MODELING OF CRUDE OIL VISCOSITY: CLASSICAL AND NEURAL REGRESSION TECHNIQUES

Adel M. Elsharakwy* **and Ridha B. Gharbi**
Department of Petroleum Engineering
College of Engineering and Petroleum
P.O. Box 5969, Safat 13060, Kuwait

ABSTRACT

Numerous research efforts have been directed towards the development of viscosity models that are capable of accurately predicting crude oil viscosity as a function of production data, and/or composition of well stream fluids. Most of these efforts were focused on developing viscosity correlation using classical regression techniques. Viscosity values of dead and live oils are required for the evaluation of hydrocarbon reserves, studying fluid flow through porous media, and the design of production equipment and pipeline. They are also involved in several dimensionless parameters to calculate flow regimes, friction factors and pressure gradients in multiphase flow problems. Generally, oil viscosity is measured isothermally at reservoir temperature and different pressures. However, viscosity data are needed at temperatures other than reservoir temperature for the design of production equipment and pipelines, and for planning thermal methods of enhanced oil recovery.

Several viscosity models were developed, in this study, using both classical regression techniques (CRT) and neural regression techniques (NRT). This paper compares for the first time models developed using these techniques. These models are developed from viscosity data collected from different oil fields. The models have also been tested using another collection of viscosity data that was not used before in the development phase. The proposed model uses stock-tank oil API gravity, gas gravity, pressure(s), and temperature(s) to predict dead, saturated, and undersaturated oil viscosity. Comparison of the methods showed that, under any conditions, NRT models predict the oil viscosity more accurately than CRT models.

INTRODUCTION

Crude oil viscosity is required for many petroleum engineering calculations such as evaluating hydrocarbon reserves and planning methods of oil recovery, calculating rate of fluid flow through reservoir rock, calculating pressure losses in production system for design of production equipment, and designing pipe lines. Crude oil viscosity is usually measured isothermally at reservoir temperature on bottom hole samples or surface recombined samples at various reservoir pressures. However, viscosity data at temperatures other than reservoir temperature and in cases where laboratory data becomes unavailable, are estimated from empirical correlations.

Several research have been published concerning the development of empirical correlations capable of predicting crude oil viscosity as a function of reservoir temperature, pressure or solution gas oil ratio, and or composition.[1-13] Most of these correlations, empirical models, were developed for a given area or a region using limited viscosity data. These correlations have limited accuracy in estimating crude oil viscosity when applied to new area or regions.[13,14]

All the published empirical correlations were developed using multiple linear or non-linear regression techniques using statistical regression packages such as SAS, SPSS, and Minitab. Recently, neural network models have been used in the petroleum industry to model reservoir fluid properties.[15-19] Gharbi and Elsharkawy[15,16] presented a neural network model using back propagation to predict bubble point pressure and oil formation volume factor for Middle East crude oil systems, and crude oil systems collected from all over the world. Lately, Gharbi[17] used a back propagation neural network model to estimate the isothermal compressibility coefficient of undersaturated Middle East reservoirs, and Elsharkawy[18] used radial basis function network to model solution gas-oil ratio, oil formation volume factor, oil density, and compressibility. General regression neural network (GRNN) was also used to model the constant volume depletion behavior (CVD) of gas condensate reservoirs.[19] Comparison

*Corresponding author
Fax (+965) 4849558, e-mail: asharkawy@kuc01.kuniv.edu.kw

between simulated behavior of the gas condensate reservoir by general regression neural network (GRNN) and equation of state showed the advantages of the GRNN.

The objective of this study is to compare artificial neural networks (ANN) with classical regression techniques (CRT) in modeling reservoir fluid properties. Crude oil viscosity is used in this study to show the performance of different schemes of neural networks and classical regression techniques. The study starts with discussion about development of empirical models using classical regression techniques (Minitab) followed by different neural network models.

CLASSICAL REGRESSION TECHNIQUES

A literature survey has indicated that empirical models developed by classical regression techniques (CRT) to predict crude oil viscosity fall in three different types. First, dead oil viscosity model is used to estimate crude oil viscosity at atmospheric condition (gas-free crude or stock tank) as a function of stock tank API gravity and reservoir temperature. Second, the saturated oil viscosity model uses solution gas oil ratio and dead oil viscosity to estimate viscosity of crude oil at pressures below the bubblepoint pressure. Third, undersaturated oil viscosity model uses saturated crude oil viscosity and pressure above the bubble point to estimate viscosity of undersaturated crudes. One of the limitations of these models which are developed by CRT is that their accuracy in estimating the saturated and undersaturated viscosity is highly dependent on estimated dead and saturated oil viscosity, respectively. In addition, in cases where PVT data becomes unavailable, solution gas oil ratio has to be estimated from other correlation that may reduce the accuracy of saturated and undersaturated oil viscosity models.

In this study, the above mentioned limitation has been avoided by using pressure which can easily be measured in the filed instead of solution gas oil ratio in addition to reservoir temperature, stock tank oil API gravity and produced gas gravity as input variable to empirical models.

Dead Oil Viscosity

PVT analysis of 59 crude oil samples from different Kuwaiti oil fields is used in this study. Viscosity data of 50 samples are used to develop the proposed classical and neural network models. The other 9 samples are used to test the validity, accuracy and performance of the proposed models. Dead crude oil viscosity, μ_{od} , is usually expressed as a function of crude oil API gravity and reservoir temperature (T) as:

$$\mu_{od} = f(API, T) \quad (1)$$

In this study, several function forms of equation (1) have been tested to model dead oil viscosity using classical regression technique. The following equation was found to best fit the data from the 50 samples:

$$\text{Log}_{10}\, \mu_{od} = 10.7580 - 3.9145 \log_{10} API - 1.9364 \log_{10} T_f \quad (2)$$

Saturated Viscosity

The saturated oil viscosity (μ_{ob}) is predicted, in this study, as a function of dead oil viscosity (μ_{od}) and reservoir pressure (p) as follows:

$$\mu_{ob} = 10^{0.82604}\, p^{-0.38678}\, \mu_{od}^{0.79903} \quad (3)$$

Undersaturated Viscosity

The undersaturated oil viscosity (μ_{oa}) is modeled using saturated oil viscosity (μ_{ob}) and the pressure difference ($P - P_b$) as:

$$\mu_{oa} = \mu_{ob} + a\,(P - P_b), \quad (4)$$

where $a = -5612 \times 10^{-8} + 9481 \times 10^{-8} \mu_{od} - 1459 \times 10^{-8} \mu_{od}^2 + 81 \times 10^{-8} \mu_{od}^{-3}$ (5)

NEURAL NETWORK TECHNIQUES

Neural networks are computing tools that consist of large number of simple, highly interconnected processors called neurons. A neuron processes an input vector by applying a transfer function to give an output, which can serve as input to other neurons. These neurons may be arranged in multiple layers as shown in Figure 1 (i.e. standard connection) or in Figure 2 (Ward nets connection). The network, shown in Figure 1, has an input layer, output layer and one hidden layer. The standard connection of back-propagation network is one in which every layer is connected or linked to the immediately previous layer (Figure 1). In Figure 2, the hidden layer network is divided in three different slabs each with different activation function. In this case, the output layer will get different views of the data.

Training a network is an important factor for the success of the neural network.[20] Several training algorithms are available in the literature. The most common of which is the back propagation training algorithm. For comparison purposes, several training methods were investigated. These methods include:
1) Back propagation, BP, using standard connection
2) Back propagation, Ward, using connection of Figure 2
3) Levenberg-Marquardt back propagation (LM)
4) General regression neural network (GRNN)

Back Propagation Neural Networks

As stated before, the back propagation algorithm is the most common training algorithm used for system learning. Back-propagation is used so that input connection weights becomes modified based on error signal arising from the output layer. The steps involved in back-propagation is illustrated below:

The first step in performing back-Propagation is to calculate the output of the last layer of the network or output layer, *n*. In general, the output from neuron *j* in layer *k* is calculated by the following:

$$a_{jk} = F_k\left(\sum_{i=1}^{N_{k-1}} w_{ijk}\, a_{i(k-1)} + b_{jk}\right) \qquad j = 1, 2, \ldots, N_k \text{ and } k = 1, 2, \ldots, n \qquad (6)$$

where for k = 0 ; $N_o = m$ and $a_{io} = I_i$ (7)

The output of the output layer can be obtained by setting k = n in eq. 7. The sum-squared error, sse, of the network is given by

$$sse = \sum_{p=1}^{P}\sum_{j=1}^{N_n} (d_j - a_{jn})_p^2 \qquad (8)$$

where d_j and a_{jn} are the desired and actual outputs from neuron j in the output layer n, respectively. The subscript p represents a specific input pattern. Several methods existed to update the weights and biases (e.g. gradient descent momentum and Levenberg-Marquardt optimization). To update weights and biases according to gradient descent momentum and an adaptive learning rate, then the next step is to calculate the gradient descent of the output layer. The gradient descent of layer k is given by

$$(d_{n-k})_p = (\Delta F_{n-k})_p (W_{n-k}^T)_p (d_{n-k})_p \qquad k = 1, 2, \ldots, n-1 \qquad (9)$$

The superscript T represents the transpose of the matrix. The neuron gradient matrix for an input pattern p is given by

$$(\Delta F_k)_p = diag\left(\frac{\partial F_k}{\partial a_{1k}}, \frac{\partial F_k}{\partial a_{2k}}, \ldots, \frac{\partial F_k}{\partial a_{N_k k}}\right) \qquad (10)$$

and the corresponding weight matrix is given by:

$$(W_k)_p = \begin{bmatrix} w_{11} & w_{12} & \Lambda & w_{1N_k} \\ w_{21} & w_{22} & \Lambda & w_{2N_k} \\ \mathrm{M} & \mathrm{M} & \Lambda & \mathrm{M} \\ w_{N_k 1} & w_{N_k 2} & \Lambda & w_{N_k N_k} \end{bmatrix}_p \quad (11)$$

Note that the third subscript of all the connection weights, *k* in eq. 12, is removed to avoid cumbersome subscripts. The values of the gradient descent for the output layer is given by

$$d_n^p = (\Delta F_n)_p (E_n)_p \quad (12)$$

where

$$(E_n)_p = [(d_1 - a_{1n}), (d_2 - a_{2n}), ..., (d_{N_n} - a_{N_n n})]_p^T \quad (13)$$

The gradient descents of the hidden layers are calculated one by one going backwards until the input layer is reached. When all the gradient descents for all the layers are calculated, the adjustments for the weights are then calculated:

$$(\Delta w_{ijk})_p = h(d_{jk} a_{ik})_p + g(\Delta w_{ijk})_{p-1} \quad (14)$$

where η is the learning rate and γ is the momentum constant. The weights can be updated as follows:

$$(w_{ijk})_p = (w_{ijk})_{p-1} + (\Delta w_{ijk})_p \quad (15)$$

The above procedure is repeated until the sum squared error is within the desired limits. In order to improve the search process, an adaptive learning procedure is used in the following fashion: If the new sse is greater than the previous one by more than ε, then the new weights are rejected and recalculated using a new learning rate. The new learning rate is given by

$$\eta_{new} = \eta_o \eta_{old} \quad (16)$$

ε and η_o are two values that can be fixed at the beginning of the computations. Typically, ε is chosen to be 5% and η_o is $1/\sqrt{2}$.

General Regression Neural Network

General regression neural networks are universal single pass neural implementation of Parzen's window based estimators (Patzen[21]) with one parameter, sometimes two, to adjust which makes training relatively, fast. The theory of such networks can be developed as follows. Let *x* be a vector of random variables and y be a scalar random variable. Let *X* be a particular measured value of *x*. In other words, *X* is the finite vector of noisy measurements of *x* and *y* represents the associated scalar value. The conditional mean of *y* given *X* is given by

$$E[y|X] = \int_{-\infty}^{\infty} y f(X, y)\, dy \Big/ \int_{-\infty}^{\infty} f(X, y)\, dy \quad (17)$$

where *f(X,y)* denotes the joint continuous probability density function (pdf) of a vector random variable *X* and a scalar random variable *y*. If *f(X,y)* is not known, an estimate $\hat{f}(X, y)$ must be used. Parzen[21] proposed a class of consistent estimators for *f(X,y).* Using window estimation, $\hat{f}(X, y)$ can be written as

$$\hat{f}(X, y) = \frac{1}{n(2p)^{p+1} s^{p+1}} \sum_{i=1}^{n} \exp(-(X - X^i)^T (X - X^i)/2s^2) \cdot \exp(-(y - y^i)^2/2\sigma^2) \quad (18)$$

where $x \in \mathbb{R}^p$ and n is the number of sample observations. Using (19), Eq. (18) becomes

$$\hat{y} = \frac{\sum_{i=1}^{n} e^{-(X-X^i)^T(X-X^i)/2\sigma^2} \int_{-\infty}^{\infty} y\, e^{-(y-y^i)^2/2\sigma^2} dy}{\sum_{i=1}^{n} e^{-(X-X^i)^T(X-X^i)/2\sigma^2} \int_{-\infty}^{\infty} e^{-(y-y^i)^2/2\sigma^2} dy} \tag{19}$$

Let $D_i^2 = (X - X^i)^T (X - X^i)$, then the estimate of the expected value $\hat{y}$ becomes

$$\hat{y}(x) = \sum_{i=1}^{n} y^i e^{-D_i^2/2\sigma^2} \Big/ \sum_{i=1}^{n} e^{-D_i^2/2\sigma^2} \tag{20}$$

These density estimators are consistent, i.e. they asymptotically converge to the underlying pdf *f(x,y)* at all points *(x,y)* provided that σ = σ(*n*) is chosen such that $\lim_{n\to\infty} \sigma(n) = 0$ and $\lim_{n\to\infty} n\sigma^p(n) = \infty$.

Therefore, σ can be visualized as a smoothing parameter. That is, if σ is made large, the estimated density is forced to be smooth and the limit becomes a multivariable Gaussian with covariance $\sigma^2 I$. On the other hand, a smaller value of σ allows the estimated density to assume non-Gaussian shapes. In such case, wild points may have too great an effect on the estimate. Thus, given the joint probability density function (pdf) of *X* and *y*, the conditional pdf can be empirically determined and, hence, the expected value can be computed. Cacoullos[22] extended Parzen's method to the multivariable case. The resulting regression procedure is implemented via four layers of parallel neural network architecture, Specht[23-24], where it called General Regression Neural Network. The GRNN is a three layer neural network with one hidden neuron for each training pattern. The number of neurons in the input layer (first slab) is the number of problem inputs and the number of neurons in the output layer (third slab) matches the number of outputs. The number of neurons in the hidden layer is usually the number of training patterns. The choice of higher number of hidden neurons may be advantageous in some problems. The GRNN training is achieved by comparing the input pattern in the *n* dimensional space of all training patterns to determine how far it is from these patterns. The predicted output is proportional to the distance of the given pattern from all training patterns. The distance metric is the familiar Euclidean distance. Also, the L_1 - norms or the city block distance may be used. Eventhough both norms are equivalent, it was found that city block distance may speed training.

In conclusion, GRNN is a memory-bound three layer neural network which provides estimates of its variables and converges to the underlying linear or nonlinear regression surface. GRNN is more advantageous with sparse and noisy data than backpropagation(Specht and Shapiro[25] & Marquez[26]) and is much faster to train. Using substantial simulations, Marquez[55] has shown that the GRNN sees through noise and distortion better than the backpropagation neural network. Compared with back propagates neural networks (BPNN), the BPNN performance can be significantly hindered by the presence of local minima and longer training.

Neural Network Models

The number of layers and neurons in each of the neural network methods were systematically varied in order to obtain the most accurate model for viscosity. In each of the methods, the input layer uses pressure, temperature, stock tank oil API gravity, and produced gas gravity as input data to the model and the output layer contains the predicted crude oil viscosity. The same 50 crude oil samples that were used in developing the classical regression model are used to train the neural network models. The other 9 samples are also used to test these models.

For the BP (standard connection) network, four layers are used: an input layer, two hidden layers, and one output layer. The two hidden layers contain 40 neurons each and the logistic activation function is used. Momentum method is used for training with learning rate (momentum rate) equal to 0.1 and initial weight of 0.3. The Ward net BP model uses 15 neurons in each of the three slabs (Figure 2). The network design used a Gaussian function on the middle slab to detect features in the mid-range of the data and use Gaussian complement in the other hidden slabs to detect features from the upper and lower extremes of the data. For the Levenberg-Marquardt back propagation network (LM), the model uses four layers (one input layer, two output layers, and one output layer). The first hidden layer contains 10 neurons and the second one contains 8 neurons. The proposed GRNN model has three slabs. The first slab contains 4 input neurons: pressure, temperature, crude oil API gravity and produced gas gravity. The middle slab has 1250 neurons and the output slab has one neuron, the predicted crude oil viscosity. The city block norm is used for convergence and the genetic breeding size of 200 was used.

RESULTS AND DISCUSSION

The viscosity models presented in this study (CRT, BP, Ward, LM or GRNN) are functions of easily measured field parameters such as crude oil API gravity, produced gas gravity, reservoir temperature, and pressure. Comparison with previously published models by CRT, the models presented in this work use reservoir pressure instead of solution gas-oil ratio as input variable. Thus, it avoids the uncertainty in estimating crude oil viscosity resulting from inaccuracy in estimating solution gas-oil ratio as an input variable from correlations.

The equations developed in this study for estimating saturated and undersaturated oil viscosity by CRT, equations 2 and 3, are strongly dependent on accuracy of estimating dead oil viscosity by equation 1. However, the viscosity models developed by neural networks do not depend on estimated value of dead oil viscosity.

Training samples

Table (1) describes the data range for the training samples used in this study. It shows that, 700 viscosity measurements were selected representing 50 crude oil samples for the training process. The data ranges from heavy crudes (API gravity of 20 degree) to very light crudes (API gravity of 40 degrees). These viscosity measurements cover a temperature range of 130 °F to 243 °F and pressure range from atmospheric pressure (0 psig) to very high pressure of 9900 psi.

Table (2) shows the average relative error percent (Er), the average absolute error percent (Ea), the standard deviation percent (Sd), and the coefficient of correlation (R) for the training samples for each of the regression techniques presented in this study. It's clear from this table, Table 2, that the GRNN has the smallest errors and highest correlation of coefficient, Ea of 2.3%, Sd of 13.43%, and R of 97.89%. The CRT, however, has the highest errors and lowest coefficient of correlations, Ea of 18.26%, Sd of 22.59% and R of 93.56%. This reflects the advantage of using the general regression neural techniques over the classical regression techniques in modeling such problem. Table 2 also shows the strength of the GRNN over the other neural network regression techniques.

Figure 3 shows histogram and normal curve of error distribution for classical and various types of neural regression techniques. Two conclusions can be drawn from this figure. The first is that the CRT has wider error range and smaller frequency of zero error than the neural regression techniques. The second conclusion is that the GRNN is performing better than the other neural network models. The GRNN has a narrow error range and highest frequency of relative error near zero.

The error distribution for various techniques has been checked against the input variables for training samples. Figure 4 shows error distribution against pressure. This figure also depicts that the GRNN has the smallest scatter of error and that the error is randomly distributed. Similar plots, not reported here, show no correlation between error and API gravity, gas gravity, or reservoir temperature.

Testing samples

Table (3) shows the properties of the samples selected for testing the CRT, BP, Ward, LM, and GRNN models. It is important to note here that the properties of the testing samples fall within the range of the training samples shown in Table (1).

Figure 5 shows comparison between measured and predicted viscosity behavior of all testing samples for classical and neural regression techniques. No overfitting exist in any case of regression. The ability of the various regression techniques to capture the physical behavior of changes crude oil viscosity as a function of pressure has been checked for all testing samples.

Figure 6, a typical example for one of the test samples, demonstrates that all the models capture the physical behavior of variation of oil viscosity vs pressure. This figure also illustrates that GRNN closely matches the experimentally measured viscosity better than the other methods. All model shows a decrease of viscosity as a function of pressure below bubble point pressure and increase of viscosity above bubble point pressure. Thus, the models are valid and well behaved.

Table 4 illustrates the error distribution for all method for the testing samples. Again the GRNN technique has the smallest error range and the highest coefficient of correlation. The GRNN has Er of 5.90%, Ea of 18.67% , Sd of 23.45% and R of 96.26%. These error levels are not high based on two facts. The first is that all viscosity model presented in the literature have been developed using all the available data, i.e. no data for testing. The second is that most previously published crude oil viscosity models that were developed by CRT have large average and absolute relative error. Beal [1] reported Er of 24.2 %, Beggs and Robinson[5] reported Ea of 13.53%, and Kartoatmodjo and Schmidt[12] reported Ea of 39.6%. Sutton and Farshad[14] studied the accuracy of published viscosity models. He reported an error of 114.3% when Beggs and Robinson was tested against 93 cases from the literature. This large error might be due to the limitation of CRT in modeling such a complex behavior of crude oil viscosity.

CONCLUSION

A comparison between classical regression and neural regression techniques was presented for the first time in this study. The comparison was achieved using 805 viscosity measurements of crude oil samples. These viscosity data were dived into two sets. The first set comprising 700 data point, training set, that was used in this study to present the proposed models. The second set containing 105 viscosity measurements, testing set, was used to test the accuracy, validity, and physical behavior of the proposed models. These models use easily measured field parameters such as reservoir pressure, temperature, oil API gravity and gas gravity to predict crude oil viscosity. Comparison between classical regression and neural regression techniques reveals that models developed by general regression neural networks (GRNN) successfully simulate the viscosity behavior of crude oil system better than those developed by classical regression techniques. The GRNN presented in this study predicts viscosity data better than any other method. These viscosity models are useful tools to predict crude oil viscosity in case measured ones are not available, at temperature other than reservoir temperature for optimum design of production system and surface facilities, and planing improved oil recovery methods.

NOMENCLATURE

Classical Regression Technique

μ_{od} = Dead oil viscosity, cp
μ_{oa} = Undersaturated oil viscosity, cp
μ_{ob} = Saturated oil viscosity, cp
API = Crude oil API gravity (water = 10)
γ_g = produced gas gravity, (air = 1.0)
T_f = Reservoir temperature, °F
P_b = Bubble point pressure, psi
P = Reservoir pressure, psi

Back Propagation Techniques

a_{jk} = Output of neuron j in layer k
F_k = Transfer function of the neuron in layer k
I = Input pattern
n = Total number of layers
N_k = Number of neurons in layer k
m = Number of components in input pattern
p = Number of input patterns
w_{ijk} = Weight for input I neuron j, and layer k
β_{jk} = Bias for neuron j in layer k
η = Learning rate

γ = Momentum rate

General Regression Neural Network

x = Vector of random variable
y = Scalar random variable
pdf = Probability density function
n = Number of observations
= Expected value of *y*
σ = Smoothing factor

REFERENCES

1. Beal, C.:"Viscosity of Air, Water, Natural Gas, Crude Oil and its Associated Gases at Oil Field Temperature and Pressures," Trans. AIME, 165, 114-127 (1946).

2. Chew, J. and Connally, C. A.:"Viscosity Correlation for Gas-Saturated Crude Oil," Trans. AIME, 216, 23-25 (1959).

3. Lohrenz, J., Bray, B. C. and Clark, C. R.: "Calculating Viscosities of Reservoir Fluids from their Composition," Trans. AIME, 231, JPT, 1170-1176 (October 1964).

4. Little, J. E. and Kennedy, H. T.:"Calculating the Viscosity of Hydrocarbon Systems with Pressure Temperature and Composition," Soc. Pet. Eng. J., Trans. AIME, 234, 157-162 (June 1968).

5. Beggs, H. D. and Robinson, J. R.:"Estimating the Viscosity of Crude Oil Systems," JPT, 1140-1141 (September 1975).

6. Vasquez, M. and Beggs, H. D.:"Correlation Fluid Physical Property Prediction," JPT, 6, 968-970 (1980).

7. Glasø, o.:"Generalized Pressure-Volume-Temperature Correlation for Crude Oil System," JPT, 2, 785-795 (1980).

8. Standing, M. B.:"VOLUMETRIC PHASE BEHAVIOR OF OIL FIELD HYDROCARBON SYSTEMS," Millet Print Inc., Dallas, TX. (1977) 70-95.

9. Khan, S. A., Al-Marhoun, M. A., Duffuaa, S. O. and Abu-Khansin, S. A.:"Viscosity Correlation for Saudi Arabian Crudes," paper SPE 15720, presented at the fifth SPE Middle east Oil Show, Manama, Bahrain, March 7-10, (1987).

10. Abdul-Majeed, G. H., Kattan, R. R., and Salman, N. H.:"New Correlation for Estimating the Viscosity of Undersaturated Crude Oils," J. Can. Pet. Technol., vol. 29, No. 3, 80-85, (May-June 1990).

11. Labedi, R.:"Improved Correlations for Predicting the Viscosity of Light Crudes," J. Pet. Sci. Eng., 8, 221-234 (1992).

12. Kartoatmodjo, F. and Schmidt, Z.:"Large Data Bank Improves Crude Physical Property Correlation," Oil & Gas J, 4, 51-55 (July 1994).

13. Elsharkawy, A. M.:" Models for predicting the viscosity of Middle East Crude oils," Fuel, (1999).

14. Sutton, R. P. and Farshad, F. F.:"Evaluation of Empirically Derived PVT Properties for Gulf of Mexico Crudes," SPE Res. Eng., 79-86. (Feb. 1990).

15 . Gharbi, R. B. and Elsharkawy, A. M.:"Neural Network Model for Estimating the PVT Properties of Middle East Crude Oils," Insitue. 20, 367-394 (1996).

16. Gharbi, R. and Elsharkawy, A.: "Umiversal Neural Network Based Models for Estimating the PVT Properties of Crude Oil Systems," paper SPE 38099, presented at the SPE Asia Pacific Oil & Gas Conference and Exhibition, 14-16 April 1997, Kuala Lumpur, Malaysia.

17. Gharbi, R.: "Estimating the Isothermal Compressibility Coefficient of Undersaturated Middle East Crudes Using Neural Networks," Energy & Fuel, Vol. 11, No. 2, pp. 372-378, (March 1997).

18. Elsharkawy, A. M. : " Modeling the properties of crude oil and gas systems using RBF network". Paper SPE 49961 presented at the SPE South Asia & Pacific, Perth, Australia, October 12-14, 1998.

19. Elsharkawy, A. M. and Foda, S. G.: " EOS simulation and GRNN modeling of the constant volume depletion behavior of gas condensate reservoirs" Energy & Fuel , Vol. 12, No. 2, 353-364, (1998).

20 Rumelhart, D.E.; Hinton, G. E., and Williams, R.J.: Learning Internal Representation by Error Propagation. *Parallel Data Processing*; The MIT Press: Cambridge, MA, 1986; Vol. 1, Chapter 8, pp 318-362.

21. Parzen, E. "On Estimation of a Probability Density Function and Mode, Annals of Mathematical Statistics," 33, 1065-1076 (1962).

22. Cacoullos, T.:"Estimation of a Multivariable Density," Ann. Inst. Statist. Math (Tokyo), Vol. 18, No. 2, 179-189 (1966).

23. Specht, D. F. Probabilistic Neural Networks, IEEE Trans. *Neural Networks*, Vol. 3, No. 1, pp. 109-118, (January 1990).

24. Specht, D. F. :"A General Regression Neural Network", IEEE Trans. *Neural Networks*, Vol. 2, No. 6, 568-576 (November 1991).

25. Specht, D. F. and Shapiro. P. D.: "Generalization Accuracy of Probabilistic Neural Networks Compared with Backpropagation Networks", Proc. International Joint Conf. on Neural Networks, Seattle, WA, 887-892, July 8-14. (1991).

26. Marquez, L. and Hill, T.:"Function Approximation Using Backpropagation and General Regression Neural Networks," Proc. 26th Hawaii International Conf. on System Sciences, Wailea, HI, 607-615, January 5 (1993).

Table 1. Data range for training samples

No. of viscosity data points = 700				
	Gas gravity	APT	Tem., F	Press., psi
Minimum.	0.807	24.51	130	0
Average	0.949	31.89	157	2153
Maximum	1.234	39.81	243	9900

Table 2. Error analysis for classical and neutral regression techniques (training samples)

	CRT	Ward	LM	GRNN	BP
Er %	3.36	0.64	2.43	2.30	-1.24
Ea %	18.26	14.00	12.17	9.39	19.18
Sd %	22.59	17.89	16.76	13.43	24.03
R %	93.56	97.03	98.86	97.89	92.64

Table 3. Data range for testing samples

No. of viscosity data points = 105				
	Gas gravity	APT	Tem., F	Press., psi
Minimum.	0.889	27.85	133	0
Average	0.947	31.91	159	2131
Maximum	0.997	35.96	241	9600

Table 4. Error analysis for classical and neutral regression techniques (testing samples)

	CRT	Ward	LM	GRNN	BP
Er %	12.51	12.38	19.12	6.53	5.90
Ea %	22.91	23.81	23.87	18.45	18.67
Sd %	28.53	32.84	33.09	23.82	23.45
R %	96.40	97.06	97.24	97.02	96.26

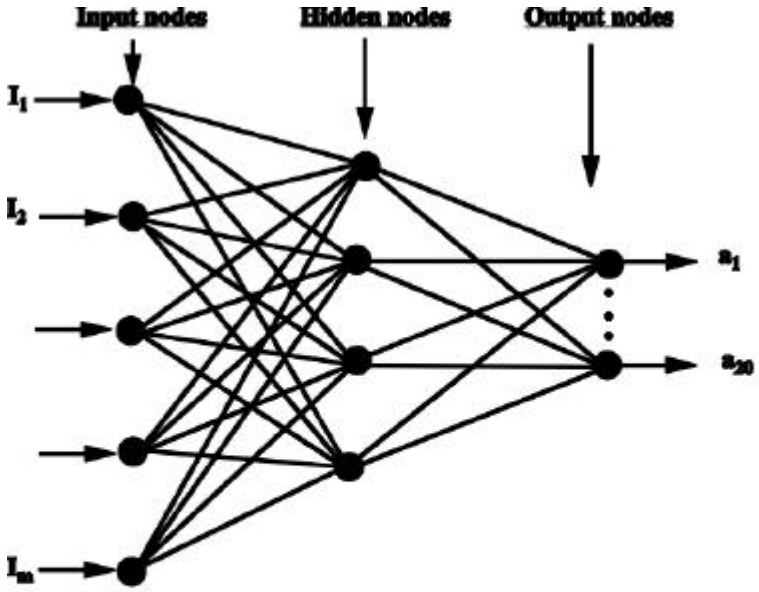

Figure 1. One-hidden layer neural-network model

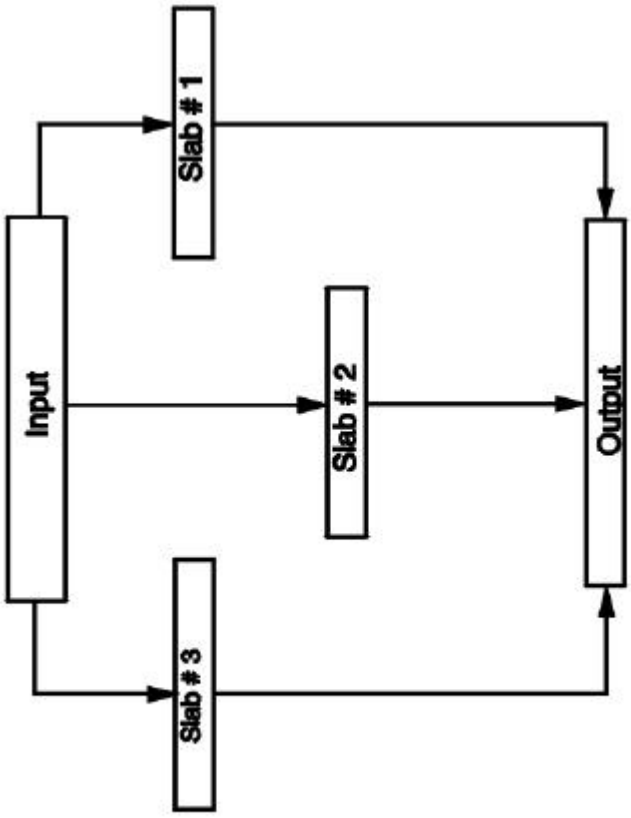

Figure 2. Network with 3 hidden slabs with different activation functions

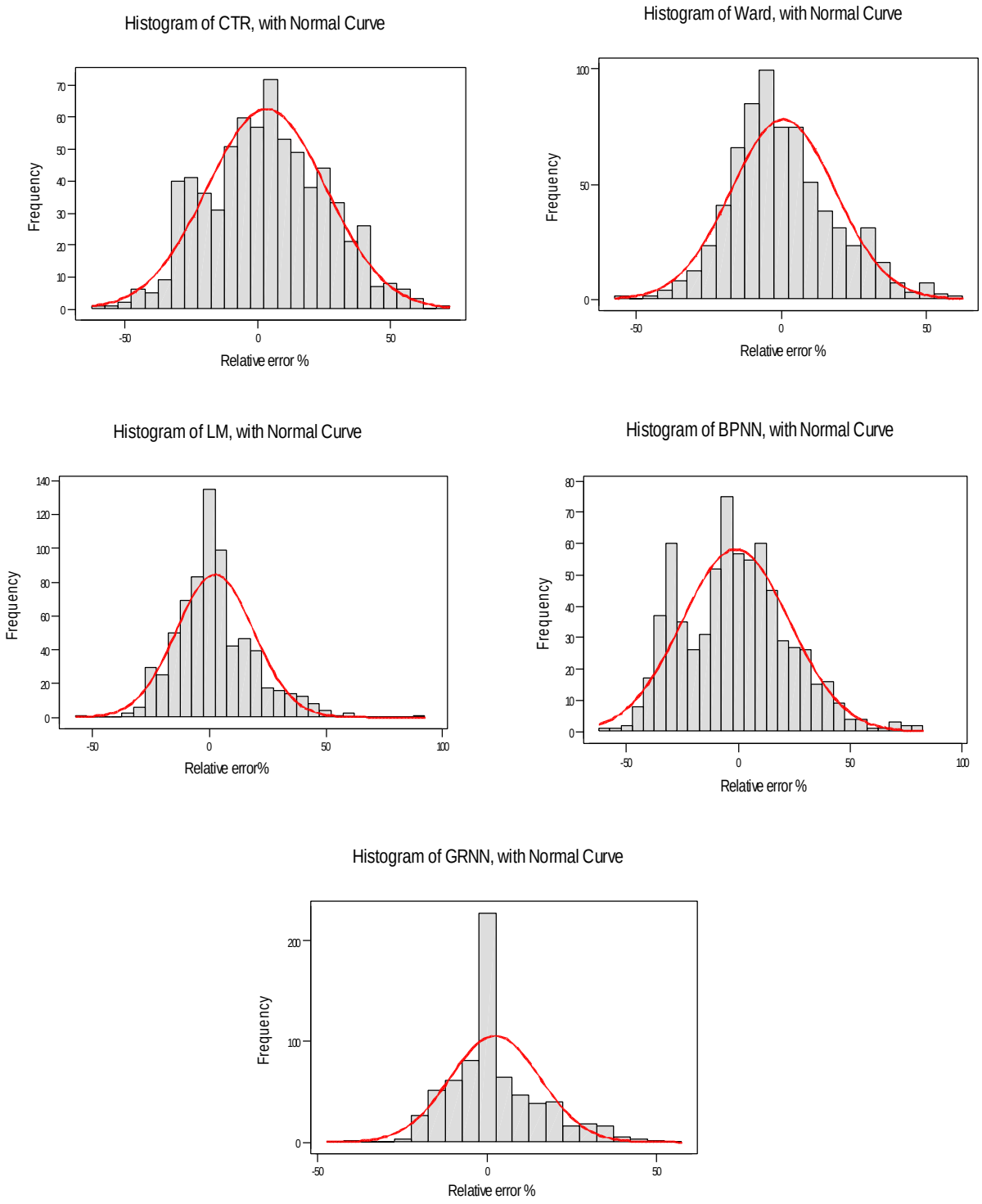

Figure 3 - Relative error Distribution for various viscosity models (training samples)

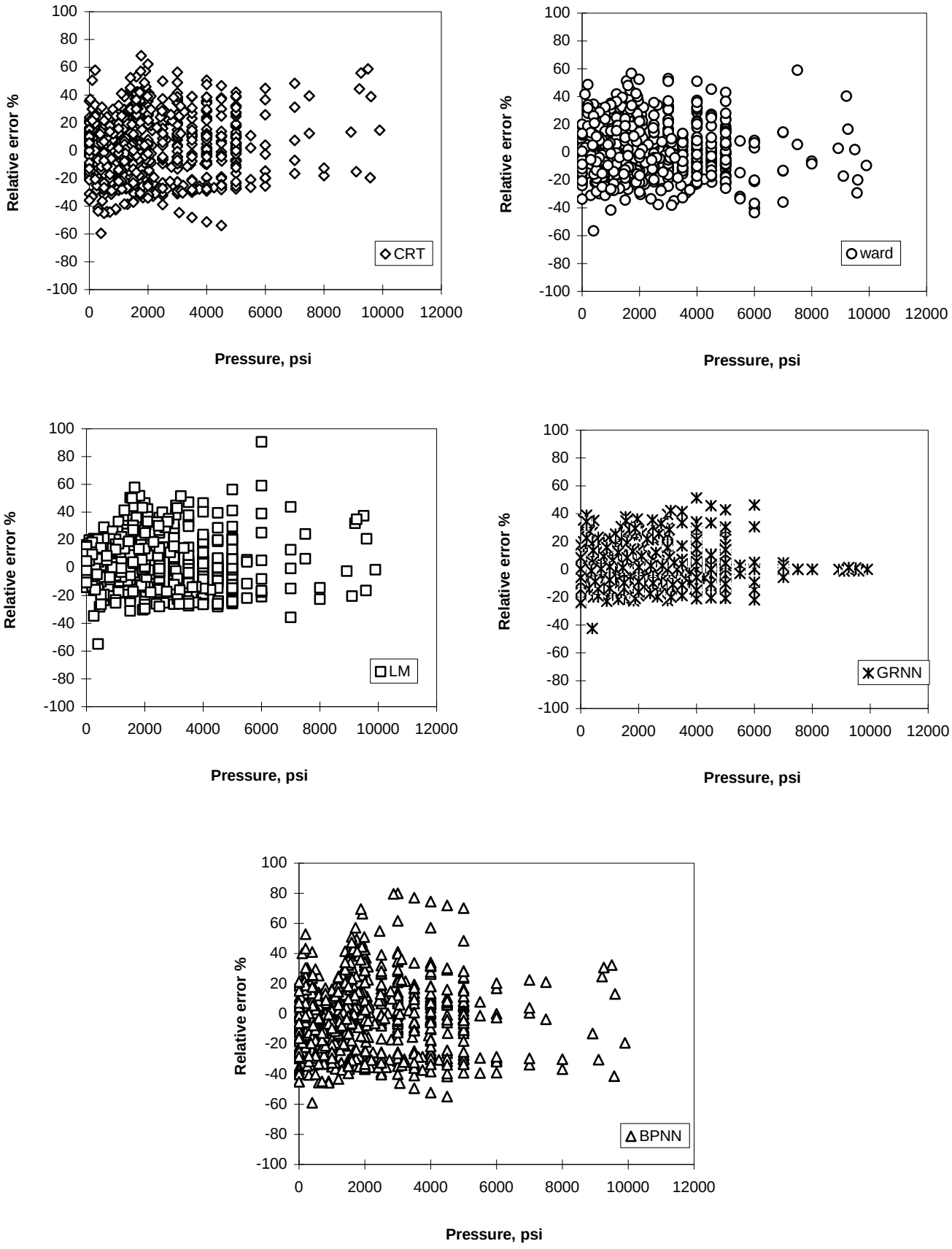

Figure 4- Relative error distribution for various viscosity models

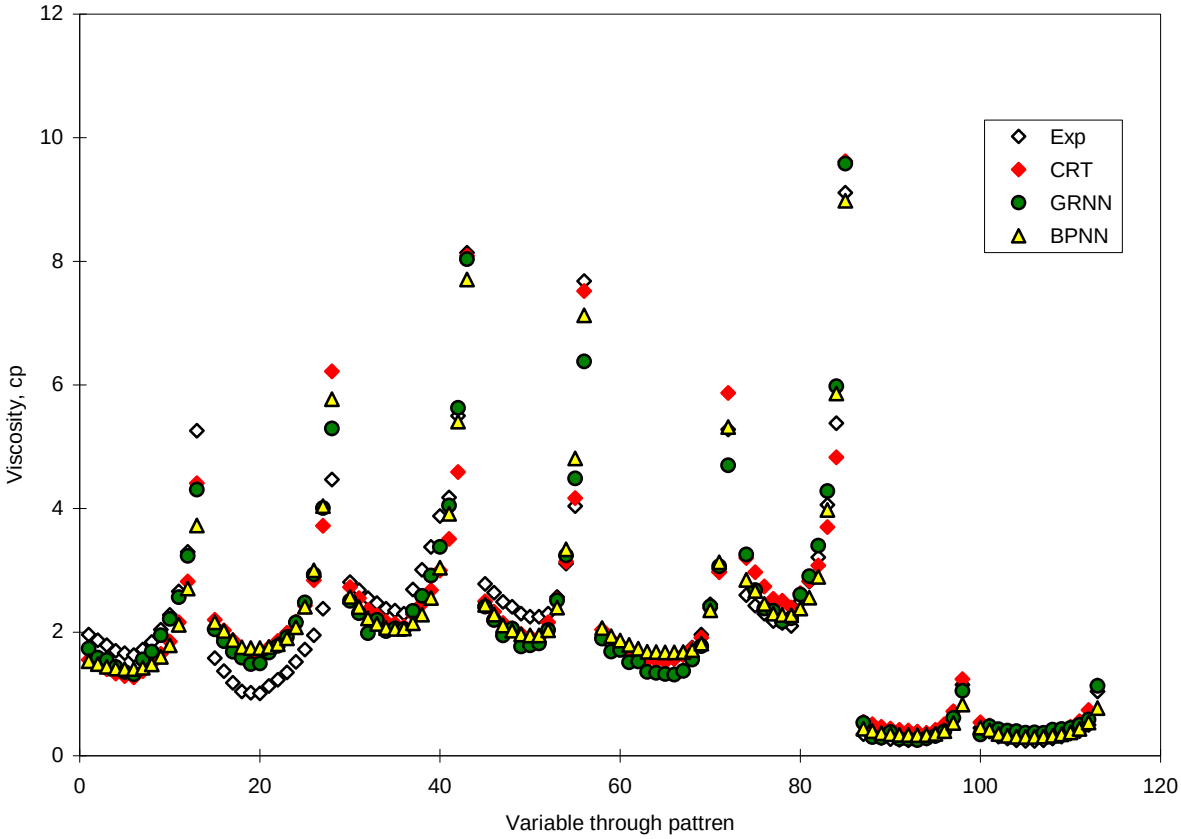

Figure 5- Comparison between measured and predicted viscsoity for testing samples

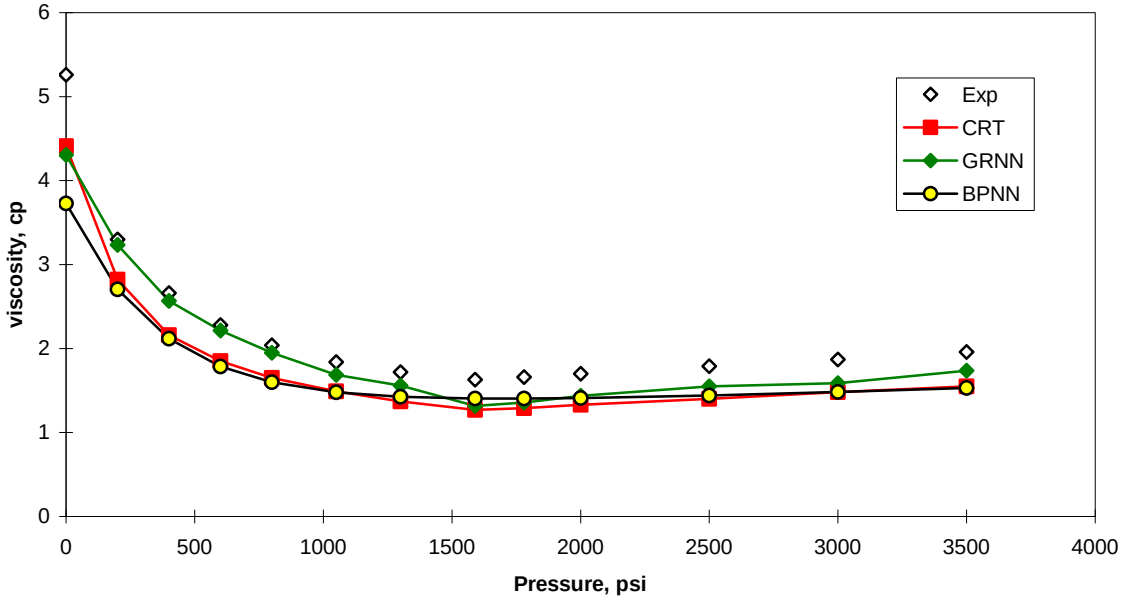

Figure 6- Measured and predicted viscosity for various models (test sample #8)