Using Probabilistic Unsupervised Neural Method for Lithofacies Identification

Salim Chikhi and Mohamed Batouche
Computer Science Department, Mentouri University, Algeria

Abstract: *This paper presents a probabilistic unsupervised neural method in order to construct the lithofacies of the wells HM2 and HM3 situated in the south of Algeria (Sahara). Our objective is to facilitate the experts' work in geological domain and to allow them to obtain the structure and the nature of lands around the drilling quickly. For this, we propose the use of the Self-Organized Map (SOM) of Kohonen. We introduce a set of labeled log's data in some points of the hole. Once the obtained map is the best deployed one (the neuronal network is well adapted to the data of the wells), a probabilistic formalism is introduced to enhance the classification process. Our system provides a lithofacies of the concerned hole in an aspect easy to read by a geology expert who identifies the potential for oil production at a given source and so forms the basis for estimating the financial returns and economic benefits. The obtained results show that the approach is robust and effective.*

Keywords: *Lithofacies, differed well logging, self-organized map, probabilistic formalism, classification, underground cores.*

Received October 4, 2003; accepted January 3, 2004

1. Introduction

Lithofacies identification is important for many geological and engineering disciplines. It consists of a rock type identification and can be used to correlate the important characteristics of a geologic unit, such as mineralogy, depositional fabric, or fossil content [21]. For petroleum reservoir characterization, the primary task is to identify lithofacies of the reservoir rocks. Conventionally, one identifies lithofacies by direct observation of underground cores which are small cylindrical rock samples retrieved from wells at selected well depths. One can think currently that, if formations were crossed completely in continuous coring, the registration of well loggings would not impose itself. In this case the geologist and the reservoir engineer would arrange some necessary land samples indeed to their studies; but while acting of the way, they would be deprived of :

- Information with difficulty accessible and expensive to get (natural radio-activity, ...).
- A continuous vision of formations.
- The possibility of quantification and automatic treatment by computer that well logging offers.
- The permanence of information.

The recuperation of cores is not always total and the risk of shift in information is not negligible. It is why a lower-cost method providing similar or higher accuracy is desirable. In this paper, we use differed well logging which consists on a set of records of a digital measurements obtained along the depth of the oil and gas wells. This method provide indirect information about the subsurface and is far less expensive. Our purpose is to describe an automated method, based on neural networks, for predicting rock characteristics from differed well-log data. Using differed well logging measurements, some researchers in geology have recently employed statistical methods [6, 23] and Back Propagation Neural Networks (BPNNs) in an attempt to improve performance in solving such geological problems [13, 24, 25, 26]. However, BPNNs have several significant disadvantages. First, convergence during training is slow and there is no guarantee of reaching the user-defined acceptable error range. Second, when test data are located outside the training data range, BPNNs cannot classify them; thus, the discriminating ability is not assured. BPNNs adequately deal with well-bounded and stable problems, because training sets may cover the entire expected input space. Unfortunately, in reservoir characterization problems, variables commonly are neither well-bounded nor stable. New lithofacies and new values of important rock properties are often encountered.

We chose a SOM neural networks with a probabilistic variant PRSOM to propose a logfacies-recognizer system (geonal). The global conception scheme of the *geonal* system is described in Figure 1. The learning algorithms of SOM and PRSOM are unsupervised, which adapt the map to a set of learning samples. Those algorithms allow realizing a partition of the data space, with each subset associated to a neuron of the map. PRSOM is a probabilistic model which associates to a neuron c of the map a spherical Gaussian density function fc, it approximates the data's density distribution using a mixture of normal distributions.

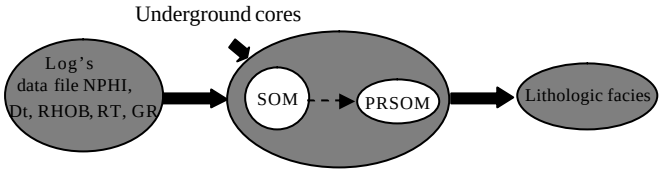

Figure 1. Global conception scheme of the geonal system.

2. Supervised and Unsupervised Methods

Neural networks learning may be broadly grouped as supervised and unsupervised. In supervised learning, the network learns from a training set consisting of inputs and the desired outputs. Learning is accomplished by adjusting the network weights so that the difference between the desired outputs and the network computed outputs is minimized. BPNNs are examples of supervised learning. Unsupervised learning [21, 25] requires only input data. During the learning process, the network weights are adjusted so that similar inputs produce similar outputs. Kohonen's Self Organizing Maps (SOMs) [17, 19], and Adaptive Resonance Theory (ART) neural networks [13, 26] are examples of unsupervised learning. They extract statistical regularities from the input data automatically rather than using desired outputs to guide the learning processes. Several researchers employed pattern recognition with unsupervised learning neural networks as pattern-recognizers; e. g. SOMs and ART neural networks [16, 22], to solve a lithofacies identification problem.

3. Physical Properties of Rocks

Differed well logging [2, 12] are records of geological properties of subsurface rock formations at depth retrieved by electrical, physical, or radioactive devices. They permit to do several measures. For the purpose of our survey we chose the following measures: Neutron Porosity log (NPHI), Sonic log (Dt), Bulk Density log (RHOB), Gamma Ray log (GR), Deep Resistivity (RT). We describe briefly the usefulness of each parameter used in this study for the identification process of rocks.

- Neutron Porosity log (NPHI): It measures the rock's reaction to fast neutron bombardment. The unit is dimensionless. The recorded parameter is an index of hydrogen for a given formation of lithology, generally of the limestone (as well as sandstone or dolomite). NPHI is useful mainly for: Lithology identificationm, Porosity evaluationm, and differentiation between liquids and gases (in combination with the density).
- Travel time or Sonic log (Dt): The unit is *µs/ft*. It measures the variations of the speed of acoustic wave propagation according to the depth. It must be done in "open" hole, that means before the pose of the protective intubation. It is useful mainly for: Determination of the porosity in a non clayey formation, and identification of the lithology (in combination with neutron and density).
- Bulk density log (RHOB): The unit is Grams per Cubic Centimeter (g/cc). It measures the bulk density of rock by measuring the bombardment of medium-high energy gamma rays. Densities of the most fluently measured evolve between 2 and 3 g/cm3. RHOB is useful mainly for: Determination of the porosity in zones to hydrocarbons or in formations containing the clay, and differentiation between liquids and gases (in combination with Neutron).
- Gamma Ray log (GR): The unit is American Petroleum Institute *(API)*. It measures the radioactivity of rocks and is useful mainly for: Geological correlations, correlations of depths, differentiation between clean zones and clayey zones, and evaluation of the content in clay of formations.
- Deep Resistivity (RT): The unit is ohm. It measures the fluid resistivities at deep vicinity of the wells by using long focusing electrodes and a distant return electrode. The rock resistivity measure is a basic data for all reservoir evaluation. It is useful mainly to have an idea about the quantity of water in the rock, therefore the porosity and saturation of the lithology, the nature and percentage of clay, nature and percentage of mineral, and the texture of the rock (fashion of pore distribution).

In this study all logs are scaled uniformly between –1 and +1 and results are given in scaled domain. Figure 2 shows the well logging measurements (logfacies) of well HM3.

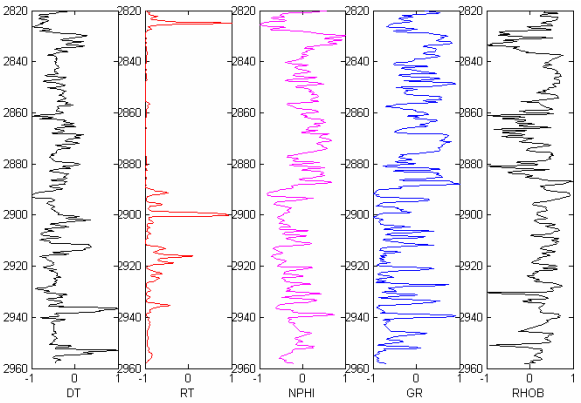

Figure 2. The logfacies of well HM3 between 2820 and 2960 meters.

4. The Self-Organizing Map

To represent the facies of the drilling holes HM2 and HM3, we used a Self-Organizing Map (SOM) [17, 19]. The SOM algorithm realizes a partition of the data space which permits to affect each vector of the data space to a particular neuron on the map. We have used available underground cores (only 477 points) as labels. Then we know the exact nature of the rocks present at the corresponding depth which is used to label each

neuron of the map (see Figure 3). Thus, the map becomes a classifier. We have used log's data from the well HM2 as a learning base for training the map and then, when the network is stable and represent relatively well the data space, we reuse the map with log's data from the well HM3. The best map obtained has a size 14x6 (see Figure 4) and the lithofacies obtained is shown in Figure 5.

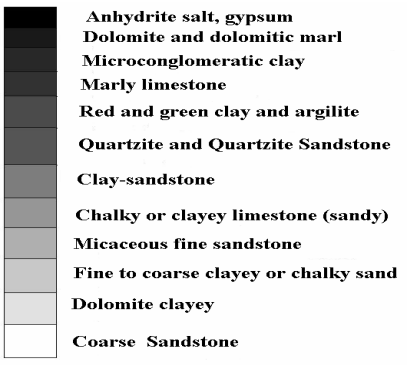

Figure 3. Legend of rock classes.

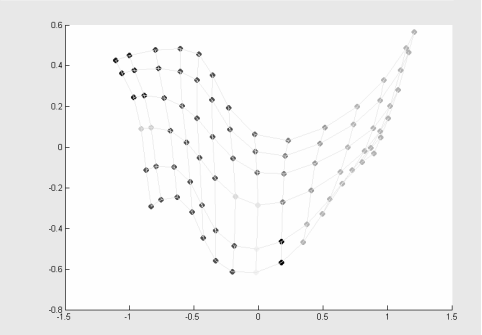

Figure 4. 3D projection of the best map (14x6) obtained for HM3 (HM2 as learning base).

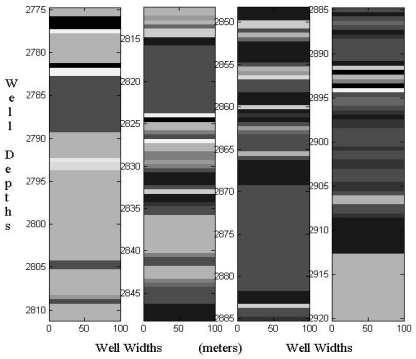

Figure 5. Lithofacies obtained for HM3 by SOM method (HM2 as learning base).

5. The PRSOM Algorithm

Now, consider the second algorithm that we used to obtain a topological map. We already have a good map and we will use this algorithm to refine a map obtained by the SOM algorithm.

5.1. Principle of the Algorithm

A first version of this algorithm can be found in [10]. PRSOM introduces the notion of density in the topological map. We used data obtained from different observation subsets, each of these subsets being governed by a normal law. Every neuron is the representative of one of these subsets. By the action of the neighborhood a neuron gives account of the subset that it represents, and also of subsets represented by its neighbors. Therefore, each neuron represents a mixture of densities. The data set is discerned as the mixture of the mixture of densities represented by every neuron. A neuron is constituted therefore of 2 elements: A vector of 5 measurements that corresponds to the mean vector, to which we add a supplementary value that represents the variance of the normal law that is associated to it [5, 10, 15]. A neuron defines a gaussien density function. To limit calculations, we consider that the variance is identical in all directions. When an observation is presented to the network, the PRSOM algorithm calculates the activation provoked by this observation in the density function of every neuron of the map. The winning neuron is the one with the strongest activation. All neurons of the map are modified after the presentation of all observations of the training set (one cycle). To modify parameters of a neuron c, average and variance, PRSOM takes in account the influence exercised by all observations of the cycle. This influence is pondered by the relation of neighborhood that exists between the winning neuron to which is affected the observation and the neuron c of which we want to modify parameters. The sum of influences exercised by observations is normalized then by the sum of neighborhood relations binding the neuron c to the neuron winning every observation.

As in the SOM algorithm a "temperature" adjusts the influence of the neighborhood. In PRSOM, this temperature is calculated at the beginning of every cycle to remain constant during all the cycle.

5.2. PRSOM Algorithm

First of all, it is necessary to fix some parameters:

- The shape and the dimension of the map.
- The number of neurons on the map.
- The maximum temperature that generates the neighborhood of size maximum.
- The minimum temperature that generates the neighborhood of size minimum.
- The number of cycles wished.

Then each cycle of the algorithm takes place according to the principle shown in Figure 6.

5.2.1. Assessment of the Temperature of the Cycle

The first operation of the PRSOM training cycle consists in valuing the temperature that will be constant during all the cycle:

$$h(t) = hMax * \left(\frac{hMin}{hMax}\right)^{\frac{1}{cycleMax - 1}} \quad (1)$$

where:
h(t): The temperature at the iteration t.
hMin and hMax: The minimum and the maximum temperature parameters fixed in the beginning of algorithm.
N: The number of observations in the training set.
cycleMax: The maximum number of cycles.

5.2.2. Competition Phase

All observations of the training set are studied in order to choose the winning neuron for each of them. When we present an observation to the network, we look at the behavior of every neuron towards it. For this, we make the hypothesis that the observation is descended of the mixture of densities covered by the neuron. A neuron being provided of an average and a variance, we can calculate the value of the activation provoked by the observation in the normal law of the neuron (equation (2)). The winner neuron is the one for which this activation is the strongest (equation (3)).

$$Rbf(z,c) = \frac{1}{\left(\sqrt{2p} * s_c\right)^n} * \exp\left[-0{,}5 * \sqrt{\sum_{i=1}^{n}\left[\frac{\left(z_{[i]} - w_{c[i]}\right)}{s_c}\right]^2}\right] \quad (2)$$

$$g = X(x) = \underset{C}{argmax}\,(Rbf(z, c)) \quad (3)$$

where:
c: Any neuron of the map.
w_c: The mean vector associated to the neuron c.
$w_{c[i]}$: The i^{th} component of the vector w_c.
s_c: The variance of the neuron c.
z : The observation vector presented to the map input.
$z_{[i]}$: The i^{th} component of the vector z.
g: The winning neuron of the vector z.
n: The dimension of the observation space, in our case n = 5.

5.2.3. Adaptation Phase

To proceed to the calculation of the parameters average and variance of every neuron c of the map, PRSOM consider every observation of the training set. Several heaps are incremented by the impact of every observation:

- Values of each vector component of the input observation are first treated separately. They are pondered by the tie of neighborhood that binds the winning neuron to the neuron c that we want to modify, then they are added in zones of separated heaps. These heaps are destined to the calculation of the different components of the mean vector of the neuron *c* (equation (5)).
- The quadratic distance heap between the observation z and its winning neuron, weighted by the neighborhood, is destined to the calculation of the variance of the c neuron.

The heap of neighborhood relations is used to normalize the calculation of the different components of the mean vector, as well as the calculation of the variance (equation (6)). These values are not modified in an adaptive way all along the treatment, as the SOM fact. To each end of cycle, PRSOM calculates the average and the variance of every neuron globally so that its associated normal law regains the mixture of densities defined by itself and its neighbors.

The influence zone around a neuron depends on the topological distance. The importance of the contribution brought by an observation depends on a function of neighborhood that is a kernel function K similar to the function of neighborhood used in SOM. The metrics used to establish the distance between two neurons is the topological distance that is to say the difference between indexes permitting to situate neurons on the map. In the implementation, we used a very simplified function of neighborhood which is given by equation (4). We ponder by this function of neighborhood the contribution of the observation to the calculation of the mean and variance of the neuron according to the equations (5) and (6).

$$K_{h(t)}(d(c,g)) = \exp\left[-0{,}5 * \frac{d(c,g)}{h(t)}\right] \quad (4)$$

$$w_c = \frac{\sum_z \left(K_{h(t)}(d(c, j(z))) * z_{[i]}\right)}{\sum_z K_{h(t)}(d(c; c(z)))} \quad (5)$$

$$s_c = \sqrt{\frac{\sum_z \left[K_{h(t)}(d(c, c(z))) * \sum_{i=1}^{n}\left(z_{[i]} - w_{c(z)[i]}\right)^2\right]}{\left[\sum_z K_{h(t)}(d(c, c(z)))\right]^n}} \quad (6)$$

where:
c: The neuron to modify.
g: The winning neuron of the influential observation.
$h^{(t)}$: The temperature of the t cycle.
$K^{h(t)}$: The function of neighborhood to the t cycle.
d(g,c): The topological distance between the neuron c. to modify and the winning neuron g.
w_c: The mean vector associated to the neuron c.
s_c: The variance of the neuron c.

5.3. Learning and Processing

The data of wells HM2 and HM3 are normalised between –1 and +1 (Figure 2). Then vectors corresponding to the 5 log measures: RHOB, PEF, NPHI, GR, and DT are all presented (Figure 1) to the SOM algorithm. We use also log's data from the well

HM2 as a learning base for training the map and then, we reuse the network, obtained in 200 cycles, with log's data from well HM3. Then this map is considered by the PRSOM algorithm.

Initially, averages are maintained constant to allow every neuron to value its variance. Then, every neuron is provided of a normal law. As shown in Figure 7, 40 cycles are sufficient for the quantification error stabilization (see next section). Then, averages are freed in order to allow a simultaneous updating of variances and averages. One thousand cycles are launched with a very weak temperature, varying from 3 to 1 so that to disturb the least possible the map structure. Figure 6 presents the classifier performances according to the map size and the used algorithm. We divided the training set into 3 groups:

- Observations that correspond to zones of underground cores. We have for these data the exact class that it is necessary to get. We will call these 477 data "Core Set".
- Observations that correspond to a zone for which the expert provided an approximation of rock classes composing the facies. We will call "Expert Set" the totality of these 706 data. The Expert Set only concerns the HM2 well.
- Well logging measures for which we didn't have cores, nor evaluation of the expert. We cannot provide any quantitative appreciation of results for these data.

Table 1 contains 3 columns: SOM, PRSOM V (update variances with constant averages), and PRSOM V + M (simultaneous updating of variances and averages). We see that the only variance establishment does not modify greatly the results. On the other hand modifications brought by the simultaneous updating of averages and variances are important. We note that the PRSOM algorithm almost always improves topological map performances developed by the SOM algorithm.

5.4. The Numeric Criteria

Table 1 shows the method performances by considering "qualitative" errors i. e. by considering the geological aspect of the obtained lithofacies. Here, we measure the PRSOM performances by considering "quantitative" errors. We value the quality of the training through a criteria that we called Quantification error (*Qe*) that permits to appreciate the quality of regroupings. It is about the root squared of intra-class variance. The calculation of the *Qe* measures the distance between neurons of the map and observations that they represent (equation (7)). The reduction of *Qe* gives account of the neuron adaptation to partitions that they defined. Figure 8 shows that *Qe* begins to stabilize from 40 cycles, we have the certainty that the learning process is correctly performed. We stop the execution of PRSOM if *Qe* stabilizes completely or if the maximum number of cycles is reached.

$$Qe = \sqrt{\frac{\sum_{z}\sum_{i}\left(z[i]-w_g[i]\right)^2}{N}} \qquad (7)$$

where:

z: Any observation of the training set.

g: The winning neuron for the z observation.

w_g: The weight vector associated to the winning neuron g.

$w_g[i]$: The i^{th} component of the vector wg.

N: The number of observations forming the training set.

1. Evaluate the temperature h of the cycle according to the equation (1).
2. For every observation z of the traioning set:
   - Present the observation z to the classifier.
   - Calculate the value of the activation produced in the function of density of every neuron c, according to the equation (2).
   - Choose the winning neuron g according to the equation (3).
   - Read the following observation.
3. At the end of the cycle:
   For every neuron c of the map:
   For every observation z of the training set:
   - Establish the topological distance d (c,g) between the neuron c and the winning neuron g of this observation z.
   - Evaluate the neighborhood between the winning neuron g and the neuron c, according to the equation (4).
   - Add the contribution of the observation z to heaps constituting numerators and denominators of equations (5) and (6).

   Calculate the mean vector of the neuron c, according to the equation (5).
   Calculate the variance of the neuron c, according to the equation (6).
4. Return to 1 until to all cycles wanted have been done.

Figure 6. The PRSOM algorithm.

Table 1. Improvement of performances by PRSOM algorithm.

Size of the Map	Percentage of Error Number over the Core Set (%) 477 Observations			Percentage of Error Number over the (Core + Expert) Set (%) 1183 Observations			Total Number of Neurons	Number of Used Neurons
	SOM	PRSOM V	PRSOM V+M	SOM	PRSOM V	PRSOM V + M		
7x13	14,69	13,50	10,18	32,74	34,16	29,98	91	91
7x20	11,37	13,27	10,90	36,65	35,05	30,42	140	138
6x14	12,32	11,13	08,5	36,47	35,58	25,98	84	84
10x10	13,50	12,32	09,71	37,07	35,76	30,33	100	100
20x20	07,81	08,76	05,21	33,27	28,29	26,42	400	350
25x25	05,1	05,92	00,71	28,20	24,73	19,57	625	515

5.5. PRSOM Algorithm Properties

5.5.1. Convergence Proof

During the execution of the algorithm we obtain a sequence of weight sets W^0, W^1,…, W^t,…, and a sequence of variance sets s^1,.., s^t,… such that for all t iteration, the cost function E verifies [5, 10]:

$$E(?^t, W^t, s^t) = E(?^{t-1}, W^t, s^t) = E(?^{t-1}, W^{t-1}, s^{t-1})$$

The competition phase or (affectation phase) is executed by using the affectation function ? (equation (3)). The *argmax* function returns the index c for which the probability is the highest. Then, we obtain the inequality: $E(?^t, W^t, s^t) = E(?^{t-1}, W^t s^t)$. The adaptation phase or (minimisation phase) is executed by using the steepest descent method which allows to obtain the inequality: $E(?^{t-1}, W^t s^t) = E(?^{t-1}, W^{t-1} s^{t-1})$. Since the function E (?, W, s) decreases at each iteration, the algorithm converges in a limited number of iterations. The stationary point is a local minimum of E (?, W, s).

5.5.2. Similarity with Radial-Bias Networks

Originally, the radial-bias functions approach is an exact interpolation technique in a multidimensional space. The problem of exact interpolation consists in associating with accurateness to each input vector its desired output value.

This type of interpolation is not valid in the case of data with noise because the interpolation function would oscillate strongly to adjust to the noise that is part of data. A certain number of modifications of the exact interpolation procedure gave birth to neuronal models based on radial-bias functions and called radial-bias networks. The goal is to get a smoothed interpolation function of which the number of basis functions is not represented anymore by the data set size but by a certain number of representatives obtained by a data quantification. If k is the number of these representatives, for an input z, the output y of such a network has the form:

$$y(z) = \sum_{c=1}^{k} l_c \, f_c(z) + l_0$$

where $\{f_c, c = 1, \ldots, k\}$ is the set of the k basis functions, and $?_0$ is a bias. In practice, the most used form of the basis functions is the function:

$$f_c(z) = exp\left[- \frac{\| z - W_c \|^2}{2(s_c)^2} \right] \qquad (8)$$

We can combine the unsupervised PRSOM algorithm and radial-bias functions to solve problems for which we possess an information on the expected result like underground cores and some expertise in our application. The radial-bias network architecture is characterized by two layers of weights whereas the PRSOM algorithm turns on a topological map and performs the first learning phase taking into account a topological structure. We have used a specific radial-bias function in the affectation process (competition phase) which we have called *Radial-bias function* (*Rbf*) in equations (2) and (3). It is the only resemblance between the two methods.

5.5.3. Simulated Annealing Algorithm

Simulated annealing is a random-search technique for combinatorial optimisation problems, such as minimizing functions of too many variables. A parameter called *temperature* permits escaping from a local minimum. To obtain solutions close to the optimum, the temperature is decreased according to an appropriate law defining in this fashion the *temperature schedule* which can have a Logarithmic or an exponential form. Many researchers have shown interest to the problem of improving the Simulated Annealing Algorithm (SAA), [7, 14]. The PRSOM algorithm presented in this paper is differently conceived and is used for a classification problem. It consists to project a large multidimensional data space into finite partitions on a topological map. The PRSOM algorithm use a neighborhood system controlled by a parameter *h* called *temperature* which is determined by equation (1) at the beginning of each cycle and remains constant during all the cycle. The *temperature h* plays an important role for keeping the same neighborhood and to disturb the least possible the map topology established by the SOM algorithm.

6. Results and Interpretation

In order to have a best idea of expected results, an expert provided us, within sight of the logfacies, with an estimate of the types of rocks awaited for part of well HM3. The expert reads the curves of the measurements to evaluate the class of corresponding rocks. This correspondence is not always very well defined. So that we have sometimes up to 4 possible cases for the estimate of only one point of well. It is thus possible for the expert to provide “precise” limits between the various benches of rock. These limits are sometimes evaluated with a margin of a few meters. These estimates having been proposed starting from the reading of the logfacies, they do not bring the same information as the underground cores. For this reason, expert,' knowledge can be erroneous since it does not make it possible to determine in a very fine way the changes of rock when the layers are very thin. However, with these estimates, it will be possible to quantify the performances of our classifier; Figure 7 shows that the PRSOM quantitative error *Qe* stabilizes between 0,22 and 0.26 from 40 cycles, Figure 8 shows that the neurons represent well the input data space, and Figure 9 shows that the lithofacies obtained is enough close to the global model of wells being in a

specific zone in the south of Algeria susceptible to contain hydrocarbons [2, 3]. Moreover, the analysis of the found facies confirms the good adaptation of the network chosen to the treated data because the data already labeled at the beginning by underground cores did not return in confusion between them (two differently labeled observations do not belong to the same class of rocks). This is not the case for some observations labeled by the expert. In fact, the difference between the expert and the classifier is that the expert generally analyses the data without taking into account the values of the observations labeled by the coring whereas the classifier makes a general classification on the data then propagates the labels starting from the data of coring.

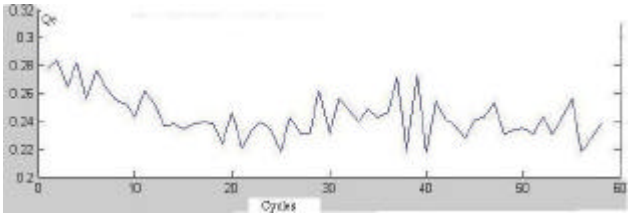

Figure 7. PRSOM quantitative error *Qe*.

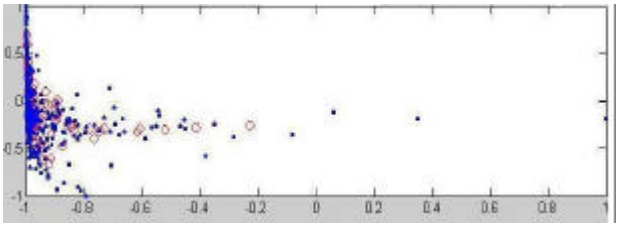

Figure 8. Neurons on the map (o) and corresponding input data vectors (*).

7. Conclusion

The lithology controls the strategies for reservoir management and it is with porosity the primary key to make a reliable reservoir model. Neural network analysis is one of the latest technologies available to the petroleum industry. The proposed method enhanced the performance of the geology expert to find quickly the lithology of a given well. Then, many studies and analysis (lateral reservoir characterization, rocks porosity, permeability, reservoir volume, hydrocarbon distribution,…) will be simplified. Using SOM and PRSOM neural network techniques, we have successfully estimated the lithology of well HM3. The PRSOM neural network using data from 2 wells (HM2 and HM3), provided a better and finest identification. Indeed, layers of rocks found in facies of Figure 9 are thinner than those found in facies of Figure 5. Some points as log's selection and expert interaction may be improved. We also need some precise information about the exact situation of the wells. The results described here open the field of neural research for log's data's study, generalization of neural networks to a drilling field and, why not, construction of an universal classification tool. We also think that if we quantitatively combine seismic and well log data with stratigraphical information [1, 4, 18, 20] we would generate reliable lithology and rock-property models. It is also interesting to consider the uncertainty [11] related to such earth science applications by using neural-fuzzy [24], fuzzy ART [13], or fuzzy-ARTMAP [26] approaches.

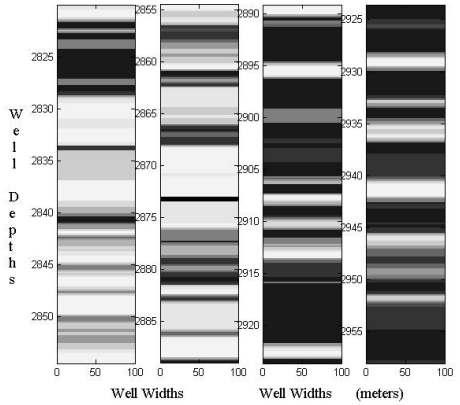

Figure 9. Lithofacies obtained for HM3 by SOM and PRSOM methods.

Acknowledgements

The Authors would like to thank Dr. Shout H., Earth Sciences Department at Mentouri University of Constantine. Data have been provided by ENAFOR. Special thanks to its technical direction.

References

[1] Adegbenga O. E., "High Resolution Sequence Stratigraphic and Reservoir Characterization Studies of D-07, D-08 and E-01 Sands, Block 2 Meren Field, Offshore Niger Delta," *in Proceedings of the American Association of Petroleum Geologists (AAPG) Conference and Exhibition*, Barcelona, Spain, September 2003.

[2] Aider H., "Contribution des Diagraphies Différées à L'évaluation de Paramètres des Réservoirs," *Second Scientific and Technical Days of the Sonatrach*, Algiers, pp 115-130, April 1996.

[3] Aliev M., Ait Laoussine N., Avrov V., Aleksine G., Barouline G., Lakov-lev B., Korj M., Kouvykine J., Makarov V., Mazanov V., Medvedev E., Mkrtchiane O., Moustafinov R., Oriev L., Oroudjeva D., Oulmi M., and Said A., "Structures Géologiques et Perspectives en Pétrole et en Gaz du Sahara Algérien," *Sonatrach Editions*, Algiers, 1971.

[4] Aminzadeh F., "Challenging Problems in the Oil Industry and Applications of AI and Soft Computing," *in Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, Mexico, August 2003.

[5] Anouar F., "Modélisation Probabiliste des Cartes Auto-Organisées: Application en Classification et en Régression," *PhD Thesis*, National Conservatory of Arts and Professions, CNAM, Paris V, 1996.

[6] Baddari K. and Nikitin A. A., "Statistical Processing of Geophysical Data," *The Second Scientific and Technical Days of the Sonatrach*, Algiers, pp 205-220, April 1996.

[7] Barbaroux J. M., Germinet F., and Tcheremchantsev S., "Generalized Fractal Dimensions: Equivalences and Basic Properties," *Journal de Mathématiques Pures et Appliquées*, September 2000.

[8] Blume M. and Esener S. C., "An Efficient Mapping of Fuzzy Art onto a Neural Architecture," *Reaserch Report*, University of California at San Diego, La Jolla, CA, 1997.

[9] Bougrain L. and Alexandre F., "Unsupervised Connectionist Algorithms for Clustering an Environmental Data Set: A Comparison," *Neurocomputing*, vol. 28, pp. 177-189, 1999.

[10] Chikhi S. and Shout H., "Using Probabilistic Neural Networks to Construct Well Facies," *WSEAS Transactions on Systems*, vol. 2, no. 4, pp. 839-843, October 2003.

[11] Claude C. S., "Practical Analysis for Reservoir Uncertainty," *in Proceedings of the AAPG Conference and Exhibition*, Barcelona, Spain, September 2003.

[12] Derguini K., "Applications Sédimentologiques des Diagraphies," *The Algerian Geology Journal*, pp.1-29, September 1995.

[13] Dreyfus G., Martinez J. M., Samuelides M., Gordon M. B., Badran F., Thiria S., and Hérault L., "Réseaux de Neurones. Méthodologie et Applications," *Eyrolles Editions*, Paris, 2002

[14] Durand M. D. and White S. R., "Trading Accuracy for Speed in Parallel Simulated Annealing with Simultaneous Moves," *Parallel Computing*, vol. 26, no. 1, pp 135-150, January 2000

[15] Frayssinet D., "Utilisation des Réseaux de Neurones en Traitement des Données de Diagraphies: Prédiction et Reconstitution de Faciès Lithologiques," *Master Thesis*, National Conservatory of Arts and Professions, Paris, 2000.

[16] Gedeon T. D., Kuo H., and Wong P. M., "Rule Extraction Using Fuzzy Clustering for a Sedimentary Rock Data Set," *International Journal of Fuzzy Systems*, vol. 4, no. 1, pp. 600-605, March 2002.

[17] Gottlib-Zeh S., Briqueu L., and Veillerette A., "Indexed Self-Organization Map: A New Calibration System for a Geological Interpretation of Logs," *in Proceedings of the IAMG'99*, VI, Lippard, Naess and Sinding-Larsen, Editons Norway, pp. 183-188, 1999.

[18] Jerry F. L., "Applying Sequence Stratigraphic Methods to Constructing Petrophysical Models," *in Proceedings of the AAPG Conference and Exhibition*, Barcelona, Spain, September 2003.

[19] Kohonen T., "Self Organization and Associative Memory," *Springer Series in Information Sciences*, Sprinter Verlag, Berlin, vol. 8, 1988.

[20] Platon E., Gary A., and Glenn J., "Pattern Matching in Facies Analysis from Well Log Data: A Hybrid Neural Network-Based Application," *in Proceedings of the AAPG Conference and Exhibition*, Barcelona, Spain, September 2003.

[21] Rabaute A., "Obtenir une Représentation en Continu de La Lithologie et de La Minéralogie: Exemples d'Applications du Traitement Statistique de Données de Diagraphie Aux Structures Sédimentaires en Régime de Convergence de Plaques," *PhD Thesis*, Montpellier II University, 1999.

[22] Tamhane D., Wong P. M., and Aminzadeh F., "Integrating Linguistic Descriptions and Digital Signals in Petroleum Reservoirs," *International Journal of Fuzzy Systems*, vol. 4, no. 1, pp 586-591, March 2002.

[23] Thiria S., Lechevallier Y., Gascuel O., and Canu S., "Statistiques et Méthodes Neuronales," *Dunod Editions*, Paris, 1997.

[24] Wong K. W., "A Neural Fuzzy Approach for Well Log and Hydrocyclone Data Interpretation," *PhD Thesis*, Curtin University of Technology, December 1999.

[25] Yang H., Chen H. C., and Fang J. H., "Determination of Carbonate Lithofacies Using Hierarchical Neural Networks," *in Proceedings of the Artificial Neural Networks in Engineering Conference (ANNIE`96)*, Missouri, USA, pp. 481-486, 1996.

[26] Yong H.T. and Marzuki K., "Comparison of Fuzzy Artmap and MLP Neural Networks for Hand Written Character Recognition," *Symposium on Artificial Intelligence in Real Time Control (AIRTC`97)*, pp. 363-371, 1997.

Salim Chikhi is a PhD student in computer science. He received his MSc degree in computer systems from the University of Constantine, Algeria, in collaboration with Glasgow University, UK. Very soon he is going to defend his PhD thesis. His research interests include neural networks, hybrid systems and soft computing application in petroleum reservoir modeling.

Mohamed Batouche received his MSc and PhD degrees in computer science from the Institut National Polytechnique de Lorraine, Nancy, France in 1989 and 1993, respectively. Currently, he is a full professor at the University of Constantine, Algeria. His research areas include artificial intelligence and pattern recognition.