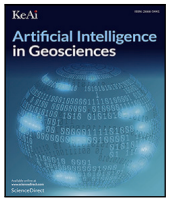
Artificial Intelligence in Geosciences 5 (2024) 100072

Contents lists available at ScienceDirect

Artificial Intelligence in Geosciences

journal homepage: www.keaipublishing.com/en/journals/artificial-intelligence-in-geosciences



Reconstruction of lithofacies using a supervised Self-Organizing Map: Application in pseudo-wells based on a synthetic geologic cross-section

Carreira V.R. [b,*], Bijani R. [a], Ponte-Neto C.F. [b]

[a] Departamento de Geologia e Geofísica, Universidade Federal Fluminense (UFF), Niterói, Brazil
[b] Coordenação de Geofísica, Observatório Nacional (ON-MCTIC), Rio de Janeiro, Brazil

ARTICLE INFO

Keywords:
Self-Organizing Maps
Supervised machine learning
Synthetic well-log data
Classification of lithofacies

ABSTRACT

Recently, machine learning (ML) has been considered a powerful technological element of different society areas. To transform the computer into a decision maker, several sophisticated methods and algorithms are constantly created and analyzed. In geophysics, both supervised and unsupervised ML methods have dramatically contributed to the development of seismic and well-log data interpretation. In well-logging, ML algorithms are well-suited for lithologic reconstruction problems, once there is no analytical expressions for computing well-log data produced by a particular rock unit. Additionally, supervised ML methods are strongly dependent on a accurate-labeled training data-set, which is not a simple task to achieve, due to data absences or corruption. Once an adequate supervision is performed, the classification outputs tend to be more accurate than unsupervised methods. This work presents a supervised version of a Self-Organizing Map, named as SSOM, to solve a lithologic reconstruction problem from well-log data. Firstly, we go for a more controlled problem and simulate well-log data directly from an interpreted geologic cross-section. We then define two specific training data-sets composed by density (RHOB), sonic (DT), spontaneous potential (SP) and gamma-ray (GR) logs, all simulated through a Gaussian distribution function per lithology. Once the training data-set is created, we simulate a particular pseudo-well, referred to as classification well, for defining controlled tests. First one comprises a training data-set with no labeled log data of the simulated fault zone. In the second test, we intentionally improve the training data-set with the fault. To bespeak the obtained results for each test, we analyze confusion matrices, logplots, accuracy and precision. Apart from very thin layer misclassifications, the SSOM provides reasonable lithologic reconstructions, especially when the improved training data-set is considered for supervision. The set of numerical experiments shows that our SSOM is extremely well-suited for a supervised lithologic reconstruction, especially to recover lithotypes that are weakly-sampled in the training log-data. On the other hand, some misclassifications are also observed when the cortex could not group the slightly different lithologies.

1. Introduction

The human brain has evolved amazingly during lifetimes, especially when it comes to the amount of information to be processed (Mao, 1996; Hall et al., 2014). Following this guideline, the artificial intelligence (AI) emerges as a prolific science field whose basis are totally inspired by human intellect and behavior. Any technique that enables computers to simulate human intelligence can be considered an AI method (Lachaux et al., 2020; Lechner et al., 2020). Most of technological segments of modern society are connected to AI, like in forensic investigations, robotics, health care treatments, social media and geosciences (Weyn et al., 2019, 2020).

As a subgroup of AI, machine learning (ML) addresses statistical concepts to create computer programs that are capable of gaining experience (Michie et al., 1994; Levy, 1997; MacKay, 2005). One relevant aspect of ML methods lie in the ability of solving complex problems exploring only information comprising the data-set (Ruvini and Dony, 2000; Wu et al., 2019; Dramsch, 2020). Under this assumption, some ML methods are developed with a notorious capability of locating specific patterns into the data by its own. Each pattern is then referred to as cluster and the ML algorithm works in clustering the data. This is called unsupervised learning (Dayan et al., 1999; Baştanlar and Ozuysal, 2014; Li and Phung, 2014). On the other hand, in the supervised learning framework, the ML algorithms identify reliable

* Corresponding author.
E-mail addresses: victorcarreira@id.uff.br (Carreira V.R.), rodrigobijani@id.uff.br (Bijani R.), cosme@on.br (Ponte-Neto C.F.).
URL: https://www.on.br (Carreira V.R.).

https://doi.org/10.1016/j.aiig.2024.100072
Received 23 May 2023; Received in revised form 18 December 2023; Accepted 31 January 2024
Available online 10 February 2024

patterns directly from a tagged data-set, during the training stage. After that, every unlabeled data are ready to be classified (Kotsiantis et al., 2007; Schrider and Kern, 2018). Several works have been developed in recent decades based on different ML methods, such as K-means (Lloyd, 1982), Support Vector Machine (SVM) (Cortes and Vapnik, 1995), Linear Discriminant Analysis (LDA) (Fisher, 1936), Gaussian Mixture (GM) (Reynolds, 2009), Artificial Neural Networks (ANNs) (McCulloch and Pitts, 1943), Single and Multi-Layer Perceptrons (SLP and MLP, respectively) (Raudys, 1998; Kanal, 2001; Krogh, 2008; Günther and Fritsch, 2010; Pastukhov and Prokofiev, 2016; Gonçalves et al., 2017), Self-Organizing Maps (SOMs) (Kohonen, 1989).

The ANNs are totally inspired by the human brain behavior, such as perceptual interpretation, abstraction, learning and memory (Calderón-Macías et al., 2000). Following a similar guideline, SOMs mimic the working of a cerebral cortex, in which specific human characteristics require more neurons than others. A SOM algorithm is based on a fully-connected network with several possible geometric configurations (Haykin, 2001). Such arrangements are associated with vertices, which are also known as artificial neurons. The connection among neurons are referred to as edges and each one governs the neural interactions (Kohonen, 2013). Those artificial neurons change their weights during an iterative procedure. Classical SOM uses unsupervised learning to update weights during a training process. More recent evidence (Papadimitriou et al., 2002) reveals that supervised SOM together with synthetic data improved the results significantly related to those obtained with the unsupervised SOM.

Some of the above-mentioned methods have been applied to solve a variety of Geophysical problems (Guillen et al., 2015; Kumar and Kishore, 2006; Bestagini et al., 2017; Costa et al., 2019). For example, Kuyuk et al. (2012) applied both k-means and Gaussian Mixture methods to identify specific seismic events in the vicinity of Istanbul, Turkey. Dias et al. (2018) present a comparison among unsupervised ML methods for studying the problem of segmentation of fractures. To do so, images of simulated fractures are considered for a better segmentation analysis. Following this guideline, Neyamadpour et al. (2009) investigate the applicability of a ANN, implemented in Matlab, to invert a Wenner–Schlumberger geoelectrical data. Kuroda et al. (2012) classify electrofacies using SOMs in a supervised framework. For the training stage, well-log data and lithologic information are considered from Namorado Oilfield in Campos Basin, Southeast Brazil. Konaté et al. (2015) predict porosity values of crystalline crusts by means of well-log data. Both Feed-forward retro-propagation and the radial basis functions networks are compared in this porosity problem. Perol et al. (2018) develop a stochastic convolutional network for evaluating seismic risks. A two-dimensional tensor representing the seismic wave is the input of the network, for a fixed data window. Shoji et al. (2018) combined probabilities and a convolutional networks to classify volcanic dusts. To do so, images are converted to signal and inserted into the network. Carneiro et al. (2012) uses a SOM to infer how clustering outcrops rock units, in Anapu-Tuerê province inside the Amazon region, using aerialgeophysical data and spatial analysis. Pastukhov and Prokofiev (2016) uses kohonen-SOM to train a multi-layer perceptron (MLP). This pairwise strategy creates new models for data clustering, which contains unique sets of attributes used in validation and training tests. Sahoo and Jha (2017) uses a hybrid SOM with Genetic Algorithm (GA) to characterizing lithology in groundwater basins using well log data. Concerning only metamorphic rocks such as orthogneiss, paragneiss, eclogite, amphibolite and ultramafic rocks (Valentín et al., 2019) relates borehole images with petrophysical properties characterizing reservoir rocks in a facies scale using deep residual convolutional neural networks (ResNetRock). Gonçalves et al. (2017) uses a series of classification algorithms such as k-nearest neighbors, Naive Bayes, Random Forest, Sequential minimal optimization and Multi Layer Perceptron to identify carbonate rocks based on relaxation times in Nuclear Magnetic Resonance logging data. Alternatively, Saporetti et al. (2019) integrate the gradient boosting method (i.e., a collection of decision tree models) with a differential evolution algorithm for formation lithology identification using data from both Daniudui and Hangjinqi gas fields, in China.

Well-logging is a valuable geophysical method for water and hydrocarbon investigations beneath the Earth's surface. Basically, different physical properties are measured during the drilling process (Ellis and Singer, 2012). These precious acquired data endorse for a meticulous interpretation of potential exploration zones, once there is a natural affinity between the log data and the causative rock unit in subsurface. Despite the notorious importance of well-logging to applied geophysics, several complications may arise during the acquisition process, such as collapsing of wells, equipment entrapment (Pashin et al., 2018). Additionally, the lithologic information in a well may not be accurate during the drilling procedure. To overcame such limiting aspects, simulation of well-log data is extremely recommended. For example, da Silva et al. (2015) used some empirical expressions and linear regression concepts to estimate sonic and density logs. Following this guideline, Chagas et al. (2010) present an empirical expression for sonic logs based on multi-variate statistics. Additionally, some undesired artifacts are removed from the simulated data by statistical analysis. Zhang et al. (2018) make use of an recursive artificial neural network to estimate missing and/or corrupted logs. The advantage of using such iterative approach lies in a good performance of the method facing a reduced training data-set. Kostorz (2021) presents a data restructure and a quasi-probabilistic interpolation technique capable of smoothing noisy data. The strategy provides a statistical characterization of the lithologic classification problem.

In this research we design a new and complete workflow for solving the lithologic classification problem from well-log data. Firstly, a complex synthetic scenario is designed inspired by the interpreted geologic section of Solimões sedimentary basin, in Amazonas (Mohriak et al., 2008). This is a prolific exploratory onshore basin in Brazil, mainly composed by conglomerates, shalestones, sandstones, dolomites, diabases, crystalline, halites, granites, basalts and normal faults. Based on the above-mentioned rock units, we use Gaussian distribution functions for simulating density (RHOB), gamma-ray (GR), spontaneous potential (SP) and sonic (DT) logs directly from a compiled data-set. For a more realistic data-set, a smoothing filtering using moving average is applied to each simulated well-log data, especially in contacts among different electrofacies. We then implement the supervised version of a Self-Organizing Map (SOM) to solve the lithologic reconstruction problem. During the supervision stage, each rock unit corresponds to a coding number, and then each log-data is precisely labeled to giving raise the training data-set. In our context, the labeled log-data can work as a prolific geologic prior information to the classification of lithofacies, once the data pattern presented in logs are strongly related to the causative lithologies (Dvorkin and Wollner, 2017; Dvorkin, 2020). With such synthetic configuration, we offer more accurate data-sets to be used in our SSOM. To verify the relevance of a good training data-set for obtaining reasonable predicted models by a supervised machine learning method, we define two different tests for classifying a particular pseudo-well of the geologic cross-section. A fault zone is simulated by using the mixture law (Nery, 2013) comprising two specific rock units. The pseudo-well is intentionally design under this complex scenario for a more in-deep investigation on the potentialities and limitations of the reconstructed models. Firstly, we setup a training data-set with no log data produced to the fault zone. After that, a second test is configured with an improved training data-set, by adding the log data of normal fault comprising the pseudo-well, which are calculated through Mixture law (Nery, 2013). With this test, we verify the capability of our supervised SOM in recovering all lithologies of the pseudo-well, including the rock units of the fault zone. We end up this work by analyzing errors, precision and accuracy of each test. With these complete set of controlled tests, we explore the potentialities and limitations of a supervised SOM to reconstruct lithofacies in a well.

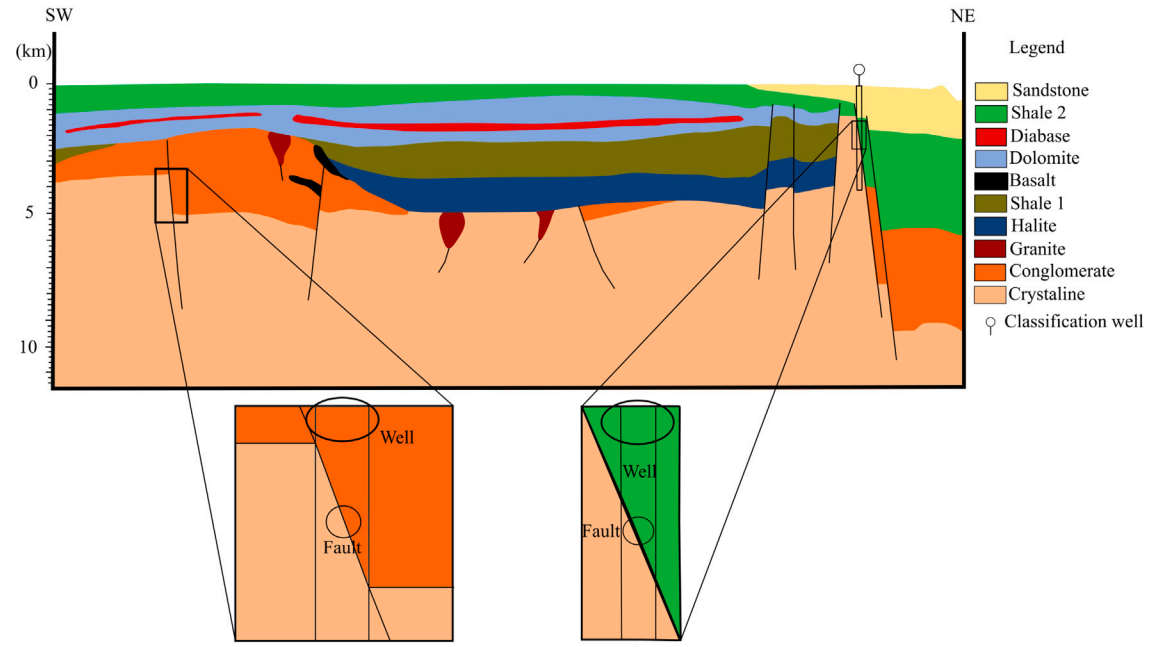


Fig. 1. The geologic cross-section representing the on-shore syneclise sedimentary section by Mohriak et al. (2008). The zoom boxes highlight two simulated normal faults involving Conglomerate, Crystalline and Shale 2. The circle at the top of cross-section indicates the classification well to be considered in this work.

2. Synthetic geological scenario

To aim the scope of this work, we first design a synthetic geological scenario comprising a cross-section of Solimões Sedimentary Basin in North Brazil, an intracratonic basin previously interpreted by Mohriak et al. (2008), as can be seen in Fig. 1. Horts, Grabens, normal and reverse faults are the major geologic structures presented in Solimões sedimentary basin. Moreover, Fig. 1 also present the rock units comprising the geologic cross-section. As can be seen, a list of ten different rock units comprising the cross-section are highlighted, such as Sandstone, Diabase, Dolomite, Basalt, Halite, Granite, Conglomerate and Crystalline. Particularly, shales are numbered to distinguish two different depositional periods (Mohriak et al., 2008). Additionally, a zoom box of two normal faults simulating a contact zone of two rock units (i.e., conglomerate-crystalline and shale2-crystalline) are also shown in Fig. 1.

Once the geologic cross-section is drawn, we are free to simulate different log data from a realistic geologic scenario, which is fundamental for a more complete analysis of the supervised SOM to be presented in this work (i.e., Section 3). As can be seen in Fig. 1, we define a classification well to be investigated in the synthetic examples designed for this work. The classification well presents a normal fault that produces a very particular log-data pattern. In the following section, we suggest a simple strategy to simulate such geologic condition.

2.1. Simulation of logs produced by faults

For simulating the normal faults observed in the geologic cross-section, we implement the mixture law (Nery, 2013) for two pairs of rock units. Let $\mathbf{R_1}$ be a set of physical properties produced by a specific rock unit $\mathbf{R_1} = [x_1^1, x_2^1, x_3^1, x_4^1]$ and $\mathbf{R_2}$ the same set of physical properties produced by another rock unit $\mathbf{R_2} = [x_1^2, x_2^2, x_3^2, x_4^2]$. A log-data under fault F can be simulated by the following expression:

$$F = [\phi \mathbf{R}_1^m + (1 - \phi)\mathbf{R}_2^m]^{1/m}, \quad (1)$$

where ϕ is the fault angle. For the linear case, $m = 1$ represents a normal fault. Eq. (1) combines the volumetric component of the rock unit and the specific physical property (Nery, 2013). The m factor represents the geometric distribution in a specific mixture. $\mathbf{R_1}$ and $\mathbf{R_2}$ carry out the amount of rock properties presented in each portion of the well, held in percentages.

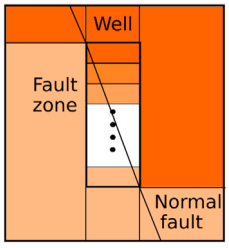


Fig. 2. A simple sketch of a normal fault simulated by the mixture law. The transition of colors from top to bottom indicates the mixture of rocks to be considered.

Table 1
Percentages of rock mixtures involving the two simulated normal faults. FZas and FZbs are the Fault IDs.

Fault ID	Crystalline	Shale 2	Conglomerate
FZa(20)	20%	–	80%
FZa(40)	40%	–	60%
FZa(60)	60%	–	40%
FZa(80)	80%	–	20%
FZb(20)	20%	80%	–
FZb(40)	40%	60%	–
FZb(60)	60%	40%	–
FZb(80)	80%	20%	–

To represent a normal fault by means of the mixture law, we discretize the fault zone into N identical blocks, as can be seen in Fig. 2. Each one corresponds to a particular percentage of log data produced by the rock unit immediately above and below the fault zone. As can be seen in Fig. 1, two normal faults are highlighted in our workflow. One involving conglomerate and crystalline, while the second is composed by shale2 and crystalline (see Table 1).

Once the synthetic cross-section based on a true sedimentary basin is defined, we can move further and simulate well-log data from a more realistic geologic scenario.

Table 2
Physical-property values and the standard deviations of each rock unit for simulating the log data.

Physical properties				
Rock	$RHOB$ g/cm^3	$GR\ API$	SP mV	DT µs/m
Conglomerate	2.30 ± 0.046	20.0 ± 2.0	-40.0 ± 4.0	110 ± 11.0
Shale 2	2.48 ± 0.049	110.0 ± 11.0	70.0 ± 7.0	550 ± 55.0
Dolomite	2.52 ± 0.05	40.0 ± 4.0	-60.0 ± 6.0	142 ± 14.2
Diabase	2.80 ± 0.056	30.0 ± 3.0	90.0 ± 9.0	50 ± 5.0
Crystalline	2.75 ± 0.055	40.0 ± 4.0	70.0 ± 7.0	55 ± 5.5
Halite 1	2.58 ± 0.051	230.0 ± 23.0	75.0 ± 7.5	520 ± 52.0
Halite	2.16 ± 0.043	11.0 ± 1.1	6.0 ± 0.6	216 ± 21.6
Granite	2.67 ± 0.053	150.0 ± 15.0	-60.0 ± 6.0	68 ± 6.8
Sandstone	2.31 ± 0.046	20.0 ± 2.0	120.0 ± 12.0	170 ± 17
Basalt	2.87 ± 0.057	15.0 ± 1.5	50.0 ± 5.0	65 ± 6.5
Shale 1	2.58 ± 0.05	230.6 ± 25.5	70.0 ± 7.8	520.5 ± 60.0

2.2. Synthetic log data

To simulate log data, we first promote a compilation of literature information (Bassiouni et al., 1994) combined with rigorous analysis of real log data from similar sedimentary basins. After that, we define means and standard deviations of physical-property values for each rock unit within the cross-section, as can be seen in Table 2.

Shale 1 simulates a shale rock enriched by organic matter with high values of Gamma-ray. On the other hand, shale 2 simulates a rock unit composed by few organic matter.

Once the values of physical properties for each rock unit are compiled, a particular well-log data can be entirely simulated by correlating the rock-unit color with the values in Table 2. Fig. 3(a) and (b) show the lithologies presented in a pseudo-well and a theoretical RHOB log produced by our synthetic approach, respectively.

According to Lindberg et al. (2015), the measured log data have a smooth effect due to instrumental convolution. The top and bottom of rock layers are heterogeneous, which implies in a smooth logging at interfaces. To simulate this particular feature, we apply a simple moving average into the synthetic log data. This procedures gives a smooth transitions to rock layers into the model. In our experiments, we set a moving average comprising three data-samples, without overlapping.

The last part of our simulation consists of applying some random noise to the smoothed logs. To do so, we simply add a Gaussian noise vector of zero mean and a particular standard deviation value to the smoothed synthetic log. In our numerical experiments, we set different standard deviation values due to the large log ranges. Additionally, each log is computed at a ratio of 0.01 measure per meter. In this work, we simulate four specific logs: density ($RHOB$ - g/cm^3), gamma-ray ($GR - Ci/g$), spontaneous potential (SP – mV), and sonic (DT – µs/ m). Table 2 illustrates all log-properties and uncertainty values per rock unit.

In summary, the following steps are required for modeling synthetic log data:

1. Draw a geologic cross-section;
2. Convert colors into rock-unit IDs;
3. Define means and standard deviations of physical-property values for each rock unit within the geologic model;
4. Define the pseudo-well to be simulated;
5. Create a theoretical log (e.g., Fig. 3b)
6. Apply a moving window to smooth interfaces, as can be seen in Fig. 3c;
7. Apply a Gaussian noise smoothed log (e.g., Fig. 3d);

3. Methodology

Self Organizing Maps (SOMs) (Kohonen, 2013) are machine learning methods that mimic the working of a cortex brain. These algorithms follow Penfield Homunculus, which states that complicated activities require more neurons than the simple tasks. Under this assumption, we have implemented a supervised version of a SOM, called Supervised Self-Organizing Map (SSOM), to classify lithofacies using simulated well-log data. For a decent comprehension of the SSOM implemented in this work, we design a synthetic geological cross-section based on the one interpreted by Mohriak et al. (2008), from which different training well-log data-sets are simulated according Section 2. For comparisons, two specific training data-sets are intentionally established by a concatenation of synthetic log-drillings comprising the interpreted geologic cross-section. The SSOM is trained using the two training data-sets and then applied one at a time to the classification well. Basic statistics are used to quantify the obtained classification marks. In the following sections, we explain the main concepts of the implemented SSOM and the adopted supervision strategy.

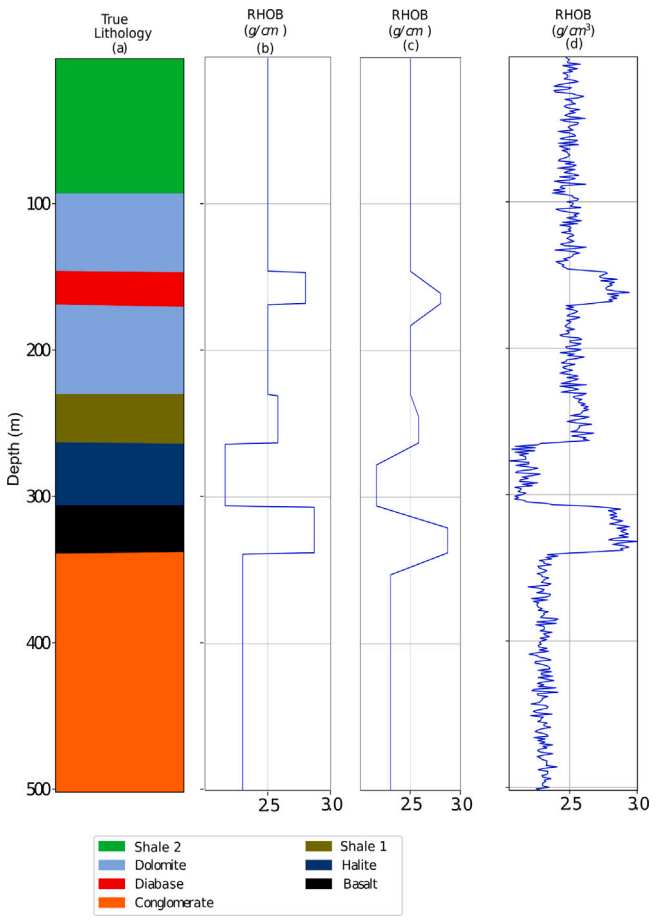


Fig. 3. An example of a (a) lithologic log, (b) the theoretical RHOB log, (c) the smoothing RHOB log produced by the moving average procedure and (d) the Gaussian noise vector added to the smoothing RHOB log.

Fig. 4 presents a workflow of the entire methodology to be detailed throughout this section.

3.1. SSOM: Model structure and hyperparameters

SOMs are machine learning types composed by networks that are distributed on a hyper-plane inside a hyperspace of features. In our problem statement, features are inputs referred to as different well-log data. We developed a standard supervised SOM, referred herein as SSOM, based on a torus geometry (Ventrella, 2011), as exhibited in Fig. 5(a). A torus cross-section provides a two-dimensional domain composed by a rectangular grid of neurons, also known as nodes

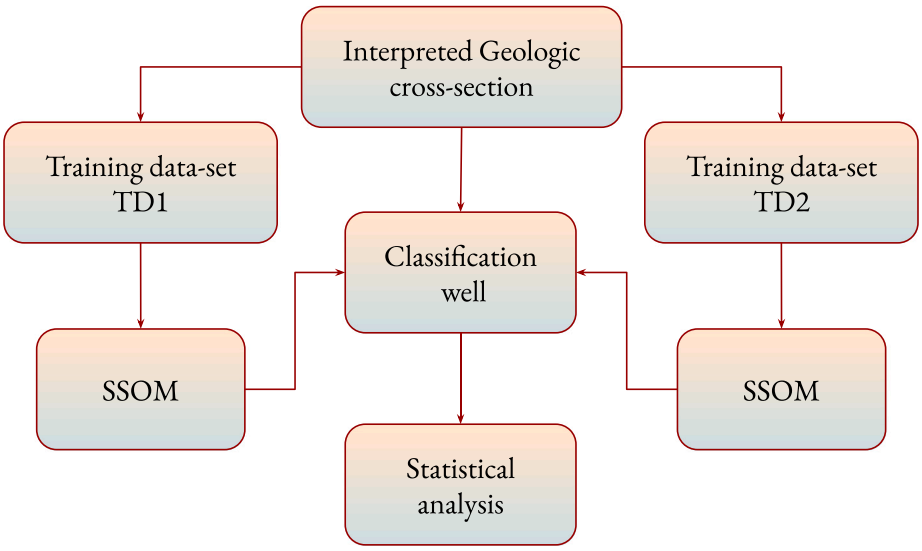


Fig. 4. Flowchart of the entire methodology adopted in this work.

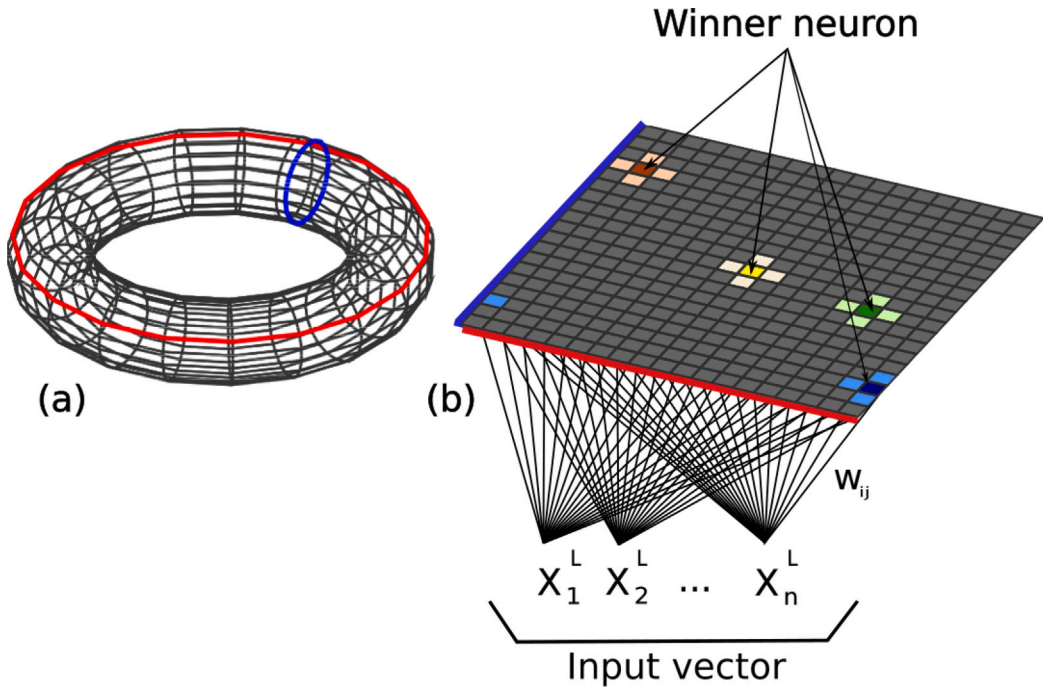


Fig. 5. (a) toroidal geometry of our SSOM. The red and blue lines are ideally two-dimensional. (b) Model structure of our implemented SSOM. A two-dimensional view of the toroid, representing the rectangular cortex and the neighborhoods.

(Fig. 5b). This particular configuration guarantees that all nodes are connected and avoids boundary problems (Ventrella, 2011; Kohonen, 2013). More about the model structure of SOMs are detailed in Hoan (2016) and Han et al. (2019). The input vector is represented by different log-data:

$$\mathbf{x}_i^L = [x_1^L, x_2^L, \ldots, x_n^L]. \quad (2)$$

In our context, $n = 4$ once we are working with $RHOB$, SP, GR and DT synthetic logs. Superscript ^{L} indicates labeled data (i.e., both logs and lithofacies are known). Another crucial hyperparameter of our SSOM is the weight matrix $\mathbf{W}_i$, which elements are randomly selected between $[0, 1]$ at the very beginning of the training process. Additionally, the number of neurons comprising the cortex is also previously required. Ideally, this hyperparameter is directly related to the amount of data to be trained (Adibifard et al., 2014). In this work, we also computed the number of obsolete neurons (i.e., nodes that are never used during the supervision) as a quality-control parameter for defining the ideal size of the rectangular grid. In theory, if the number of obsolete neurons is zero, the cortex is fully operational. Other relevant parameter to be defined is the total number of iterations of the supervision stage, which is also a relevant hyperparameter for a proper use of our SSOM.

Fig. 6(a) presents a simple sketch of our rectangular cortex, composed by the winner neuron and four neighboring neurons. Fig. 6(b)

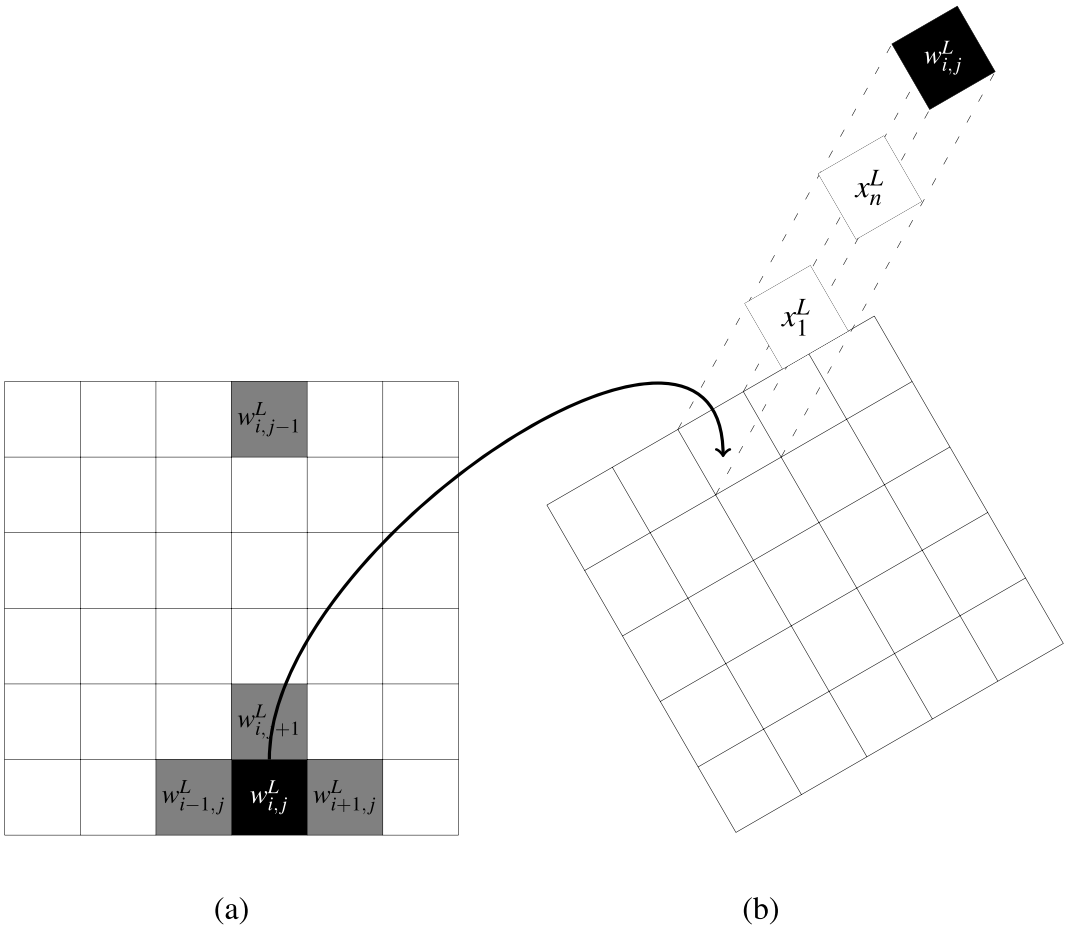


Fig. 6. (a) A standard rectangular grid of neurons. The black square represents a winner neuron and the gray blocks are the four neighboring neurons. Node $w^L_{i,j-1}$ on the opposite side of the rectangular grid is the downward neighbor. (b) x^L_1 represents the first labeled log data and x^L_n the nth labeled log-data.

highlights the concept of dimensionality reduction, in which all log values are transformed into weights and stored in a neuron.

3.2. SSOM: Supervision stage

Basically, a Self-Organizing Map is composed of both input and output layers. The former comprises the data-set and the latter is associated with neurons in a cerebral cortex (Hoan, 2016). The connection between the two layers is achieved by computing a set of weights organized under the form of a two-dimensional matrix of neurons. Then, each neuron comprising the cortex is represented by a weight value, which should be modified in an iterative procedure. At the very first iteration, all weights are randomly set and then modified until an acceptable convergence rate is achieved. In our context, weights are referred to as the Euclidean distances in the physical-property domain (i.e., each log-data), as stated in the following expression:

$$d_i(\mathbf{X_i}, \mathbf{W_i}) = \|\mathbf{X}_i - \mathbf{W}_i\| = \sqrt{(\sum_{i=1}^{n} x_i^L - w_{i,j}^L)^2}, \tag{3}$$

where x_i^L is the ith labeled log data and $w_{i,j}^L$ is a specific weight value comprising the ith input data and the jth layer neuron (Hoan, 2016).

The election of a neuron with least weight value, refereed to as the winning neuron, is accomplished by measuring the Euclidean distance between the ith well-log data and the jth neuron weight, as highlighted in Fig. 6. The weights of both winning neuron and neighboring nodes are updated by the following iterative expression:

$$\mathbf{w}_{i,j}^L(t+1) = \mathbf{w}_{i,j}^L(t) + \eta(t)[x_i^L - \mathbf{w}_{i,j}^L(t)], \tag{4}$$

where x_i^L is the ith labeled log data comprising the training data-set. Eq. (4) defines a linear expression for computing new weight values. Symbol $\eta(t)$ is the learning rate, defined as:

$$\eta(t) = \frac{1}{k}\left(1 - \frac{t}{T}\right), \quad k \neq 0 \tag{5}$$

where t is the current iteration, k is a damping coefficient, to be set by trial and error, to differentiate weight values of the winner and the neighboring neurons during the iterative procedure (i.e., $k = 1$ for the winner and $k \neq 1$ for the neighbors). T is the total number of iterations, previously defined by the user. Eq. (5) imposes a larger variability to the weights of winner neurons at the initial iterations. This is strictly related to a higher learning capability at the beginning of the iterative process, which mimics the behavior of a human brain (Kohonen, 1989). To avoid bad weights, all log data should be normalized. In our applications, we use the MinMax normalization strategy (Jayalakshmi and Santhakumaran, 2011), which is defined as:

$$x_i^L = \frac{x_i^L - min(x_i^L)}{max(x_i^L) - min(x_i^L)}, \tag{6}$$

where x_i^L is the ith labeled log data to be normalized. Outliers are previously removed to avoid spurious data. So, the supervision of our SOM basically sets the labels of the training data-set to the weights of Eq. (4). So, the training data-set is classified with the current set of labeled weights and the errors in classification are counted. The convergence is reached when the errors are minimal and do not change significantly from consecutive iterations. After that, we run the SSOM using a different input data and the classification is then performed

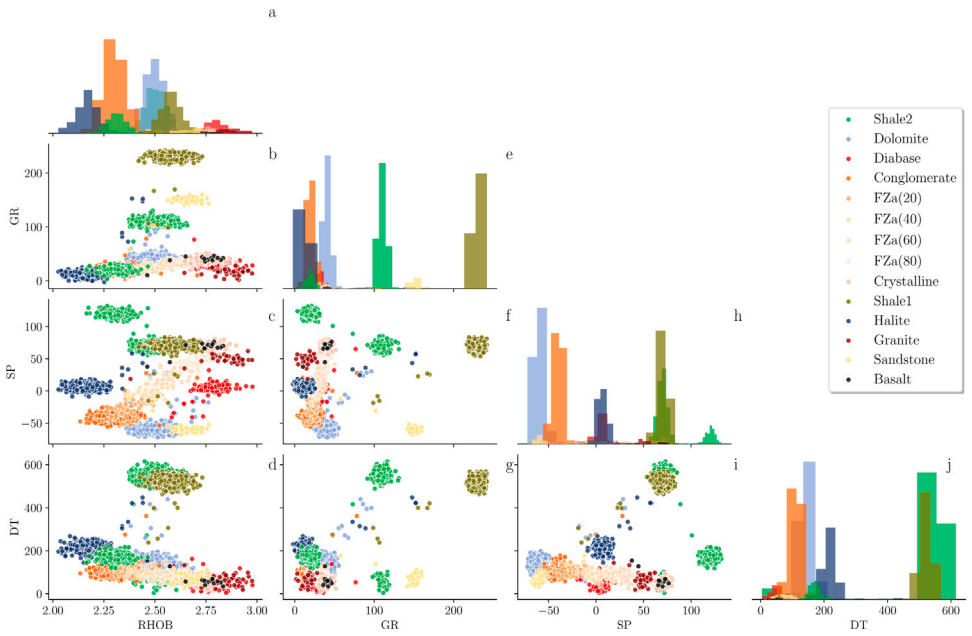


Fig. 7. Dispersion analysis for the training data-set one (TD1).

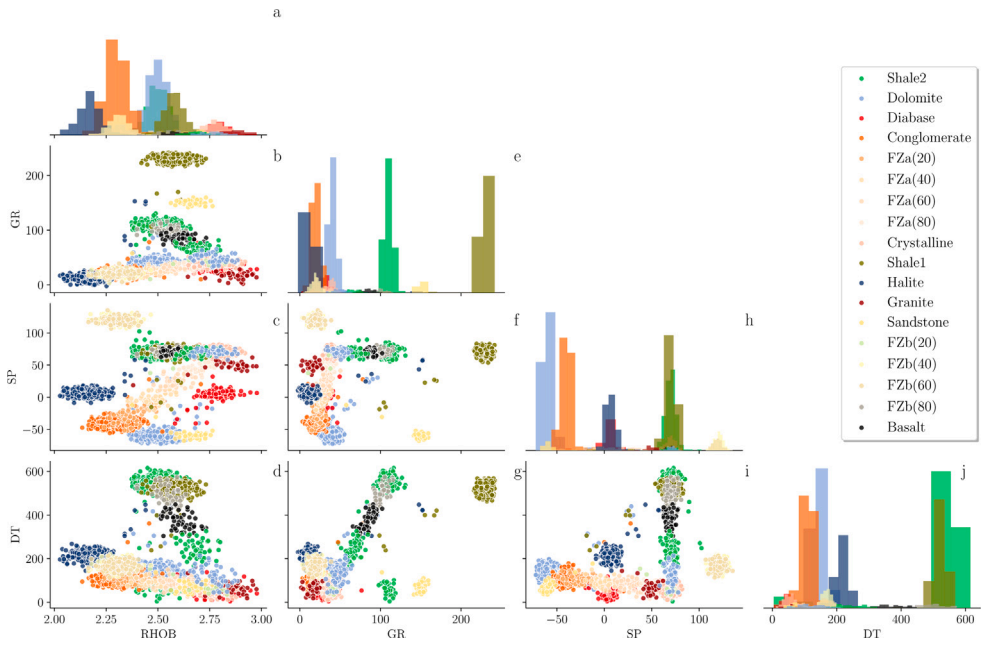


Fig. 8. Dispersion analysis for the training data-set two (TD2).

simply by computing the Euclidean distances (i.e., Eq. (3)) between each normalized log-data of the classification well and the trained and labeled set of weights. With this, we reinforce the relevance of a promising supervision stage for a proper use of a Self-organizing map.

3.3. Statistical analysis: Confusion matrix

In machine learning problems, a very efficient way to quantify the classification marks is through the Confusion matrix (Townsend, 1971). Commonly set as a 2 × 2 Matrix in a binary problem, the confusion matrix is represented by a table of true and predicted values for all classes.

True positives (TPs) are obtained when the classification of the ith class is correct. Similarly, true negatives are related to an accurate classification of the jth class. False positives and negatives are misclassifications of ith and jth classes, respectively (Townsend, 1971; Deng et al., 2016).

Exploiting the confusion matrix elements, we can define the accuracy (A) value by the following expression:

$$A = \frac{TP + TN}{n}, \tag{7}$$

where TP are the true positives, TN are the true negatives and n is the total amount of data. Eq. (7) quantifies the accuracy rate of the classification method. Additionally, if one needs a measure of the reliability of the acquired classification, a relevant parameter can be

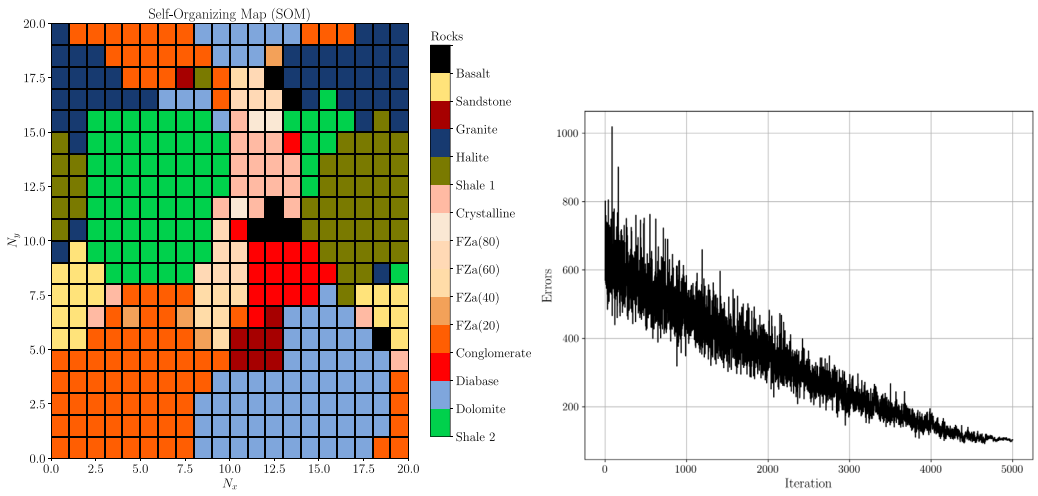


Fig. 9. (a) The rectangular distribution of 400 neurons in the cortex after supervision stage using TD1. Colors highlight each lithology identified in the cortex. Each cell is a trained neuron, whose training procedure is represented by a convergence curve in (b).

computed by the following:

$$P = \frac{TP}{TP + FP}, \tag{8}$$

where FP are false positives and P is referred to as Precision. Eq. (8) can be interpreted as a reliability metric related to the true-positive classes (i.e., the ith group).

3.4. Training data-sets

Supervised machine learning methods are extremely dependent on the quality of labeled data. Commonly, a more in-deep analysis based on data-mining procedures to define a promising training data-set is required. In our case, the pseudo-well formulation presented in Section 2 avoids the tedious screening of promising training data-sets. Based on the synthetic geologic scenario highlighted in Section 2, we define RHOB, GR, SP and DT logs for each lithology into the interpreted geologic cross-section. Additionally, to deal with the simulated fault zones exhibited in Fig. 1, we compute Eq. (1).

In our numeric experiments, we design two particular training data-sets to verify the relevance of a promising supervision. The first one consists of a decent amount of data for all lithologies presented in the cross-section and no data of the fault zone represented by the mixture of Crystalline and Shale2 (i.e., FZa). In the second training data-set, we add logs of this particular geologic structure simulated by mixture law, as previously explained.

3.4.1. Training data-set one (TD1)

Fig. 7 show a complete dispersion analysis of training data-set one, composed by 3091 data-samples. There are some overlapping for RHOB logs, specially for the case of similar lithologies, such as shale1 and shale2, dolomite and halite, as can be seen in histograms of Fig. 7a. On the other hand, GR crossplots show an important separability of shale1 and shale2, which is not observed in SP and DT plots. Additionally, we can see that there is few data about Fault zones (i.e., FZa20, FZa40, FZa60 and FZa80) in TD1. On the other hand, there is no log data of Fault Zones composed by the mixture between shale2 and crystalline (i.e., FZb20, FZb40, FZb60 and FZb80).

With this first training data-set, we simulate a particular task in a classification problem: a supervision guided by an incomplete training data-set.

3.4.2. Training data-set two (TD2)

Fig. 8 show a complete dispersion analysis of training data-set two, which is composed by 3539 samples. In this case, all rock units comprising both fault zones FZa(20), FZa(40), FZa(60), FZa(80) and FZb(20), FZb(40), FZb(60), FZb(80) are entirely considered in the training data-set two.

With these two particular training data-sets, we better explore the potentialities and limitations of SSOM for classification of lithofacies. Additionally, we also contribute with more synthetic-based experiments about supervised machine learning.

4. Results and discussions

To bespeak the capability of our SSOM in recovering rock units of the classification well (see Fig. 1), we design two controlled tests. In the first example, the training data-set one (TD1), composed by no log data about the fault zone Zb, simulated by the mixture law of shale2 and crystalline is considered. In the second numerical experiment, we use the complete training data-set two (TD2) for an improved supervision of our SSOM. We also highlight all rocks of the fault zone with gray color for a better visualization. Log plots and accuracy are also presented for a complete discussion of the recovered lithofacies. In all examples, we set a rectangular cortex with 400 neurons and all weights are randomly selected at the very beginning of the supervision procedure. The supervision stage runs over $T = 5000$ iterations and the damping coefficient for neighbors is $k = 1.9$ in both tests. For simulating all log data used in our experiments, for both training and classification data-set, we use a moving average of 3 data-samples.

4.1. Classification using TD1

Fig. 9(a) shows the SSOM after being supervised by training data-set one (TD1). We can observe that most of the rock units presented in the geologic cross-section are accurate represented in the cortex, such as Sandstone, Shale 2 and Crystalline. Some other rock units, such as basalt, granite, FZa(20) and FZa(40) are represented by very few neurons. In this particular case, the log-data computed by the mixture of shale 2 and crystalline are not presented in the TD1. In this particular case, the SSOM might confounds some specific rock units due to the spacial distribution of colors throughout the cortex. For example, crystalline (light pink) and basalt (black), conglomerate (orange) and dolomite (light blue), shale1 (dark green) and granite (dark red) are neighboring neurons of the cortex, which means that these units are

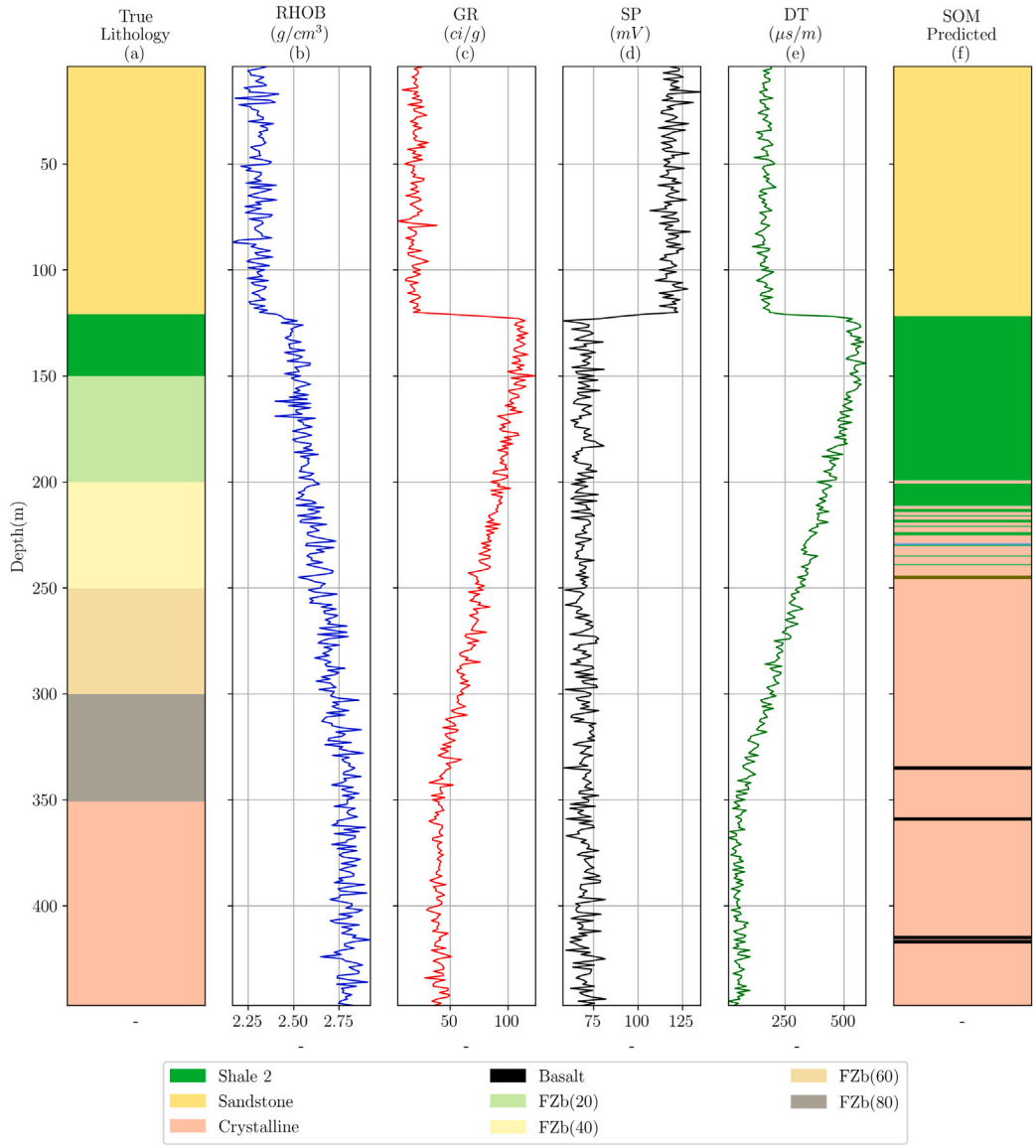


Fig. 10. Logplots for the Classification well using TD1. Colors indicate each rock unit comprising the well. (a) True Lithologic log, (b) RHOB log, (c) GR log, (d) SP log, (e) DT log, (f) Classified lithologic log using SSOM.

similar for SSOM. Additionally, we observe some discrepancies into the interfaces of different rock units, which might be related to the distribution of neighbors into the rectangular cortex. Despite, the convergence curve of Fig. 9(b) highlights a decent supervision procedure after 5000 iterations.

Fig. 10 show all logplots presented in this first test using TD1 as the training data-set. As expected, our SSOM did not recover the fault zone represented by the mixture of crystalline and shale 2, once the log data produced by such lithotypes are entirely missing in the training data-set one (TD1). Fig. 10(b–e) present log data used for classification. RHOB, GR and DT logs are quite sensitive to the transition from shale 2 to crystalline. An interesting aspect of Fig. 10(f) lies in small artifacts observed inside the fault zone, involving Dolomite, shale 1 and shale 2. At the bottom of well, thin layers of basalt are also encountered, despite this rock unit is not presented into the true well. This aspect shows that the SSOM misclassifies thin layers of basalt (around 350 and 400 meters depth) and dolomite (within 250 and 300 meters depth). This pattern is related to the similarities of such rock units with others better identified in the cortex, as stated in Fig. 9(a).

The shallow package of sandstone presented in the true well is accurately recovered by our SSOM. As expected, the SSOM using TD1 did not recovered the fault zone FZb accurately. Instead, similar rock units mapped in the cortex were classified. With this first example, we reinforce the relevance of an accurate training data-set for a successful classification using a supervised Self-Organizing Map.

The statistics for this first test shows an overall error of 45.1% (204 wrong classified samples out of 451). The accuracy obtained by Eq. (7) is around 54%, which is mainly due to not recovering the entire fault zone Zb. As a final remark, we can affirm that the SSOM showed expected results for an incomplete training data-set. In the following example, we improve the training data set with accurate information regarding the rock units within the classification well.

4.2. Classification using training data-set two

Fig. 11(a) shows the SSOM after supervision using the TD2. Now we can see some clusters in the cortex dedicated to the normal fault presented in the classification well. The SSOM is accurately represented by all lithologies comprising the well, such as shale 1, Halite and the faults FZa and FZb simulated by the mixture law. This configuration indicates a natural ability of our SSOM in classifying all rock units

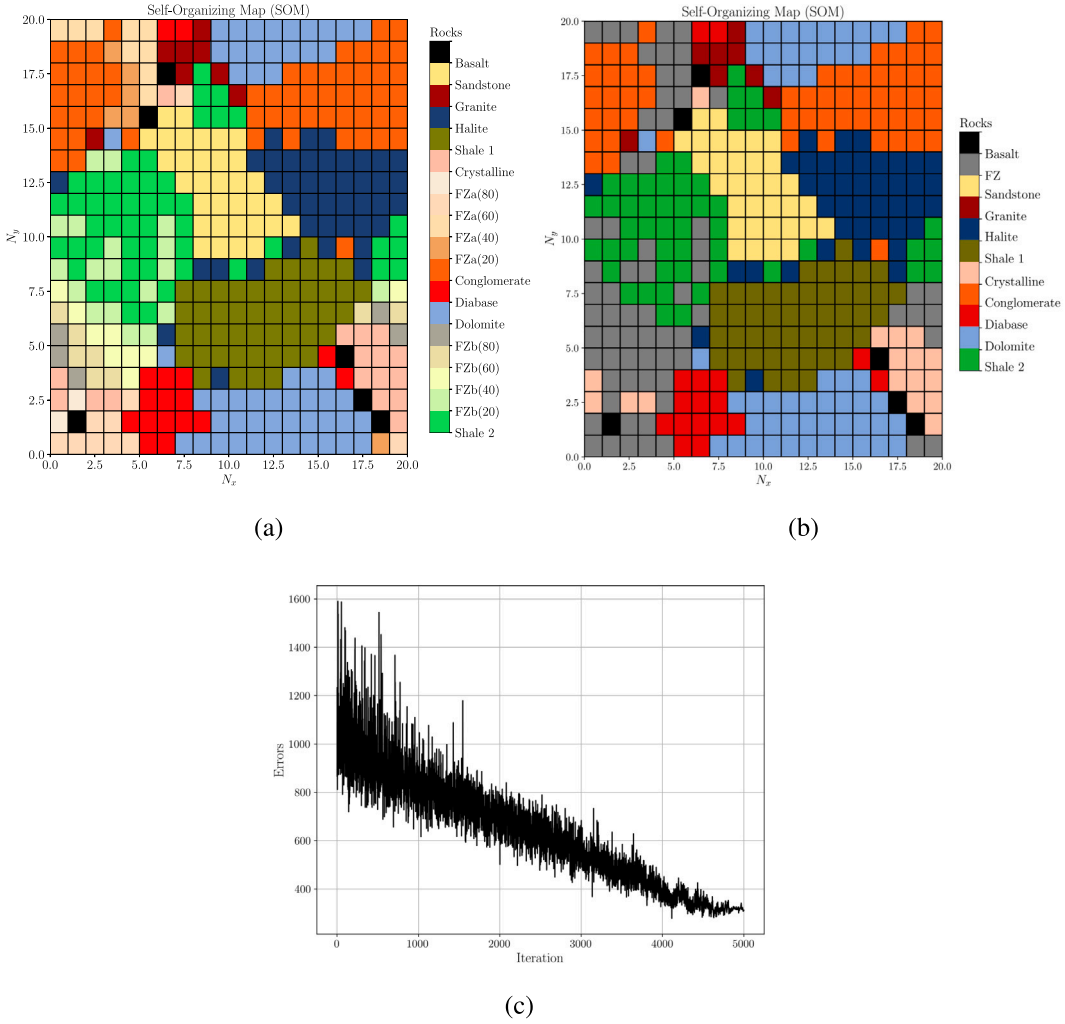


Fig. 11. (a) SSOM trained with TD2. Colors highlight each rock unit into the cortex. (b) The rectangular cortex showing all faults (i.e., FZa and FZb) in gray. (c) Convergence curve of the training procedure after 5000 iterations.

of the well. On the other hand, we can also observe some particular unclustered units, such as basalt (black), FZb(20) (light green), FZa(4) and crystalline (light pink) and shale 2. So, some misclassifications can be observed comprising the above-mentioned lithologies. With this analysis, we explore the importance of working with controlled training data-set. To facilitate the visualization and also the interpretation of the classified units, Fig. 11(b) highlights all supervised faults by dark gray. We can see that the fault zone, conglomerate, dolomite, shale 1, halite and sandstone produced well-defined clusters in the rectangular cortex. Additionally, Fig. 11(c) shows a reasonable convergence curve obtained during the supervision stage.

Fig. 12 shows all log plots presented in this second test. As expected, our SSOM is now capable of recovering the fault zone FZa represented by the mixture of shale 2 and crystalline. This is now possible once the log data produced by such lithotypes are entirely represented in the training data-set two (TD2). Fig. 12(b–e) present the logs used for classification. An interesting aspect of Fig. 10(f) lies in small artifacts observed inside the fault zone, especially in contacts among different mixtures. This might be related to the smoothing strategy applied to the synthetic log data. The shale 2 package around 150 m depth is classified with some thin layers of FZb (20), which is expected. At the bottom of well, a large package of crystalline is now recovered without artifacts. Once again, the shallow package of sandstone is entirely recovered.

Table 3
Performance comparison of our SSOM with different training data-sets. Accuracy (i.e., Eq. (7)) and error are computed in percentage. The sampling errors are calculated by the ratio between the number of misclassifications over the total amount of classified data.

	Accuracy(%)	Error(%)	Sampling-errors (n/n_t)
SSOM with TD1	54	45	204/451
SSOM with TD2	86	13	61/451

Fig. 10(g) shows the faults of classification well in dark gray. With this, we can observe that the entire fault zone is almost perfectly recovered, apart from very thin interbedded layers. After analyzing the second example, we can affirm that the SSOM is totally capable of recovering this particular geologic structure only if an accurate training data-set is considered.

The statistics for this test shows now an error of 13 % (61 wrong classified samples out of 451). The accuracy obtained by Eq. (7) is improved to 86.3%, which is mainly due to the small artifacts at the interfaces of different rock units inside the fault zone. As a final remark, the SSOM worked fine in recovering supervised lithologies, including the ones simulated by mixture law (i.e., Eq. (1)).

For a better comparison of the classification performance, Table 3 presents the error analysis of both tests.

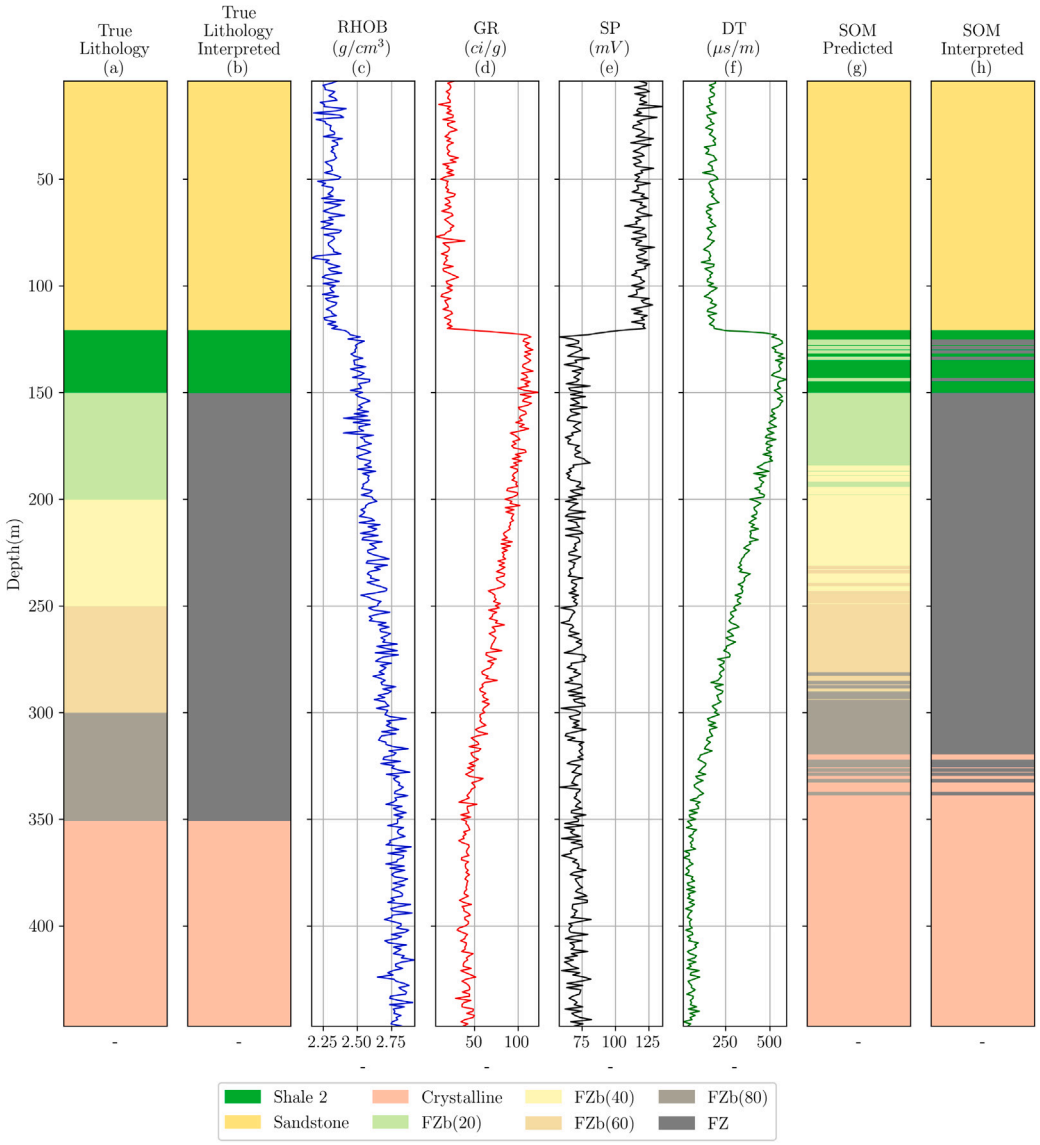


Fig. 12. Log plots for the classification well. Colors into the legend indicate each rock unit comprising the well. (a) Lithologic log, (b) Lithologic log representing the fault zone by dark gray, (c) RHOB log, (d) GR log, (e) SP log, (f) DT log, (g) Predicted lithologic log and (h) Predicted lithologic log highlighting the fault zone in dark gray.

5. Conclusions

In this work we design a complete workflow for solving a lithologic identification problem by using a supervised Self-Organizing Map. First, we design synthetic log data from a previous interpreted geologic cross-section of a specific sedimentary basin of Brazil. In this scenario, rock units and fault zones are depicted by colors in a sketch. Additionally, the faults are simulated by the mixture law involving the neighboring rocks. We then drill a pseudo-well in this particular geologic scenario and simulate density, gamma-ray, spontaneous potential and sonic logs. Two different training data-sets are considered for supervising the implemented Self-Organizing Map. The numerical experiments show a good accuracy, especially when the training procedure is performed with the more complete data-set. Additionally, more challenging regions in the pseudo-well, such as the fault composed by Shale 2 and Crystalline could be reasonably recovered by the supervised Self-Organizing Map presented in this work. On the other hand, when the training data-set is not representative of the true lithologies, the accuracy becomes an issue and so the classification. A relevant aspect of the investigation proposed herein lies in the total control of the relation among lithologies and the log data, which is something extremely challenging to achieve in real data applications. We hope that this work would definitively be helpful for interpreters in determining particular targets in a well, such as horizons or reservoirs. Additionally, we also propose more discussion about the pros and cons of supervising machine learning methods. For the future, improvements into the log-data simulations are required for more realistic data-sets. Another perspective lies in altering some of the relevant parameters of the Self-Organizing Map, such as proposing different cortex geometries, defining other neighbor connections and also computing different learning rates during the supervision stage.

CRediT authorship contribution statement

Carreira V.R.: Data curation, Investigation, Methodology, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing. **Bijani R.:** Data curation, Investigation, Methodology, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing. **Ponte-Neto C.F.:** Data curation, Investigation, Methodology, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

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

Acknowledgments

First author thanks Fundação de Amparo à Pesquisa do Estado do Rio de Janeiro (FAPERJ), Brazil and Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), Brazil for the scholarship. A special thanks to Mario Martins Ramos, Fernando Vizeu and Wagner Lupinacci for valuable suggestions during the development of this research. Authors also thanks editors and reviewers for precious comments and recommendations during the revision of this paper.

References

Adibifard, M., Tabatabaei-Nejad, S., Khodapanah, E., 2014. Artificial Neural Network (ANN) to estimate reservoir parameters in Naturally Fractured Reservoirs using well test data. J. Pet. Sci. Eng. 122, 585–594. http://dx.doi.org/10.1016/j.petrol.2014.08.007, URL: http://linkinghub.elsevier.com/retrieve/pii/S0920410514002563.

Bassiouni, Z., et al., 1994. Theory, Measurement, and Interpretation of Well Logs, vol. 4, Henry L. Doherty Memorial Fund of AIME, Society of Petroleum Engineers.

Baştanlar, Y., Ozuysal, M., 2014. Introduction to machine learning. Methods Mol. Biol. (Clifton N.J.) 1107, 105–128. http://dx.doi.org/10.1007/978-1-62703-748-8_7, arXiv:0904.3664v1.

Bestagini, P., Lipari, V., Tubaro, S., 2017. A machine learning approach to facies classification using well logs. In: Seg Technical Program Expanded Abstracts 2017. Society of Exploration Geophysicists, pp. 2137–2142.

Calderón-Macías, C., Sen, M.K., Stoffa, P.L., 2000. Artificial neural networks for parameter estimation in geophysics [Link]. Geophys. Prospect. 48 (1), 21–47.

Carneiro, C.d.C., Fraser, S.J., Crósta, A.P., Silva, A.M., Barros, C.E.d.M., 2012. Semi-automated geologic mapping using self-organizing maps and airborne geophysics in the Brazilian Amazon. Geophysics 77 (4), K17–K24. http://dx.doi.org/10.1190/geo2011-0302.1.

Chagas, E.S., Russo, S., Simon, V., 2010. Geração de perfil sônico sintético em poços de petróleo através dos modelos de regressão não lineares usando a profundidade como variável regressora. Sci. Plena 6 (12 (b)).

Cortes, C., Vapnik, V., 1995. Support-vector networks. Mach. Learn. 20 (3), 273–297.

Costa, I.S.L., Tavares, F.M., de Oliveira, J.K.M., 2019. Predictive lithological mapping through machine learning methods: a case study in the Cinzento Lineament, Carajás Province, Brazil. J. Geol. Surv. Braz. 2 (1), 26–36.

Dayan, P., Sahani, M., Deback, G., 1999. Unsupervised learning. In: The MIT Encyclopedia of the Cognitive Sciences. MIT Press, pp. 857–859.

Deng, X., Liu, Q., Deng, Y., Mahadevan, S., 2016. An improved method to construct basic probability assignment based on the confusion matrix for classification problem. Inform. Sci. 340–341, 250–261. http://dx.doi.org/10.1016/j.ins.2016.01.033.

Dias, L.O., Bom, C.R., Márcio, P., Marcelo, P., Faria, E.L., Correia, M.D., Bom, R., Faria, L., Correia, M.D., 2018. Comparação de métodos de segmentação de fraturas em imagem acústica de perfilagem petrofísica. Notas Téc. 8 (3), 7–19.

Dramsch, J.S., 2020. Chapter One - 70 years of machine learning in geoscience in review. In: Moseley, B., Krischer, L. (Eds.), Machine Learning in Geosciences. In: Advances in Geophysics, vol. 61, Elsevier, pp. 1–55. http://dx.doi.org/10.1016/bs.agph.2020.08.002, URL: https://www.sciencedirect.com/science/article/pii/S0065268720300054.

Dvorkin, J., 2020. Rock physics: Recent history and advances. Geophys. Ocean Waves Stud. 1–24.

Dvorkin, J., Wollner, U., 2017. Rock-physics transforms and scale of investigation. Geophysics 82 (3), MR75–MR88.

Ellis, D.V., Singer, J.M., 2012. second ed. Well Logging for Earth Scientists, vol. 33, Springer, New York, pp. 3–8. http://dx.doi.org/10.1073/pnas.0703993104, arXiv:arXiv:1011.1669v3.

Fisher, R.A., 1936. The use of multiple measurements in taxonomic problems. Ann. Eugen. 7 (2), 179–188.

Gonçalves, E.C., da Silva, P.N., Silveira, C.S., Carneiro, G., Domingues, A.B., Moss, A., Pritchard, T., Plastino, A., Azeredo, R.B.d.V., 2017. Prediction of carbonate rock type from NMR responses using data mining techniques. J. Appl. Geophys. 140, 93–101. http://dx.doi.org/10.1016/j.jappgeo.2017.03.014.

Guillen, P., Larrazabal*, G., González, G., Boumber, D., Vilalta, R., 2015. Supervised learning to detect salt body. In: SEG Technical Program Expanded Abstracts 2015. Society of Exploration Geophysicists, pp. 1826–1829.

Günther, F., Fritsch, S., 2010. Neuralnet : Training of neural networks. R J. 2, 30–38, URL: http://journal.r-project.org/archive/2010-1/RJournal{_}2010-1{_}Guenther+Fritsch.pdf.

Hall, P., Dean, J., Kabul, I.K., Silva, J., 2014. An Overview of Machine Learning with SAS® Enterprise Miner™. Vol. 2, SAS Institute Inc.

Han, L., Guijun, Y., Dai, H.-y., Yang, H., Xu, B., Li, H., Long, H., Li, Z., Xiaodong, Y., Zhao, C., 2019. Combining self-organizing maps and biplot analysis to preselect maize phenotypic components based on UAV high-throughput phenotyping platform. Plant Methods 15, http://dx.doi.org/10.1186/s13007-019-0444-6.

Haykin, S., 2001. Neural Networks: Principles and Practice, second ed. McMaster University, Hamilton, Ontario - Canada.

Hoan, N.Q., 2016. Improving feature map quality of SOM based on adjusting the neighborhood function. Int. J. Comput. Sci. Inf. Secur. (IJCSIS) 14 (9).

Jayalakshmi, T., Santhakumaran, A., 2011. Statistical normalization and back propagation for classification. Int. J. Comput. Theory Eng. 3 (1), 1793–8201.

Kanal, L.N., 2001. Perceptrons. In: Encyclopedia of Computer Science. pp. 11215–11218.

Kohonen, T., 1989. Self-organizing feature maps. In: Self-Organization and Associative Memory. Springer, pp. 119–157.

Kohonen, T., 2013. Essentials of the self-organizing map. Neural Netw. 37, 52–65. http://dx.doi.org/10.1016/j.neunet.2012.09.018.

Konaté, A.A., Pan, H., Khan, N., Ziggah, Y.Y., 2015. Prediction of porosity in crystalline rocks using artificial neural networks: an example from the Chinese continental scientific drilling main hole. Stud. Geophys. Geod. 59 (1), 113–136.

Kostorz, W., 2021. A practical method for well log data classification. Comput. Geosci. 25 (1), 345–355.

Kotsiantis, S.B., Zaharakis, I., Pintelas, P., 2007. Supervised machine learning: A review of classification techniques. Emerg. Artif. Intell. Appl. Comput. Eng. 160 (1), 3–24.

Krogh, A., 2008. What are artificial neural networks? Nature Biotechnol. 26 (2), 195–197. http://dx.doi.org/10.1038/nbt1386, URL: http://www.ncbi.nlm.nih.gov/pubmed/18259176.

Kumar, B., Kishore, M., 2006. Electrofacies classification–a critical approach. In: 6th International Conference & Exposition on Petroleum Geophysics, New Delhi, India. pp. 822–825.

Kuroda, M.C., Vidal, A.C., Leite, E.P., Drummond, R.D., 2012. Electrofacies characterization using self-organizing maps. Braz. J. Geophys. 30 (3).

Kuyuk, H., Yildirim, E., Dogan, E., Horasan, G., 2012. Application of k-means and Gaussian mixture model for classification of seismic activities in Istanbul. Nonlinear Process. Geophys. 19 (4), 411–419.

Lachaux, M.-A., Roziere, B., Chanussot, L., Lample, G., 2020. Unsupervised translation of programming languages. arXiv preprint arXiv:2006.03511.

Lechner, M., Hasani, R., Amini, A., Henzinger, T.A., Rus, D., Grosu, R., 2020. Neural circuit policies enabling auditable autonomy. Nat. Mach. Intell. 2 (10), 642–652.

Levy, S., 1997. The computer. Newsweek 130 (22), 28, URL: file:///D:/Dropbox/Whitfield/History/SemesterTwo/ChangingTimes/Research/EBSCOhost2.htm.

Li, H., Phung, D., 2014. Scikit-learn: Machine learning in Python. J. Mach. Learn. Res. 39 (2014), i–ii.

Lindberg, D.V., Rimstad, E., Omre, H., 2015. Inversion of well logs into facies accounting for spatial dependencies and convolution effects. J. Pet. Sci. Eng. 134, 237–246. http://dx.doi.org/10.1016/j.petrol.2015.09.027.

Lloyd, S., 1982. Least squares quantization in PCM. IEEE Trans. Inform. Theory 28 (2), 129–137.

MacKay, D.J.C., 2005. Information Theory, Inference, and Learning Algorithms David J.C. MacKay, vol. 100, pp. 1–640. http://dx.doi.org/10.1198/jasa.2005.s54.

Mao, J., 1996. Why artificial neural networks? Communications 29, 31–44. http://dx.doi.org/10.1109/2.485891, URL: http://ieeexplore.ieee.org/xpls/abs{_}all.jsp?arnumber=485891.

McCulloch, W.S., Pitts, W., 1943. A logical calculus of the ideas immanent in nervous activity. Bull. Math. Biophys. 5 (4), 115–133.

Michie, E.D., Spiegelhalter, D.J., Taylor, C.C., 1994. Machine learning , neural and statistical classification. Technometrics 37 (4), 459. http://dx.doi.org/10.2307/1269742.

Mohriak, W., Szatmari, P., Anjos, S., 2008. Salt: Geology and Tectonic. Exemples on Brazilian's Sedimentary Basins, first ed. Beca, São Paulo, SP., (in Portuguese).

Nery, G.G., 2013. Perfilagem Geofísica em Poço Aberto: Fundamentos Básicos Com Ênfase em Petróleo. SBGf, Rio de Janeiro.

Neyamadpour, A., Taib, S., Abdullah, W.W., 2009. Using artificial neural networks to invert 2D DC resistivity imaging data for high resistivity contrast regions: A MATLAB application. Comput. Geosci. 35 (11), 2268–2274.

Papadimitriou, S., Mavroudi, S., Vladutu, L., Pavlides, G., Bezerianos, A., 2002. The supervised network self-organizing map for classification of large data sets. Appl. Intell. 16 (3), 185–203.

Pashin, J.C., Pradhan, S.P., Vishal, V., 2018. Chapter thirteen - formation damage in coalbed methane recovery. In: Yuan, B., Wood, D.A. (Eds.), Formation Damage During Improved Oil Recovery. Gulf Professional Publishing, pp. 499–514. http://dx.doi.org/10.1016/B978-0-12-813782-6.00013-0, URL: https://www.sciencedirect.com/science/article/pii/B9780128137826000130.

Pastukhov, A.A., Prokofiev, A.A., 2016. Kohonen self-organizing map application to representative sample formation in the training of the multilayer perceptron. St. Petersburg Polytech. Univ. J.: Phys. Math. 2 (2), 134–143. http://dx.doi.org/10.1016/j.spjpm.2016.05.012.

Perol, T., Gharbi, M., Denolle, M., 2018. Convolutional neural network for earthquake detection and location. Sci. Adv. 4 (2), 2–10. http://dx.doi.org/10.1126/sciadv.1700578.

Raudys, Š., 1998. Evolution and generalization of a single neurone: I. Single-layer perceptron as seven statistical classifiers. Neural Netw. 11 (2), 283–296. http://dx.doi.org/10.1016/S0893-6080(97)00135-4.

Reynolds, D.A., 2009. Gaussian mixture models. In: Encyclopedia of Biometrics. Vol. 741, Springer, Berlin, pp. 659–663.

Ruvini, J.-D., Dony, C., 2000. APE: learning user's habits to automate repetitive tasks. In: Proceedings of the 5th International Conference on Intelligent User Interfaces. pp. 229–232.

Sahoo, S., Jha, M.K., 2017. Pattern recognition in lithology classification: modeling using neural networks, self-organizing maps and genetic algorithmsReconnaissance des caractéristiques d'une classification lithologique: modélisation utilisant des réseaux neuronaux, des cartes aut. Hydrogeol. J. 25 (2), 311–330. http://dx.doi.org/10.1007/s10040-016-1478-8.

Saporetti, C.M., da Fonseca, L.G., Pereira, E., 2019. A lithology identification approach based on machine learning with evolutionary parameter tuning. IEEE Geosci. Remote Sens. Lett. 16 (12), 1819–1823.

Schrider, D.R., Kern, A.D., 2018. Supervised machine learning for population genetics: a new paradigm. Trends Genet. 34 (4), 301–312.

Shoji, D., Noguchi, R., Otsuki, S., Hino, H., 2018. Classification of volcanic ash particles using a convolutional neural network and probability. Sci. Rep. 8 (1), 1–12. http://dx.doi.org/10.1038/s41598-018-26200-2.

da Silva, A.A.N., Bahia, B.F., Sant'ana, T.C.S., Holz, M., 2015. Modelagem de perfis geofísicos sintéticos para possibilitar a amarração sísmica-poço na Bacia do Recôncavo. In: 14th International Congress of the Brazilian Geophysical Society & EXPOGEF, Rio de Janeiro, Brazil, 3-6 August 2015. Brazilian Geophysical Society, pp. 1121–1126.

Townsend, J.T., 1971. Erratum to: Theoretical analysis of an alphabetic confusion matrix. Percept. Psychophys. 10 (4), 256. http://dx.doi.org/10.3758/BF03212817.

Valentín, M.B., Bom, C.R., Coelho, J.M., Correia, M.D., de Albuquerque, M.P., de Albuquerque, M.P., Faria, E.L., 2019. A deep residual convolutional neural network for automatic lithological facies identification in Brazilian pre-salt oilfield wellbore image logs. J. Pet. Sci. Eng. 179 (April), 474–503. http://dx.doi.org/10.1016/j.petrol.2019.04.030.

Ventrella, J., 2011. Glider dynamics on the sphere: Exploring cellular automata on geodesic grids. J. Cell. Autom. 6 (2–3), 245–256.

Weyn, J.A., Durran, D.R., Caruana, R., 2019. Can machines learn to predict weather? Using deep learning to predict gridded 500-hPa geopotential height from historical weather data. J. Adv. Modelling Earth Syst. 11 (8), 2680–2693.

Weyn, J.A., Durran, D.R., Caruana, R., 2020. Improving data-driven global weather prediction using deep convolutional neural networks on a cubed sphere. J. Adv. Modelling Earth Syst. 12 (9), e2020MS002109.

Wu, C.-J., Brooks, D., Chen, K., Chen, D., Choudhury, S., Dukhan, M., Hazelwood, K., Isaac, E., Jia, Y., Jia, B., et al., 2019. Machine learning at facebook: Understanding inference at the edge. In: 2019 IEEE International Symposium on High Performance Computer Architecture. HPCA, IEEE, pp. 331–344.

Zhang, D., Yuntian, C., Jin, M., 2018. Synthetic well logs generation via Recurrent Neural Networks. Pet. Explor. Dev. 45 (4), 629–639.