Classification Using Linguistic Descriptions in a Petroleum Engineering Application

Tom D. Gedeon[1], Tao Lin[3,] Patrick M. Wong[2] and Dilip Tamhane[2]

Abstract-- There are many classification problems in petroleum reservoir characterization, an example being the recognition of lithofacies from well log data. Data classification is not an easy task when the data are not of numerical origin. This paper compares a number of approaches to classify porosity into groups (e.g. very poor, poor, fair, etc.) using petrographical characteristics, which are often described in linguistic statements in core analysis.

Index Terms-- linguistic classification, soft computing, petroleum engineering

I. Introduction

Understanding the form and spatial distribution of heterogeneities in sedimentary rock properties, such as porosity, is fundamental to the successful characterisation of petroleum reservoirs. Poor understanding of lithofacies distribution results in inaccurate definitions of reserves and improper management schemes. Mapping the continuity of major lithofacies is of great importance in reservoir characterisation studies. It is, however, impossible to start this mapping exercise until the major types of lithofacies have been recognised and identified.

Lithofacies recognition is often done in drilled wells where suitable well logs and core samples are available. Techniques, such as k-means cluster analysis [1], discriminant analysis [2], artificial neural networks [3], and fuzzy logic methods [4] are popular pattern recognition methods for classifying well log data into discrete classes. These methods, however, cannot be applied without a prior understanding of the lithological descriptions of the core samples extracted at selected well depths. Core descriptions are usually available from routine core analysis reports in exploration and appraisal wells.

The recognition of major lithofacies is not an easy task in heterogeneous reservoirs. Rock characteristics such as petrophysical, depositional (or sedimentary), and diagenetic (or textural) features are common parameters that are used to define lithofacies. However, geologists with different field experiences create different lithofacies groupings based on the same observational information. These diverse definitions occur because only a series of qualitative or linguistic statements are provided in lithological descriptions. Thus a subjective decision must be made on how many dominant lithofacies are present and what these lithofacies are.

The objective of this paper is to introduce a systematic approach for the handling of linguistic descriptions of core samples, by using a number of approaches to classify porosity into groups using petrographical characteristics. The three techniques used are an expert system approach, a supervised clustering approach, and a neural network approach. We first review the basics of lithological descriptions and describe each technique. We then demonstrate these techniques using a data set available for an oil well in a reservoir located in the North West Shelf, offshore Australia. We then apply the methods to porosity classification based on core descriptions, and validate the model using unseen cases with known porosity.

II. Lithological descriptions

Classifying geological data is a complicated process because linguistic statements dominates the results of core analysis studies. The problem is worse for lithological descriptions. Each core sample is usually described by a number of petrographic characters. These characters are described in terms of linguistic petrographic terms, such as grain size, sorting, and roundness. A typical statement for a core sample could be:

> "Sst: med dk gry f-med gr sbrndd mod srt arg Mat abd Tr Pyr Cl Lam + bioturb abd"

which means, "Sandstone: medium, dark gray, fine-medium grain, sub-rounded, moderate sorting, abundant argillaceous matrix, trace of pyrite, calcareous laminae, and abundant bioturbation".

Although these statements are subjective, they do provide important indications about the relative magnitudes of various lithohydraulic properties (e.g. porosity and permeability). It is, however, difficult to establish an objective relationship between, say, porosity levels (e.g. low, medium or high) and the petrographic characters.

III. Data

An oil well located in the North West Shelf, offshore Australia, provided a routine core analysis report for this field study. There were 226 core plug samples taken from a total of 54 metres of cores obtained from three intervals. The reservoir is composed of sandstones, mudstones, and carbonate cemented facies. The porosity and permeability

1 School of Information Technology, Murdoch University, Perth
2 School of Petroleum Engineering, University of N.S.W., Sydney
3 CMIS, CSIRO, Canberra

values ranged from 2 to 22 percent and from 0.01 millidarcy to 5.9 darcies, respectively.

The report includes porosity measurements from helium injection as well as detailed core (lithological) descriptions on each sample. The lithological descriptions were summarised into six porosity-related sets: grain size, sorting, matrix, roundness, bioturbation, and laminae. Each character was described by a number of attributes. A total of 49 attributes were used. Table 1 tabulates the character-attributes relationships used in this study.

Character (attribs)	Descriptions	
		Attributes
Grain size (12)	The general dimensions (e.g. ave. diameter or volume) of the particles in a sediment or rock, or of the grains of a particular mineral that made up a sediment or rock.	
		Very Fine, Very-Fine to Fine, Fine, Fine to Medium, Medium, Fine to Coarse, Medium to Fine, Medium to Coarse, Fine to Very Coarse, Coarse to Very Coarse, Very Fine with Coarse Quartz, Fine with Coarse Quartz.
Sorting (6)	The dynamic process by which sedimentary particles have some particular characteristic (eg. simil. of size, shape, or specific gravity).	
		Well, Moderate to Well, Moderate to Poor, Moderate, Poor to Moderate, Poor.
Matrix (14)	The smaller or finer-grained, continuous material enclosing, or filling the interstices between, the larger grains or particles of a sediment or sedimentary rock.	
		Argillaceous (Arg), Sideritic (Sid), Siliceous (Sil), Sid with Arg, Sid with Sil, Arg with Sil, Sil with Arg, Carbonaceous, Calcareous, Pyritic with Arg, etc.
Roundness (8)	The degree of abrasion of a clastic particle as shown by the sharpness of its edges and corners as the ratio of the average radius of curvature of the maximum inscribed sphere.	
		Sub-angular (subang), Angular (Ang) to Subang, Subang to Sub-rounded (subrnd), Subrnd to Ang, Subang, Subrnd, etc.
Bioturb-ation (6)	The churning and stirring of a sediment by organisms.	
		Abundant bioturbation (bioturb), Increase bioturb, Bioturb, Decrease bioturb, Minor bioturb, Trace of bioturb.
Lamina (10)	The thinnest or smallest recognisable unit layer of original deposition in a sediment or sedimentary rock	
		Irregular argular, Irregular Calcareous, Trace of Calcareous, Less Traces, Argillaceous, Calcareous, Irregular Silt, Thick, Irregular.

Table 1. Character and attributes used for porosity classif.

The objective of this study is to demonstrate how intelligent techniques can be applied in classifying linguistic descriptions of core samples into various porosity classes. We will first develop the knowledge base, implemented for the three methods as expert system, clustering diagram or neural network weights, respectively. The knowledge base is developed using a number of known porosity cases (training data). The knowledge base will then be tested using an unseen set of core descriptions (test data). The performance can be evaluated by comparing the predicted porosity classes with the actual classes using the correct-recognition rate (i.e. number of correct classifications divided by total number of samples).

In the following sections the three techniques are briefly described, followed by the results section, our conclusions, and suggestions for future work.

IV. Expert System Technique

We have used an expert system knowledge acquisition and maintenance technique, to establish new rules (acquire knowledge) and to update existing rules (maintain knowledge) when suitable observations are obtained. Knowledge is added to the system only in response to a case where there is an inadequate (i.e. none) or incorrect classification. The notion of basing classification on keystone cases has previously been used in petrography [5]. In cases of an incorrect classification, a human expert needs to provide a justification, in terms of the difference(s) associated with the case that shows the error or prompts the new rules, that explains why his/her interpretation is better than the interpretation given for such cases. Hence, the approach is able to adapt new rules or knowledge without violating previously established rules, and hence, all rules are consistent within the system. Rules are formulated in the following form:

IF [conditions] THEN [conclusion].

The basic logic is simple and interpretable. There is only one requirement to develop the rule bases: all the cases must be described with a fixed set of descriptive characters. The rules can be viewed as binary decision trees. Each node in the tree is a rule with any desired conjunctive conditions. Each rule makes a classification, the classification is passed down the tree, and the final classification is determined by the last rule that is satisfied. The technique is very simple and has no further complications beyond the description given here. Its benefits derive from its simplicity, and its applicability without the need for an expert system specialist to build the knowledge base. There are some deficiencies, which we describe in the context of our results.

V. Supervised Clustering

A supervised clustering technique was also used. Clustering techniques are generally non-supervised. The benefit of the supervised approach is that the expert can label as acceptable clusters which make suitable distinctions in the data classification. Clusters which are not suitable can be labelled for further clustering. A portion of the data

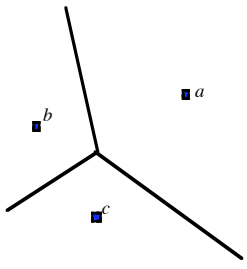
is held out (as for all the three techniques used) from the technique so that the success rate can be validated using this unseen data.

Visual Clustering Classifier (VC+) is a visual system through which users can conduct clustering operations to generate classification models. Clustering as an unsupervised learning mechanism has been widely used for clustering analysis [6]. Clustering operations divide data entities into homogenous groups or clusters according to their similarities. As a clustering algorithm, k-means algorithm measures the similarities between data entities according to the distances between them. Lin and Fu [7] applied a k-means based clustering algorithm for the classification of numerical data entities. To apply clustering algorithm to data mining applications, two important issues need to be resolved: large data set and categorical attribute. Extended from k-means algorithm, k-prototype algorithm [8] has resolved these two issues.

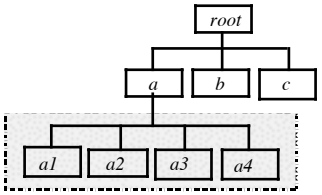
This k-prototype algorithm is based on an assumption that the similar data entities should be located closer than other data entities. Those similar data entity groups are normally called clusters. A classification divides a data set into a few groups that are normally called classes. The classes are determined either by human experts or a few data fields of the data entities, such as the application discussed in this paper. Therefore clusters and classes are not equivalent. To apply k-prototype algorithm for classification, the class distribution of the data entities in the generated clusters must be considered.

Two steps are required for the development of a classification model using VC+: cluster hierarchy construction; and classification model generation. Once the training data set has been loaded into VC+, a root cluster node for the cluster hierarchy is generated. The root contains the entire training data set. The user can apply the clustering operation on the data set to generated clusters that will be the children nodes of the root node. A leave cluster node in the cluster hierarchy will be further partitioned if the shape of distribution is not good or there is not a dominant class in the data entities in this cluster.

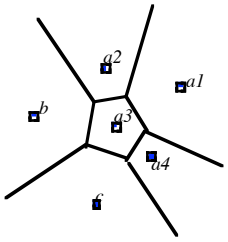
Figure 2 illustrates the procedure for generating a classification model. Firstly three clusters that have centers: a, b and c are generated by a clustering operation on root node. The cluster hierarchy will be generated. This cluster hierarchy will be expended after node a is further partitioned.



(a) Clustering result on root.



(b) Cluster hierarchy.



(c) Result of the clustering on node a.

Figure 2. Cluster hierarchy construction.

If there is a dominant class in the data entities in a leave cluster node, the center of this cluster will be marked as this class. The classification model generated by VC+ consists of all the leave nodes that have been marked. The class of the cluster in the classification model which has the shortest distance to a given data entity will determine the class of this data entity. If there is no dominant class for the data entities in a leave node and this leave node cannot be further partitioned due to the number of data entities contained, this leave node will be left unmarked and will not be included in the classification model.

To apply k-prototype clustering for classification, there are many non-deterministic criteria that direct affect the classification result, such as the number of clusters, the start cluster centers, and the chosen features. However it is out of computational power if all of the combination of these criteria were taken considered. VC+ provides various visualization tools to display data entities, statistical results and also allows users to compare the results of different clustering operations. In this fashion, users' expertise can be incorporated with the procedure for generating classification models.

VC+ adopts visualization techniques to incorporate users' expertise in the procedure for the generation of classification models. This approach increases the exploration space of the mining system. This approach has advantages in handling noise and outliners.

VI. NEURAL NETWORK

A standard 12 input x 7 hidden x 4 output neural network was used. The input data was encoded by means of a linguistic encoding technique into 12 numeric input variables.

The simplest case is for "Sorting", where the characters of Poor – Poor-moderate – Moderate-poor – Moderate – Moderate-well – Well-moderate – Well are easy to place in a sequence, and allocated values evenly distributed from 0

to 1. Neural network inputs for the standard backpropagation algorithm used in some 70% of applications worldwide, are usually normalised to this range.

For some of the fields more complicated encoding was necessary. For example, in the case of a circular linguistic term ordering, two variables are required to be able to encode the values. The values of the sine and cosine for an even distribution around a circle is required. This is illustrated for the input property Sphericity and Roundness in Figure 3.

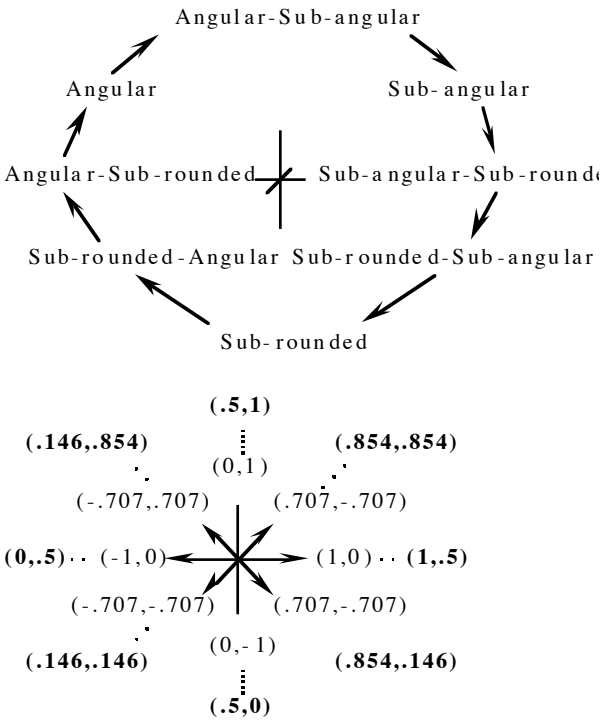


Figure 3. Circular encoding of Roundness (Sphericity) top, normalisation to inputs shown above

As there are eight values, the familiar points of 0°, 45°, 90° and so on are used. The (sin,cos) tuples are shown in Figure 3. The values are in the range from -1 to 1, which are then normalised to the range 0 to 1. The property of this circular encoding is that for all of the adjacent points the sum of the absolute values of the changes to the values is the same.

VII. Results

The experiments were run using the full date set, split 2/3 training, 1/3 testing, using all three techniques. The overall results were very similar. The supervised clustering algorithm produced 64.2% accuracy, the neural network result on the test set was 60%, and the expert system result was 59.7%.

Note that the expert system required some user effort in manual pre-processing to discover plausible rules and sequencing the data appropriately, to compensate for missing parameters. This is due to the system relying on cornerstone cases, which is prone to bias from the sequence of presentation of examples. Qualitatively, this appeared to be a greater cognitive burden than the equivalent task of encoding the inputs for the neural network, as that encoding had to be done once only and did not require perusal of the entire training set and the attempted extraction of significant patterns.

Some extra experiments were performed using the expert system technique to discover the significance of such user pre-processing.

In the first of these extra experiments, very specific rules were created for each pattern, choosing all of the available non-null characters. This produced a result of 51.6% on the test set. This indicates that the previous effort in manual pre-processing had some significant effect, and the difficulty of doing this.

The next experiment was to include the null fields for each pattern in each rule. Thus, if for a pattern no "Sorting" character was reported, the rule specified that the value for this field be "None". This produced a result of 38.7%, verifying our belief that the system was providing some generalisation, and demonstrating the importance of making sensible rules. At the same time, we discovered the minimum possible error on the test set (with this split of the data) of 15% as there are some patterns with identical characters and different category.

VIII. Conclusion

We have used three techniques for using linguistic information from core analysis reports for classification. We have found that the use of pre-processing and clustering, and fuzzy output encodings both improve the results, which are otherwise unsatisfactory from the expert system technique without a major cognitive effort on the part of the user.

To be fair, the expert system produced results using symbolic inputs essentially the same as the neural network on the numerically encoded inputs. This suggests that with the use of this encoding further improvements may be achieved. The benefit of expert system technique is that a rule trace is possible for every decision, so failures can be accounted for and successes understood by users. This tends to be an issue in the wider use of neural networks, where the "black box" nature of predictions are unacceptable, mistrusted or merely not preferred.

The next stage in our work will be to properly integrate the three techniques. Thus, a neural network will be used to learn the significant properties of the data, which can then be examined and verified by the use of the clustering technique, and the training file constructed for the expert system technique. Even further down the track, we can envisage an on-line interactive use of the three techniques. Thus, when a new rule is required in the expert system, the neural network can be run on the as yet uncategorised patterns remaining to suggest some rules, and the clusters of patterns correctly or incorrectly classified be visualised on screen.

The use of these techniques systematically will allow the incorporation of such linguistic information with numeric well logs for improved results.

IX. REFERENCES

[1] Wolff, M., and Pelissier-Combescure, J. (1982) FACIOLOG: Automatic electrofacies determination, Society of Professional Well Log Analysts, 23rd Annual Logging Symposium, Paper FF.

[2] Jian, F.X., Chork, C.Y., Taggart, I.J., McKay, D.M., and Barlett, R.M. (1994) A genetic approach to the prediction of petrophysical properties. Journal of Petroleum Geology, vol. 17, no. 1, 71-88.

[3] Gedeon, T.D., Wong, P.M., Huang, Y. and Chan, C. (1997) "Adaptive Dimensional Fuzzy-Neural Interpolation for Spatial Data," Journal of Mathematical Modelling and Scientific Computing, vol. 8: 15 pages.

[4] Wong, P.M., Gedeon, T.D. and Taggart I.J. (1997) Fuzzy ARTMAP: A new tool for lithofacies recognition. AI Applications, vol. 10, no. 3, 29-39.

[5] Griffith, C.M. (1987) Pigeonholes and Petrography. In Pattern Recognition and Image Processing, Aminzadeh, F. (ed.), Geophysical Press, 539-557.

[6] Jain A. K. and R. C. Dubes, (1988) Algorithms for Clustering Data, Prentice Hall.

[7] Lin, Y. K. and K. S. Fu, (1983) Automatic Classification of Cervical Cells Using a Binary Tree Classifier, Pattern Recognition, Vol. 16, No.1, 68-80.

[8] Huang, Z. Extension to the k-Means Algorithm for Clustering Data Sets with Categorical Values, Data Mining and Knowledge Discovery, Vol. 2, Pages 283—304, 1998.