Hybrid Functional Networks for Oil Reservoir PVT Characterisation

1Munirudeen A. Oloso, 2Mohamed G. Hassan, 3Mohamed B. Bader-El-Den, 4James M. Buick

aSchool of Engineering, University of Portsmouth, Portsmouth, United Kingdom
bSchool of Computing, University of Portsmouth, Portsmouth, United Kingdom
E-Mails: 1munirudeen.olo@port.ac.uk; 2mohamed.hassan@port.ac.uk; 3mohamed.bader@port.ac.uk; 4james.buick@port.ac.uk

Abstract— Predicting pressure-volume-temperature (PVT) properties of black oil is one of the key processes required in a successful oil exploration. As crude oils from different regions have different properties, some researchers have used API gravity, which is used to classify crude oils, to develop different empirical correlations for different classes of black oils. However, this manual grouping may not necessarily result in correlations that appropriately capture the uncertainties in the black oils. This paper proposes intelligent clustering to group black oils before passing the clusters as inputs to the functional networks for prediction. This hybrid process gives better performance than the empirical correlations, standalone functional networks and neural network predictions.

Keywords—PVT; API gravity; clustering; functional networks; neural network; black oil

1. Introduction

The API gravity ($\gamma_{API}$) is one of the important pressure-volume-temperature (PVT) properties of crude oils. Crude oil is classified based on this property to determine its heaviness which consequently determines its marketability. Table 1 shows a typical oil classification based on $\gamma_{API}$ (Dandekar, 2013; De Ghetto et al., 1995). Knowledge of the $\gamma_{API}$ and other PVT properties such as bubblepoint pressure ($P_b$), oil formation volume factor ($B_o$) and oil viscosity are important for determining future production or oil reserves from petroleum wells.

<table>
<thead>
<tr>
<th>Classification</th>
<th>API Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Light</td>
<td>$API &gt; 31.1$</td>
</tr>
<tr>
<td>Medium</td>
<td>$22.3 \leq API \leq 31.1$</td>
</tr>
<tr>
<td>Heavy</td>
<td>$API &lt; 22.3$</td>
</tr>
<tr>
<td>Extra Heavy</td>
<td>$API &lt; 10.0$</td>
</tr>
</tbody>
</table>

Some of the PVT properties, e.g. $\gamma_{API}$, can easily be measured or determined onsite while others, e.g. $P_b$ and $B_o$, are ideally determined through laboratory experimentation. However, this laboratory analysis, which requires special expertise, is expensive and time consuming. Hence, there is a need for a less rigorous, cheaper and quicker solution.

For a long period of time, the petroleum industry has used equations of states (EOS) for determining these PVT properties. However, the EOS are considered computationally complex and require extensive detailed compositions of reservoir fluids. Consequently, many empirical correlations have been developed to meet the industrial demands for less complex, quicker, cheaper and acceptable solutions.

Empirical correlations for predicting PVT properties are generally developed by performing linear or non-linear regression analysis using easily acquired crude oil properties as the inputs. In order to improve the performance of the correlations, some researchers have used $\gamma_{API}$ to group crude oils into two or more groups and determine correlations for each group (Kartoatmodjo & Schmidt, 1994).

For more reliable and improved prediction performance of these PVT properties, other researchers have implemented machine learning [ML] algorithms to predict different PVT properties (Osman & Al-Marhoun, 2005; Khoukhi et al., 2011; Gharbi et al., 1999; El-Sebakhy et al., 2007). Also, to address the problem in modelling ANN, a recursive least square algorithm has been used for learning feedforward ANN to model crude oil blending process (de Jesús Rubio, 2016). However, none of the previous works found in the literature which have applied ML techniques in this field, have taken diversity of API or other input properties into consideration.

For instance, if the data that are used to train an ML algorithm have more light crude oils than heavy oils, then the model is confined to the constraints of the light oils. This type of data is called “imbalanced data set” which is well known in the classification problem (Ramyachitra & Manikandan, 2014).

This paper proposes a hybrid solution of K-Means clustering and functional networks (FN) for predicting crude oil PVT properties. K-means clustering is used to generate clusters of the input dataset before using functional networks to perform the prediction of the actual target variables, $P_b$ and the oil formation volume factor at bubblepoint pressure ($B_o$). The performances of the hybrid solution (K-Means+FN) is compared with the standalone FN, artificial neural network (ANN) and selected empirical correlations which are either commonly used in the petroleum industry or recently developed.

The rest of the paper is organised as follows. Section 2 discusses the empirical correlations and ML methods that have
been used to predict different PVT properties. A brief explanation of K-Means clustering and functional networks in comparison with neural networks is given in section 3. This is followed by the proposed hybrid solution in section 4. Results and discussion of the performances of the hybrid model and other compared models are done in section 5 with a brief analysis of sensitivity of FN learning parameters. The conclusion is given in section 6.

1.1. Research Contribution

In this paper, a hybrid of K-Means cluster and FN has been developed to estimate oil PVT properties. This paves way for intelligent grouping or clustering of crude oils. Also unlike the existing practice where only the $\gamma_{API}$ of crude oils is used for grouping them, all the independent variables are used for grouping the crude oils.

2. Methods for Predicting Oil Reservoir PVT Properties

Aside the EOS, there are two main categories of methods for predicting oil reservoir PVT properties in the literature: empirical correlations and ML techniques.

2.1. Empirical Correlations for Predicting Oil PVT Properties

Many correlations have been developed for the estimation of $P_b$ and $B_{ob}$. Usually, the procedure involves linear and/or non-linear regression analyses on the available datasets.

Standing developed graphical correlations for both $P_b$ and $B_{ob}$ based on 105 sets of experimental data (Standing, 1947). These graphical correlations were later expressed by equations (Standing, 1977). The input variables that Standing used for the $P_b$ correlation are: gas solubility ($R_s$), gas specific gravity ($\gamma_g$), $\gamma_{API}$ and reservoir temperature ($T$). For the $B_{ob}$ correlation, the correlating parameters were: $R_s$, $\gamma_g$, oil specific gravity ($\gamma_o$) and $T$. On this basis, many other correlations have been developed for both $P_b$ and $B_{ob}$ (De Ghetto et al., 1995; Al-Shammasi, 2001; Al-Marhoun, 1988; Vazquez & Beggs, 1980; Almehaideb, 1997; Petrosky Jr & Farshad, 1998; Jarrahiyan et al., 2015). It is noted that some of these correlations have replaced $\gamma_{API}$ with $\gamma_o$ and vice-versa. These two variables are related by equation (1).

$$\gamma_{API} = \frac{141.5}{\gamma_o} - 131.5 \quad (1)$$

2.2. Machine Learning Techniques for Predicting Oil PVT Properties

The quest for better PVT prediction models to improve the current performances of the empirical correlations has driven applications of different ML techniques in PVT characterisation. The most commonly used ML technique for PVT prediction is artificial neural network (ANN) and its variants (Talebi et al., 2014). While many of the authors have reported good performance from the developed ANN models, others have pointed out that ANN may not necessarily perform better than the empirical correlations (Al-Shammasi, 2001). It should be noted that the performance of any ML model depends on careful selection of its learning parameters.

The “black box” representation of ANN has made it unattractive for adoption in industrial PVT application. Sequel to this, some representative ML and evolutionary techniques have been used, such as support vector machine (SVM), genetic algorithm (GA), adaptive neuro fuzzy system (ANFIS), functional networks (FN) and so on (Khoukhi et al., 2011; El-Sebakhy et al., 2007; Hajizadeh, 2007).

Generally, an ML solution usually involves minimization of the error in the learning algorithm. It is notable that many of the machine learning algorithms work by performing a local search that may become stuck in a local minima which causes the model to perform poorly when presented with new data (Dietterich, 2000). An ML model could be stuck in local minima as a result of inefficient learning parameters or imbalanced datasets. However, this problem has not been adequately addressed in prediction of oil PVT properties with ML techniques, though some efforts have been made in empirical correlation development with some possible accuracy improvement (Kartoatmodjo & Schmidt, 1994). However, this involves manual grouping of crude oils and generating different correlations for the groups. This paper proposes intelligent clustering of data before applying an ML technique to the generated clusters.

3. Hybrid Machine Learning System

Hybrid ML system is normally developed so that the consisting sub-systems complement each other. The aim of a hybrid system is to fill the gap that a single method cannot necessarily fill. This paper implements a hybrid of K-means clustering and FN.

3.1. K-means Clustering

K-means is a widely used data mining technique. It puts a number of input observations into a number of clusters which must have been defined a priori. There is no defined rule for choosing an optimal number of clusters for a given dataset. The goal of clustering is to allow natural grouping of data (Jain, 2010). The implementation, pros and cons of the K-Means algorithm have been discussed in the literature (Jain, 2010).

3.2. Implementation of Functional Networks

FN were introduced as a powerful alternative to neural networks (Castillo, 1998; Castillo et al., 2000). Unlike ANN, FN have the advantage that they use domain knowledge in addition to data knowledge. The network initial topology can be derived based on the modelling of the properties of the real world. Once this topology is available, functional equations allow one to obtain a much simpler equivalent topology.

Several other improvements on the ANN algorithm for different applications can be found in the literature (Chairez, 2016; de Jesús Rubio et al., 2017; Aljarah et al., 2016; Liu et al., 2016).

Simplified general topologies for ANN and FN are shown in Fig. 1 and 2. In these figures, $X_1$, $X_2$ and $X_3$ are the inputs into the network. $X_4$ and $X_5$ are the outputs of the hidden layer. $W_{mn}$ ($m=4,5; n=1,2,3,4,5$) are the weights while $Y$ is the output in both cases.
There are some quite significant differences between ANN and FN. Notably, the functions in FN are truly learned during the structural learning unlike the ANN where neuron functions are assumed to be fixed and known, and only the weights are learned. The implemented FN is shortly described below.

Given a data set \( \{ x_{ij}, y_i; i = 1,2, \ldots n & j = 1,2,3,4 \} \) where \( x_{ij} \) are the predictors and \( y_i \) is the output. Mathematically, the relationship can be given by

\[
Y = f(x_{1}, x_{2}, x_{3}, x_{4}) \tag{2}
\]

Note that \( j \) refers to the number of inputs which is 4 here. The general form of a functional network that learns from the data can be given as follows (Castillo et al., 2000).

\[
y_i = \sum_{r=1}^{n} \sum_{r=1}^{m} c_r \varphi_r(x_{ij}), \quad i = 1,2, \ldots, n \tag{3}
\]

where \( \varphi_r \) are the linear independent functions which are used to learn the coefficients \( c_r \). Some possible functions for \( \varphi_r \) are:

1. Polynomial function:
   \[
   \varphi = \{1, x, x^2, \ldots, x^m\} \tag{4}
   \]
2. Exponential Function:
   \[
   \varphi = \{e^x, e^{-x}, \ldots, e^{mx}, e^{-mx}\} \tag{5}
   \]
3. Fourier Function:
   \[
   \varphi = \{1, \sin(x), \cos(x), \sin(mx), \cos(mx)\} \tag{6}
   \]
4. Logarithm Function:
   \[
   \varphi = \{1, \log(x+2), \log(x+3), \ldots, \log(x+m)\} \tag{7}
   \]

The aim is to get \( \hat{Y} \) which is an estimate of \( Y \) such that the square of the error is minimised. That is:

\[
\min_{\hat{Y}} \frac{1}{2} \sum_{i=1}^{n} (y_i - \hat{Y}_i)^2 \tag{8}
\]

Hence, the aim is to produce an estimate \( \hat{Y} \) that gives minimal error \( \epsilon \) which can be represented as:

\[
\epsilon = \min(Y - \hat{Y}) \tag{9}
\]

This final equation can be solved using least square optimization. Several useful analyses and applications of functional networks are found in the literature (de Jesus Rubio et al., 2012; Elsebakhi et al., 2015; Asafa et al., 2015).

### 3.3. Proposed Hybrid K-means and Functional Networks

The proposed hybrid implementation is shown in Fig. 3. The clusters serve as inputs to the FN. FN has been specifically chosen for the hybrid modelling as it has been shown to perform very well on both small and large data sets (Castillo et al., 2000). It is important to note that the clustering takes all the predictors into consideration to generate the clusters unlike the manual grouping based on only \( Y_{API} \) grouping, which is done for some empirical correlations.

1. Input: Get the data \( X \) for clustering
2. Data clustering with K-Means to get clusters, \( D_c \), \( c=1, \ldots, k \) where \( k \) is total number of clusters
3. BEGIN FN Network
4. FOR \( m=3 \) to 10
5. Initialise the functional model e.g. \( \varphi = \{1, x, x^2, \ldots, x^m\} \)
6. FOR each cluster \( D_c \) compute the functional components \( C \)
7. END FOR \( m \) loop
8. END each cluster loop
9. Combine and evaluate the prediction from the clusters.

### 3.4. Experimental Work

A total of approximately 1400 data points were available for the simulation. 327 data points were collected from different published papers (Al-Marhoun, 1988; Omar & Todd, 1993; Dokla & Osman, 1992; Bello et al., 2008). The remaining data are unpublished and they are from different sources such as GeoMark Research and Shell Company. The data comes from diverse crude oils across the globe.

In the hybrid systems, four input clusters have been generated which are passed to the FN. All the stated learning functions were tested and the polynomial function of degrees three to five gave the best results for both the hybrid K-Means+FN and the standalone FN. For the ANN model, different activation functions, number of hidden layers and neurons were explored. The best ANN model has been achieved with sigmoid activation, one hidden layer and ten neurons. The input variables for both \( P_b \) and \( B_{ob} \) are presented and compared with the standalone FN, ANN, popular and recently proposed empirical correlations. Mainly, two statistical error measures, \( RMSE \) and \( E_a \) are used to compare the prediction capability of the models. Whenever there
is a tie between these two, then $E_{\text{max}}$ and $\text{CC}$ are used respectively for comparison. The lower the error parameters: $\text{RMSE}$, $E_a$ and $E_{\text{max}}$, the better is the model. On the other hand, the higher the $\text{CC}$, the better is the performance of the model.

4.1. Comparison of Hybrid K-Means+FN with other Models

For $P_b$ prediction, the proposed hybrid K-Means+FN model gives both minimum $\text{RMSE}$ (344.8264) and $E_a$ (11.0829) as shown in Table 2. K-Means+FN hybrid model also has the least $E_{\text{max}}$ (81.8879) which implies that it has the least tendency of over-fitting. Likewise, K-Means+FN has the highest $\text{CC}$ (0.9652) which is an indication that its predicted output are more correlated with the target than others.

Among all these compared models for $P_b$ prediction, the least performance is given by the correlation of (Jarrahian et al., 2015). From the results, both ANN and standalone FN also perform better than all the listed empirical correlations. The results for the $B_{ob}$ prediction are shown in Table 3. Clearly, the hybrid K-Means+FN gives the best performance with the least $\text{RMSE}$ (0.0489) and $E_a$ (1.3856). Also the K-Means+FN hybrid has the minimum $E_{\text{max}}$ (31.8356) and maximum $\text{CC}$ (0.9807). A very significant improvement in performance can be seen in the hybrid system compared to both ANN and standalone FN. However, both ANN and standalone FN trail one of the correlations (Petrosky Jr & Farshad, 1998) in performance with respect to both $\text{RMSE}$ and $E_a$.

### Table 2

<table>
<thead>
<tr>
<th>Prediction Method</th>
<th>CC</th>
<th>$\text{RMSE}$</th>
<th>$E_r$</th>
<th>$E_a$</th>
<th>$E_{\text{min}}$</th>
<th>$E_{\text{max}}$</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standing (1947)</td>
<td>0.9057</td>
<td>616.2760</td>
<td>-3.4038</td>
<td>17.5340</td>
<td>0.0190</td>
<td>548.9583</td>
<td>0.3414</td>
</tr>
<tr>
<td>Al-Marhoum (1988)</td>
<td>0.9187</td>
<td>812.2788</td>
<td>-12.9768</td>
<td>20.0810</td>
<td>0.0084</td>
<td>388.6760</td>
<td>0.2222</td>
</tr>
<tr>
<td>Vazquez &amp; Beggs (1980)</td>
<td>0.9091</td>
<td>765.8636</td>
<td>-13.0168</td>
<td>20.6291</td>
<td>0.0212</td>
<td>603.2238</td>
<td>0.4345</td>
</tr>
<tr>
<td>Kartoatmodjo &amp; Schmidt (1991)</td>
<td>0.8911</td>
<td>882.2240</td>
<td>-11.1193</td>
<td>21.9222</td>
<td>0.0095</td>
<td>602.7154</td>
<td>0.4991</td>
</tr>
<tr>
<td>Dokla &amp; Osman (1992)</td>
<td>0.8596</td>
<td>678.0152</td>
<td>-5.3107</td>
<td>21.8189</td>
<td>0.0093</td>
<td>511.0192</td>
<td>0.7033</td>
</tr>
<tr>
<td>Petrosky Jr &amp; Farshad (1998)</td>
<td>0.9347</td>
<td>793.0799</td>
<td>10.7982</td>
<td>39.7123</td>
<td>0.0022</td>
<td>1357.4589</td>
<td>0.0706</td>
</tr>
<tr>
<td>Velarde et al. (1999)</td>
<td>0.9328</td>
<td>529.5030</td>
<td>-5.2997</td>
<td>14.9442</td>
<td>0.0385</td>
<td>443.4241</td>
<td>0.1801</td>
</tr>
<tr>
<td>Al-Shammasi (2001)</td>
<td>0.8962</td>
<td>579.3566</td>
<td>-2.2827</td>
<td>18.9615</td>
<td>0.0334</td>
<td>576.1059</td>
<td>0.3534</td>
</tr>
<tr>
<td>Dindoruk &amp; Christman (2004)</td>
<td>0.9247</td>
<td>540.7703</td>
<td>-6.8919</td>
<td>18.2234</td>
<td>0.0300</td>
<td>467.5842</td>
<td>0.4716</td>
</tr>
<tr>
<td>Khamehchi et al. (2009)</td>
<td>0.8947</td>
<td>1036.8969</td>
<td>-24.4490</td>
<td>29.5174</td>
<td>0.0142</td>
<td>654.9375</td>
<td>0.8214</td>
</tr>
<tr>
<td>Arabloo et al. (2014)</td>
<td>0.9104</td>
<td>551.5070</td>
<td>3.4806</td>
<td>16.5968</td>
<td>0.0001</td>
<td>461.4342</td>
<td>0.4964</td>
</tr>
<tr>
<td>Jarrahian et al. (2015)</td>
<td>0.8597</td>
<td>2004.4451</td>
<td>73.7602</td>
<td>73.8764</td>
<td>25.8628</td>
<td>96.8158</td>
<td>0.1077</td>
</tr>
<tr>
<td>ANN</td>
<td>0.9457</td>
<td>410.1738</td>
<td>-1.2703</td>
<td>13.0577</td>
<td>0.0268</td>
<td>117.0041</td>
<td>0.6855</td>
</tr>
<tr>
<td>Standalone FN</td>
<td>0.9522</td>
<td>390.5335</td>
<td>-5.6662</td>
<td>14.5618</td>
<td>0.0057</td>
<td>154.4435</td>
<td>0.2733</td>
</tr>
<tr>
<td>Cluster + FN (Present work)</td>
<td>0.9652</td>
<td>344.8264</td>
<td>-2.9911</td>
<td>11.0829</td>
<td>0.0056</td>
<td>81.8879</td>
<td>0.1188</td>
</tr>
</tbody>
</table>

### Table 3

<table>
<thead>
<tr>
<th>Prediction Method</th>
<th>CC</th>
<th>$\text{RMSE}$</th>
<th>$E_r$</th>
<th>$E_a$</th>
<th>$E_{\text{min}}$</th>
<th>$E_{\text{max}}$</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standing (1947)</td>
<td>0.9574</td>
<td>0.0744</td>
<td>-0.4097</td>
<td>2.2852</td>
<td>0.0007</td>
<td>68.1578</td>
<td>0.1468</td>
</tr>
<tr>
<td>Vazquez &amp; Beggs (1980)</td>
<td>0.9374</td>
<td>0.0866</td>
<td>0.4498</td>
<td>3.5164</td>
<td>0.0047</td>
<td>63.0057</td>
<td>0.2384</td>
</tr>
<tr>
<td>Al-Marhoum (1988)</td>
<td>0.9611</td>
<td>0.0716</td>
<td>0.5751</td>
<td>2.2557</td>
<td>0.0007</td>
<td>53.0750</td>
<td>0.0499</td>
</tr>
<tr>
<td>Kartoatmodjo &amp; Schmidt (1991)</td>
<td>0.9624</td>
<td>0.0679</td>
<td>0.3174</td>
<td>2.1148</td>
<td>0.0025</td>
<td>56.5184</td>
<td>0.0770</td>
</tr>
<tr>
<td>Dokla &amp; Osman (1992)</td>
<td>0.9594</td>
<td>0.0759</td>
<td>0.5801</td>
<td>3.1316</td>
<td>0.0012</td>
<td>65.1839</td>
<td>0.0101</td>
</tr>
<tr>
<td>Al-Marhoum (1992)</td>
<td>0.9646</td>
<td>0.0655</td>
<td>0.2331</td>
<td>1.9317</td>
<td>0.0038</td>
<td>57.4667</td>
<td>0.0962</td>
</tr>
<tr>
<td>Omar &amp; Todd (1993)</td>
<td>0.9434</td>
<td>0.0896</td>
<td>-0.3410</td>
<td>3.1626</td>
<td>0.0119</td>
<td>77.3916</td>
<td>0.2419</td>
</tr>
<tr>
<td>Almehaideb (1997)</td>
<td>0.9472</td>
<td>0.0834</td>
<td>0.2608</td>
<td>3.8067</td>
<td>0.0055</td>
<td>54.3641</td>
<td>0.0846</td>
</tr>
<tr>
<td>Petrosky Jr &amp; Farshad (1998)</td>
<td>0.9642</td>
<td>0.0654</td>
<td>-0.0645</td>
<td>1.4851</td>
<td>0.0001</td>
<td>53.0785</td>
<td>0.0902</td>
</tr>
<tr>
<td>Al-Shammasi (2001)</td>
<td>0.9490</td>
<td>0.0851</td>
<td>-2.6715</td>
<td>3.7926</td>
<td>0.0206</td>
<td>67.3687</td>
<td>0.1473</td>
</tr>
<tr>
<td>Dindoruk &amp; Christman (2004)</td>
<td>0.9086</td>
<td>0.1128</td>
<td>0.4307</td>
<td>3.3049</td>
<td>0.0012</td>
<td>95.3749</td>
<td>0.0869</td>
</tr>
<tr>
<td>Ikieniskimama &amp; Ajienka (2012)</td>
<td>0.9629</td>
<td>0.0670</td>
<td>-0.1210</td>
<td>2.1352</td>
<td>0.0013</td>
<td>55.7279</td>
<td>0.0258</td>
</tr>
</tbody>
</table>
4.2. Sensitivity of Functional Network Parameters

In this section, the sensitivity of the FN parameters in the overall performance of the hybrid K-Means+FN models is examined. In the previous section, the best achievable model for the hybrid has been used. Boxplots for the four most important statistical parameters are presented in Figs. 4-7.

It is noted that the performance of the hybrid models, largely influenced by the linear independent function and its degree for the FN, can vary over a wide range. For both \( P_b \) and \( B_{ob} \), polynomial function of degrees three to five gave the best performances.

From the box plots in Figs. 4-7, it can be seen that performances of the hybrid K-Means+FN varied significantly for \( CC \), \( RMSE \), \( E_a \) and \( E_{max} \). For \( P_b \), \( CC \) ranges from 0.9079 to 0.9652, \( RMSE \) from 344.8264 to 552.5147, \( E_a \) from 11.0829% to 31.2093% and \( E_{max} \) from 81.8879% to 2911.8493%. For the \( B_{ob} \), \( CC \) ranges from 0.9614 to 0.9807, \( RMSE \) from 0.0489 to 0.0689, \( E_a \) from 1.3856% to 3.2082% and \( E_{max} \) from 31.8356% to 61.7369%.

![Fig. 4. \( P_b \) Boxplot for K-Means+FN and \( CC/RMSE \)](image)

![Fig. 5. \( P_b \) Boxplot for K-Means+FN and \( E_a/E_{max} \)](image)

![Fig. 6. \( B_{ob} \) Boxplot for K-Means+FN and \( RMSE/E_a \)](image)

![Fig. 7. \( B_{ob} \) Boxplot for K-Means+FN and \( E_a/E_{max} \)](image)
5. Conclusion

A hybrid system using K-Means clustering and functional networks has been developed to predict crude oil PVT based on worldwide data from a wide range of different crude oils with diverse thermodynamic properties. The clustering part of the hybrid is inspired by the common API grouping of crude oils.

Functional networks which resemble neural networks in architecture have been used for the actual prediction. The neurons in a functional network are functions defined by the modeller and it does not suffer from the ‘black box’ indictment, though it is more computationally demanding.

For the two PVT properties, $P_b$ and $B_{ob}$, that have been modelled in this work, the proposed hybrid system outperforms all the compared empirical correlations, feed forward neural network and standalone functional networks. The results show that the clustering of the data before prediction by the functional networks has significantly improve the results compared to the standalone functional networks. The attempt to solve the impact of imbalanced data set by pre-processing the data into clusters to prevent domination by most predominant crude oil with similar properties has shown its significance.

It is also clear that the learning parameters of the functional networks must be carefully selected to get a good high performance. It should be noted that different data sets in different problem will probably be modelled by different linear independent functions as expected. Hence, different ones must be explored in each problem to attain the model that appropriately captures the uncertainties in the input dataset.

Other PVT properties such as viscosity and gas-oil ratio can also benefit from this approach of clustering before prediction. Also, the K-Means clustering can be used with other ML techniques such as regression tree, SVM and extreme learning machine. Experimentation with other clustering techniques can also be considered in future research.

Lastly, the statistical descriptions of the data used in each cluster are given in the appendix along with the corresponding coefficients of the functional networks. New clusters and functional networks can be generated for significant data variability for new datasets.

Appendix

Statistical Measures for the performance Analysis

A.1. Average percent relative error

$$E_r = \frac{1}{n} \sum_{i=1}^{n} E_i$$

Where,

$$E_i = \left( \frac{x_{exp} - x_{pred}}{x_{exp}} \right) \times 100$$

A.2. Average absolute percent relative error

$$E_a = \frac{1}{n} \sum_{i=1}^{n} |E_i|$$

A.3. Maximum absolute percent relative error

$$E_{max} = \max_{i} |E_i|$$

A.4. Standard Deviation

$$SD = \sqrt{\frac{1}{(n-1)} \sum_{i=1}^{n} (E_r - E)^2}$$

Where,

$$E = \frac{1}{n} \sum_{i=1}^{n} E_i$$

A.5. Root mean squared error

$$RMSE = \left( \frac{1}{n} \sum_{i=1}^{n} E_i \right)^{0.5}$$

Table B.1

<table>
<thead>
<tr>
<th>Variable</th>
<th>Mean</th>
<th>Min</th>
<th>Max</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_s$</td>
<td>1389.232</td>
<td>628.000</td>
<td>2637.000</td>
<td>452.535</td>
</tr>
<tr>
<td>$\gamma_g$</td>
<td>0.707</td>
<td>0.592</td>
<td>0.899</td>
<td>0.085</td>
</tr>
<tr>
<td>$Y_{API}$</td>
<td>37.926</td>
<td>22.300</td>
<td>48.200</td>
<td>4.283</td>
</tr>
<tr>
<td>T</td>
<td>208.838</td>
<td>114.000</td>
<td>300.000</td>
<td>32.599</td>
</tr>
<tr>
<td>$P_b$</td>
<td>4649.996</td>
<td>3796.000</td>
<td>7142.700</td>
<td>669.333</td>
</tr>
<tr>
<td>$B_{ob}$</td>
<td>1.737</td>
<td>1.269</td>
<td>2.588</td>
<td>0.296</td>
</tr>
</tbody>
</table>

Table B.2

<table>
<thead>
<tr>
<th>Variable</th>
<th>Mean</th>
<th>Min</th>
<th>Max</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_s$</td>
<td>256.273</td>
<td>10.000</td>
<td>840.000</td>
<td>150.634</td>
</tr>
<tr>
<td>$\gamma_g$</td>
<td>0.951</td>
<td>0.556</td>
<td>1.367</td>
<td>0.179</td>
</tr>
<tr>
<td>$Y_{API}$</td>
<td>32.341</td>
<td>11.600</td>
<td>55.000</td>
<td>6.926</td>
</tr>
<tr>
<td>T</td>
<td>163.000</td>
<td>74.000</td>
<td>275.000</td>
<td>48.248</td>
</tr>
<tr>
<td>$P_b$</td>
<td>940.504</td>
<td>70.000</td>
<td>1593.000</td>
<td>430.858</td>
</tr>
<tr>
<td>$B_{ob}$</td>
<td>1.183</td>
<td>1.030</td>
<td>1.593</td>
<td>0.102</td>
</tr>
</tbody>
</table>

Table B.3

<table>
<thead>
<tr>
<th>Variable</th>
<th>Mean</th>
<th>Min</th>
<th>Max</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_s$</td>
<td>872.562</td>
<td>60.0</td>
<td>1870.000</td>
<td>334.794</td>
</tr>
<tr>
<td>$\gamma_g$</td>
<td>0.744</td>
<td>0.570</td>
<td>1.101</td>
<td>0.104</td>
</tr>
<tr>
<td>$Y_{API}$</td>
<td>36.457</td>
<td>17.40</td>
<td>50.400</td>
<td>6.894</td>
</tr>
<tr>
<td>T</td>
<td>189.219</td>
<td>100.0</td>
<td>327.000</td>
<td>43.938</td>
</tr>
<tr>
<td>$P_b$</td>
<td>3240.123</td>
<td>2559.0</td>
<td>3000.000</td>
<td>345.396</td>
</tr>
<tr>
<td>$B_{ob}$</td>
<td>1.487</td>
<td>1.075</td>
<td>2.088</td>
<td>0.222</td>
</tr>
</tbody>
</table>

Table B.4

<table>
<thead>
<tr>
<th>Variable</th>
<th>Mean</th>
<th>Min</th>
<th>Max</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_s$</td>
<td>627.180</td>
<td>50.000</td>
<td>1602.000</td>
<td>237.920</td>
</tr>
<tr>
<td>$\gamma_g$</td>
<td>0.821</td>
<td>0.578</td>
<td>1.298</td>
<td>0.127</td>
</tr>
<tr>
<td>$Y_{API}$</td>
<td>37.825</td>
<td>15.900</td>
<td>53.400</td>
<td>7.014</td>
</tr>
<tr>
<td>-----------</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>T</td>
<td>173.831</td>
<td>80.000</td>
<td>302.000</td>
<td>48.703</td>
</tr>
<tr>
<td>$P_b$</td>
<td>2158.456</td>
<td>1441.0</td>
<td>2800.00</td>
<td>336.62</td>
</tr>
<tr>
<td>$B_{ob}$</td>
<td>1.378</td>
<td>1.083</td>
<td>1.884</td>
<td>0.152</td>
</tr>
</tbody>
</table>

C. FN Coefficients for $P_b$ prediction

The coefficients for the four clusters are given by C1-C4 respectively.

$C_1 = \begin{bmatrix} 9.274865088 & -0.3347652512 & 2.11452 & -0.73 \ 3.822939 & -0.3 & -8.33934 & -0.73 & 1.77758 & -0.73 & 5.63703 & -0.3 & 8.900214 & -0.3 & 1.51982 & -0.3 & 1.99607 & -11 \ \end{bmatrix}$

$C_2 = \begin{bmatrix} 19.2923 & 0.0765 & 0.0166 & -0.0000 & 0.0026 & -0.3319 & -18.5494 & -4.2865 & 6.3032 & 9.1714 & -0.0076 \ \end{bmatrix}$

$C_3 = \begin{bmatrix} 17.8722 & 0.0055 & -0.0001 & 0.0153 & 0.0229 & 0.0011 & -17.6863 & 6.3229 & 0.0086 & -0.1998 & 0.0018 & -0.0000 & -0.0016 \ \end{bmatrix}$

$C_4 = \begin{bmatrix} [-54.48650495 & 19.364302 & 0.016413227 & 2.163160324 & -17.44356375 & -0.386908924 & 0.014021511 & 32.94166286 & -2.682646562 & -0.012230197 & 0.003024811 & 3.864771903 & 5.6607 & -0.05278976519 & -2.95876 & -0.05 \ \end{bmatrix}$

D. FN Coefficients for $B_{ob}$ prediction

The coefficients for the four clusters are given by D1-D4 respectively.

$D_1 = \begin{bmatrix} 0.886611058 & 0.002550228 & 0.06634445 & -7.20681 & 0.05 & -0.001789681 & 0.000946617 & 6.39284 & -0.0510944 & -10 & -0.000434836 & -4.85867 & -0.09 & 4.03943 & -10 & 0.01224 & -0.06 & 0.029586 & -0.08 \ \end{bmatrix}$

$D_2 = \begin{bmatrix} 0.614771607 & 0.000511528 & 0.566965523 & 0.000561991 & -0.243882029 & -1.41053 & -0.07 & 4.93548 & -0.09 & 7.30253 & -11 \ \end{bmatrix}$

$D_3 = \begin{bmatrix} 2.747385085 & 2.53824 & -0.05 & 0.069158418 & 2.4943055 & -0.001145203 & 0.870892037 & 0.098897059 & 9.91554 & -0.07 & 1.11654 & -0.04 & -0.035432841 & 4.741975 & -0.06 & 1.37956 & -0.163507 & -0.07 & 6.59292 & -0.04 & 5.70017 & -0.05 & -0.628488 & -0.07 & 1.24723 & -0.08 \ \end{bmatrix}$

$D_4 = \begin{bmatrix} 1.790041513 & 4.76633 & -0.04 & 0.044052727 & -0.868929191 & 5.58737 & -0.04 & 0.061333617 & -0.7291 & -0.07 & 7.55122 & 0.05 & 0.119961335 & 0.020082851 & 1.19762 & -0.10 & 4.82085 & 0.07 & 4.47238 & -0.04 & 4.10756 & -0.05 & 4.10455 & -0.07 & 4.17728 & -0.08 \ \end{bmatrix}$

Acknowledgment

Oloso Munirudeen thanks the Petroleum Technology Development Fund, Nigeria, for sponsoring his PhD research at the University of Portsmouth. Special thanks also go to the Shell Petroleum Development Company, Nigeria, GeoMark Research and Professor T.A Blasingame of Texas A& M University for supplying part of the data used in this work.

References


Kartoatmodjo, T., & Schmidt, Z. (1994). Large data bank improves crude physical property correlations. *Oil and Gas Journal; (United States), 92*(27).


