A COMPARISION OF PCA/ICA FOR DATA PREPROCESSING IN A GEOSCIENCE APPLICATION

Patrick M. Wong[1], *Seungjin Choi* [2] *and Yanping Niu* [1]

[1] School of Petroleum Engineering, University of New South Wales, Sydney, NSW 2052, Australia
[2] Department of Computer Science and Engineering, POSTECH, Pohang, Korea

pm.wong@unsw.edu.au; seungjin@postech.ac.kr; z2246837@student.unsw.edu.au

ABSTRACT

This paper presents a performance comparison of a variety of data preprocessing algorithms in a geoscience application. The selected algorithms are principal component analysis (PCA) and three different independent component analyses (FLEXICA, JADE and SOBI). These algorithms are applied to a set of electrical and radioactive signals obtained from a drilled well in Indonesia. Standard backpropagation neural networks are used to perform pattern (flow unit) classification from raw or preprocessed data. The results show that use of the preprocessed data gives more confident results than those obtained from the raw data. Among the preprocessing algorithms, FLEXICA seems to slightly outperform the others. The study also present a technological framework for combining results from different techniques and it shows that further improvement was achieved.

1. INTRODUCTION

Classification of rock types is a complex problem in petroleum reservoir geology and engineering. This problem involves the classification of sedimentary rock quality (in terms of hydraulic properties in porous media) based on a series of electrical and radioactive measurements obtained in drilled holes [1]. In heterogeneous reservoirs, the measurements can fluctuate continuously as a function of drilled depths. The resulting curves (signals versus depths) are often known as "logs." The classification of log responses is a practical way to perform rock typing at different depths, and hence providing a "rock log" (commonly known as a "lithology log"). We may also express rock qualities as discrete flow units. For instance, flow unit 1 (FU-1) has low quality, and flow unit 2 (FU-2) has medium quality.

Classification of log responses can be performed in unsupervised or supervised mode. The practical advantage of unsupervised classification is that it can make use of all the log data, but the resulting typing may not relate to the rock quality. Supervised classification has the opposite advantage and disadvantage. It can give meaningful typing, but the number of training patterns is often small, due to the expensive costs of retrieving rock samples from downhole to the surface. There are also additional costs to perform laboratory testing on these samples in order to quantify the rock quality. In practice, rock samples are few and the lack of data poses a great challenge for data mining and experimental design.

In this paper, we will look at the use of a popular supervised tool of backpropagation neural networks for flow unit classification from well logs in an Indonesian reservoir. Since the data is complex (see later sections), it becomes necessary to preprocess the data prior to feeding the raw data to the neural networks. We will apply the existing techniques of principal component analysis (PCA) and independent component analysis (ICA) to extract hidden features from the log responses. We will compare the performance of the use of raw data and hidden features for solving the classification problem. In the next section, we will review the existing PCA and ICA methods. A case study will be presented in later sections, followed by results and conclusions.

2. DATE PREPROCESSING

2.1 Background

Blind source separation (BSS) or independent component analysis (ICA) is a statistical method which aims at finding latent variables (hidden variable, sources) that are believed to generate the data. The ICA exploits the statistical independence among latent variables, so its task is to decompose *m*-dimensional multivariate observation data $\mathbf{x}(t)$ into a linear sum of statistically independent components. BSS is closely related to the ICA, and its task is to recover unknown sources, given only observation data. BSS/ICA considers a linear data model where the data vector $\mathbf{x}(t)$ is assumed to be generated by:

$$\mathbf{x}(t) = \mathbf{A}\,\mathbf{s}(t)$$

where $\mathbf{A} \in \Re^{m\times n}$ is called the mixing matrix (each column of which corresponds to the basis vector) and $\mathbf{s}(t) \in \Re^{n}$ is the source vector (whose elements correspond to basis coefficients). In other words, it seeks for a linear transformation with basis coefficient being statistically independent. Unlike most linear transforms (e.g. Fourier transform), both basis vectors and coefficients are learned from data only. If the data $\mathbf{x}(t)$ consists of linear mixtures of sources $\{s_i(t)\}$, then the BSS can recover unknown sources $\{s_i(t)\}$, given only a finite number of observations $\{\mathbf{x}(t)\}, t = 1, \ldots, N$.

2.2 Principal component analysis (PCA)

PCA is a classical multivariate data analysis method that is useful in linear feature extraction and data compression. It is essentially equivalent to Karhunen-Loeve transformation and closely related to factor analysis. All these methods are based on 2nd-order statistics of the data.

The PCA finds a linear transformation $\mathbf{y}=\mathbf{W}\mathbf{x}$ such that the retained variance is maximized. It can be also viewed as a linear transformation that minimizes the reconstruction error [2]. Each row vector of **W** corresponds to the normalized orthogonal eigenvector of the data covariance matrix.

One simple approach to PCA is to use singular value decomposition (SVD). Let us denote the data covariance matrix by $\mathbf{R}_{\mathbf{x}}(0)=E\left\{\mathbf{x}(t)\mathbf{x}^{\mathrm{T}}(t)\right\}$. Then the SVD of $\mathbf{R}_{\mathbf{x}}(0)$ gives:

$$\begin{aligned}\mathbf{R}_{\mathbf{x}}(0) &= \mathbf{U}\mathbf{D}\mathbf{U}^{\mathrm{T}} \\ &= [\mathbf{U}_s, \mathbf{U}_n]\begin{bmatrix}\mathbf{D}_s & \\ & \mathbf{D}_n\end{bmatrix}[\mathbf{U}_s, \mathbf{U}_n]^{\mathrm{T}}\end{aligned}$$

where **U** is the eigenvector matrix (i.e. modal matrix) and **D** is the diagonal matrix whose diagonal elements correspond to the eigenvalues of $\mathbf{R}_{\mathbf{x}}(0)$ (in descending order). Then the PCA transformation from *m*-dimensional data to *n*-dimensional subspace is given by choosing the first *n* column vectors, i.e., *n* principal component vector **y** is given by:

$$\mathbf{y}=\mathbf{U}_s^{\mathrm{T}}\,\mathbf{x}$$

2.3 Flexible ICA (FLEXICA)

In general, the dimension of source vector (latent variable vector) is less than that of observation data. Thus we first perform the dimensionality reduction by data sphering. In fact the data sphering (whitening) project the data onto its subspace as well as normalizing its variance. In other words, the data sphering transformation **Q** is given by:

$$\mathbf{Q}=\mathbf{D}_s^{-\frac{1}{2}}\,\mathbf{U}_s^{\mathrm{T}}$$

The whitened vector $\mathbf{z}\in\Re^n$ is given by:

$$\mathbf{z}=\mathbf{Q}\,\mathbf{x}$$

The orthogonal factor **V** in ICA can be found by minimizing the mutual information in **z**. The natural gradient in orthogonality constraint [3] or relative gradient (EASI algorithm) [4] leads to the learning algorithm that has the form:

$$\Delta\mathbf{V}=\eta_t\left\{\mathbf{I}-\mathbf{y}\,\mathbf{y}^{\mathrm{T}}-\varphi(\mathbf{y})\,\mathbf{y}^{\mathrm{T}}+\mathbf{y}\,\varphi^{\mathrm{T}}(\mathbf{y})\right\}\mathbf{V}$$

where $\mathbf{y}=\mathbf{V}\,\mathbf{z}$ and $\varphi(\mathbf{y})$ is an elementwise non-linear function whose i^{th} element is given by:

$$\varphi_i(y_i)=-\frac{d\log p_i(y_i)}{d\,y_i}$$

where $\{p_i(\cdot)\}$ are the probability density functions of sources. Then the ICA transformation **W** is given by:

$$\mathbf{y}=\mathbf{W}\,\mathbf{x}$$

where $\mathbf{W}=\mathbf{V}\,\mathbf{Q}$.

Since we do not know the probability density functions of sources in advance, we have to rely on the hypothesized density functions. The flexible ICA [5] adopts a generalized Gaussian density which is able to approximate all kinds of uni-modal distributions. For the generalized Gaussian density model, the nonlinear function is given by $\varphi_i(y_i)=|y_i|^{\alpha_i}\,sign(y_i)$. The flexible ICA exploits the relation between the Gaussian exponent and the kurtosis in order to select a proper value of the Gaussian exponent. See [5] for details.

2.4 JADE and SOBI

The JADE [6] and SOBI [7] are popular BSS methods based on the joint approximate diagonalization. In both JADE and SOBI, the data is whitened first, then an orthogonal factor of the mixing matrix is found by a linear transformation that jointly diagonalizes a full set of 4th-order cumulant matrix (in JADE) or multiple time-delayed correlation matrices (in SOBI). Unlike the other methods, JADE exploits only 4th-order independence and SOBI utilizes 2nd-order uncorrelatedness with temporal correlations. When sources are spatially uncorrelated but temporally correlated, the SOBI is a very efficient method.

2.5 Application in neural networks

The above data preprocessing algorithms provide a transformation of raw data vectors into their independent component vectors. Use of independent inputs is theoretically better in any statistical techniques. When the component vectors are used in neural networks, it is equivalent to adding an extra layer of hidden neurons with connection weights determined by the above algorithms, rather than the learning algorithm used in neural networks. Improvements are often observed in practice as the hidden features of the raw data are extracted after the transformation.

3. CASE STUDY

3.1 Objective and data descriptions

The objective of this paper was to compare the performance of various data preprocessing systems for flow unit classification in a petroleum reservoir. The data came from a well drilled in Indonesia [8][9]. The well has 27 rock samples retrieved downhole, in which the training patterns were obtained. The inputs of the training patterns were four different well logs, namely GR (gamma ray response), RHOB (bulk density), NPHI (neutron porosity) and RT (deep resistivity). The target output was the flow unit type. In this reservoir, the expert geologist used four possible types to denote the flow units, ranging from “FU-1” (low quality) to “FU-4” (good quality). The typing was done based on the observation and hydraulic measurement obtained from the rock samples. Based on the use of this training set, this study applied supervised classification and to generate a “FU log” for the entire well, which has 301 input patterns.

Flow unit classification is a highly non-linear problem in reservoir modeling. Figure 1 shows four scatter-plots of the training data used in this study. The actual flow unit types are displayed by the corresponding symbols. For simplicity, no units are displayed for the well logs. As shown, none of the well logs alone has good discriminative power. Many previous studies have confirmed the suitability of neural networks for solving reservoir problems, because the conventional

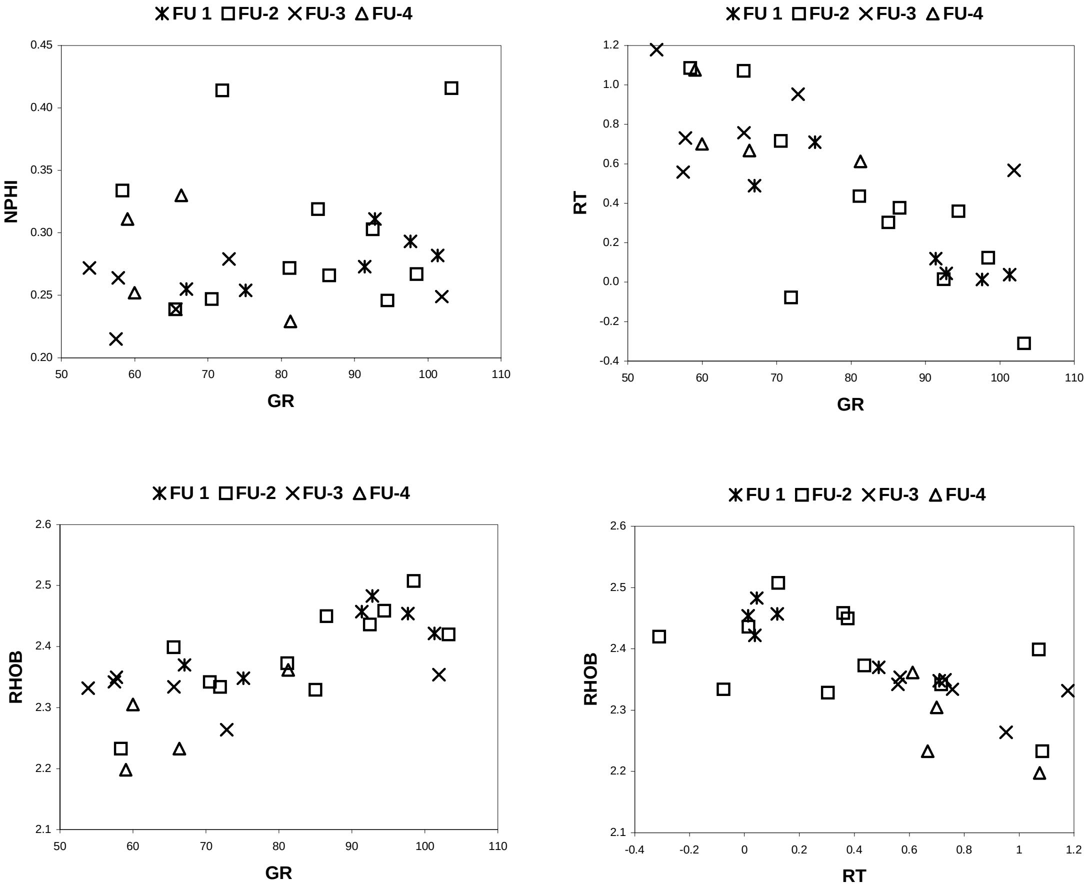

Fig. 1. Scatter-plots of input data.

classifiers are overly linear and parametric. Detailed reviews can be found in [10][11].

3.2 Neural network setup

In this study, we had 27 training patterns with 4 inputs and 4 classes (outputs). Since the training set was small, it was difficult to develop a generalized network by any means. We therefore applied two rules of thumb: 1) the number of network weights is approximately equal to the number of training patterns; and 2) the network stops learning when the classification accuracy does not improve in 30 consecutive epochs. According to our first rule, the maximum number of hidden neurons was three and this resulted in 31 weights (including biases).

3.3 Confidence measure

Once the network is trained, the network can be applied to perform classification for any input vectors. The confidence of the results is an important issue for any prediction problems. One simple way to quantity the confidence of the neural classifier is to calculate the entropy of the predictions:

$$H_j = -\sum_i p(y_{ij}) \log p(y_{ij})$$

where H_j is the entropy of the output vector *j* and y_{ij} is the corresponding activation at output neuron *i*. The larger the *H*, the smaller the confidence. Note that y_{ij} has to be normalized such that $\sum_i y_{ij} = 1$. For a four-class problem, the maximum entropy is approximately $H = -4 \times 0.25 \times \log 0.25 = 0.602$.

In this study, we will use H_j to access the prediction confidence. We will also use the average entropy $\overline{H}$ for the entire data set with *n* input vectors:

$$\overline{H} = \frac{1}{n} \sum_j^n H_j$$

Note that $\overline{H}$ can be used to compare the performance of different neural networks trained by different training sets.

3.4 Data preprocessing

As we had four input dimensions, we applied the PCA and ICA methods and ended with four PCs and four ICs. For comparison purposes, we retained all the four components as inputs to identical neural networks. All the three different ICA methods (FLEXICA, JADE and SOBI) were used. A total of five 4-3-4 neural networks were trained (including RAW and PCA). The results from the raw data were treated as the “bottom-line” (base) case.

3.5 Results

All the five networks converged to reasonable accuracy. The classification accuracy of the training set RAW was 70%, while all the preprocessed training sets gave 74%. Since no blind tests were performed, we examined the entropy values for the entire well using the 301 input vectors. The corresponding FU logs are displayed in Figure 2. Note that depths are measured in feet subsea, the solid line represents the flow unit (FU) along the well and the dotted line represents the entropy (*H*).

The average entropy values ($\overline{H}$) and performance ranking are tabulated in Table 1. From the confidence assessment, it was clear that the RAW classifications were the least confident ($\overline{H} = 0.344$), especially in the deeper part of the well (see also Figure 2). Use of JADE was particularly good for the

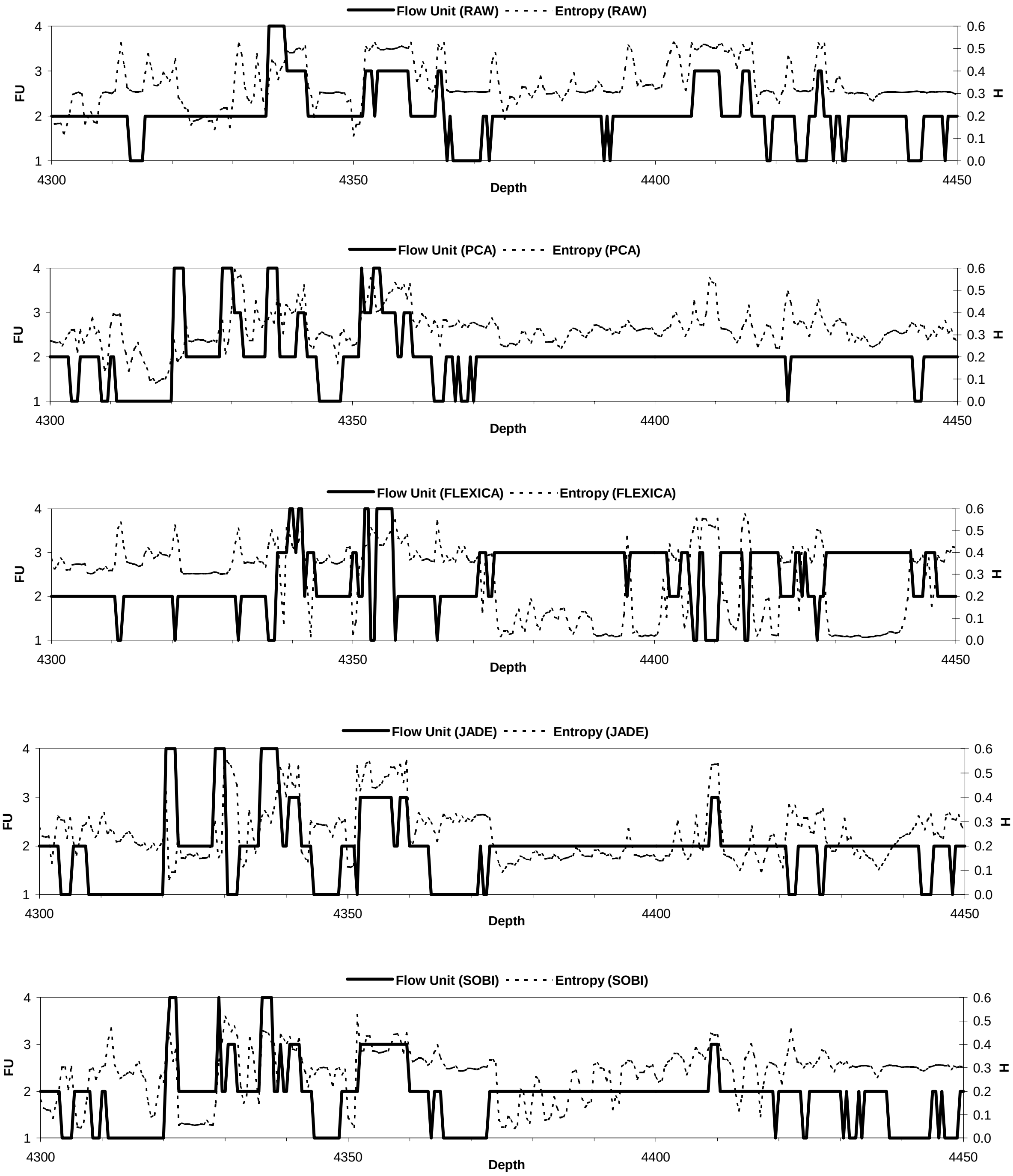

Fig. 2. FU logs from different methods.

entire well on average (see also Figure 2) with the lowest average entropy ($\overline{H} = 0.251$).

Methods	Average Entropy	Ranking
RAW	0.344	5
PCA	0.324	4
FLEXICA	0.275	2
JADE	0.251	1
SOBI	0.279	3

Table 1. Entropy analysis of the five methods.

Since all the methods use different criteria to perform classification, it is possible to combine all their individual advantages by building a hybrid model. In this study, we aggregated the results by taking the FU classification from the method with the minimum entropy. Figure 3 shows the resulting FU log, together with the location of the optimum method along the well. It is important to note that the average entropy was $\overline{H} = 0.182$, which was the lowest among all the other methods as shown in Table 1.

Table 2 shows the percentage contribution and performance ranking of each method in the hybrid model. From this simple analysis, it is clear that FLEXICA contributed the most classifications to the hybrid model, followed by SOBI and JADE. However, FLEXICA did not work well in the shallower part of the well (4300–4370 ft). On average, FLEXICA seemed to slightly outperform the other two ICA methods.

Methods	Contribution	Ranking
RAW	6.6%	5
PCA	7.3%	4
FLEXICA	33.6%	1
JADE	24.9%	3
SOBI	27.6%	2

Table 2. A contribution analysis of the hybrid model.

This paper relied heavily on the use of entropy to measure the confidence of the predictions using the neural network output activations. Like all other confidence indicators, entropy cannot measure the “absolute” confidence of the predictions. It measures only if the trained neural network is confident in the predictions. We therefore should be aware of the limitations and check if the results conform to the expert knowledge.

In summary, PCA gave better results than the use of raw data, and all the ICA methods (FLEXICA, JADE and SOBI) performed even better in this study. The hybrid FU log was geologically realistic, especially in the middle-to-bottom section of the reservoir. The rapid changes of rock types represent the inherent heterogeneity in the reservoir, which cannot be obtained from any single method alone. Moreover, since the hybrid FU log had the lowest average entropy, it was taken as the final results for further reservoir calculations. Finally, it is important to emphasize that although the technological framework is valid, the present conclusions are only valid for the data set we applied. More vigorous studies on larger data sets are required to truly compare the performance of different preprocessing algorithms.

4. CONCLUSIONS

This paper presents the use of standard backpropagation neural networks to perform classification of multidimensional signals obtained from a drilled well in Indonesia. The signals are classified as discrete flow units, which relate to the fluid flow potential of sedimentary rocks. A number of different data preprocessing algorithms commonly used in blind source separation are compared. The results show that the neural networks trained by the independent components perform better than those trained by the principal components and the raw data. More confident classifications are derived from aggregating all the results obtained from different methods.

5. REFERENCES

[1] B.P. Moss, “The partitioning of petrophysical data: a review,” In: Developments in Petrophysics, Lovell, M.A. and Harvey, P.K. (eds.), Geological Society Special Publication No. 122, 181-252, 1997.

[2] K.I. Diamantaras and S.Y. Kung, Principal Component Neural Networks: Theory and Applications. John Wiley & Sons, INC, 1996.

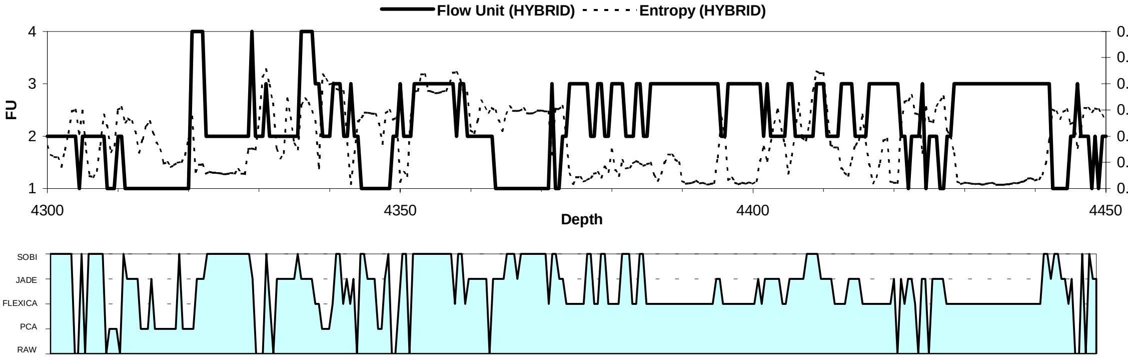

Fig. 3. FU logs from the hybrid method.

[3] S. Amari, “Natural gradient for over- and under-complete bases in ICA,” Neural Computation, 11(8), 1875-1883, 1999.

[4] J.F. Cardoso and B.H. Laheld, “Equivariant adaptive source separation,” IEEE Trans. Signal Processing, 44(12), 3017-3030, Dec. 1996.

[5] S. Choi, A. Cichocki and S. Amari, “Flexible independent component analysis,” Journal of VLSI Signal Processing, 26, 25-38, Aug. 2000.

[6] J.F. Cardoso and A. Souloumiac, “Blind beamforming for non Gaussian signals,” IEE Proceedings-F, 140(6), 362-370, 1993.

[7] A. Belouchrani, K. Abed-Merain, J.F. Cardoso and E. Moulines, “A blind source separation technique using second order statistics,” IEEE Trans. Signal Processing, 45, 434-444, Feb. 1997.

[8] A.G. Bruce, P.M. Wong, Tunggal, B. Widarsono and E. Soedarmo, “Use of artificial neural networks to estimate well permeability profiles in Sumatera, Indonesia,” The 27th Annual Conference of the Indonesian Petroleum Association, Jakarta, Feb. 1-3, 10 pp., 2000.

[9] P.M. Wong, D. Tamhane and F. Aminzadeh, “A soft computing approach to integrate well logs and geological clusters for petrophysical prediction,” Third Conference and Exposition on Petroleum Geophysics, New Delhi, Feb. 23-25, 4 pp., 2000.

[10] A.G. Bruce, P.M. Wong, Y. Zhang, H.A. Salisch, C.C. Fung and T.D. Gedeon, “A state-of-the-art review of neural networks for permeability prediction,” APPEA Journal, 40(1), 343-354, 2000.

[11] D. Tamhane, P.M. Wong, F. Aminzadeh and M. Nikravesh, “Soft computing for intelligent reservoir characterization,” SPE Asia Pacific Conference on Integrated Modelling for Asset Management, Yokohama, Apr. 25-26, 11 pp., 2000.