, for a better interpretation of the uncertainty and to avoid overfitting, a large data set is suggested, because it might include a broader set of geologic features that might be observable in the field. The cost of the MC method with a small data set is a few minutes; in this case, uncertainty could be captured, but the most likely model might be inaccurate and the correlation between measured data and predictions might not be optimal. The cost of the MC method increases linearly with the number of simulations. To obtain more accurate and precise solutions, we tested several simulations between 10^4 and 10^6; however, 1,000,000 realizations for a simple 1D example still require several hours. The MCMC method with one chain provides a good compromise in terms of accuracy and computational cost; however, testing several transition matrices would require several runs with different chains, increasing the computational cost of this approach.

The choice of some initial parameters might affect the results. In theory, Monte Carlo acceptance/rejection sampling provides the most accurate and precise result, but it might require an unfeasibly large number of simulations. The trade-off between accuracy/precision and computational cost might impact the conclusions of the study. The choice of evaluation criteria is also critical: Some methods provide less accurate solutions but with more realistic geologic features (e.g., transitions from thin layers to thick layers), some others provide the best solution in the least-

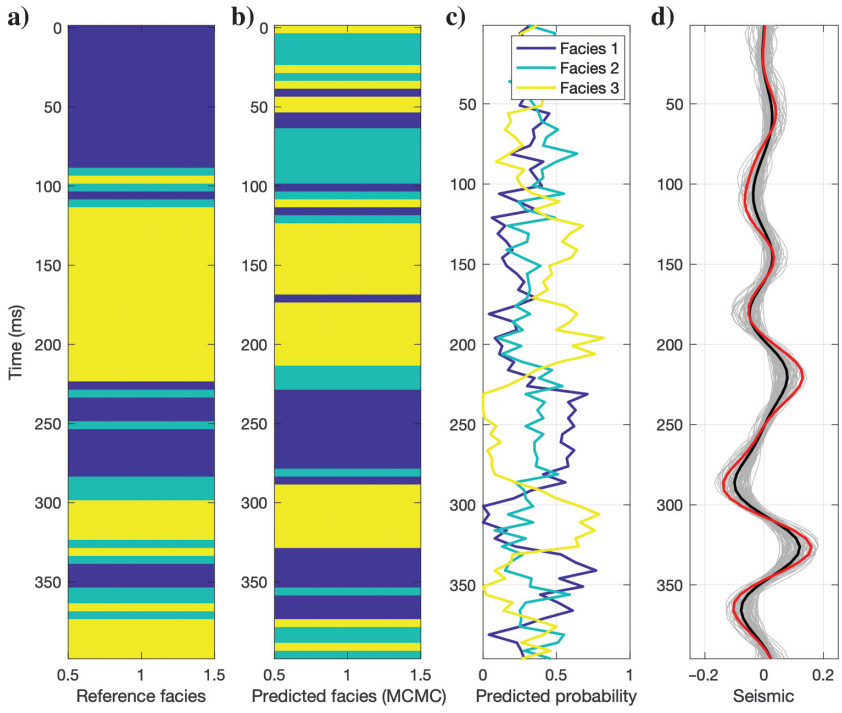

Figure 13. Inversion results using the Markov chain Monte Carlo (case MCMC): (a) reference facies classification, (b) predicted facies, (c) predicted facies probability, and (d) seismic response (the predicted data in black versus the reference data in red).

squares sense, and others might underestimate the uncertainty. How to assess accuracy and precision is also subjective. Optimal solutions in terms of least-squares error might not show the expected heterogeneity and might differ from optimal solutions in terms of correlation. If we replace the correlation measure for the mismatch with the L1 or L2 norm, some of the conclusions of this study might be different. Similarly, if we implement different variations of the method (Metropolis-Hastings for the MCMC study or CNN for the machine-learning approach), the results might be slightly different. Overall, probabilistic methods based on MCMC provide a better quantification of the uncertainty, whereas machine-learning algorithms provide a more accurate solution in terms of data mismatch and model validation with the training data set.

The proposed methods are more computationally demanding than standard facies classification methods that do not account for the spatial correlation of facies. However, pointwise facies classification methods generally lead to geologically unrealistic facies transitions and frequencies. Instead, the proposed methods impose a spatial correlation model, in the form of a prior statistical model (for the Monte Carlo method) or as a spatially correlated training data set (for machine-learning algorithms). Spatially correlated models allow preserving the geologic realism of facies sequences to mimic the stratigraphy and the depositional processes. The extension to 3D models can be achieved according to two approaches: (1) trace-by-trace classification or (2) global inversion with spatial correlation. The trace-by-trace inversion approach is computationally efficient but does not account for the lateral correlation of the facies. A global inversion with lateral correlation allows mimicking the continuity of facies and geobodies; however, the application in 3D is extremely challenging. Transition probabilities in the lateral directions are difficult to estimate and can be anisotropic and the dimensions of the training data sets would drastically increase.

The training data set in the RNN method implementation and the simulated realizations in the Monte Carlo approach were generated according to a first-order Markov chain to control the structure of the facies sequence and their proportions. The predicted seismic response was generated using a convolutional model. More advanced geostatistical methods could be used for the simulation of the facies sequences, such as multipoint geostatistics or process-based modeling, as well as more rigorous rock physics and wave-propagation models could be used for the computation of the seismic response. This would allow integrating additional geologic and physical constraints in the classification problem. Comparisons with higher order Markov chain methods have been recently proposed by Talarico et al. (2019) and Tian et al. (2019).

Table 6. Comparison of computational efficiency.

Method	Time (s)
RNN — case ML1 (1000 models)	772
RNN — case ML1 (10,000 models)	14,551
RNN — case ML12	77,200
RNN — case ML13	14,791
MC — 10,000 realizations	261
MC — 100,000 realizations	2617
MC — 1,00,000 realizations	26,173
MCMC — 1 chain	11,734
MCMC — 10 chains	117,342

CONCLUSION

We implemented and compared machine learning and Monte Carlo methods for the facies classification problem from seismic data. In all of the methods, the training data set and the simulated model were generated according to geologic rules, summarized in the vertical transition matrix for the facies sequence, and physical models (i.e., rock-physics models) for generation of the predicted data sets. Both categories of methods provide satisfactory results in terms of facies predictions. Overall, we conclude that machine-learning methods can lead to more accurate results if the training data set is large enough, but these methods might not be adequate to represent the uncertainty in the solution. The machine-learning algorithms have been tested using a training data set with a large number of samples; however, multiple runs of the same algorithms with different training sub-data sets lead to similar solutions. In practical applications, training data sets could be obtained by combining measured data with predictions obtained using statistical rock-physics models. Monte Carlo methods provide a better quantification of the uncertainty but might lead to less accurate solutions if the prior information is not informative enough, for example, when the transition matrix is unknown, and the number of simulations is not large enough.

ACKNOWLEDGMENTS

The author acknowledges the School of Energy Resources of the University of Wyoming and CERENA of the Instituto Superior Técnico for their support.

DATA AND MATERIALS AVAILABILITY

Data associated with this research are available and can be obtained by contacting the corresponding author.

APPENDIX A

SAMPLING ALGORITHM

Assuming n possible facies, the first step requires a uniform random sampling of n values in $[0, 1]$ with the constraint that their sum is equal to 1, such that the corresponding conditional probability is a valid probability density function. This procedure is repeated for each of the n rows of the transition matrix. This procedure generates n random n-element row vectors of values, $[p_{i,1}, \ldots, p_{i,n}]$ for $i = 1, \ldots, n$, uniformly distributed in the $(n-1)$-dimensional space of solutions, each with a fixed sum, equal to 1, and subject to a restriction $0 \leq p_{i,j} \leq 1$, where $p_{i,j}$ is defined as the probability of observing facies i at a given location k given facies j at the previous location $k-1$, that is, $p_{i,j} = P(f_k = i | f_{k-1} = j)$, for $i, j = 1, \ldots, n$.

After sampling the transition matrix, we sample the facies profile of N samples by adopting a sequential approach: We initially sample the first sample f_1, at the top of the sequence, from the prior probability of the facies (which is obtained by multiplying the transition matrix by itself several times until convergence); then, we

sample the subsequent values $\{f_k\}_{k=2,\ldots,N}$ from the probability $P(f_k|f_{k-1})$.

We then assign a porosity value by sampling facies-dependent Gaussian distributions estimated from the data, apply a linearized rock-physics model to compute P-impedance, and compute the predicted seismograms using a convolutional model of a wavelet and the reflection coefficient obtained from the calculated impedances. In the proposed implementation, we accept all of the realizations with a correlation coefficient between the predicted model response and the observed data higher than 0.9. According to this criterion, the realizations can also be ranked based on the degree of correlation between the prediction and the data. The proposed method is also known as the approximate Bayesian computation method (Vrugt and Sadegh, 2013; Chiachio et al., 2014; Olson and Kleiber, 2017). The simulation method is similar to the stochastic pseudo-well approach introduced by Connolly and Hughes (2016), but in our implementation, we stochastically sample the transition matrix before simulating the facies.

Because facies classification is generally performed in a limited interval, the number of samples in the segment of the seismic trace might contain a small number N of samples. Such a number might be too small for the facies realization to be representative of the corresponding transition matrix; therefore, we propose to generate a larger trace of M samples and select a subset of consecutive N samples with the best correlation with the measured data.

Alternatively, we propose an MCMC method in which we sample the facies realizations conditional to the data according to the Metropolis algorithm. The proposal distribution $p(\mathbf{f}^{\text{new}}|\mathbf{d})$ is built as a convex combination of the prior probability distribution $P(\mathbf{f})$ and the probability distribution $P(\mathbf{f}^{i-1}|\mathbf{d})$ of the last accepted realization $\mathbf{f}^{i-1}$:

$$P(\mathbf{f}^{\text{new}}|\mathbf{d}) = wP(\mathbf{f}^{i-1}|\mathbf{d}) + (1-w)P(\mathbf{f}), \tag{A-1}$$

weighted by the correlation between the measured data $\mathbf{d}$ and the predicted data $\mathbf{d}^{i-1}$ obtained from the realization $\mathbf{f}^{i-1}$, $w = \text{corr}(\mathbf{d}, \mathbf{d}^{i-1})$. At each location k, the value f_k^{new} of the new facies realization $\mathbf{f}^{\text{new}}$ is sampled by integrating the probability in equation A-1 and the transition probabilities from the given transition matrix:

$$P(f_k^{\text{new}}|\mathbf{d},\mathbf{f}) = \prod_{k=1}^{M} P(f_k|f_{k-1})P(f_k^{\text{new}}|\mathbf{d}), \tag{A-2}$$

with $P(f_1|f_0) \coloneqq P(f_1)$. The new proposed realization $\mathbf{f}^{\text{new}}$ is accepted with probability

$$a = \min\left(1, \frac{p(\mathbf{d}|\mathbf{f}^{\text{new}})}{p(\mathbf{d}|\mathbf{f}^{i-1})}\right). \tag{A-3}$$

The simulation method is similar to the probability perturbation methods proposed by Caers and Hoffman (2006), but in our implementation, the new facies profile is sampled by integrating the prior information in the transition matrix of the Markov chain. To account for the uncertainty in the transition matrix in the Markov chain, one could run r independent chains using the r transition matrices corresponding to the best r realizations obtained in the Monte Carlo acceptance-rejection sampling or from a feasibility study.

REFERENCES

Avseth, P., T. Mukerji, and G. Mavko, 2005, Quantitative seismic interpretation: Cambridge University Press.

Bestagini, P., V. Lipari, and S. Tubaro, 2017, A machine learning approach to facies classification using well logs: 87th Annual International Meeting, SEG, Expanded Abstracts, 2137–2142, doi: 10.1190/segam2017-17729805.1.

Boggs, S., 2001, Principles of sedimentology and stratigraphy: Prentice-Hall.

Caers, J., and T. B. Hoffman, 2006, The probability perturbation method: A new look at Bayesian inverse modeling: Mathematical Geology, **38**, 81–100, doi: 10.1007/s11004-005-9005-9.

Caers, J., and X. Ma, 2002, Modeling conditional distributions of facies from seismic using neural nets: Mathematical Geology, **34**, 143–167, doi: 10.1023/A:1014460101588.

Chen, C., and R. N. Hiscott, 1999, Statistical analysis of facies clustering in submarine-fan turbidite successions: Journal of Sedimentary Research, **69**, 505–517, doi: 10.2110/jsr.69.505.

Chiachio, M., J. L. Beck, J. Chiachio, and G. Rus, 2014, Approximate Bayesian computation by subset simulation: SIAM Journal on Scientific Computing, **36**, A1339–A1358, doi: 10.1137/130932831.

Chung, J., C. Gulcehre, K. Cho, and Y. Bengio, 2014, Empirical evaluation of gated recurrent neural networks on sequence modeling: arXiv:1412.3555.

Chung, J., C. Gulcehre, K. Cho, and Y. Bengio, 2015, Gated feedback recurrent neural networks: International Conference on Machine Learning, 2067–2075.

Coléou, T., M. Poupon, and K. Azbel, 2003, Unsupervised seismic facies classification: A review and comparison of techniques and implementation: The Leading Edge, **22**, 942–953, doi: 10.1190/1.1623635.

Cong, S., and Y. Liang, 2009, PID-like neural network nonlinear adaptive control for uncertain multivariable motion control systems: IEEE Transactions on Industrial Electronics, **56**, 3872–3879, doi: 10.1109/TIE.2009.2018433.

Connolly, P. A., and M. J. Hughes, 2016, Stochastic inversion by matching to large numbers of pseudo-wells: Geophysics, **81**, no. 2, M7–M22, doi: 10.1190/geo2015-0348.1.

Da Veiga, S., and M. Le Ravalec, 2012, Maximum likelihood classification for facies inference from reservoir attributes: Computational Geosciences, **16**, 709–722, doi: 10.1007/s10596-012-9283-5.

de Matos, M. C., P. L. Osorio, and P. R. Johann, 2006, Unsupervised seismic facies analysis using wavelet transform and self-organizing maps: Geophysics, **72**, no. 1, P9–P21, doi: 10.1190/1.2392789.

Doyen, P. M., 2007, Seismic reservoir characterization: EAGE.

Dubois, M. K., G. C. Bohling, and S. Chakrabarti, 2007, Comparison of four approaches to a rock facies classification problem: Computers & Geosciences, **33**, 599–617, doi: 10.1016/j.cageo.2006.08.011.

Elfeki, A., and M. Dekking, 2001, Markov chain model for subsurface characterization: Theory and applications: Mathematical Geology, **33**, 569–589, doi: 10.1023/A:1011044812133.

Fjeldstad, T., and H. Omre, 2017, Bayesian inversion of convolved hidden Markov models with applications in reservoir prediction: arXiv:1710.06613.

Gilks, W. R., S. Richardson, and D. Spiegelhalter, 1995, Markov chain Monte Carlo in practice: Chapman and Hall/CRC.

Graf, W., S. Freitag, M. Kaliske, and J. U. Sickert, 2010, Recurrent neural networks for uncertain time-dependent structural behavior: Computer-Aided Civil and Infrastructure Engineering, **25**, 322–323, doi: 10.1111/j.1467-8667.2009.00645.x.

Grana, D., T. Mukerji, J. Dvorkin, and G. Mavko, 2012, Stochastic inversion of facies from seismic data based on sequential simulations and probability perturbation method: Geophysics, **77**, no. 4, M53–M72, doi: 10.1190/geo2011-0417.1.

Guillen, P., G. Larrazabal, G. González, D. Boumber, and R. Vilalta, 2015, Supervised learning to detect salt body: 85th Annual International Meeting, SEG, Expanded Abstracts, 1826–1829, doi: 10.1190/segam2015-5931401.1.

Hall, B., 2016, Facies classification using machine learning: The Leading Edge, **35**, 906–909, doi: 10.1190/tle35100906.1.

Hall, M., and B. Hall, 2017, Distributed collaborative prediction: Results of the machine learning contest: The Leading Edge, **36**, 267–269, doi: 10.1190/tle36030267.1.

Hastie, T., R. Tibshirani, and J. Friedman, 2002, The elements of statistical learning: Springer.

Huang, L., X. Dong, and T. E. Clee, 2017, A scalable deep learning platform for identifying geologic features from seismic attributes: The Leading Edge, **36**, 249–256, doi: 10.1190/tle36030249.1.

Jang, E., S. Gu, and B. Poole, 2016, Categorical reparameterization with Gumbel-Softmax: arXiv:1611.01144.

Jin, Y., X. Wu, J. Chen, Z. Han, and W. Hu, 2018, Seismic data denoising by deep-residual networks: 88th Annual International Meeting, SEG, Expanded Abstracts, 4593–4597, doi: 10.1190/segam2018-2998619.1.

Krumbein, W. C., and M. F. Dacey, 1969, Markov chains and embedded Markov chains in geology: Mathematical Geology, **1**, 79–96, doi: 10.1007/BF02047072.

Li, W., 2018, Classifying geological structure elements from seismic images using deep learning: 88th Annual International Meeting, SEG, Expanded Abstracts, 4643–4648, doi: 10.1190/segam2018-2998036.1.

Li, Y., and R. Anderson-Sprecher, 2006, Facies identification from well logs: A comparison of discriminant analysis and naïve Bayes classifier: Journal of Petroleum Science and Engineering, **53**, 149–157, doi: 10.1016/j.petrol.2006.06.001.

Lin, F. J., R. J. Wai, W. D. Chou, and S. P. Hsu, 2002, Adaptive backstepping control using recurrent neural network for linear induction motor drive: IEEE Transactions on Industrial Electronics, **49**, 134–146, doi: 10.1109/41.982257.

Lindberg, D. V., and D. Grana, 2015, Petro-elastic log-facies classification using the expectation–maximization algorithm and hidden Markov models: Mathematical Geosciences, **47**, 719–752, doi: 10.1007/s11004-015-9604-z.

Maniar, H., S. Ryali, M. S. Kulkarni, and A. Abubakar, 2018, Machine-learning methods in geoscience: 88th Annual International Meeting, SEG, Expanded Abstracts, 4638–4642, doi: 10.1190/segam2018-2997218.1.

Martinez, W. L., and A. Martinez, 2007, Computational statistics handbook with Matlab: Chapman & Hall.

Mikolov, T., M. Karafiát, L. Burget, J. Černocký, and S. Khudanpur, 2010, Recurrent neural network-based language model: 11th Annual Conference of the International Speech Communication Association.

Mosegaard, K., 1998, Resolution analysis of general inverse problems through inverse Monte Carlo sampling: Inverse Problems, **14**, 405–426, doi: 10.1088/0266-5611/14/3/004.

Mosegaard, K., and A. Tarantola, 1995, Monte Carlo sampling of solutions to inverse problems: Journal of Geophysical Research, **100**, 12431–12447, doi: 10.1029/94JB03097.

Mukerji, T., P. Avseth, G. Mavko, I. Takahashi, and E. F. González, 2001, Statistical rock physics: Combining rock physics, information theory, and geostatistics to reduce uncertainty in seismic reservoir characterization: The Leading Edge, **20**, 313–319, doi: 10.1190/1.1438938.

Negahdari, A., M. Ziaii, and J. Ghiasi-Freez, 2014, Application of discriminant analysis for studying the source rock potential of probable formations in the Lorestan Basin, Iran: International Journal of Mining and Geo-Engineering, **48**, 31–54, doi: 10.22059\IJMGE.2014.51805.

Olson, B., and W. Kleiber, 2017, Approximate Bayesian computation methods for daily spatiotemporal precipitation occurrence simulation: Water Resources Research, **53**, 3352–3372, doi: 10.1002/2016WR019741.

Park, B. S., S. J. Yoo, J. B. Park, and Y. H. Choi, 2008, Adaptive neural sliding mode control of nonholonomic wheeled mobile robots with model uncertainty: IEEE Transactions on Control Systems Technology, **17**, 207–214, doi: 10.1109/TCST.2008.922584.

Pascanu, R., T. Mikolov, and Y. Bengio, 2013, On the difficulty of training recurrent neural networks: International Conference on Machine Learning, 1310–1318.

Roy, A., V. Jayaram, and K. J. Marfurt, 2013, Active learning algorithms in seismic facies classification: 83rd Annual International Meeting, SEG, Expanded Abstracts, 1467–1471, doi: 10.1190/segam2013-0769.1.

Saggaf, M. M., M. N. Toksöz, and M. I. Marhoon, 2003, Seismic facies classification and identification by competitive neural networks: Geophysics, **68**, 1984–1999, doi: 10.1190/1.1635052.

Sak, H., A. Senior, and F. Beaufays, 2014, Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition: arXiv:1402.1128.

Sambridge, M., and K. Mosegaard, 2002, Monte Carlo methods in geophysical inverse problems: Reviews of Geophysics, **40**, 1–29, doi: 10.1029/2000RG000089.

Sen, M. K., and P. L. Stoffa, 1996, Bayesian inference, Gibbs sampler and uncertainty estimation in geophysical inversion: Geophysical Prospecting, **44**, 313–350, doi: 10.1111/j.1365-2478.1996.tb00152.x.

Strecker, U., and R. Uden, 2002, Data mining of 3D poststack seismic attribute volumes using Kohonen self-organizing maps: The Leading Edge, **21**, 1032–1037, doi: 10.1190/1.1518442.

Talarico, E., W. Leao, and D. Grana, 2019, Comparison of recursive neural network and Markov chain models in facies inversion: 81st Annual International Conference and Exhibition, EAGE, Extended Abstracts, doi: 10.3997/2214-4609.201902275.

Tartakovsky, D. M., and B. E. Wohlberg, 2004, Delineation of geologic facies with statistical learning theory: Geophysical Research Letters, **31**, 1–5, doi: 10.1029/2004GL020864.

Tian, M., H. Omre, and H. Xu, 2019, Well logs inversion into lithology classes: Comparing Bayesian inversion and machine learning: 81st Annual International Conference and Exhibition, EAGE, Extended Abstracts, doi: 10.3997/2214-4609.201902207.

Tieleman, T., and G. Hinton, 2012, Lecture 6.5 — RMSProp, COURSERA: Neural networks for machine learning: Technical Report.

Vrugt, J. A., and M. Sadegh, 2013, Toward diagnostic model calibration and evaluation: Approximate Bayesian computation: Water Resources Research, **49**, 4335–4345, doi: 10.1002/wrcr.20354.

Wang, G., and T. R. Carr, 2012, Methodology of organic-rich shale lithofacies identification and prediction: A case study from Marcellus Shale in the Appalachian Basin: Computers & Geosciences, **49**, 151–163, doi: 10.1016/j.cageo.2012.07.011.

Wang, G., T. R. Carr, Y. Ju, and C. Li, 2014, Identifying organic-rich Marcellus Shale lithofacies by support vector machine classifier in the Appalachian Basin: Computers & Geosciences, **64**, 52–60, doi: 10.1016/j.cageo.2013.12.002.

West, B. P., S. R. May, J. E. Eastwood, and C. Rossen, 2002, Interactive seismic facies classification using textural attributes and neural networks: The Leading Edge, **21**, 1042–1049, doi: 10.1190/1.1518444.

Wong, P. M., F. X. Jian, and I. J. Taggart, 1995, A critical comparison of neural networks and discriminant analysis in lithofacies, porosity and permeability predictions: Journal of Petroleum Geology, **18**, 191–206, doi: 10.1111/j.1747-5457.1995.tb00897.x.

Wu, P. Y., V. Jain, M. S. Kulkarni, and A. Abubakar, 2018, Machine learning-based method for automated well-log processing and interpretation: 88th Annual International Meeting, SEG, Expanded Abstracts, 2041–2045, doi: 10.1190/segam2018-2996973.1.

Zhang, L., and C. Zhan, 2017, Machine learning in rock facies classification: An application of XGBoost: International Geophysical Conference, 1371–1374.

Zhao, T., V. Jayaram, A. Roy, and K. J. Marfurt, 2015, A comparison of classification techniques for seismic facies recognition: Interpretation, **3**, no. 4, SAE29–SAE58, doi: 10.1190/INT-2015-0044.1.