A HIDDEN MARKOV MODEL APPROACH FOR LITHOLOGY IDENTIFICATION FROM LOGS

Maria Padron **, Sonia Garcia-Salicetti*, Daniel Barraez**, Bernadette Dorizzi*, Sylvie Thiria****

*Institut National des Télécommunications (INT), Evry, France; **Universidad Central de Venezuela (UCV), Caracas, Venezuela; ***LODYC, Université Pierre et Marie Curie, Paris, France

Maria.Padron@int-evry.fr, Sonia.Salicetti@int-evry.fr, dbarraez@euler.ciens.ucv.ve, Bernadette.Dorizzi@int-evry.fr, Sylvie.Thiria@lodyc.jussieu.fr

1. INTRODUCTION

We present a new statistical method of identifying lithologies relying on wireline log measurements made on two holes from the french site of Marcoule. Since several years, statistical techniques have appeared as a powerful tool to classify complex and heterogeneous reservoir lithology: Multivariate Statistics [Doveton (1994)], Discriminant Analysis [Busch (1987)] and, more recently, Neural Networks have been applied to this problem. Concerning Neural techniques, Multilayer Perceptrons (MLPs) are used to classify lithologies, either relying on well-logs directly [Samuel (1992)], or sometimes after using Kohonen Maps to determine the lithologies of the reservoir [Saggaf (2000)]. Also, Self-Organizing Maps have been used to reconstruct the lithologic facies of a drilling hole [Frayssinet (2000), Anouar (1997)].

The goal of this study is to identify lithologies from logs, relying on information about rocks porosity and permeability. To this end, we propose an original approach based on Hidden Markov Models (HMMs). Indeed, we consider a log serie of a drilling hole as a *sequence* of measures, and propose to model it in the statistical framework given by HMMs. The reason is that, in this way, we can take into account contextual dependencies between measures made at different levels of the drilling hole, while performing lithology identification. In particular, in complex reservoirs, several lithologies are mixtured, and it is extremely difficult, even for a human expert, to determine which is the lithology relying only on the log measures taken at a given level. In this framework, contextual information may be of importance to improve classification at a given level. HMMs are indeed well-known statistical models in other applicative areas (like speech recognition, on-line handwriting recognition, etc..). They appear to be a powerful tool to exploit contextual information when performing classification locally in a sequence of observations. In such applications, the signal is temporal and non stationary; the context of a single observation brings information about the evolution of the signal in time. Our purpose in the present work is to envisage this approach for sedimental series deposited during time.

This work is structured as follows: HMMs are briefly presented in Section 2, as well as their application to lithology identification. For that, we start describing the applicative context in detail. Then, the model is described in Section 3, and results are then presented and discussed in Section 4.

2. HMM FOR LITHOLOGY IDENTIFICATION

2.1 The applicative context: the Marcoule site

The Marcoule site is in the south of France, in the Gard area, near Bagnols-sur-Cèze. A hundred million years ago, this site was covered by an ocean and aside the mountains of the Massif Central. It is why the subsoil is composed of both facies of continental origin (resulting from erosion of the cristallin formations of the Massif Central) and of marine origin. The subsoil is made of clayey and sandy sedimental series, which have been deposited at Cretaceous.

Data come from two drilling holes, named MAR402 and MAR203. The profile of the Marcoule site shows a tilt of the soil between these two holes; because of this tilt, the facies encountered in MAR203 are encountered in the inferior half of the well MAR402. For that reason, MAR402 is in fact more complete than MAR203 from a geological point of view: some facies present in MAR402 are absent of MAR203. This is an important fact in our study. Also, core data from holes is only available at certain levels of the holes. For this reason, we use labels resulting from a previous research work obtained with a Kohonen map on the Marcoule site [Frayssinet (2000)]. MAR203 is drilled until 891 mts and MAR402 until 1530 mts. We use three logs that are PEF (photoelectric effect), RHOB (relative density in gr/cm3), and GR (Gamma-Ray in API numbers). In the drilling, the measures are taken every half-foot (15.24 cm). Nevertheless, the study of signals shows that their vertical resolution is rather of around 50 cms. In the drilling hole MAR203, we have 5590 measures' levels, and in MAR402, 9962 measures' levels.

According to [Frayssinet (2000)], twelve lithologies were determined in the Marcoule site: Limestones (C), Marls (M), glauconitic Sandstone (Gga), Shales (A), Other Shales (A1), Silts (S1), Other Silts (S), coarse Sandstone (Ggs), Sandstone (G), sandy Limestones (Cg), sandy Breccia (B) and Lignite (L).

2.2 Hidden Markov Models

In the last years, Hidden Markov Models have become a useful tool in non stationary signal recognition [Rabiner (1989), Rabiner (1993)]. HMMs are statistical models based on the classical Markov chains. Consider a stochastic process (q_t) which is described at time t as being in one of a set of N states, S_1, S_2,…,S_N. The process (q_t) is a Markov chain if, in order to make a prediction at time t on what is going to happen in the future, it is useless to know anything more about the whole past up to time t -1, i.e.

$$P(q_t = s_j | q_{t-1} = s_i, q_{t-2} = s_k, \ldots, q_1 = s_m) = P(q_t = s_j | q_{t-1} = s_i) \qquad (1)$$

We only consider the homogeneous Markov Chain, that is those processes in which the right-hand side of (1) (namely the transition probability from state q_{t-1} to state q_t) is independent of time. The matrix of state transitions probabilities $A=\{a_{ij}\}$ and the initial distribution π are the relevant information in order to describe the time evolution of the process:

$$a_{ij} = P(q_t = s_j | q_{t-1} = s_i) \text{ , } 1 \le i,j \le N \qquad (2)$$

$$\pi_i = P(q_1 = s_i), \qquad 1 \le i \le N \qquad (3)$$

The Markov Chain defined in this way is called an observable Markov model since the output of the process are the states. In many interesting problems in which the signal is non stationary, the states of the Markov Chain are hidden, not directly observable, and the observations are the random signals emitted by the states. A Hidden Markov Model (HMM) is therefore a double stochastic process characterized by:

- the number N of states $\{S_1, S_2, \ldots, S_N\}$ in the model;

- the state transition probability distribution;

$$a_{ij} = P(q_t = s_1 | q_{t-1} = s_i) \qquad 1 \le i,j \le N \qquad (4)$$
$$\sum_{j=1}^{N} a_{ij} = 1$$

- the initial distribution;

$$\pi_i = P(q_1 = s_i), \qquad 1 \le i \le N . \qquad (5)$$

- the set of observation signal densities, $B=\{b_j\}$, where b_j is the observation signal density when the process is in state *j*.

A HMM provides the mechanism for a random system which may be described as follows. At time t=1, the initial state $q_1 = S_i$ will be chosen at random, according to the initial distribution probability π. In this state S_i, a signal O_1 will be observed according to the observation signal density b_i. At time t=2, the process changes to another state S_j according to the transition matrix a_{ij}, and so on. Note that a complete specification of a HMM is given by the specification of probability measures A, B and π. In the following, λ=(A,B,π) denotes the complete set of parameters specifying the HMM λ. For a complete description of training procedures in a HMM, see [Rabiner (1989), Rabiner (1993)].

To identify lithologies, we first train a HMM per class (per lithology). During this step, called training, for each lithology, we optimize the models' parameters (A, B, π) that best explain a given set of observation sequences, called training database. Afterwards, in a second step, called recognition, a sequence of logs is, at the same time, segmented and recognized by the lithologies' HMMs. The Viterbi algorithm [Rabiner (1989)] gives indeed the sequence of states with highest likelihood for this observation sequence. This allows to segment the observation sequence corresponding to a log serie of a hole, in different lithologies, while such lithologies are recognized. These two steps will be detailed in section 3.

3. TRAINING

3.1 Structure of the Model

There are different types of HMMs: discrete or continuous HMMs, regarding the nature of the state emission probability laws, left-right HMMs, or parallel ones, or ergodic HMMs, regarding to the topology of the model [Rabiner (1993)], that is the transitions that are authorized between the states of the HMM.

We model each lithology by an ergodic and gaussian continous HMM. Ergodicity permits to envisage transitions from every state to any other state of the HMM (see Figure 1*)*. Also, we used a mixture of gaussian densities to approximate the distribution of the observations, represented by the logs. This means that each observation O_t (each log) at time t, a vector of dimension 3 (the PEF, RHOB and GR logs), is emitted by state j with probability:

$$b_j(O_t) = \sum_{m=1}^{M} c_{jm} \eta[O_t, \mu_{jm}, U_{jm}], \quad 1 \le j \le N \qquad (6)$$

where c_{jm} is the mixture coefficient for the *m*th mixture component in state *j* and η the gaussian density function, with mean μ_{jm} and covariance matrix U_{jm} for the *m*th mixture component in state j, that is $\eta(O_t, \mu_{jm}, U_{jm})$:

$$\eta(O_t, \mu_{jm}, U_{jm}) = (2\pi)^{-n/2} (\det(U_{jm}))^{-1/2} \exp\left[-\frac{(o_t - \mu_{jm})^t U_{jm}^{-1} (o_t - \mu_{jm})}{2} \right] \qquad (7)$$

The mixture coefficients c_{jm} satisfy the stochastic constraints:

$$\sum_{m=1}^{M} c_{jm} = 1 \qquad 1 \le i \le N, 1 \le m \le M \qquad (8)$$
$$c_{jm} \ge 0$$

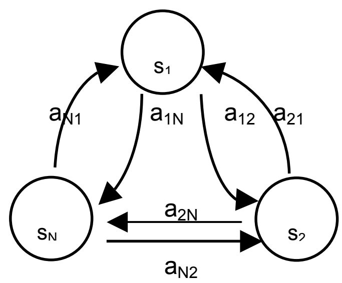

Figure 1. Ergodic HMM of a given lithology (N states)

3.2 Isolated Training

We first consider a training paradigm in which isolated log sequences of each class (lithology) are used to train the corresponding HMM. We call this particular training paradigm "Isolated Training". For this, we cut the complete sequence of logs of hole 402 (the hole used for training purposes) into segments, where each segment corresponds to a different lithology. Each resulting sequence of observations in a given lithology has a size T≤16. We used the Baum-Welch algorithm to estimate the parameters of each lithology HMM $\lambda=(A,B,\pi)$. To summarize, this algorithm maximizes iteratively $P(O \mid \lambda)$, the likelihood of the observation given the model. A local maximum is attained after a given number of iterations of the training database. This algorithm works in the following iterative form:

1.- Initialization of the model: the transition and initial probabilities are defined equiprobable. Also, the number of observations in each log sequence are distributed equitably in the states of the HMM. The same is done in each state regarding the number of gaussian densities of the state emission probability law.

2.- After each iteration of the training database, we reestimate $\hat{\lambda} = (\hat{A}, \hat{B}, \hat{\Pi})$ as follows:

- for the initial probability, the expected frequency in state s_i at time t=1 is computed as:

$$\hat{\pi}_i = \gamma_1(i) \qquad 1 \le i \le N \tag{9}$$

where $\gamma_t(i)$ is the a posteriori probability of being in state i at time t:

$$\gamma_t(i) = P(q_t = i \mid O, \lambda) \tag{10}$$

- for transition probabilities, the expected number of transitions from state s_i to state s_j, divided by the expected number of transitions from s_i is computed:

$$\hat{a}_{ij} = \frac{\sum_{t=1}^{T-1} \varepsilon_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)}, \qquad 1 \le i,j \le N \tag{11}$$

where:

$$\varepsilon_t(i,j) = P(q_t = i, q_{t+1} = j \mid O, \lambda) \tag{12}$$

that is, the a posteriori probability to be in state i at time t and in state j at time t+1;

- for the state emission probabilities, the parameters of each gaussian density function and the mixture coefficients are reestimated. The mixture coefficient reestimation for the gaussian density k in state j is the following [Rabiner (1989)]:

$$\hat{c}_{jk} = \frac{\sum_{t=1}^{T} \gamma(j,k)}{\sum_{t=1}^{T} \sum_{k=1}^{M} \gamma_t(j,k)} \tag{13}$$

where $\gamma_t(j,k)$ is the probability of being in state j at time t with the kth mixture component accounting for O_t

$$\gamma_t(j,k) = \gamma_t(j) \left[\frac{\eta[O_t, \mu_{jm}, U_{jm}]}{\sum_{m=1}^{M} c_{jm}\, \eta[O_t, \mu_{jm}, U_{jm}]} \right] \tag{14}$$

Finally, the mean and the covariance matrix of the gaussian density k in state j are reestimated as follows:

$$\hat{\mu}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k) O_t}{\sum_{t=1}^{T} \gamma_t(j,k)} \tag{15}$$

$$\hat{U}_{jm} = \frac{\sum \gamma_t(j,k)(O_t - \mu_{jk})(O_t - \mu_{jk})'}{\sum_{t=1}^{T} \gamma_t(j,k)} \tag{16}$$

3.- Training is stopped when the average log-likelihood on the training database is stabilized, that is when:

$$\left| \frac{\overline{P(O \mid \lambda^{r+1})} - \overline{P(O \mid \lambda^{r})}}{\overline{P(O \mid \lambda^{r})}} \right| < c \tag{17}$$

where c is 10^{-3} or 10^{-4} according to the class (lithology) that we consider. Baum [Baum (1970)] showed that at the *r*th iteration of this algorithm, we have:

$$P(O \mid \lambda^{r+1}) \ge P(O \mid \lambda^{r})$$

for each observation sequence, until a local maximum is reached.

3.3 Contextual Training after Isolated Training

After isolated training is performed, we consider this as an initialization for another type of training, that we call "Contextual Training".The interest of such training is to introduce in the parameter estimation process, contextual information present in the hole. Indeed, isolated training only trains each lithology model on isolated sequences of each lithology. In this new training paradigm, we consider instead longer subsequences of logs of the hole, containing *several litologies*. Then, for training purposes, we must *concatenate the HMMs* corresponding to the lithologies present in each of these subsequences. This way, parameter estimation for each of these lithology models will be influenced by the neighboring lithologies, as follows : the Viterbi algorithm [Rabiner (1989)] is used to segment the whole sequence into different lithologies, and this segmentation is exploited for training purposes, as we explain below.

The Viterbi algorithm computes the optimal path shown in Figure 2 (the "hidden" state sequence) in terms of maximal likelihood of the whole logs-subsequence (containing C, M, and L), presented to the corresponding

HMMs (those of C, M and L) (see Figure 2). The optimality criterion used on the sequence of logs is therefore global. Contextual information is introduced in the training process precisely when using the resulting optimal path to reestimate the parameters of the corresponding HMMs.

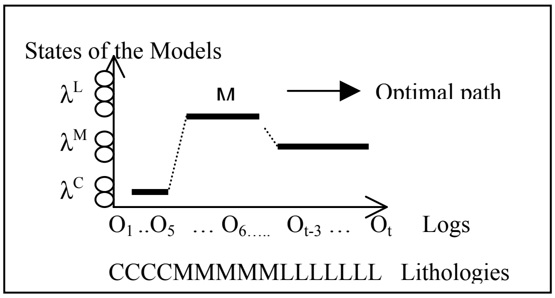

Figure 2. Viterbi path computation during Contextual Training on a logs-sequence alternating C, M and L

The Viterbi algorithm [Rabiner (1989)] is based on dynamic programming; this algorithm finds the best state sequence in the sense of maximal likelihood, for a given observation (logs) sequence. For the first t observations, the probability:

$$\delta_t(i)=\max_{q_1q_2..q_{t-1}} P\left(q_1q_2..q_t=i, O_1O_2..O_t \middle| \lambda\right) \quad (18)$$

gives the best score along the path permitting at time t to reach state s_i: A recurrence is then stated as follows:

$$\delta_{t+1}(j)=\max_i\left[\delta_t(i)a_{ij}\right]b_j(O_{t+1}) \quad (19)$$

Also, a variable $\psi_t(j)$ contains the best preceding state for state j at time t. This algorithm has three steps:

1. Initialization

$$\delta_1(i)=\pi_1 b_i(O_1) \quad (20)$$

$$\Psi_1(i)=0 \qquad 1\le i\le N$$

2. Recursion:

$$\delta_t(j)=\max\left[\delta_{t-1}(i)a_{ij}\right]b_j(o_t) \qquad 2\le t\le N \quad (21)$$

$$\psi_t(j)=\arg\max\left[\delta_{t-1}(i)a_{ij}\right] \qquad 1\le j\le N$$

3. Termination

$$P^*=\max_i\left[\delta_t(i)\right] \qquad 1\le i\le N \quad (22)$$

$$q_T^*=\arg\max_i\left[\delta_t(i)\right]$$

The *optimal state sequence* is obtained by "backtracking", as follows :

$$q_t^*=\psi_{t+1}(q_{t+1}^*) \qquad t= T-1,T-2,\ldots,1 \quad (23)$$

As mentioned before, after this segmentation step, the reestimation of the HMMs' parameters is performed. This training paradigm is well-known as the Segmental K-Means algorithm [Rabiner (1993)]. According to the segmentation given by the optimal state sequence, all the observations (logs) attributed to a given state are affected to a given gaussian density in this state. This is done by the computation of the distance of each observation to the means of all the gaussians in this state. For each observation, the gaussian realizing the minimum distance is affected to such observation. Then, parameter reestimation can be performed for the HMMs, as follows:

- for the mean of the mixture component *m*, for a given sequence of the training database,

$$\hat{\mu}_{jm}=\frac{\sum_{t=1}^{T}\delta(q_t^*-j)O_t\delta(O_t\in c_m)}{\sum_{t=1}^{T}\delta(q_t^*-j)\delta(O_t\in c_m)} \quad (24)$$

there δ is the Kronecker function, c_m denotes cluster *m* (the observations affected to the mixture component *m*), and O_t denotes the current observation at time t.

- for the covariance matrix of mixture component *m*:

$$\hat{U}_{jm}=\frac{\sum_{t=1}^{T}\delta(q_t^*-j)\delta(O_t\in c_m)(O_t-\hat{\mu}_{jm})(O_t-\hat{\mu}_{jm})^t}{\sum_{t=1}^{T}\delta(q_t^*-j)\delta(O_t\in c_m)} \quad (25)$$

The mixture coefficient is reestimated as the number of observations (logs) affected to cluster *m* of state *j* divided by the number of observations affected to state *j* :

$$\hat{c}_{jm}=\frac{\sum_{t=1}^{T}\delta(q_t^*-j)\delta(O_t\in c_m)}{\sum_{t=1}^{T}\delta(q_t^*-j)} \quad (26)$$

3.4 Contextual Training after random initialization

Another type of training was also tested: it consists of Contextual Training by the Segmental K-Means algorithm described in Section 3.3, when the HMMs are not initialized by Isolated Training (previously described in Section 3.2). In this framework, the training database (a set of sequences of length T≤25 of MAR402) is used once (one epoch) to initialize the HMMs, using the "correct path" instead of the optimal path computed by the Viterbi algorithm. The "correct path" is in fact the path that we obtain when we associate to each observation (a log) its correct label (the corresponding lithology). With this "correct segmentation" of each sequence of the training database, we obtain a first estimation of the HMMs' parameters by the Segmental K-Means algorithm.

3.5 Convergence criteria

Two convergence criteria are considered: the first one is based on the stabilization of the average log-likelihood per class; the second one is based on the stabilization of the performance of each lithology HMM (the percentage of correct classification).

4. TESTING THE SYSTEM

Classification is performed on the other hole (MAR203) using the Viterbi algorithm: the complete sequence of logs measured in MAR203 is presented to the 12 lithology HMMs at the same time, to compute the optimal state sequence in the Viterbi sense. To this end, transitions are authorized from any state of any HMM to any state of any other HMM.

4.1 Testing after Isolated Training

The following table (Table 1) shows the number of log data available per lithology in MAR402 ("Data" column in Table 1) and the number of resulting isolated sequences of each lithology ("Total" column in Table 1), after data in MAR402 is cut into segments (of maximum length 16) of each lithology. Notice that lithologies S, M and Cgs have the fewest number of sequences for training their respective HMMs. On the other hand, lithology L has the highest number of sequences for training purposes.

Class	Data	Total	Class	Data	Total
C	918	58	S	204	13
M	473	30	Cgs	408	30
Cga	857	54	G	588	37
A	772	49	Cg	1122	70
A1	1079	68	B	549	35
S1	715	45	L	2205	138

Table 1. Data description for Isolated Training

Results on MAR203 after Isolated Training in sequences of each class extracted from MAR402, are presented in Table 2: column "C" is the class (lithology) now numbered from 1 to 12 (designing respectively C, M, Gga, A, A1, S1, S, Ggs, G, Cg, B, L). Column "S" is the number of states in the HMM, column "M" is the number of gaussian densities (mixture components) per state of the HMM, column "Data" gives the number of logs per class in MAR203, column "LL" is the number of training iterations made, column "E" is the value of constant c in formula (17) to stop training by the Baum-Welch algorithm, and column "%" is the percentage of logs correctly classified in MAR203.

According to the number of sequences per class, the criterion to stop training uses a different value of constant c (10^{-3} or 10^{-4}). Different tests were made changing the number of gaussians (M), but we present only the results with M=1, as they are the best. Results are globally good, they vary from 44.44% to 96.39%.

This can be explained by the fact that there are not enough isolated sequences to train the HMMs in the configuration in which emission probabilities are mixtures of gaussian densities. Indeed, this framework implies much more parameters to estimate (several covariance matrices, several means, mixture coefficients). It is why in the following (Sections 4.2 and 4.3), all the experiments are performed in the framework of one gaussian density per state of the HMMs.

C	S	M	Data	LL	E	%
1	3	1	305	25	e-4	96.39
2	2	1	178	23	e-3	81.46
3	3	1	338	15	e-4	55.62
4	2	1	1354	18	e-4	91.06
5	2	1	797	26	e-4	64.74
6	3	1	369	28	e-4	44.44
7	2	1	349	29	e-4	65.98
8	3	1	391	24	e-4	61.68
9	3	1	549	17	e-4	63.81
10	3	1	431	24	e-4	63.81
11	3	1	130	22	e-4	66..92
12	3	1	399	12	e-4	74.69

Table 2 . Test results for M=1 after Isolated Training

4.2 Testing after Contextual Training with HMMs initialized by Isolated Training

In this framework, transition probabilities between different lithology HMMs are introduced in the computation of the optimal path by the Viterbi algorithm. These transition probabilities are *fixed during training and testing*; they are estimated on the hole MAR402 by relative frequencies. Their role is to favour some inter-model transitions, according to what is observed in MAR402.

For Contextual Training, we used sequences of length $T \leq 25$ of MAR402. When training is stopped according to the performance criterion per HMM (stabilization of the percentage of correct classification per class), convergence is reached after 39 epochs (iterations of the training database), and after 51 epochs for the criterion of likelihood stabilization.

Table 3 shows results on MAR203 for both convergence criteria: column "Q" gives in fact the values in which the performance becomes stable in the "training hole" (MAR402), and column "%Q" gives the corresponding results in the "test hole" (MAR203). Analogously, column "LL" gives the average value of the log-likelihood per class at convergence (when this value becomes stable), and column "%LL" gives the corresponding results in the "test hole" (MAR203).

We first notice that the percentage of correct classification is improved in half of the lithologies (classes 2,3,6,7,8,11) compared to the results obtained after Isolated Training.

The other lithologies (classes 1, 4, 5, 9, 10, 12) for which results are degraded, are very mixtured in the drilling holes, that is a single or very few observations (logs) of such lithologies are often found between other lithologies. For this reason, *only few sequences of such lithologies have a significant length during Contextual Training*. This is particularly visible for lithology 12 (L), for which plenty of data are available, but such data are spread in the drilling holes at most measures' levels. Indeed, this lithology, of vegetal origin, is very weak and tends to get damaged during the data acquisition process, spreading itself at most measures' levels.

In other words, Contextual Training is effective when subsequences of each lithology appearing in the context of other classes are of significant length.

C	S	M	Data	Q	%Q	LL	%LL
1	3	1	305	70.26	80.65	-6.82	80.65
2	2	1	178	74.20	91.01	-75.75	91.01
3	3	1	338	66.27	57.10	1.73	57.98
4	2	1	1354	84.45	62.62	-5.42	62.62
5	2	1	797	59.87	24.09	-29.24	24.09
6	3	1	369	64.19	55.55	-75.11	55.55
7	2	1	349	71.56	86.53	-27.35	86.53
8	3	1	391	67.08	44.75	-5.6	45.01
9	3	1	549	59.86	37.34	26.84	37.34
10	3	1	431	72.90	61.71	-18.63	59.62
11	3	1	130	91.43	93.84	-14.5	93.84
12	3	1	399	39.81	67.91	-52.73	67.91

Table 3. Test after Contextual Training when HMMs are initialized by Isolated Training

4.3 Testing after only Contextual Training

In this framework, as detailed in Section 3.4, the training database is used once (one epoch) to initialize the HMMs, using the "correct path" instead of the Viterbi path. With this "correct segmentation" of the sequences of the training database, we obtain a first estimation of the HMMs' parameters by the Segmental K-Means algorithm. Our goal is to evaluate the influence of this initialization, done in a contextual way, when followed by Contextual Training. Results are given in Table 5. 48 iterations (epochs) of the training database were necessary to stop training, for both convergence criteria (described in Section 3.5). In both cases, results are the same.

Table 4 shows that for 2/3 of the classes, the results are improved compared to those presented in Table 3. Also, the degradation of class 12 (L) is confirmed after this contextual initialization. Compared to Isolated Training, two classes are strongly improved: class 2 (M) and class 11 (B), and some classes like classes 6 and 7 (S1 and S), and class 10 (Cg) are globally unchanged. This may reveal that the latter are quite difficult to model.

C	S	M	Data	Q	%Q	LL	%LL
1	3	1	305	83.44	90.16	-0.35	90.16
2	2	1	178	73.36	93.82	-67.75	93.82
3	3	1	338	45.85	38.46	-88.54	38.46
4	2	1	1354	87.30	64.69	-13.16	64.69
5	2	1	797	57.92	39.14	-43.12	39.14
6	3	1	369	68.11	44.44	-97.74	44.44
7	2	1	349	61.76	60.74	-6.20	60.74
8	3	1	391	56.45	49.82	-3.32	49.82
9	3	1	549	32.82	38.25	-839.9	38.25
10	3	1	431	72.37	62.87	-23.26	62.87
11	3	1	130	91.07	97.69	-10.15	97.69
12	3	1	399	35.60	58.39	-64.13	58.39

Table 4. Test after only Contextual Training

5. CONCLUSIONS

We have proposed an original approach based on HMMs to identify lithologies in a drilling hole. This statistical approach considers a sequence of logs measured in a drilling hole as a time serie. This permits to introduce some contextual information present in the sequence of logs when estimating the parameters of the statistical models of each lithology. A lithology is modeled by a gaussian ergodic HMM and trained in three different ways: Isolated Training (in which data are separated per lithology for training), Contextual Training after initializing the HMMs by means of Isolated Training, and only Contextual Training in which the HMMs are even initialized in a contextual way. The last paradigm improves results for 2/3 of the classes relatively to the second one. Some classes are difficult to model in any of such paradigms: class 6 and 7 (S1 and S), and class 10 (Cg). Also, we noticed that Contextual Training is uneffective for those classes whose sequences are not of significant length when taken in the context of other lithologies. For that reason, only two classes show the real interest of Contextual Training relatively to Isolated Training: class 2 (M) and class 11 (B). On the other hand, class 12 (L) shows another limit of our approach: this class appears in the context of all the others because it is spread at all the levels of the drilling holes. Results for this class get degraded with contextual training, and when the initialization is also done contextually, they are even more degraded.

These preliminary tests show that, while in general the introduction of context improves the classification accuracy, one has to be careful with some classes with a very changing context. Our further work will explore this aspect in more details and propose some efficient strategy to cope with this phenomenon.

6. ACKNOWLEDGEMENTS

This work was partially supported by the French-Venezuelan Action ECOS-Nord N° V99M01 (N° 99000267), CDCH-UCV grant N° 03.11.4258.98 and Agenda Petroleo Accion Modelaje Estocastico Aplicado.

7. REFERENCES

Anouar F., Badran F., Thiria S., 1997: Self-Organized Map, A Probabilistic Approach, *Proceedings of the Workshop on Self-Organized Maps*, Helsinki University of Technology, Espoo, Finland, June 4-6.

Baum, L H., Petrie T. , Soules G., Weiss N., 1970: A maximization technique ocurring in the statistical analysis of probabilistic functions of Markov chains, *Annals of Mathematical Statistics*, **41**, nº 1, 164-171.

Busch, J. M., Fortney, W. G. , Berry L. N., 1987: Determination of lithology from well logs by statistical analysis, *SPE Formation Evaluation*, vol. 2, 412-418.

Davis, J. C., 1986: Statistic and data analysis in geology, *2nd ed..New York, John Wiley*, 273p.

Delfiner, P., Peyret O., Serra O., 1987: Automatic determination of lithology from well logs. *SPE Formation Evaluation*, vol. 2, 303-310.

Doveton J. H., 1994: Geologic log analysis using computer methods. *Am Assoc. Petroleum Geologists, Computer Methods in Geology,* No 2, 169p.

Frayssinet D., Thiria S., Badran F., Briqueu L., 2000: Use of Neural Networks in Log's Data Processing-Prediction and Rebuilding of lithologic facies. *Petrophysics meets Geophysics*, Paris, France, November 6-8.

Juang. B. H. , L. R. Rabiner, 1990: The Segmental K-Means algorithm for estimating parameters of Hidden Markov Models". *IEEE Transactions on Acoustics, Speech and Signal Processing*, **38**, nº 9, 1639-1641.

Rabiner L.R., 1989: A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, **77**, No 2, 257-285.

Rabiner, L.R., 1993: Fundamentals of Speech Recognition. *Prentice Hall Signal Processing Series*, Prentice Hall.

Rogers. S. J., J. H. Fang, C. L. Karra, and D. A. Stanley, 1992: Determination of Lithology from Well Logs using a Neural Network. *Am Assoc. Petroleum Geologists Bulletin*, **76**, No 5, 731-739.