Soft Computing for Intelligent Reservoir Characterization

D. Tamhane, SPE, P.M. Wong, SPE, University of New South Wales, F. Aminzadeh, SPE, FACT Inc. & dGB-USA, M. Nikravesh, SPE, Energy and Geoscience Institute (EGI)-University of Utah & Zadeh Institute of Information Technology.

Abstract
This paper presents an overview of soft computing techniques for reservoir characterization. The key techniques include neurocomputing, fuzzy logic and evolutionary computing. A number of documented studies show that these intelligent techniques are good candidates for seismic data processing and characterization, well logging, reservoir mapping and engineering. Future research should focus on the integration of data and disciplinary knowledge for improving our understanding of reservoir data and reducing our prediction uncertainty.

Introduction
Accurate prediction of reservoir performance is a difficult problem. This is mainly due to the failure of our understanding of the spatial distribution of lithofacies and petrophysical properties. Because of this, the recovery factors in many reservoirs are unacceptably low. The current technologies based on conventional methodologies are inadequate and/or inefficient. In this paper, we propose the next generation of reservoir characterization tools for the new millennium – soft computing.

Reservoir characterization plays a crucial role in modern reservoir management. It helps to make sound reservoir decisions and improves the asset value of the oil and gas companies. It maximizes integration of multi-disciplinary data and knowledge and improves the reliability of the reservoir predictions. The ultimate product is a reservoir model with realistic tolerance for imprecision and uncertainty. Soft computing aims to exploit such a tolerance for solving practical problems.

Soft computing is an ensemble of various intelligent computing methodologies which include neurocomputing, fuzzy logic and evolutionary computing. Unlike the conventional or hard computing, it is tolerant of imprecision, uncertainty and partial truth. It is also tractable, robust, efficient and inexpensive. In reservoir characterization, these intelligent techniques can be used for uncertainty analysis, risk assessment, data fusion and data mining which are applicable to feature extraction from seismic attributes, well logging, reservoir mapping and engineering. Figure 1 shows schematically the flow of information and techniques to be used for intelligent reservoir characterization. The main goal is to integrate soft data such as geological data with hard data such as 3D seismic and production data to build a reservoir and stratigraphic model. While some individual methodologies (esp. neurocomputing) have gained much popularity during the past few years, the true benefit of soft computing lies on the integration of its constituent methodologies rather than use in isolation.

This paper firstly outlines the unique roles of the three major methodologies of soft computing – neurocomputing, fuzzy logic and evolutionary computing. We will summarize a number of relevant and documented reservoir characterization applications. Lastly we will provide a list of recommendations for the future use of soft computing. This includes the hybrid of various methodologies (e.g. neural-fuzzy or neuro-fuzzy, neural-genetic, fuzzy-genetic and neural-fuzzy-genetic) and the latest tool of “computing with words” (CW). CW provides a completely new insight into computing with imprecise, qualitative and linguistic phrases and is a potential tool for geological modeling which is based on words rather than exact numbers. An appendix is also provided for introducing the basics in soft computing.

Neurocomputing
Neurocomputing represents general computation with the use of artificial neural networks. An artificial neural network is a computer model that attempts to mimic simple biological learning processes and simulate specific functions of human nervous system. It is an adaptive, parallel information processing system which is able to develop associations, transformations or mappings between objects or data. It is also...
the most popular intelligent technique for pattern recognition to date.

The basic elements of a neural network are the neurons and their connection strengths (weights). Given a topology of the network structure expressing how the neurons (the processing elements) are connected, a learning algorithm takes an initial model with some “prior” connection weights (usually random numbers) and produces a final model by numerical iterations. Hence “learning” implies the derivation of the “posterior” connection weights when a performance criterion is matched (e.g. the mean square error is below a certain tolerance value). Learning can be performed by “supervised” or “unsupervised” algorithm. The former requires a set of known input-output data patterns (or training patterns), while the latter requires only the input patterns.

Figure 2 depicts a typical structure of a neural network, showing three layers of neurons. The lines represent how the neurons are connected. Each line is represented by a weight value. In this case, the inputs are passed to each layer and the results are obtained at the output layer. This is commonly known as the feedforward model, in which no lateral or backward connections are used. The full technical details can be found in Bishop³.

Applications. The major applications of neurocomputing are seismic data processing and interpretation, well logging and reservoir mapping and engineering.

Good quality seismic data is essential for realistic delineation of reservoir structures. Seismic data quality depends largely on the efficiency of data processing. The processing step is time consuming and complex. The major applications include first arrival picking, noise elimination, structural mapping, horizon picking and event tracking. A detailed review can be found in Nikravesh and Aminzadeh⁷.

For interwell characterization, neural networks have been used to derive reservoir properties by crosswell seismic data. In Chawathé et al.¹⁵, the authors used a neural network to relate five seismic attributes (amplitude, reflection strength, phase, frequency and quadrature) to gamma ray (GR) logs obtained at two wells in the Sulimar Queen field (Chaves County). Then the GR response was predicted between the wells and was subsequently converted to porosity based on a field-specific porosity-GR transform. The results provided good delineation of various lithofacies.

Feature extraction from 3D seismic attributes is an extremely important area. Most statistical methods are failed due to the inherent complexity and nonlinear information content. Figure 3 shows an example use of neural networks for segmenting seismic characters thus deducing information on the seismic facies and reservoir properties (lithology, porosity, fluid saturation and sand thickness). A display of the level of confidence (degree of match) between the seismic character at a given point versus the representative wavelets (centers of clusters) is also shown. Combining this information with the seismic model derived from the well logs while perturbing for different properties gives physical meaning of different clusters.

Monson and Pita³³ applied neural networks to find relationships between 3D seismic attributes and well logs. The study provided realistic prediction of log responses far away from the wellbore. Boaduxiii also used similar technology to relate seismic attributes to rock properties for sandstones. In Nikravesh et al.⁷, the author applied a combination of k-means clustering, neural networks and fuzzy c-means (a clustering algorithm in which each data vector belongs to each of the clusters to a degree specified by a membership grade) techniques to characterize a field that produces from the Ellenburger Dolomite. The techniques were used to perform clustering of 3D seismic attributes and to establish relationships between the clusters and the production log. The production log was established away from wellbore. The production log and the clusters were then superimposed at each point of a 3D seismic cube. They also identified the optimum locations for new wells based on the connectivity, size and shape of the clusters related to the pay zones (see Figure 4).

The use of neural networks in well logging has been popular for nearly one decade. Many successful applications have been documented⁵⁴,⁵⁵,⁵⁶,⁵⁷. The most recent work by Bruce et al.⁵⁸ presented a state-of-the-art review of the use of neural networks for predicting permeability from well logs. In this application, the network is used as a nonlinear regression tool to develop transformation between well logs and core permeability. Such a transformation can be used for estimating permeability in uncored intervals and wells. One example is shown in Figure 5. In this work, the permeability profile was predicted by a Bayesian neural network⁶. The network was trained by a training set with four well logs (GR, NPHI, RHOB and RT) and core permeability. The network also provided a measure of confidence (the standard deviation of a Gaussian function): the higher the standard deviation (“sigma”), the lower the prediction reliability. This is very useful for understanding the risk of data extrapolation. The same tool can be applied to estimate porosity and fluid saturations. Another important application is the clustering of well logs for the recognition of lithofacies⁶⁵. This provides useful information for improved petrophysical estimates and well correlation.

Neurocomputing has also been applied to reservoir mapping. In Wong et al.⁶⁴ and Wang et al.⁶⁵, the authors applied a radial basis function neural network to relate conceptual distribution of geological facies (in the form of hand drawings) to reservoir porosity. It is able to incorporate the general property trend provided by local geological knowledge and to simulate fine-scaled details when used in conjunction with geostatistical simulation techniques. In Caers⁶⁶ and Caers and Journel⁶⁷, the authors trained a neural network to recognize the local conditional probability based on multiple-point information retrieved from a “training image,” which can be any densely populated image (e.g. outcrop data, photographs, hand drawings, seismics, etc.).
conditional probability was used in stochastic simulation with a Markov Chain sampler (e.g. Markov Chain Monte Carlo). These methodologies can be applied to produce 3D model of petrophysical properties using multiple seismic attributes and conceptual geological maps. This is a significant advantage compared to the conventional geostatistical methods which are limited to two-point statistics (e.g. variograms) and simple objects (e.g. channels).

In Nikravesh et al.xxiv, the authors used neural networks to predict field performance and optimize oil recovery by water injection in the Lost Hill Diatomite (Kern County). They constructed several neural networks to model individual well behavior (wellhead pressure and injection-production history) based on data obtained from the surrounding wells. The trained networks were used to predict future fluid production. The results matched well with the actual data. The study also led to the best oil recovery with the minimum water injected.

### Fuzzy Logic

Fuzzy logic was first introduced by Zadehxxvii almost 35 years ago. Unlike the classical logic which only allows an element either belong or not belong to a particular set, fuzzy logic expresses the degree of membership of the element in each set. It mimics the ability of the human mind to effectively employ modes of reasoning that are approximate rather than exact. It is appropriate to deal with the nature of uncertainty in system and human errors which are not included in current reliability theory. The major significance of fuzzy logic is to simulate human ways of thinking in a formal manner which is better than previous logical tools of mathematics.

Fuzzy logic provides a completely new way of modeling complex and ill-defined systems. The major concept of fuzzy logic is the use of a linguistic variable, that is a variable whose values are words or sentences in a natural or synthetic language. This also leads to the use of fuzzy if-then rules, in which the antecedent and consequents are propositions containing linguistic variables.

Figure 6 shows how fuzzy if-then rules can be used to represent a complex function. In this case, the rules are expresses as “patches” (knowledge) that try to cover the curve. It is obvious that more rules (patches) will give a better coverage and accuracy. If the rules are so precise, they are no longer fuzzy and the patches collapse to crisp points. In reservoir geology however, most observed data are imprecise and ambiguous. Hence, fuzzy logic is a good candidate for solving most reservoir problems.

### Applications

In recent years, fuzzy logic, or more generally, fuzzy set theory, has been applied extensively in many reservoir characterization studies. This is mainly due to the fact that reservoir geology is mainly a descriptive science which uses mostly uncertain, imprecise, ambiguous and linguistic informationxxvii. Fuzzy set theory has the ability to deal with such information and to combine them with the quantitative observations. The applications are many, including seismic and stratigraphic modeling and formation evaluation.

In Boisxxix, he proposed to use fuzzy set theory as a pattern recognition tool to interpret a seismic section. The algorithm produced a synthetic seismic section by convoluting a geological model with a representative impulse (by deconvolution or signature of the source), which were both of subjective and fuzzy nature. The synthetic section was then compared with the original seismic section in a fuzzy context. In essence, the algorithm searches for the appropriate geological model from the observed seismic section by an iterative procedure, which is a popular way for solving inverse problems.

In Baygun et al.xx, the authors used fuzzy logic as a classifier for the delineation of geological objects in a mature hydrocarbon reservoir with many wells. The authors showed that fuzzy logic can be used to extract dimensions and orientation of geological bodies, and geologists can use such a technique for reservoir characterization in a practical way which bypassed many tedious steps.

In Nordlundxxi, the author presented a study on dynamic stratigraphic modeling using fuzzy logic. In stratigraphic modeling, it is possible to model several processes simultaneously in space and time. Although many processes can be modeled using conventional mathematics, modeling the change of deposition and erosion on surfaces is often difficult. Formalizing geological knowledge is a difficult exercise as it involves handling of several independent and complex parameters. In addition, most information is qualitative and imprecise, which are unacceptable for direct numerical treatment. In the paper, the author showed a successful use of fuzzy rules to model erosion and deposition. The fuzzified variables included the depth of the reservoirs, the distances to source and shore, a sinusoidal sea-level curve, tectonic subsidence rate and simulation time with depositional surface. From the study, the author demonstrated that a few (10) fuzzy rules could produce stratigraphies with realistic geometry and facies.

In Cuddyxxxii, the author applied fuzzy logic to solve a number of petrophysical problems in several North Sea fields. The work included lithofacies and permeability from well logs. Lithofacies prediction was based on the use of a possibility value (Gaussian function with a specific mean and variance) to represent a well log belonging to a certain lithofacies. The lithofacies that was associated with the highest combined fuzzy possibility (multiplication of all values) was taken as the most likely lithofacies for that set of logs. A similar methodology was applied to predict permeability by dividing the core permeability values into ten equal bin sizes on a log scale. The problem was converted into a classification problem. All the results suggested that the fuzzy approach had given better petrophysical estimates compared to the conventional techniques.

Fang and Chenxxxiii also applied fuzzy rules to predict porosity and permeability from five compositional and textural characteristics of sandstone in the Yacheng Field (South China
Sea). The five input attributes were the relative amounts of rigid grains, ductile grains and detrital matrix, grain size and the Trask sorting coefficient. All the porosity and permeability data were firstly divided into certain number of clusters by fuzzy c-means. The corresponding sandstone characteristics for each cluster were used to general the fuzzy linguistic rules. Each fuzzy cluster produced one fuzzy if-then rule with five input statements. The formulated rules were then used to make linguistic prediction by combining individual conclusion from each rule. If a numerical output was desired, a defuzzification algorithm could be used to extract a crisp output from a fuzzy set. The results showed that the fuzzy modeling gave better results compared to those presented in Bloch.

In Huang et al., the authors presented a simple but practical fuzzy interpolator for predicting permeability from well logs in the North West Shelf (offshore Australia). The basic idea was to simulate local fuzzy reasoning. When a new input vector (well logs) was given, the system would select two training vectors which were nearest to the new input vector and build a set of piece-wise linear inference rules with the training values, in which the membership value of the training values was one. The study used well log and core data from two wells and the performance was tested at a third well, where actual core data were available for comparison. The accuracy of the permeability predictions at the test well was although similar to that obtained from the authors’ previous neural-fuzzy technique, the fuzzy approach was >7,000 times faster in terms of CPU time.

In Nikravesh and Aminzadeh, the authors applied a neural-fuzzy approach to develop an optimum set of rules for nonlinear mapping between porosity, grain size, clay content, P-wave velocity, P-wave attenuation and permeability. The rules developed from a training set were used to predict permeability in another data set. The prediction performance was very good. The study also showed that the integrated technique discovered clear relationships between P-wave velocity and porosity, and P-wave attenuation and clay content, which were useful to geophysicists.

**Evolutionary Computing**

Evolutionary computing represents computing with the use of some known mechanisms of evolution as key elements in algorithmic design and implementation. A variety of algorithms have been proposed. They all share a common conceptual base of simulating the evolution of individual structures via processes of parent selection, mutation, crossover and reproduction. The major one is the genetic algorithms (GAs).

GAs are efficient global optimization methods for solving ill-behaved, nonlinear and discontinuous problems. Other optimization methods, such as simulated annealing and gradient descent algorithms, are local in nature, adopting an iterative procedure using partial derivatives to improve on some initial model. These methods can lead to a strong dependence on the initial model and are prone to entrapment in local minima. Moreover the calculation of derivatives can be difficult and further add to instability if numerical approximations are used. In contrast, no calculation of partial derivatives or matrix inversion is required in GAs and hence their performance is relatively insensitive to the initial model. The solution can also be evolved in reasonable time using today's desktop computers.

GAs work by firstly encoding the parameters of a given estimator as chromosomes (binary or floating-point). This is followed by populating a range of potential solutions. Each chromosome is evaluated by a fitness function. The better parent solutions are reproduced and the next generation of solutions (children) is generated by applying the genetic operators (crossover and mutation). The children solutions are evaluated and the whole cycle repeats until the best solution is obtained.

The methodology is in fact general and can be applied to optimizing parameters in other soft computing techniques, such as neural networks. In Yao, the author gave an extensive review of the use of evolutionary computing in neural networks with more than 300 references. Three general areas are: evolution of connection weights; evolution of neural network architectures; and evolution of learning rules.

Figure 7 shows a simple example of what evolutionary computing could be used to optimize the neural network architecture. In this diagram, the initial neural network contains five neurons with no connections (Figure 7a). By implementing the architecture into a suitable form of chromosomes, evolutionary computing could produce a complex but optimized architecture in which humans would never attempt to use. For example, Figure 7b shows the final architecture with a combination of feedforward (1 to 3), backward (4 to 2), lateral (1 to 2) and recurrent (3 to 3) connections. Similarly, evolutionary computing can certainly help to discover better solutions for other soft computing techniques.

**Applications.** Most geoscience applications began in early 1990s. Gallagher and Sambridge presented an excellent overview on the use of GAs in seismology. Other applications include geochemical analysis, well logging and seismic interpretation.

Fang et al. first used GAs to predict porosity and permeability from compositional and textural information and the Archie parameters in petrophysics. The same authors later used the same method to map geochemical data into a rock’s mineral composition. The performance was much better than the results obtained from linear regression and nonlinear least-squares methods.

In Huang et al., the authors used GAs to optimize the connection weights in a neural network for permeability prediction from well logs. The study showed that the GA-trained networks (neural-genetic model) gave consistently smaller errors compared to the networks trained by the conventional gradient descent algorithm (backpropagation). However, GAs were comparatively slow in convergence. In Huang et al., the same authors initialized the connection...
weights in GAs using the weights trained by backpropagation. The technique was also integrated with fuzzy reasoning, which gave a hybrid system of neural-fuzzy-genetic
true. This improved the speed of convergence and still obtained better results.

Another important feature of GAs is its capability of extracting fuzzy rules. However, this becomes unpractical when the data sets are large in size. To overcome this, a new encoding technique has been presented recently, which is based on the understanding of biological DNA. Unlike the conventional chromosomes, the length of chromosome is variable and it is flexible to insert new parts and/or delete redundant parts. In Yashikawa et al.xlvi and Nikravesh et al.xlvii, the authors used a hybrid system of neural-fuzzy-DNA model for knowledge extraction from seismic data, mapping the well logs into seismic data and reconstruction of porosity based on multi-attributes seismic mapping.

**Future Trends**

This paper reviews a number of applications of intelligent techniques in reservoir characterization. The results have shown great promises. Many commercial packages are now emerging in various application domains. However, the true benefit of soft computing, which is to use the intelligent techniques in combination (hybrid) rather than isolation, has not been demonstrated in a full extent. This section will address two particular areas for future research: hybrid systems and computing with words.

**Hybrid Systems.** So far we have seen the primary roles of neurocomputing, fuzzy logic and evolutionary computing. Their roles are in fact unique and complementary. Many hybrid systems can be built. For example, fuzzy logic can be used to combine results from several neural networks; GAs can be used to optimize the number of fuzzy rules; linguistic variables can be used to improve the performance of GAs; and extracting fuzzy rules from trained neural networks. Although some hybrid systems have been built, this topic has not yet reached maturity and certainly requires more field studies.

In order to make full use of soft computing for intelligent reservoir characterization, it is important to note that the design and implementation of the hybrid systems should aim to improve prediction and its reliability. At the same time, the improved systems should contain small number of sensitive user-definable model parameters and use less CPU time. The future development of hybrid systems should incorporate various disciplinary knowledge of reservoir geoscience and maximize the amount of useful information extracted between data types so that reliable extrapolation away from the wellbores could be obtained.

**Computing with Words.** One of the major difficulties in reservoir characterization is to devise a methodology to integrate qualitative geological description. One simple example is the core descriptions in standard core analysis. These descriptions provide useful and meaningful observations about the geological properties of core samples. They may serve to explain many geological phenomena in well logs, mud logs and petrophysical properties (porosity, permeability and fluid saturations). Yet, these details are not utilized due to the lack of a suitable computational tool. Gedeon et al.xlviii provided one of the first attempts to relate these linguistic descriptions (grain size, sorting, matrix, roundness, bioturbation and lamina) to core porosity levels (very poor, poor, fair and good) using intelligent techniques. The results were promising and drawn a step closer to Zadeh’s idea on computing with words⁹.

Computing with words (CW) aims to perform computing with objects which are propositions drawn from a natural language or having the form of mental perceptions. In essence, it is inspired by remarkable human capability to manipulate words and perceptions and perform a wide variety of physical and mental tasks without any measurement and any computations. It is fundamentally different from the traditional expert systems which are simply tools to “realize” an intelligent system, but are not able to process natural language which is imprecise, uncertain and partially true. CW has gained much popularity in many engineering disciplinesxlix. In fact, CW plays a pivotal role in fuzzy logic and vice-versa. Another aspect of CW is that it also involves a fusion of natural languages and computation with fuzzy variables.

In reservoir geology, natural language has been playing a very crucial role for a long time. We are faced with many intelligent statements and questions on a daily basis. For example: “if the porosity is high then permeability is likely to be high”; “most seals are beneficial for hydrocarbon trapping, a seal is present in reservoir A, what is the probability that the seal in reservoir A is beneficial?”; and “high resolution log data is good, the new sonic log is of high resolution, what can be said about the goodness of the new sonic log?”

CW has much to offer in reservoir characterization because most available reservoir data and information are too imprecise. There is a strong need to exploit the tolerance for such imprecision, which is the prime motivation for CW. Future research in this direction will surely provide a significant contribution in bridging reservoir geology and reservoir engineering.

**Final Remarks**

This paper reviews a number of soft computing techniques for solving a variety of reservoir characterization problems. The applications include the use of neurocomputing, fuzzy logic and evolutionary computing for seismic data processing and interpretation, well logging, reservoir mapping and engineering. The results are so far promising. The future of soft computing should focus on the development of hybrid systems, which combine the primary strengths of individual techniques, for extracting useful relationships between reservoir data types and performing reliable extrapolation away from the wellbores. The latest tool of computing with words, which has great potential for geological modeling, should be emphasized in the near future.
Acknowledgments
Partial funding for this research was provided by the Gas Research Institute under GRI contract No. 5097-21-4101, project manager Robert W. Siegfried, Zadeh Institute for Information Technology (ZIFIT), Phase I and II pilot study, and “Intelligent Computing System for Reservoir Analysis and Risk Assessment of Red River Formation, Williston Basin, North Dakota (Class II Reservoir) Study” for Luff Exploration Company. The authors express their thanks to Dr. Roy Adams from EGI-University of Utah for his many useful comments and suggestions. The appendix, “A Basic Primer on Soft Computing Terminology,” was provided by Dr. Roy Adams.

References

xxvi. Nikravesh, M., Kovscek, A.R., Murer, A.S. and


Figure 1. Intelligent-integrated reservoir characterization.

Figure 2. A typical neural network model.

Figure 3. A seismic segmentation example (courtesy of dGB).

Figure 4. Optimal well placement.

Figure 5. Permeability profile from Bayesian neural network:
a) gamma ray curve; b) permeability prediction with core data; c) sigma (standard deviation) curve.
Figure 6. Use of fuzzy rules to represent a complex function.

Input Data

1 3 5 Prediction

a) Structure at 1st generation.

Input Data

1 3 5 Prediction

2 4

b) Structure after nth generation.

Figure 7. Evolutionary neural networks.

Appendix 7

A Basic Primer on Soft Computing Terminology

Neural Networks. Neural networks are systems that "... use a number of simple computational units called "neurons" ... and each neuron"... processes the incoming inputs to an output. The output is then linked to other neurons." (p. 63, von Altrock 1995). Neurons are also called "processing elements."

Weight. When used in reference to neural networks, "weight" defines the robustness or importance of the connection (also known as a link or synapse) between any two neurons. Medsker (p. 12, 1994) notes that weights "... express the relative strengths (or mathematical value) of the various connections that transfer data from layer to layer."

Backpropagation learning algorithm. In the simplest neural networks, information (inputs and outputs) flows only one way. In more complex neural networks, information can flow in two directions, a "feedforward" direction and a "feedback" direction. The feedback process is known as "backpropagation." The technique known as a "backpropagation learning algorithm" is most often used to train a neural network towards a desired outcome by running a "training set" of data with known patterns through the network. Feedback from the training data is used to adjust weights until the correct patterns appear. Hecht-Nielsen (p. 124-137, 1990) and Medsker (p. 16, 1994) provide additional information.

Perceptron. There are two definitions of this term (p. 3-10, Hecht-Nielsen 1990). The "perceptron" is a classical neural network architecture. In addition, processing elements (neurons) have been called "perceptrons."

Fuzziness and fuzzy. It is perhaps best to introduce the concept of "fuzziness" using Zadeh's original definition of fuzzy sets (Zadeh 1965): "A fuzzy set is a class of objects with a continuum of grades of membership. Such a set is characterized by a membership (characteristic) function which assigns to each object a grade of membership ranging between zero and one." Zadeh (1973) further elaborates that fuzzy sets are "... classes of objects in which the transition from membership to non-membership is gradual rather than abrupt." Fuzzy logic is then defined as the "... use of fuzzy sets defined by membership functions in logical expressions" (p. 14, von Altrock 1995). Fuzziness and fuzzy can then be defined as having the characteristics of a fuzzy set.

Neuro-fuzzy. This is a noun that looks like an adjective. Unfortunately, "neuro-fuzzy" is also used as an adjective, e.g. "neuro-fuzzy logic" or "neuro-fuzzy systems." Given this confusing situation, a useful definition to keep in mind is: "The combination of fuzzy logic and neural net technology is called "NeuroFuzzy" and combines the advantages of the two technologies." (p. 63, von Altrock 1995). In addition, a neuro-fuzzy system is a neural network system that is self-training, but uses fuzzy logic for knowledge representation, the rules for behavior of the system, and for training the system.

Crisp sets and Fuzzy sets. "Conventional (or crisp) sets contain objects that satisfy precise properties required for membership." (p.1, Bezdek and Pal 1992). Compare this to their definition that "fuzzy sets" "... contain objects that satisfy imprecise properties to varying degrees..." Each member of a crisp set is either "true" or is "false," whereas each member of a fuzzy set may have a certain degree of truth or a certain degree of falseness or may have of some degree of each!

Neural Networks

Details of neural networks are available in the literature (Cybenko 1989; Hecht-Nielsen 1989; Widrow and Lehr 1990; Kohonen 1987, 1997; and Lin and Lee 1996) and therefore only the most important characteristics of neural networks will be mentioned. The typical neural network (Figure 2) has an input layer, an output layer, and at least one hidden layer. Each layer is in communication with the succeeding layer via a set of connections of various weights, i.e. strengths. In a neural network, nonlinear elements are called various names,
including nodes, neurons, or processing elements. A biological neuron is a nerve cell that receives, processes, and passes on information. Artificial neurons are simple first-order approximations of biological neurons.

Consider a single artificial neuron with a transfer function, \( y_l^{(i)} = f(z^{(i)}) \), connection weights, \( w_j \), and a node threshold, \( \theta \). For each pattern \( i \),

\[
z^{(i)} = x_1^{(i)} w_1 + x_2^{(i)} w_2 + \ldots + x_N^{(i)} w_N + \theta \quad \forall i = 1 \ldots P
\]

[A.1]

All patterns may be represented in matrix notation as,

\[
\begin{bmatrix} z^{(1)} \\ z^{(2)} \\ \vdots \\ z^{(P)} \end{bmatrix} = \begin{bmatrix} x_1^{(1)} & x_2^{(1)} & \ldots & x_N^{(1)} & 1 \\ x_1^{(2)} & x_2^{(2)} & \ldots & x_N^{(2)} & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_1^{(P)} & x_2^{(P)} & \ldots & x_N^{(P)} & 1 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix} + \begin{bmatrix} \theta \\ \theta \\ \vdots \\ \theta \end{bmatrix}
\]

[A.2]

and

\[ y_l = f(z) \]

[A.3]

The transfer function, \( f \), is typically defined by a sigmoid function such as the hyperbolic tangent function,

\[ f(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}. \]

During learning, the information is propagated back through the network and used to update the connection weights (backpropagation algorithm). The objective function for the training algorithm is usually set up as a squared error sum,

\[ E = \frac{1}{2} \sum_{i=1}^{P} (y_l^{(i)}_{\text{observed}} - y_l^{(i)}_{\text{prediction}})^2 \]

[A.4]

This objective function defines the error for the observed value at the output layer, which is propagated back through the network. During training, the weights are adjusted to minimize this sum of squared errors.

k-means Clustering

An early paper on k-means clustering was written by MacQueen (1967). k-means is an algorithm to assign a specific number of centers, \( k \), to represent the clustering of \( N \) points (\( k < N \)). These points are iteratively adjusted so that each point is assigned to one cluster, and the centroid of each cluster is the mean of its assigned points. In general, the k-means technique will produce exactly \( k \) different clusters of the greatest possible distinction. The algorithm is summarized in the following:

1. Consider each cluster consisting of a set of \( m \) samples that are similar to each other: \( x_1 \ldots x_m \)
2. Choose a set of clusters \( \{y_1 \ldots y_k\} \).
3. Assign the \( m \) samples to the clusters using the minimum Euclidean distance rule.
4. Compute a new cluster so as to minimize the cost function.
5. If any cluster changes, return to step 3; otherwise stop.
6. End.

Fuzzy c-means Clustering

Bezdek (1981) presents comprehensive coverage of the use of fuzzy logic in pattern recognition. Fuzzy techniques can be used as an alternative method for clustering. Fuzzy clustering partitions a data set into fuzzy clusters such that each data point can belong to multiple clusters. Fuzzy c-means (FCM) is a well-known fuzzy clustering technique that generalizes the classical (hard) c-means algorithm, and can be used where it is unclear how many clusters there should be for a given set of data.

Subtractive clustering is a fast, one-pass algorithm for estimating the number of clusters and the cluster centers in a set of data. The cluster estimates obtained from subtractive clustering can be used to initialize iterative optimization-based clustering methods and model identification methods. The algorithm is summarized in the following:

1. Consider a finite set of elements \( X = \{x_1 \ldots x_n\} \) or \( j = 1 \ldots n \).
2. Select a number of clusters \( c \).
3. Choose an initial partition matrix, \( U^{(0)} \):

\[ U = \begin{bmatrix} u_{ij} \\ \vdots \\ u_{ij} \end{bmatrix} \]

where \( u_{ij} \) expresses the degree to which the element of \( x_j \) belongs to the \( i \)th cluster, \( \sum_{i=1}^{c} u_{ij} = 1 \) \( \forall j = 1 \ldots n \) and \( 0 < \sum_{j=1}^{n} u_{ij} < 1 \) \( \forall i = 1 \ldots c \).
4. Choose a termination criteria, \( \varepsilon \).
5. Set the iteration index \( l \) to 0.
6. Calculate the fuzzy cluster centers using \( U^{(l)} \) and an objective function.
7. Calculate the new partition matrix \( U^{(l+1)} \) using an objective function
8. Calculate \( \Delta = \left| U^{(l+1)} - U^{(l)} \right| = \max_{i,j} |u_{ij}^{(l+1)} - u_{ij}^{(l)}| \).
9. If \( \Delta > \varepsilon \), then set \( l = l + 1 \), return to step 6; otherwise stop.
10. End.
Neural Network Clustering
Kohonen (1987, 1997) wrote two fundamental books on neural network clustering. The self-organizing map technique known as Kohonen’s self-organizing feature map (Kohonen, 1997) can be used as an alternative for clustering purposes. This technique converts patterns of arbitrary dimensionality (the pattern space) into the response of one- or two-dimensional arrays of neurons (the feature space). This unsupervised learning model can discover any relationship of interest such as patterns, features, correlations, or regularities in the input data, and translate the discovered relationship into outputs. The algorithm is summarized in the following:

1. Consider the network structure as shown in Figure A.1.
2. Use the following similarity match:
   \[ \| x - w_i \| = \min_j \| x - w_j \| \]
3. The learning rule is defined as:
   \[ w_{i(t+1)} = \begin{cases} 
   w_i^{(t)} + \alpha (x^{(t)} - w_i^{(t)}) & \text{if } x^{(t)} \text{ is in the neighborhood set of the winner node } i \text{ at the time step } t \\
   w_i^{(t)} & \text{otherwise.} 
   \end{cases} \]
   with \( \alpha \) defined as a learning constant.

Appendix References