



# Applications of Soft Computing Methods in Environmental Engineering

# 82

Kaan Yetilmezsoy

## Contents

Introduction .....	2002
Description of Soft Computing Methods .....	2004
Artificial Neural Networks (ANN) .....	2005
Fuzzy Logic (FL) .....	2010
Adaptive Neuro-Fuzzy Inference System (ANFIS) .....	2015
Support Vector Machines (SVM) .....	2018
Implementation of Soft Computing Methods in Environmental Engineering .....	2020
ANN-Based Applications for Water and Wastewater Treatment .....	2020
ANN-Based Applications for Air Quality/Pollution Control/Forecasting .....	2021
FL-Based Applications for Water and Wastewater Treatment .....	2023
FL-Based Applications for Air Quality/Pollution Control/Forecasting .....	2025
ANFIS-Based Applications for Water and Wastewater Treatment .....	2026
ANFIS-Based Applications for Air Quality/Pollution Control/Forecasting .....	2027
SVM-Based Applications for Water and Wastewater Treatment .....	2027
SVM-Based Applications for Air Quality/Pollution Control/Forecasting .....	2028
Illustrative Soft Computing Examples for Environmental Engineers .....	2030
Example 1 .....	2030
Solution of Example 1 .....	2032
Example 2 .....	2034
Solution of Example 2 .....	2036
Conclusion .....	2040
References .....	2040

---

K. Yetilmezsoy (✉)

Department of Environmental Engineering, Faculty of Civil Engineering, Yildiz Technical University, Istanbul, Esenler, Turkey

e-mail: [yetilmez@yildiz.edu.tr](mailto:yetilmez@yildiz.edu.tr)

© Springer Nature Switzerland AG 2019

C. M. Hussain (ed.), *Handbook of Environmental Materials Management*,  
[https://doi.org/10.1007/978-3-319-73645-7\\_149](https://doi.org/10.1007/978-3-319-73645-7_149)

2001

---

**Abstract**

Soft computing has been extensively studied and applied in the last three decades for scientific research and engineering computing. In environmental engineering, researchers and engineers have successfully employed different methods of soft computing for modeling of various real-life environmental problems. In this study, applications of core soft computing techniques, such as artificial neural networks (ANN), fuzzy logic (FL), adaptive neuro fuzzy inference systems (ANFIS), and support vector machines (SVM), are investigated and important mathematical aspects of these methods are highlighted. Considering the concepts and methods, this study briefly reviews applications of soft computing techniques in the field of environmental engineering, especially in water/wastewater treatment and air quality/pollution control/forecasting. A brief introduction to complexity of environmental problems and the general definition soft computing concept are presented in the first section of this chapter. The second section comprises four subsections and presents mathematical background of four different soft computing methods. Section “[Implementation of Soft Computing Methods in Environmental Engineering](#),” which is consisted of eight subsections, reviews successful applications of soft computing-based prediction models implemented in the field of environmental engineering and summarizes the important findings obtained in these studies. At the end of the overview of the published works on soft computing applications in different environmental areas, in the last section, some special illustrative soft computing examples and the respective MATLAB<sup>®</sup>-based solutions are presented for environmental engineers.

---

**Keywords**

Adaptive neuro fuzzy inference systems (ANFIS) · Aggregation · Air quality/pollution control/forecasting · Algorithm · Artificial neural networks (ANN) · Backpropagation · Bayesian regulation · Broyden–Fletcher–Goldfarb–Shanno (BFGS) · Centroid · Classification · Defuzzification · Early stopping · Environmental engineering · Feed-forward · Firing strength · Fletcher–Reeves · Fuzzification · Fuzzy inference system (FIS) · Fuzzy logic (FL) · Fuzzy operator · Gradient descent · Hessian matrix · Implication · Jacobian matrix · Kernel functions · Lagrange multipliers · Levenberg–Marquardt · Linguistic · Logarithmic sigmoid · MATLAB<sup>®</sup> · Membership function · Modeling · Momentum factor · Normalized layer (N) · Polak–Ribière · Powell–Beale · Prediction · Product layer ( $\pi$ ) · Quasi–Newton · Scaled conjugate gradient · Soft computing · Support vector machines (SVM) · Tangent sigmoid · Training · Water and wastewater treatment

---

**Introduction**

The real-life environmental problems are very complex and highly dependent on several process configurations, different influent characteristics, and various operational conditions. For a sustainable control of environmental-related problems, the proposed systems must be continuously monitored and properly controlled due to

possible instabilities in circumstance conditions. Although statistical models may be able to establish a relationship between the input and the output variables without detailing the causes and effects in the formation of pollutants, they are not capable of capturing the inherent nonlinear nature of the environmental problems. For this reason, the complicated inter-relationships among a number of system factors in the process may be explicated through a number of attempts in developing representative and powerful prediction models allowing the investigation of the key variables in greater detail. At this point, soft computing-based control of real-time process variables may provide several potential advantages, such as protection of the system from possible risks associated with significant fluctuations in influent characteristics, optimization of the process at a reasonable cost, providing a rapid evaluation and estimation of pollutant loads and emissions on energetic basis, and also development of a continuous early-warning strategy without requiring a complex formulation and laborious parameter estimation procedures (Yetilmezsoy et al. 2011a, b, 2015).

The principal soft computing technologies can be categorized as fuzzy algorithms, neural networks, supporting vector machines, evolutionary communication, machine learning, and probabilistic reasoning (Jang and Topal 2014). McCulloch and Pitts (1943) introduced an initial model of an artificial neural network (ANN), which was recognized as the first study of artificial intelligence. It has been widely accepted as an approach, which acts like a “black-box” model derived from a simplified concept of the human brain, for prediction, control systems, classification, optimization, and decision-making in various fields (Antwi et al. 2017). In 1965, fuzzy logic (FL) theory was proposed by Zadeh (1965) as a new soft computing methodology in order to address uncertainty and subjectivity (i.e., human experience and intuition) within the framework of fuzzy sets which could be described by linguistic variables and membership functions according to a fuzzy rule-based system (Assimakopoulos et al. 2013). In 1993, soft computing became a formal area of computer science and many new and hybrid algorithms, i.e., adaptive neuro fuzzy inference systems (ANFIS) (Jang 1993), were introduced with the help of advanced computer technology (Jang and Topal 2014). The SVM method, developed by Vapnik (1995), can provide an effective novel approach to overcome the inherent drawbacks such as over-fitting training, local minima, and poor generalization performance of ANN when studying with large initial data. Since SVM implements Structural Risk Minimization Principle (SRMP), instead of the Empirical Risk Minimization Principle (ERMP) like feed-forward neural networks, its process leads to better generalization than conventional methods (Yeganeh et al. 2012). The main advantage of the SVM over multilayer perceptron (MLP) or neuro-fuzzy network is its good generalization ability, acquired at relatively small number of learning data and at large number of input nodes (high dimensional problem) (Osowski and Garanty 2007; Yeganeh et al. 2012).

Among soft computing techniques, ANN provides configurations made up of interconnecting artificial neurons that mimic the properties of biological neurons. It is used in a wide range of applications as a multilayer feed-forward network with back propagation learning algorithm. A typical neural network includes three layers:

first, the input layer, second, the output layer, and third, the hidden layer or intermediate layer (Gocic et al. 2015). Other alternative methodologies have also emerged from artificial intelligence, such as FL, which is currently being tested in real environmental problems. Its success is mainly due to its closeness to human perception and reasoning, as well as its intuitive handling and simplicity, which are important factors for handling of imprecise data (Kotti et al. 2013). This method develops multivalued, nonnumeric linguistic variables for modeling human reasoning in an imprecise environment. Nevertheless, it is noted that both ANN and FL control sometimes exist some shortages. For instance, ANN may have limitations in performing heuristic reasoning of the domain problem; on the other hand, the FL control may be difficult to design and adjust automatically. Moreover, the use of artificial neural networks is, however, challenged by the difficulty of network design and parameterization. Many factors affect the performance of ANN that include network topology, training algorithm and parameters setting, and network architecture. Likewise, the outcome of fuzzy classification highly depends on the predefined fuzzy rules (Dwarakish and Nithyapriya 2016). So ANFIS is designed as a fuzzy neural network model, it can use the both advantages. ANFIS consists of both ANN and FL including linguistic express of membership functions and *if-then* rules of Takagi and Sugeno's type (Mingzhi et al. 2009). SVM is another novel soft learning algorithm that has been recently realized for a wide range of applications in the field of soft computing, hydrology, and environmental studies. It emerged as a set of supervised generalized linear classifiers and often provide higher classification accuracies than multilayer perceptron ANN. It is essentially a kernel-based procedure and relatively new machine learning method that has been recently applied as one of the leading techniques for pattern classification and function approximation (Gocic et al. 2015; Huang et al. 2010; Pai et al. 2011; Singh et al. 2011).

This chapter is aimed at bringing forward original and the recent trends and efforts in the application of some soft computing methods in environmental engineering. It is especially interested in describing the successful application and advances in soft computing-based modeling of real-world environmental processes. The sections of this chapter summarize various applications of (1) artificial neural networks (ANN), (2) fuzzy logic (FL) control systems, (3) adaptive neurofuzzy inference systems (ANFIS), and (4) support vector machines (SVM) for modeling of various environmental problems based on water and wastewater treatment and air quality/pollution control/forecasting.

---

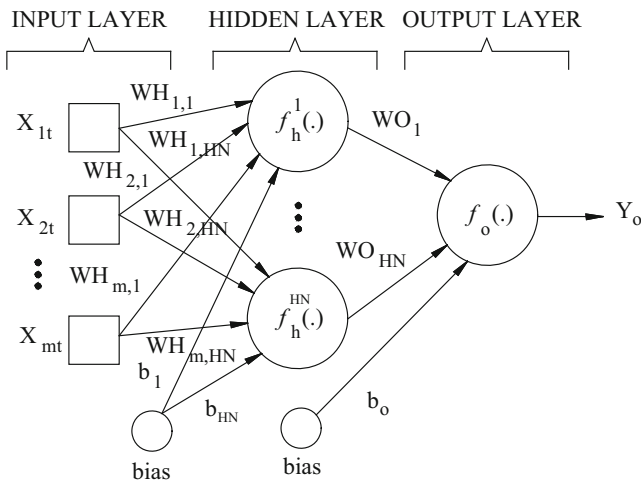
## Description of Soft Computing Methods

In this section, the basis of the widely used AI-based techniques, such as ANN, FL, ANFIS, and SVM, are briefly summarized and important mathematical aspects of these methods are highlighted. Moreover, computational issues, advantages, and particular theoretical principles are described, and some methodological techniques are discussed to make a comparative assessment of the present AI-based prediction models.

## Artificial Neural Networks (ANN)

To better control a specific environmental process, a robust mathematical tool for predicting the process performance must be developed based on past observations of certain key parameters. Modeling a multivariate system is highly difficult due to the complexity of the environmental processes exhibiting nonlinear behavior that are difficult to describe by linear mathematical models (Hamed et al. 2004). Although deterministic models (also called white-box models) may provide insight into the mechanism, they require hard work before being applied to a specific environmental process. As an alternative to physical models, artificial neural networks (ANNs) are a valuable forecast tool in environmental sciences. They can be used effectively due to their learning capabilities and their low computational costs (Wieland et al. 2002). Because of their reliable, robust, and salient characteristics in capturing the nonlinear relationships between variables (multi-input/output) in multivariate systems, numerous applications of ANN-based models have been successfully utilized in the field of environmental engineering in the past decade (Yetilmezsoy and Demirel 2008).

The ANN-based models are meant to interact with objects in the real world in the same way that the biological nervous system does. The calibration of ANN-based models is easier than the white-box models as fewer parameters are used in the model development process. For this reason, artificial intelligence techniques using ANN have recently become immensely popular and attractive mathematical tools for both modeling and controlling of several complex environmental processes. When the measured variables begin showing difference with the response of ANN, the model can be retrained using the newer data used for cross-checking. These facts and the quality of the results they provide make the ANN-based models more attractive than conventional models (Agirre-Basurko et al. 2006).



**Fig. 1** Simple schematic of an ANN model (Adapted from Hamed et al. 2004)

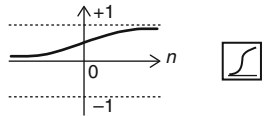
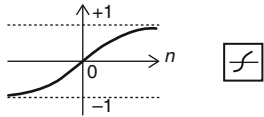
A simple diagram of an ANN model is depicted in Fig. 1. As seen in Fig. 1, each neuron is connected to several of its neighbors, with varying coefficients or weights representing the relative influence of the different neuron inputs to other neurons. The weighted sum of the inputs are transferred to the hidden neurons, where it is transformed using an activation function such as a tangent sigmoid activation function. In turn, the outputs of the hidden neurons act as inputs to the output neuron where they undergo another transformation. The output of a feed-forward ANN with one hidden layer and one output neural network is given as follows (Hamed et al. 2004; Antwi et al. 2017):

$$Y_o = f_o \left[ \sum_{j=1}^{HN} WO_j \times f_h \left( \sum_{i=1}^m WH_{ij} \times X_{it} + b_j \right) + b_o \right] \tag{1}$$

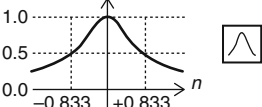
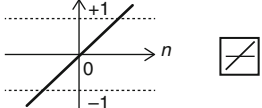
where  $WH_{ij}$  is the weight of the link between the  $i$ th input and the  $j$ th hidden neuron,  $m$  is the number of input neurons,  $WO_j$  is the weight of the link between the  $j$ th hidden neuron and the output neuron,  $f_h$  is the hidden neuron activation function,  $f_o$  is the output neuron activation function,  $b_j$  is the bias of the  $j$ th hidden neurons,  $b_o$  is the bias of the output neuron,  $X_{it}$  is the input variable, and  $HN$  is the number of hidden neurons.

Hamed et al. (2004) reported that the tangent sigmoid (*tansig*) activation functions for the input and hidden neurons are needed to introduce nonlinearity into the network in order to make nets more powerful than plain perceptrons. Moreover, the authors reported that a linear activation function, such as linear transfer function (*purelin*), could be selected for the output neuron since it is appropriate for continuous valued targets.

The logarithmic sigmoid function *logsig*( $x$ ) produces outputs between 0 and 1 as the node’s net input goes from negative to positive infinity. Alternatively, the *tansig*( $x$ ) as transfer function can be used. Sigmoid outputs nodes are often employed for pattern recognition problems, while linear or *purelin*( $x$ ) transfer function is applied for function fitting problems shows the *purelin* transfer function (Ghaedi and Vafaei 2017). The mathematical definitions of some widely used differentiable activation or transfer functions are given as follows (Yetilmezsoy and Sapci-Zengin 2009; Ghaedi and Vafaei 2017):

Function graphs	Mathematical definitions	
	$y = \text{logsig}(x) = f(x) = \frac{1}{(1+e^{-x})}$	(2)
	$y = \text{tansig}(x) = f(x) = \frac{2}{(1+e^{-2x})} - 1 = \frac{(1-e^{-2x})}{(1+e^{-2x})}$	(3)

(continued)

Function graphs	Mathematical definitions	
	$y = radbas(x) = f(x) = e^{-x^2}$	(4)
	$y = purelin(x) = f(x) = x$	(5)

Among the many types of ANNs, backpropagation (BP) networks have recently been considered as one of the simplest and most widely used network models (Cai et al. 2009). The learning process of a BP network consists of two main iterative steps: forward computing of data stream and backward propagation of error signals. During forward computing, original data are transmitted from the input layer to the output layer through the hidden processing layer, with the neurons of each layer only affecting the neurons of the succeeding layer. One of the main advantages of BP networks over other types of networks is that if the desired output cannot be obtained from the output layer, the error is propagated backwards through the network against the direction of forward computing (Cai et al. 2009; Liu and Meng 2009). According to the error signal of BP, the network changes the network connection of all layers to determine the best weight set and realize the correct network output (Liu and Meng 2009). Therefore, with these two steps performing iteratively, the error between network output and desired output can be minimized using the delta rule (Cai et al. 2009).

The network training is a process by which the connection weights and biases of the ANN are adapted through a continuous process of simulation by the embedded network’s environment. The training function applies the inputs to the new network, calculates the outputs, compares them to the associated targets, and calculates a mean square error. If the error goal is met, or if the maximum number of epochs is reached, the training is stopped and the training function returns the new network and a training record. Otherwise, the training goes through another epoch. During the adaptation phase, the training algorithm receives part of the data (inputs and outputs) and automatically develops the ANN model. After development, the model could generate the appropriate responses for simulations with varying levels of data input. When the learning is complete, the neural network is used for prediction. The primary goal of training is to minimize an error function by searching for a set of connection strengths and biases that causes the ANN to produce outputs equal or close to the targets. In other words, the training aims at estimating the parameters ( $WH_{ij}$ ,  $WO_j$ ,  $b_j$ , and  $b_o$ ) by minimizing an error function (Yetilmezsoy et al. 2011a).

As data set was trained, the input pattern given to the input layers of the network would compute the output in the output layer. The BP learning rule defined a method to adjust the weights of the networks (Antwi et al. 2017). A BP algorithm as one of the strongest learning algorithms is a gradient descent algorithm that can be employed to learn these multilayer feed-forward networks with differentiable transfer functions.

The learning method is based on a gradient search, with a criterion of errors between the values of network output and desired output (Ghaedi and Vafaei 2017):

$$E = \sum_{i=1}^N (O_n - O_d)^2 \quad (6)$$

where  $E$  is the total sum squared error of all data in the training set, in which  $O_n$  is the network output for the  $n$ th data and  $O_d$  is the desired output.

In the training process, the weights of all the connecting nodes are modified until the required error level is obtained or the maximum number of iteration is reached. In order to minimize the total error of the network trained by BP algorithm, the weights are adjusted according to the following equation (Ghaedi and Vafaei 2017; Pendashteh et al. 2011):

$$\Delta w_{ki}^n(m+1) = -\eta \times \frac{\partial E}{\partial \Delta w_{ki}^n} + \mu \times \Delta w_{ki}^n(m) \quad (7)$$

where  $\Delta w_{ki}^n(m)$  is the correction of the weight at the  $m$ th learning step,  $\eta$  is the training rate (a small parameter to alter the correction each time), and  $\mu$  is the momentum factor (decrease an oscillation and helps quick convergence). Network learning adjusts using suitable values of these parameters.

Because gradient decent usually slows down near minima, so the Levenberg–Marquardt algorithm (LMA) method can be used to obtain faster convergence. LMA is a blend of simple gradient descent and the Gauss–Newton method. The algorithm for parameter updating is presented by the following equation (Pendashteh et al. 2011):

$$\Delta w = -[J^T J + \mu I]^{-1} J^T \varepsilon \quad (8)$$

where  $\varepsilon = [e_1 \ e_2 \ \dots \ e_p]^T$  is the error vector.  $\mu$  is a positive constant,  $I$  is the identity matrix, and  $J$  is the Jacobian matrix given by (Pendashteh et al. 2011):

$$J = \begin{bmatrix} \partial e_1 / \partial w_1 & \partial e_1 / \partial w_2 & \dots & \partial e_1 / \partial w_N \\ \partial e_2 / \partial w_1 & \partial e_2 / \partial w_2 & \dots & \partial e_2 / \partial w_N \\ \vdots & \vdots & \dots & \vdots \\ \partial e_p / \partial w_1 & \partial e_p / \partial w_2 & \dots & \partial e_p / \partial w_N \end{bmatrix} \quad (9)$$

In general, ANNs are sensitive to the number of neurons in their hidden layers. Too few neurons may lead to underfitting. Conversely, too many neurons may contribute to overfitting, wherein all training points fit well, although the fitting curve may take wild oscillations between the points. In this case, the error on the training set is driven to a very small value, however, when new data are presented to the network, the error becomes enlarged. Although the network has memorized the training examples, it has not learned to generalize to new situations. This can be prevented either by training



with Bayesian regulation, a modification of the Levenberg–Marquardt algorithm (LMA), or by using early stopping with any of the other training routines. In turn, this requires that the user pass a validation set to the training algorithm, in addition to the standard training set (Akkoyunlu et al. 2010). However, in practice, it is difficult to know which training algorithm will perform fastest for a given problem. It will depend on many factors, including the complexity of the problem and the number of data points in the training set (Yetilmezsoy and Saral 2007).

In general, on networks that contain up to a few hundred weights, the LMA will have the fastest convergence. It has found to be the fastest method for training moderate sized feedforward ANN, where the training rate is 10 to 100 times faster than the usual gradient descent BP method (Al-Daoud 2009). However, when the number of network weights is large, the requirement for computation and memory becomes significant. Since in LMA, inversion of square matrix  $J^T J + \mu I$  is involved; thus, a large memory space is required to store the Jacobian matrix and the Hessian matrix ( $J^T J$ ) along with inversion of approximated Hessian matrix in each iteration (Pendashteh et al. 2011). The Quasi–Newton methods are often the next fastest algorithms on networks of moderate size, while the Broyden–Fletcher–Goldfarb–Shanno (BFGS) Quasi–Newton BP algorithm is generally faster than the conjugate gradient algorithms. Of the conjugate gradient algorithms, the Powell–Beale procedure requires the most storage, but usually has the fastest convergence. Meanwhile, the Polak–Ribière has performance similar to the Powell–Beale, the storage requirements for which (4 vectors) are slightly larger than for the Fletcher–Reeves (3 vectors). The Fletcher–Reeves generally converges in fewer iterations than the Resilient backpropagation algorithm (Rprop). Although more computation is required in each iteration, the Rprop and the scaled conjugate gradient algorithm do not require a line search and have small storage requirements. They are reasonably fast and are very useful for large problems. The variable learning rate algorithm is usually much slower than the other methods and has approximately the same storage requirements as Rprop; however, it can still be useful for some problems. The one-step secant algorithm requires less storage and computation per epoch than does the BFGS algorithm; however, it requires slightly more storage and computation per epoch than do the conjugate gradient algorithms. This algorithm can be considered a compromise between the Quasi–Newton algorithms and the conjugate gradient algorithms. In the batch gradient methods, the weights and biases are updated in the direction of the negative gradient of the performance function. The scaled conjugate gradient (SCG) algorithm uses a step size scaling mechanism and avoids a time-consuming line-search per learning iteration, which takes the algorithm faster than other second order conjugate gradient algorithms, Quasi–Newton algorithms, and heuristics algorithms. Therefore, this method shows superlinear convergence on most problems (Zakaria et al. 2010). The loss on the optimality of the estimates/predictions produced by some other training algorithms may be attributed to the combinatorial nature and nonlinear structure of the considered problem. Therefore, the complexity analysis of the present problem can be validated by the results of several training algorithms used in the benchmark comparison.

Based on the above-mentioned facts, it can be noted that the performance of the various algorithms can be affected by the accuracy required of the

approximation, which is dependent on the mean square error, versus that of several representative algorithms. When the problem formulation has a combinatorial nature, the definition of each process parameter results in a complex interaction of variables used in the calculations. A number of benchmark comparisons of the various training algorithms are needed in order to choose the best-suited algorithm for obtaining a good performance on the laborious interactive and nonlinear problems. In general, the LMA will have the fastest convergence on combinatorial function approximation (or nonlinear regression) problems (Akkoyunlu et al. 2010).

Since ANN-based models contain no preconceptions regarding what the model shape will be, they are ideal for cases with low system knowledge. They are useful for functional prediction and system modeling where the physical processes are not understood or are highly complex. Consequently, it is believed that ANN-based techniques, which have recently been applied to various environmental problems, may provide a good alternative to statistical and theoretical techniques, as well as to iterative problems, because of their speed and capability of learning, robustness, nonlinear characteristics, nonparametric regression capabilities, generalization properties, and ease of working with regards to high-dimensional data.

## Fuzzy Logic (FL)

The fuzzy logic system based on linguistic expressions includes uncertainty rather than numerical probabilistic, statistical, or perturbation approaches. Fuzzy set theory (Zadeh 1965) was introduced to provide a definition for uncertainties caused by imprecision and vagueness present in real-world applications (Ozcan et al. 2009; Nasiri and Huang 2008). Rihani et al. (2009) reported that fuzzy logic has recently become a useful tool for modelling highly complex systems whose behaviors are not well understood. For instance, considering the complex qualitative relationships among the variables in a water-in-oil emulsion system, the fuzzy logic methodology has the advantage of the relatively simple mathematical calculations in linguistic terms instead of complicated equations used in the conventional methods. Since a fuzzy logic-based model does not need to handle tedious empirical formulations and complex mathematical expressions, this technique provides a transparent and a systematic analysis for the interpretation of dynamic behavior of a water-in-oil emulsion-based problem by a set of logical connectives (Yetilmezsoy et al. 2012). The key idea in fuzzy logic, in fact, is the allowance of partial belongings of any object to different subsets of a universal set instead of belonging to a single set completely. This is an artificial intelligence method utilizes fuzzy sets and linguistic terms to describe the complex qualitative relationships between model components (Ozcan et al. 2009; Nasiri and Huang 2008; Rihani et al. 2009).

There are basically five parts of the fuzzy inference process:

In the first step (fuzzification), crisp numerical inputs and outputs are divided into different fuzzy categories associated with linguistic terms (i.e., low, high, big, small, too-cold, cold, warm, hot, too-hot, young, old, etc.), where the output is

always a fuzzified degree of a specific membership function within the range from 0 to 1 (Jantzen 1999; Altunkaynak et al. 2005; Sozen et al. 2004). Instead of a definition for the developed fuzzy set categories such as moderately low, low, moderate, moderately high, high, etc., the membership functions can be defined as A, B, C, D, E, etc., to simplify processing of the rules (Yetilmezsoy et al. 2012). Since multiple measured crisp inputs first have to be mapped into the specific fuzzy membership functions, Sozen et al. (2004) reported that the fuzzification process requires good understanding of all the variables. Before the rules can be evaluated, the inputs must be fuzzified according to each of these linguistic sets.

In second step, after the inputs are fuzzified, the fuzzy operator (AND or OR) in different pieces of the antecedent is performed in the fuzzy inference system (FIS) for each fuzzy rule. It is noted that the fuzzy rule base contains some rules that include all possible fuzzy relations between inputs and output variables (or actions and conclusions). In fuzzy set theory, there are no mathematical equations and model parameters, and therefore, all the uncertainties, nonlinear relationships, and model complications are included in the descriptive fuzzy inference procedure in the form of *if-then* (*if* premise *then* consequent) logical statements, called fuzzy rules (Rihani et al. 2009; Akkurt et al. 2004; Acaroglu et al. 2008). If the antecedent of a given rule has more than one part, the fuzzy operator is applied to obtain one number that represents the result of the antecedent for that rule. This number is then applied to the output function. The input to the fuzzy operator is two or more membership values from fuzzified input variables. The output is a single truth value. Two kinds of built-in AND methods (*min* (minimum) and *prod* (product):  $prod(a,b) = ab$ ), and two kinds of built-in OR methods (*max* (maximum) and *probor* (the probabilistic OR method:  $probor(a,b) = a + b - ab$ )) can be used in the fuzzy logic toolbox (Altunkaynak et al. 2005; Sozen et al. 2004; Akkurt et al. 2004).

In the third step, an implication process from the antecedent to the consequent is performed in the FIS. This procedure is defined as the shaping of the consequent (a fuzzy set) based on the antecedent (a single number). The input for the implication process is a single number given by the antecedent, and the output is a fuzzy set (Kusan et al. 2010). Before applying the implication method, the rule's weight must be determined. Every rule has a weight (a number between 0 and 1), which is applied to the number given by the antecedent. After proper weighting has been assigned to each rule, the implication method is implemented. A consequent is a fuzzy set represented by a membership function, which weights appropriately the linguistic characteristics that are attributed to it. The consequent is reshaped using a function associated with the antecedent (a single number). The input for the implication process is a single number given by the antecedent, and the output is a fuzzy set. Implication is implemented for each rule. For this process, two built-in methods are basically supported by the fuzzy logic toolbox, and they are the same functions that are used by the AND operator: *min* (minimum), which truncates the output fuzzy set, and *prod* (product), which scales the output fuzzy set (Altunkaynak et al. 2005; Sozen et al. 2004; Akkurt et al. 2004).

In the fourth step, aggregation process is performed to fuzzy sets to obtain a single fuzzy set that represents the outputs of each fuzzy rule. Because decisions are based on the testing of all of the rules in a FIS, the rules must be combined in some manner in order to make a decision. Therefore, aggregation is the process by which the fuzzy sets that represent the outputs of each rule are combined into a single fuzzy set. The input of the aggregation process is the list of fuzzy sets that represent the outputs of each rule. Aggregation only occurs once for each output variable, just prior to the fifth and final step, defuzzification. The input of the aggregation process is the list of truncated output functions returned by the implication process for each rule. The output of the aggregation process is a fuzzy set. There are a number of aggregation methods (i.e., *max* (maximum), *sum* (simply the sum of each rule's output set), *probor*, etc.) supported by the FIS (Altunkaynak et al. 2005; Sozen et al. 2004; Akkurt et al. 2004). The nature of the information retrieval dictates that the determination of the ranking should be done based on all of the rules. In this case, the sum aggregation method appears to be a much better fit (Rubens 2006).

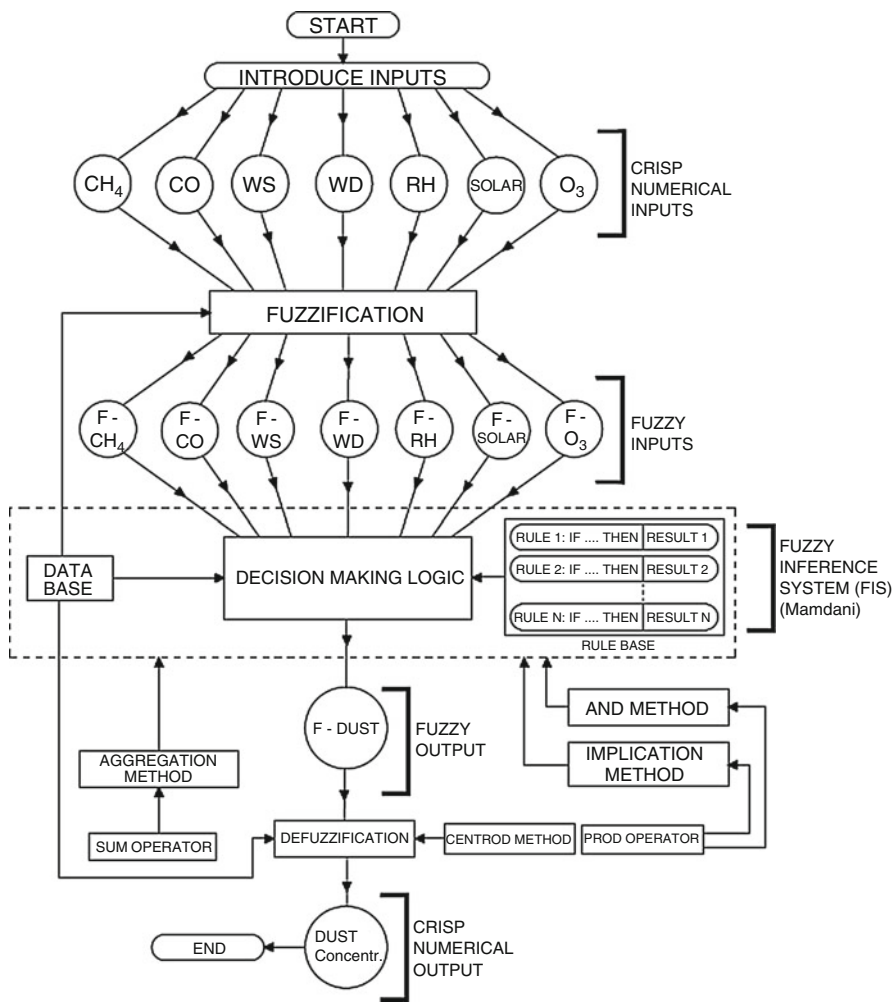
Finally, the defuzzifier produces the crisp values corresponding to the final fuzzy outputs as a conclusion (Jantzen 1999). The input for the defuzzification process is a fuzzy set (the aggregate output fuzzy set) and the output is a single number. There are many defuzzification methods such as center of gravity (COG or; centroid), bisector of area (BOA), mean of maxima (MOM), leftmost maximum (LM), rightmost maximum (RM), etc. (Nasiri and Huang 2008). In the defuzzification step, linguistic results obtained from the fuzzy inference are translated into a crisp numerical output (real value) by using the rule base provided (Kusan et al. 2010; Biyikoglu et al. 2005). In the literature, several defuzzification methods, such as center of gravity (COG or centroid), bisector of area, mean of maxima, leftmost maximum, rightmost maximum, have been reported (Jantzen 1999). It is apparent from several fuzzy logic-based studies (Turkdogan-Aydinol and Yetilmezsoy 2010; Yetilmezsoy et al. 2012; Altunkaynak et al. 2005; Akkurt et al. 2004; Rubens 2006; Sadiq et al. 2004), centroid method is most widely used defuzzification technique, since it satisfies the underlying properties of the system and exhibits the best performance. It is determined as follows (Turkdogan-Aydinol and Yetilmezsoy 2010; Yetilmezsoy et al. 2012; Sozen et al. 2004; Akkurt et al. 2004):

$$(y_i)_d = \frac{\sum_{i=1}^n \mu(y_i)y_i}{\sum_{i=1}^n \mu(y_i)} \quad (10)$$

where  $(y_i)_d$  is the defuzzified output,  $y_i$  is the output value (or the centroidal distance from the origin) in the  $i$ th subset, and  $\mu(y_i)$  is the membership value of the output value in the  $i$ th subset. For the continuous case, the summations in Eq. (5) are replaced by integrals, as given by Sadiq et al. (2004). On the basis of above-

mentioned fuzzy steps, a detailed schematic of a sample MISO (multiple inputs and single output) fuzzy system is depicted in Fig. 2.

The situations of uncertainties in fuzzy-logic are defined via giving appropriate membership functions to the elements of the set that represent the situation. The value of the variation between 0 and 1 (the highest level) for each element is called membership degree and its value in subset is called membership function (Topcu and Saridemir 2008). In fuzzy models, the shape of membership functions of fuzzy sets can be triangular, trapezoidal, bell-shaped, sigmoidal, or another appropriate form, depending on the nature of the system being studied (Acaroglu et al. 2008;



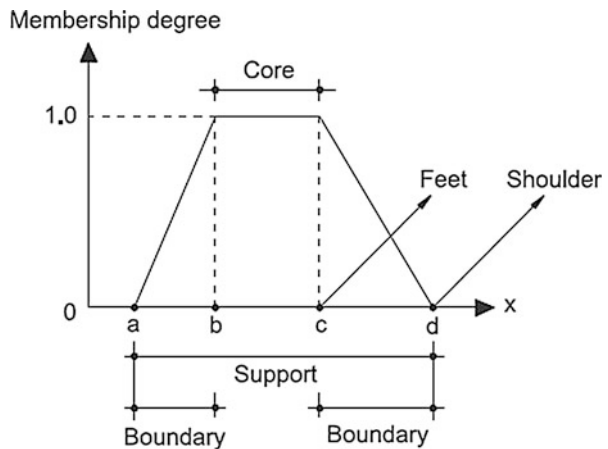
**Fig. 2** A detailed schematic of a sample MISO fuzzy system (Adapted from Yetilmezsoy and Abdul-Wahab 2012)

Metternicht and Gonzalez 2005). Among them, triangular and trapezoidal shaped membership functions are predominant in current applications of fuzzy set theory, due to their simplicity in both design and implementation based on little information (Yetilmezsoy et al. 2012; Rihani et al. 2009). A schematic overview of the trapezoidal-based membership function is given in Fig. 3. The trapezoidal curve is the membership function of a vector,  $x$ , and depends on four scalar parameters,  $a, b, c, d$ , as follows (Turkdogan-Aydinol and Yetilmezsoy 2010; Yetilmezsoy et al. 2012; Altunkaynak et al. 2005; Sozen et al. 2004; Adriaenssens et al. 2006):

$$\mu(x) = \mu(x; a, b, c, d) = \left. \begin{cases} 0, x \leq a \\ \frac{x-a}{b-a}, a < x < b \\ 1, b \leq x \leq c \\ \frac{d-x}{d-c}, c < x < d \\ 0, x \geq d \end{cases} \right\} \quad (11)$$

In the applications of the fuzzy system in both control and forecasting, there are two types of fuzzy inference systems, namely, Mamdani-type (Mamdani and Assilian 1975) and Takagi-Sugeno-type (Takagi and Sugeno 1985) fuzzy systems (Rihani et al. 2009; Ozger and Sen 2007; Sadrzadeh et al. 2009). Sadrzadeh et al. (2009) reported that each *if-then* rule produces a fuzzy set for the output variable in the Mamdani approach, and hence defuzzification step is indispensable to obtain crisp values of the output variable. Because of allowing a simplified representation and interpretation of the fuzzy rules, Mamdani’s fuzzy inference method is the most commonly applied fuzzy methodology (Turkdogan-Aydinol and Yetilmezsoy 2010; Yetilmezsoy et al. 2012; Akkurt et al. 2004; Acaroglu et al. 2008; Adriaenssens et al. 2006; Traore et al. 2005).

**Fig. 3** A schematic overview of the trapezoidal-based membership function (Adapted from Yetilmezsoy et al. 2011a)



## Adaptive Neuro-Fuzzy Inference System (ANFIS)

The ANFIS consists of two parts, antecedent and conclusion, which are connected to each other by fuzzy rules based on the network form. Since the consequent parameters are calculated forward, while the premise parameters are calculated backward, operation of the ANFIS looks like feed-forward back propagated (FFBP) ANN (Atmaca et al. 2001). Zero or first-order Sugeno inference systems or Tsukamoto inference system can be used in the fuzzy section. The output variables ( $f_i$ ) are then obtained by performing several fuzzy rules to fuzzy sets of input variables (Yetilmezsoy et al. 2011a, 2015; Atmaca et al. 2001; Cakmakci et al. 2010):

$$\text{Rule 1 : If } x \text{ is } A_1 \text{ and } y \text{ is } B_1, \text{ then } f_1 = p_1x + q_1y + r_1 \quad (12)$$

$$\text{Rule 2 : If } x \text{ is } A_2 \text{ and } y \text{ is } B_2, \text{ then } f_2 = p_2x + q_2y + r_2 \quad (13)$$

where  $p_1, p_2, q_1, q_2, r_1,$  and  $r_2$  are linear parameters, and  $A_1, A_2, B_1,$  and  $B_2$  are the nonlinear parameters.

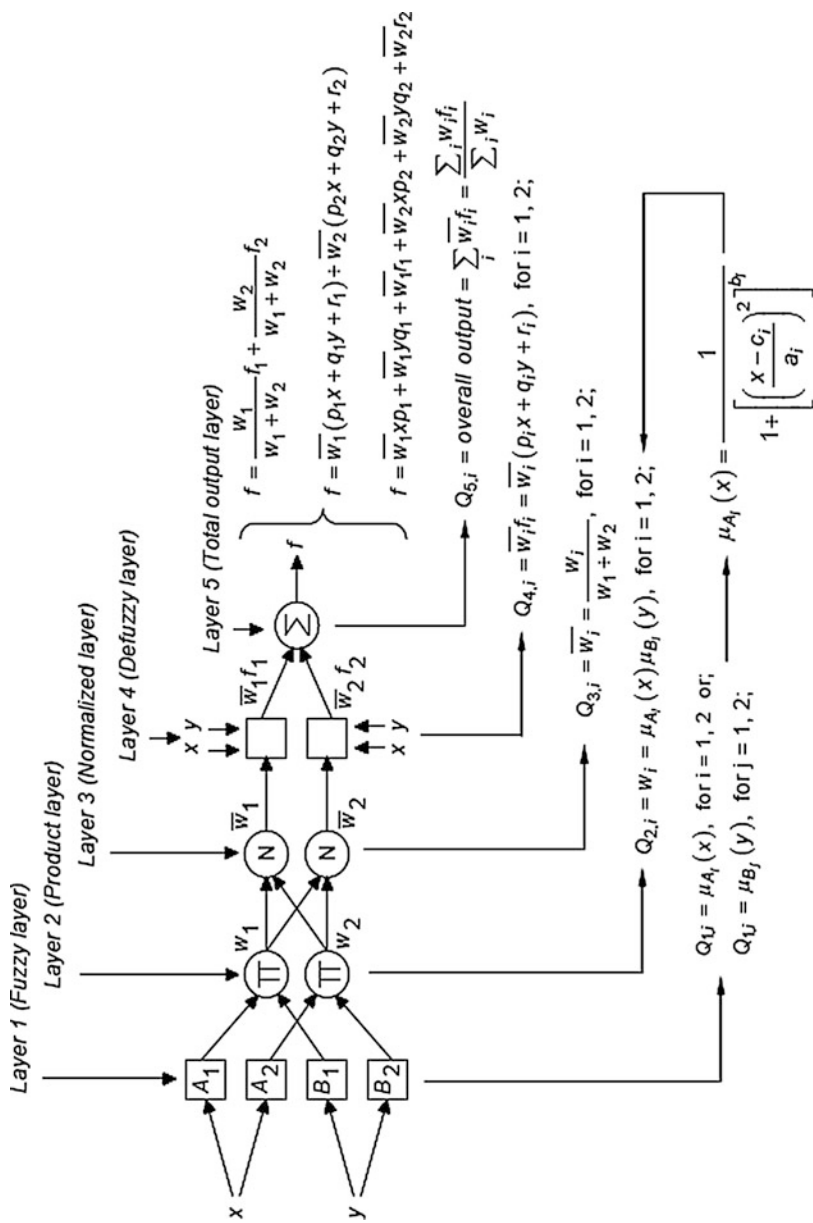
The ANFIS architecture (equivalent of a two input first-order Sugeno FIS model) including the input ( $x$  and  $y$ ) of nodes ( $A_1, A_2, B_1,$  and  $B_2$ ), membership functions ( $\mu_{A_i}(x)$  or  $\mu_{B_j}(y)$ ), membership grades (or outputs of layers) of the fuzzy sets ( $Q_{1,i}, Q_{2,i}, Q_{3,i}, Q_{4,i}, Q_{5,i}$ ), weight functions of the next layers ( $w_1$  and  $w_2$ ), normalized firing strengths ( $\bar{w}_1$  and  $\bar{w}_2$ ), and the consequent parameters ( $p_1, q_1, r_1, p_2, q_2, r_2$ ) is illustrated in Fig. 4. As seen in Fig. 4, the equivalent ANFIS architecture consists of five layers: Fuzzy layer, product layer ( $\pi$ ), normalized layer (N), defuzzy layer, and total output layer (Yetilmezsoy et al. 2011a, b, 2015; Cakmakci et al. 2010).

As seen in Fig. 4, Layer 1 is *the fuzzy layer*, in which  $x$  and  $y$  are the input of nodes  $A_1, A_2, B_1,$  and  $B_2$ , respectively.  $A_1, A_2, B_1,$  and  $B_2$  are the linguistic labels used in the fuzzy theory for dividing the membership functions. Parameters in this layer are referred to as premise parameters. Every node  $i$  in Layer 1 is an adaptive node with a specific function. Nodes in Layer 1 implement fuzzy membership functions, mapping input variables to corresponding fuzzy membership values. The membership relationship between the output and input functions of this layer can be expressed as (Yetilmezsoy et al. 2011a, b):

$$Q_i^1 = \mu_{A_i}(x), \text{ for } i = 1, 2 \text{ or;} \quad (14)$$

$$Q_i^1 = \mu_{B_i}(y), \text{ for } i = 1, 2 \quad (15)$$

where  $x$  or  $y$  is the input to node  $i$ , and  $A_i$  or  $B_i$  is the linguistic label (such as small, large, etc.) associated with this node function,  $Q_i^1$  denotes the output functions, and  $\mu_{A_i}(x)$  or  $\mu_{B_i}(y)$  usually denotes the bell-shaped membership functions with a maximum equal to 1 and a minimum equal to 0, such as (Yetilmezsoy et al. 2011a, b; Jang 1993; Esmaeizadeh and Dariane 2014):



**Fig. 4** A five-layer ANFIS architecture (equivalent of a two input first-order Sugeno FIS model consisting of two inputs and rules) (Adapted from Yetilmezsoy et al. 2015)



$$\mu_{A_i}(x) = \frac{1}{1 + \left[ \left( \frac{x-c_i}{a_i} \right)^2 \right]^{b_i}} \text{ or;} \quad (16)$$

$$\mu_{A_i}(x) = \exp \left[ - \left( \frac{x-c_i}{a_i} \right)^2 \right] \quad (17)$$

where  $(a_i, b_i, \text{ and } c_i)$  is the parameter set. As the values of these parameters change, the bell-shaped functions vary accordingly, thus exhibiting various forms of membership functions on linguistic label,  $A_i$ . In fact, any continuous and piecewise differentiable functions, such as commonly used trapezoidal and triangular-shaped membership functions, are also be used as node functions in this layer (Jang 1993).

Layer 2 is *the product layer* that consists of two fixed circle nodes labelled  $\pi$ , which multiply the incoming signals and provides the outputs of the product. The output  $w_1$  and  $w_2$  are the weight functions of the next layer. The output of this layer is the product of the input signal, which is defined as follows (Yetilmezsoy et al. 2011a, b; Jang 1993; Esmaealzadeh and Dariane 2014):

$$Q_i^2 = w_i = \mu_{A_i}(x) \cdot \mu_{B_i}(y), \text{ for } i = 1, 2 \quad (18)$$

where  $Q_i^2$  denotes the output of Layer 2. Each node output represents the firing strength of a rule.

The third layer is *the normalized layer*, whose nodes are labelled N. The  $i$ th node calculates the ratio of the  $i$ th rules firing strength to the sum of all rule's firing strengths. Its function is to normalize the weight function in the following process (Yetilmezsoy et al. 2011a, b; Jang 1993; Esmaealzadeh and Dariane 2014):

$$Q_i^3 = \bar{w}_i = \frac{w_i}{w_1 + w_2}, \text{ for } i = 1, 2 \quad (19)$$

where  $Q_i^3$  denotes the output of Layer 3. The outputs of this layer are called *normalized firing strengths*.

The fourth layer is *the defuzzy layer*, whose nodes are adaptive. Every node  $i$  in this layer is an adaptive node with a specific function. The output equation is  $\bar{w}_i(p_i x + q_i y + r_i)$ , where  $p_i, q_i, \text{ and } r_i$  denote the linear parameters or so-called *consequent parameters* of the node. The defuzzy relationship between the input and output of this layer can be defined as follows (Yetilmezsoy et al. 2011a, b; Jang 1993; Esmaealzadeh and Dariane 2014):

$$Q_i^4 = \bar{w}_i f_i = \bar{w}_i (p_i x + q_i y + r_i), \text{ for } i = 1, 2 \quad (20)$$

where  $Q_i^4$  denotes the output of Layer 4.

The fifth layer is *the total output layer*, whose node is labelled  $\Sigma$ . The output of this layer is the total of the input signals, which represents the vehicle shift decision

result. The results can be written as (Yetilmezsoy et al. 2011a, b; Jang 1993; Esmaeelzadeh and Dariane 2014):

$$Q_i^5 = \text{overall output} = \sum_i \overline{w_i} f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \quad (21)$$

where  $Q_i^5$  denotes the output of Layer 5.

Although ANN and fuzzy logic models are the basic areas of artificial intelligence concept, the ANFIS combines these two methods and uses the advantages of both methods. Since the ANFIS is an adaptive network which permits the usage of ANN topology together with fuzzy logic, it includes the characteristics of both methods and also eliminates some disadvantages of their lonely used case. Therefore, this technique it is capable of handling complex and nonlinear problems. Even if the targets are not given, the ANFIS may reach the optimum result rapidly. In addition, there is no vagueness in ANFIS as opposed to ANNs (Atmaca et al. 2001; Jang et al. 1997). Moreover, the learning duration of ANFIS is very short compared to ANN-based models. It implies that ANFIS may reach to the target faster than ANN. Therefore, when a more sophisticated system with a high-dimensional data is implemented, the use of ANFIS instead of ANN would be more appropriate to overcome faster the complexity of the problem (Atmaca et al. 2001).

In the ANFIS structure, the implication of the errors is different from that of the ANN case. In order to find the optimal result, the epoch size is not limited. In training of high-dimensional data, the ANFIS can give results with the minimum total error compared to ANN and fuzzy logic methods. Moreover, fuzzy logic method seems to be the worst in contrast to others at a first look, since the rule size is limited and the number of membership functions of fuzzy sets were chosen according to the intuitions of the expert. However, if different types of membership functions and their combinations had been tested and more membership variables and more rules had been used to enhance the prediction performance of the proposed diagnosis system, better results would have been available (Turkdogan-Aydinol and Yetilmezsoy 2010; Atmaca et al. 2001).

## Support Vector Machines (SVM)

The SVM is a linear machine of one output  $y(x)$ , working in the high dimensional feature space formed by the nonlinear mapping of the  $N$ -dimensional input vector  $\mathbf{x}$  into a  $K$ -dimensional feature space ( $K > N$ ) through the use of the nonlinear function  $\varphi(\mathbf{x})$ . The number of hidden units ( $K$ ) is equal to the number of so-called support vectors that are the learning data points, closest to the separating hyperplane. The learning task is transformed to the minimization of the error function, while keeping the weights of the network at minimum. The error function is defined through the so-called  $\varepsilon$ -insensitive loss function  $L_\varepsilon(d, y(\mathbf{x}))$  (Vapnik 1998; Osowski and Garanty 2007; Yeganeh et al. 2012):

$$L_\varepsilon(d, y(\mathbf{x})) = \begin{cases} |d - y(\mathbf{x})| - \varepsilon & \text{for } |d - y(\mathbf{x})| \geq \varepsilon, \\ 0 & \text{for } |d - y(\mathbf{x})| < \varepsilon, \end{cases} \quad (22)$$

where  $\varepsilon$  is the assumed accuracy,  $d$  is the destination,  $\mathbf{x}$  the input vector, and  $y(\mathbf{x})$  the actual output of the network under excitation of  $\mathbf{x}$  and the actual output signal of the SVM network is defined by

$$y(\mathbf{x}) = \sum_{j=1}^K w_j \varphi_j(\mathbf{x}) + b = \mathbf{w}^T \boldsymbol{\varphi}(\mathbf{x}) + b, \quad (23)$$

where  $\mathbf{w} = [w_1, \dots, w_K]^T$  is the weight vector,  $b$  the bias, and  $\boldsymbol{\varphi}(\mathbf{x}) = [\varphi_1(\mathbf{x}), \dots, \varphi_K(\mathbf{x})]^T$  the basis function vector.

The solution of the so defined optimization problem is solved by the introduction of the Lagrangian function and the Lagrange multipliers  $\alpha_i, \alpha'_i$  ( $i = 1, 2, \dots, p$ ) responsible for the functional constraints defined by (1). The minimization of the Lagrangian function has been transformed to the so-called dual problem (Vapnik 1998; Platt 1998; Osowski and Garanty 2007; Yeganeh et al. 2012):

$$\max \left\{ \sum_{i=1}^p d_i (\alpha_i - \alpha'_i) - \varepsilon \sum_{i=1}^p (\alpha_i - \alpha'_i) - \frac{1}{2} \sum_{i=1}^p \sum_{j=1}^p (\alpha_i - \alpha'_i) (\alpha_j - \alpha'_j) K(\mathbf{x}_i, \mathbf{x}_j) \right\} \quad (24)$$

at the constraints

$$\sum_{i=1}^p (\alpha_i - \alpha'_i) = 0, \quad 0 \leq \alpha_i \leq C, \quad 0 \leq \alpha'_i \leq C, \quad (25)$$

where  $K(\mathbf{x}_i, \mathbf{x}_j) = \boldsymbol{\varphi}^T(\mathbf{x}_i) \boldsymbol{\varphi}(\mathbf{x}_j)$  is an inner-product kernel defined in accordance with the Mercer's theorem (Vapnik 1998) for the learning data set  $\mathbf{x}$ . After solving the dual problem, all weights are expressed through the  $N_{sv}$  nonzero Lagrange multipliers  $\alpha_i, \alpha'_i$  and the same number of learning vectors  $\mathbf{x}_i$  associated with them. The network output signal  $y(\mathbf{x})$  can be then expressed in the form (Vapnik 1998; Osowski and Garanty 2007; Yeganeh et al. 2012):

$$y(\mathbf{x}) = \sum_{i=1}^{N_{sv}} (\alpha_i - \alpha'_i) K(\mathbf{x}, \mathbf{x}_i) + b \quad (26)$$

The most known kernel functions used in practice are radial (Gaussian), polynomial, spline, or even sigmoidal functions (Vapnik 1998; Schölkopf and Smola 2002). The most important is the choice of coefficients  $\varepsilon$  and  $C$ . Constant  $\varepsilon$  determines the margin within which the error is neglected. The smaller its value the higher accuracy of learning is required, and more support vectors will be found by the algorithm. The regularization constant  $C$  is the weight, determining the balance between the

complexity of the network, characterized by the weight vector  $w$  and the error of approximation, measured by the slack variables and the value of  $\varepsilon$  (Osowski and Garanty 2007; Yeganeh et al. 2012). For the normalized input signals, the value of  $\varepsilon$  is usually adjusted in the range ( $10^{-3}$ – $10^{-2}$ ), and  $C$  is much bigger than 1 (Osowski and Garanty 2007).

---

## Implementation of Soft Computing Methods in Environmental Engineering

In this section, successful applications of soft computing-based prediction models (ANN, FL, ANFIS, SVM) in the field of environmental engineering are examined in terms of water/wastewater treatment and air pollution related problems, and the important findings obtained in these studies are summarized.

### ANN-Based Applications for Water and Wastewater Treatment

Yetilmezsoy et al. (2013) developed two three-layer ANN models to predict biogas and methane production rates in a pilot-scale mesophilic up-flow anaerobic sludge blanket (UASB) reactor treating molasses wastewater. A tangent sigmoid transfer function (*tansig*) at the hidden layer and a linear transfer function (*purelin*) at the output layer were conducted for the proposed ANN models. After backpropagation training combined with principal component analysis (PCA), the scaled conjugate gradient algorithm (*trainscg*) was found as the best of the other training algorithms. Computational results demonstrated that compared to the conventional multiple regression-based methodology, the proposed ANN-based models produced smaller deviations and exhibited superior predictive accuracy with satisfactory determination coefficients of about 0.935 and 0.924, respectively, for the forecasts of biogas and methane production rates.

In a recent study, Podder and Majumder (2016) proposed a three-layer feed-forward back propagation (BP) ANN (4:5:1) with Levenberg–Marquardt (LM) training algorithm for predicting the phycoremediation efficiency of both As(III) and As(V) ions from wastewater using *Botryococcus braunii*. The study concluded that the proposed ANN architecture exhibited good agreements with the actual experimental and predicted values of both As(III) and As(V) and could describe the behavior of the complex reaction system with very high determination coefficient ( $R^2 = 0.99977$  and  $0.9998$  for As(III) and As(V), respectively) under different conditions.

More recently, Antwi et al. (2017) developed three-layered feedforward backpropagation (BP) ANN and multiple nonlinear regression (MnLR) models were to estimate biogas and methane yield in an upflow anaerobic sludge blanket (UASB) reactor treating potato starch processing wastewater. In the study, Quasi-Newton method and conjugate gradient backpropagation (BP) algorithms were found as the best among other training algorithms. The authors have reported that

compared with the MnLR model, BP-ANN model demonstrated significant performance, suggesting possible control of the anaerobic digestion process with the BP-ANN model.

In another recent work, Ghaedi and Vafaei (2017) reviewed important research studies of ANN on dyes adsorption from aqueous solution. The study concluded that ANN approaches could be successfully applied for the modeling and forecasting of dye adsorption process with acceptable accuracy compared to conventional linear models such as multiple linear regression (MLR) and PLS. In particular, the hybrid networks with optimization approaches were found to be more efficient to the performance of dye adsorption.

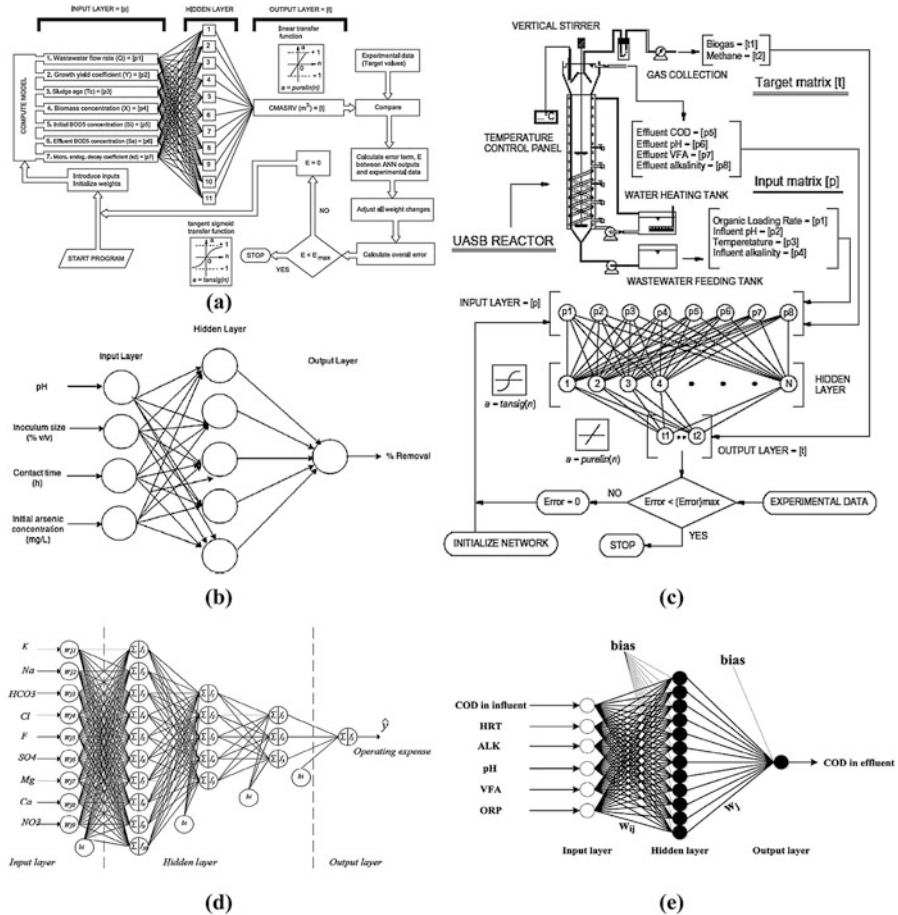
Furthermore, Qaderi and Babanezhad (2017) attempted to employ a feed-forward ANN-based model with four hidden layer and nine independent variables (i.e., concentrations of ions K, Na, Mg, Ca, Sr, Ba, CO<sub>3</sub>, HCO<sub>3</sub>, NO<sub>3</sub>, Cl, and SO<sub>4</sub>) to predict the costs of water treatment through reverse osmosis process for supplying drinking water from the available water resources. The results concerning the ANN indicated that the proposed predictive model performed desirably for estimating the costs of treating the groundwater in the region with the accuracy of approximately 98%, where the root mean square error (RMSE) percentage was 2.02% indicating an acceptable error level for the ANN model.

Finally, Hu et al. (2017) developed a three-layer backpropagation BP-ANN model to predict the chemical oxygen demand (COD) removal performance of an expanded granular sludge bed (EGSB) reactor. Activation function of hidden layer and output layer were “*tansig*” and “*purelin*” individually. Several comparisons were conducted to obtain an optimal network structure. *Divderand* function was chosen to divide the operating data into training group, testing group, and validation group. The Levenberg–Marquardt algorithm (*trainlm*) was found as the best of the tested training algorithms. The result indicated that the proposed ANN model exhibited high forecast accuracy ( $R^2 = 0.8156$ ) for the forecast of COD removal performance by EGSB system.

Apart from the above-mentioned studies, several other successful ANN modeling studies (Oliveira-Esquerre et al. 2002; Molga et al. 2006; Sahinkaya et al. 2007; Daneshvar et al. 2006; Rangasamy et al. 2007; Raduly et al. 2007; Ozkaya et al. 2007, 2008; Yetilmezsoy and Demirel 2008; Yetilmezsoy and Sapci-Zengin 2009; Yetilmezsoy 2012; Sahinkaya 2009; Pendashteh et al. 2011) have been conducted previously in various parts of the field of wastewater engineering (Fig. 5).

## **ANN-Based Applications for Air Quality/Pollution Control/Forecasting**

In the past years, it has become apparent that ANN-based prediction models have been effectively conducted on a substantial number of research activities in the field of air pollution engineering. In these investigations, several authors have developed different types of ANN models, and the results have been compared with the forecasts obtained using multiple regression models. For instance, Nunnari et al. (2004)



**Fig. 5** Various topological architectures of ANN models proposed for water and wastewater treatment (Adapted from (a) Yetilmezsoy 2010; (b) Podder and Majumder 2016; (c) Yetilmezsoy et al. 2013; (d) Qaderi and Babanezhad 2017; and (e) Hu et al. 2017)

modeled SO<sub>2</sub> concentration at a point by intercomparing several stochastic techniques such as ANN, fuzzy logic, and generalized additive techniques. Because the ANN models worked better in the prediction of critical episodes, they recommended the ANN approach for the implementation of a warning system for air quality control.

Yetilmezsoy (2006) proposed an ANN model and a new empirical model to determine optimum body diameter (OBD) of air cyclones for 505 different artificial scenarios given in a wide range of five operating variables, namely, gas flow rate, particle density, temperature, and two design parameters, namely,  $K_a$  and  $K_b$ , selected in the cyclone design. The study concluded that maximum diameter deviations from the well-known Kalen and Zenz’s model were recorded as 1.3 cm and 0.0022 cm for the empirical model and ANN outputs, respectively. Although both approaches

produced promising results, the ANN model exhibited speed and practicality, as well as a more robust and superior performance in the prediction of OBD values.

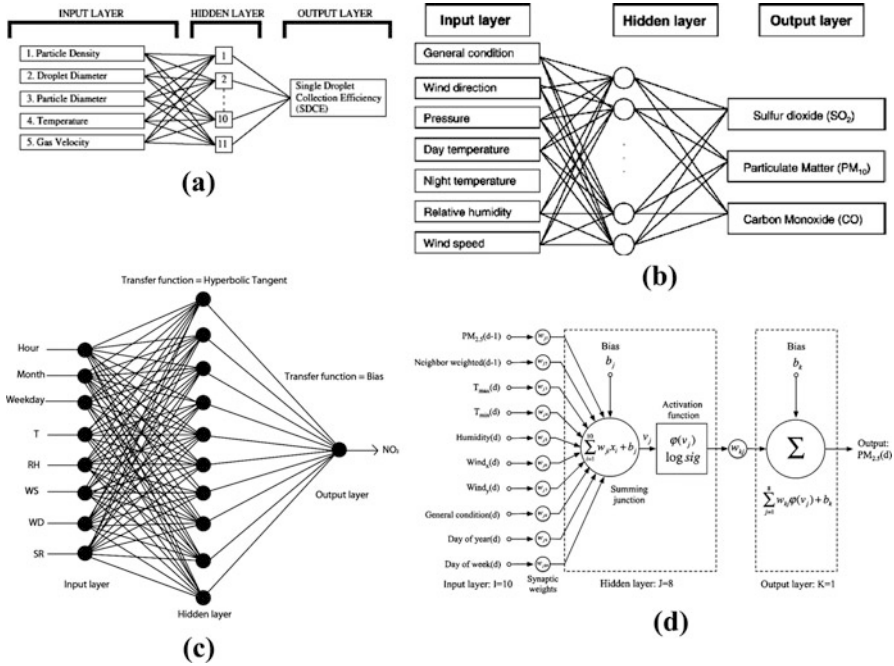
Agirre-Basurko et al. (2006) developed two multilayer perceptron (MLP)-based models and one multiple linear regression-based model to forecast ozone ( $O_3$ ) and nitrogen dioxide ( $NO_2$ ) levels in Bilbao, Spain. In their study, traffic variables were used as predictor variables in the developed models. Results indicated the MLP-based models showed remarkably better performance than the multiple linear regression model in predicting pollutant concentrations.

In another study (Yetilmezsoy and Saral 2007), an ANN-based approach and nonlinear regression analysis were performed for the determination of single droplet collection efficiency (SDCE) of countercurrent spray towers. The authors reported that predicted results obtained from the nonlinear regression analysis and the ANN model were in agreement with the theoretical data, and that all predictions proved to be satisfactory with a correlation coefficient of approximately 0.921 and 0.99, respectively. The study concluded that the development of a new mathematical model and the creation of an ANN-based model for the prediction of SDCE of countercurrent spray towers eliminated complex interactions of variables and difficult iterative calculations typically performed in the theoretical approach.

Finally, there have also been other studies (Wieland et al. 2002; Wotawa and Wotawa 2001; Abdul-Wahab and Al-Alawi 2002; Iliadis et al. 2007; Al-Alawi et al. 2008; Ozdemir et al. 2008) on the prediction of tropospheric and surface  $O_3$  concentrations reporting the advantages and adaptability properties of ANN-based models. Moreover, the use of ANN allows the prediction of daily and/or hourly particulate matter ( $PM_{2.5}$  and  $PM_{10}$ ) emissions (Chaloulakou et al. 2003; Chelani 2005; Grivas and Chaloulakou 2006; Kurt et al. 2008; Feng et al. 2015; Vakili et al. 2015; Bai et al. 2016; Biancofiore et al. 2017; Park et al. 2018) in many urban and residential areas. ANN-based models have also been used in the prediction of urban and ground-level  $SO_2$  concentrations, demonstrating successful results when considering the complex and nonlinear structure of the atmosphere (Akkoyunlu et al. 2010; Saral and Erturk 2003; Sofuoglu et al. 2006; Bai et al. 2016). Furthermore, ANN-based models have given reliable forecasts of carbon monoxide (CO) and nitrogen dioxide ( $NO_2$ ) concentrations in other studies (Kurt et al. 2008; Elangasinghe et al. 2014; Bai et al. 2016) (Fig. 6).

## FL-Based Applications for Water and Wastewater Treatment

Murnleitner et al. (2002) modelled and controlled two-stage anaerobic wastewater pretreatment using a Mamdani-type FL expert system. Hydrogen concentration together with methane concentration, gas production rate, pH, and the filling level of the acidification buffer tank were used as input variables for the FL system. With the use of the proposed FL system, very strong fluctuations in the concentration of the substrate and the volumetric loading rate could be successfully handled, and heavy overload could be avoided by taking proper control actions automatically.



**Fig. 6** Different architectures of MLP type ANN models proposed for air quality/pollution control/ forecasting (Adapted from (a) Yetilmezsoy and Saral 2007; (b) Kurt et al. 2008; (c) Elangasinghe et al. 2014; and (d) Feng et al. 2015)

In another anaerobic study, Turkdogan-Aydinol and Yetilmezsoy (2010) developed a FL-based model to predict biogas and methane production rates in a pilot-scale 90-L mesophilic up-flow anaerobic sludge blanket (UASB) reactor treating molasses wastewater. In the study, trapezoidal membership functions with eight levels were conducted for the fuzzy subsets, and a Mamdani-type fuzzy inference system was used to implement a total of 134 rules in the *if-then* format. The authors concluded that compared to nonlinear regression models, the proposed FL-based model produced smaller deviations and exhibited a superior predictive performance on forecasting of both biogas and methane production rates with satisfactory determination coefficients over 0.98.

Yetilmezsoy (2012) proposed a multiple inputs and multiple outputs (MIMO) FL-based model was proposed to estimate color and chemical oxygen demand (COD) removal efficiencies in the posttreatment of anaerobically pretreated poultry manure wastewater (PMW) effluent using Fenton’s oxidation process. The author used trapezoidal membership functions with eight levels that were conducted for the fuzzy subsets, and a Mamdani-type fuzzy inference system to implement a total of 70 rules in the *if-then* format. The product (*prod*) and the center of gravity (*centroid*) methods were applied as the inference operator and defuzzification methods, respectively. The results of the study demonstrated that a highly dynamic process, such as



Fenton's oxidation of anaerobically pretreated PMW effluent, could be successfully ( $R^2 = 0.99$  for both color and COD removals) and cost-effectively (CPU usage = 3–4% when simulating the model) modeled using FL methodology, compared to the classical regression-based method ( $R^2 = 0.772$  and  $0.861$  for color and COD removals, respectively, and CPU usage = 5–6%).

Furthermore, there have also been other studies on modeling water-in-oil emulsion formation (Yetilmezsoy et al. 2012), biological oxygen demand (BOD) removal prediction in free-water surface constructed wetlands (Kotti et al. 2013), and modeling of an integrated process for predictions of COD, total organic carbon (TOC), color, and ammonia nitrogen ( $\text{NH}_3\text{-N}$ ) removal efficiencies in the treatment of landfill leachates (young, middle-aged, and stabilized) (Sari et al. 2013) reporting robustness and cost-effectiveness of FL-based modeling tools.

## FL-Based Applications for Air Quality/Pollution Control/Forecasting

Yetilmezsoy and Abdul-Wahab (2012) proposed a prognostic approach that is based on a FL model to estimate suspended dust concentrations ( $\text{PM}_{10}$ ) in a specific residential area in Kuwait with high traffic and industrial influences. The authors employed trapezoidal membership functions with 10 and 15 levels employed for the fuzzy subsets of each model variable. A Mamdani-type fuzzy inference system (FIS) was developed to introduce a total of 146 rules in the *if-then* format. The product (*prod*) and the center of gravity (*centroid*) methods were performed as the inference operator and defuzzification methods, respectively, for the proposed FIS. The study concluded that the proposed FL model produced very small deviations from the actual results, and showed better predictive performance than an multiple regression-based exponential model with regard to forecasting  $\text{PM}_{10}$  levels, with a very high determination coefficient of over 0.99.

In a recent study, Olvera-García et al. (2016) described a new evaluation model using weighted fuzzy inference systems combined with an Analytic Hierarchy Process (AHP), providing a new air quality index (AQI) for Mexico City and its Metropolitan area. The authors evaluated six key pollutants (ozone ( $\text{O}_3$ ), sulfur dioxide ( $\text{SO}_2$ ), nitrogen dioxide ( $\text{NO}_2$ ), carbon monoxide (CO), particulate matter smaller than 10 and  $2.5 \mu\text{m}$  ( $\text{PM}_{10}$  and  $\text{PM}_{2.5}$ )) as environmental parameters according to toxicological levels, and assessed different air quality situations using a fuzzy reasoning process. They employed five score stages, such as excellent, good, regular, bad, and dangerous, in order to define a set of 174 inference rules in the *if-then* format for the proposed FIS. The results showed that a good performance of the proposed AQI against those in literature depending on the assignment of weights according to an importance level for each environmental parameter using a priority analysis based on the AHP procedure.

Additionally, some other FL-based studies on classification of air quality in Tehran, Iran (Sowlat et al. 2011), assessment and prediction of air quality in Mexico City and its Metropolitan area (Carbajal-Hernández et al. 2012), and modeling the

indoor air quality (IAQ) of the underground trains in Athens, Greece (Assimakopoulos et al. 2013), can be found in the literature.

## **ANFIS-Based Applications for Water and Wastewater Treatment**

In addition to ANN modeling studies, several ANFIS-based models have been proposed to evaluate and optimize various water and wastewater treatment processes. For instance, autoregressive integrated moving average (ARIMA) and Takagi-Sugeno (TS) fuzzy methods were used by Altunkaynak et al. (2005) for predicting future monthly water consumption values from three antecedent water consumption amounts, considered as independent variables. The TS fuzzy predicted results better than the ARIMA.

Civelekoglu et al. (2007) employed ANFIS-based models for the prediction of carbon and nitrogen removal in the aerobic biological treatment stage of a full-scale WWTP treating process wastewaters from the sugar production industry. In the study, a total of six independent ANFIS models were developed with or without PCA using the correlations among the influent and effluent data from the plant. With the use of PCA, results showed that the ANFIS modeling approach could be an effective advanced technique for performance prediction and control of treatment processes.

An ANFIS-based model was used by Firat and Gungor (2007) to estimate the flow of River Great Menderes, located west of Turkey. As a result, they discovered that ANFIS could be successfully applied for river flow estimation, providing high accuracy and reliability. Firat et al. (2009) compared two types of FIS for predicting municipal water consumption time series. Their results demonstrated that the ANFIS model is superior to Mamdani fuzzy inference systems (MFIS).

Cakmakci (2007) used an ANFIS-based technique for modeling of anaerobic digestion system of primary sludge of the Kayseri WWTP, Turkey. In the study, effluent volatile solid (VS) and methane yield were predicted by the ANFIS model using the routinely measured parameters in the anaerobic digester. The study concluded that due to highly nonlinear structure of the ANFIS model, a highly complex system such as anaerobic digestion process could be easily modeled. Filter head loss was also estimated by Cakmakci et al. (2008) using this ANFIS model. In their study, rule base sets were generated with subtractive clustering and grid partition. They determined that using a grid partition for modeling was superior to that of subtractive clustering. The correlation coefficients were greater than 0.99 in both tap and deionized water. Furthermore, filter iron removal rate was also modeled by Cakmakci et al. (2010). The best results for tap and deionized water were obtained with grid partition and subtractive clustering. The index of agreement (IA) values for tap water and deionized water were calculated as 0.996 and 0.971, and  $R^2$  values were determined as 0.99 and 0.89, respectively. The study concluded that neuro-fuzzy modeling could be successfully used to predict effluent iron concentration in sand filtration.

In another study, for a real-scale anaerobic WWTP operating under unsteady state conditions, Perendeci et al. (2008) proposed a conceptual ANFIS-based using

available on-line and off-line operational input variables to estimate the effluent COD. The study concluded that the developed ANFIS model with phase vector and history extension successfully represented the behavior of the considered treatment system.

A principal component analysis-adaptive neuro-fuzzy inference systems (PCA-ANFISs) method was used by Goodarzi et al. (2009) for the analysis of ternary mixtures of Al(III), Co(II), and Ni(II) over the range of 0.05–0.90, 0.05–4.05, and 0.05–0.95 g/mL, respectively. As a result, the method accurately and simultaneously determined the content of metal ions in several synthetic mixtures.

Finally, there have also been other computational studies (Tay and Zhang 2000; Wu and Lo 2008; Mingzhi et al. 2009; Pai et al. 2009; Erdirencelebi and Yalpir 2011; Mullai et al. 2011; Yetilmezsoy et al. 2011b; Wan et al. 2011; Pai et al. 2011; Mandal et al. 2015; Yetilmezsoy et al. 2015; Rahimzadeh et al. 2016) in the literature for modeling of various environmental problems based on water and wastewater treatment using ANFIS methodology.

### **ANFIS-Based Applications for Air Quality/Pollution Control/Forecasting**

Several adaptive neuro-fuzzy techniques emerging from the fusion of ANN and FIS have successfully found application in various areas of air pollution control. For instance, Yildirim and Bayramoglu (2006) used an adaptive neuro-fuzzy logic method to estimate the impact of meteorological factors on SO<sub>2</sub> and total suspended particulate matter (TSP) pollution levels over the city of Zonguldak, Turkey. The study concluded that the proposed ANFIS model satisfactorily forecasts the trends in SO<sub>2</sub> and TSP concentration levels, with performance levels between 75–90% and 69–80%, respectively.

An artificial intelligence-based modeling approach was conducted in another study by Noori et al. (2010) to predict daily carbon monoxide (CO) concentration in the atmosphere of Tehran, Iran, by means of developed ANN and ANFIS models. In the study, forward selection (FS) and gamma test (GT) methods were implemented for selecting input variables and developing hybrid models with ANN and ANFIS. The authors concluded that FS-ANN and FS-ANFIS models were the best models, considering  $R^2$ , mean absolute error, and developed discrepancy ratio statistics, for predicting pollution episodes.

Apart from the foregoing studies, several researchers (Shahraiyni et al. 2015; Ausati and Amanollahi 2016; Mishra and Goyal 2016; Prasad et al. 2016; Taylan 2017; Xie et al. 2017) have been successfully used ANFIS-based models in air quality/pollution control/forecasting.

### **SVM-Based Applications for Water and Wastewater Treatment**

Singh et al. (2011) used support vector classification (SVC) and support vector regression (SVR) models for (1) classification of the sampling sites with a view to

identify similar ones in the monitoring network for reducing their number for the future water quality monitoring; (2) classification of the sampling months into the groups of seasons for reducing the annual sampling frequency; and (3) to predict the biochemical oxygen demand (BOD) of the river water using simple measurable water quality variables. They studied with the data set comprised of 1500 water samples representing 10 different sites monitored for 15 years. The study concluded that The SVC model achieved a data reduction of 92.5% for redesigning the future monitoring program, and the SVR model provided a tool for the prediction of the water BOD using set of a few measurable variables.

Garcia Nieto et al. (2013) proposed a hybrid approach based on support vector regression (SVR) in combination with genetic algorithms (GA), namely, genetic algorithm support vector regression (GA-SVR) model, in forecasting the cyanotoxins presence in the Trasona reservoir, Northern Spain. The authors reported that a correlation coefficient equal to 0.98 was obtained when the hybrid GA-SVR technique was applied to the experimental data set, and the predicted results for the model demonstrated to be consistent with the history of observed actual cyanobacteria blooms from 2006 to 2011.

In another study, Liu et al. (2013) described a hybrid approach, known as real-value genetic algorithm support vector regression (RGA-SVR), to forecast aquaculture water quality in a high-density river crab culture situation. The authors concluded that the RGA-SVR forecasting method could help avoid economic losses caused by water quality problems to a certain extent. On the other hand, they reported that different types and rates of crossover and mutation should be set for different problems, since the operation of the genetic algorithm was difficult in the training process of the RGA-SVR model.

Furthermore, there have also been other recent studies on prediction of effluent concentration in a wastewater treatment plant in Ulsan Metropolitan city, Korea (Guo et al. 2015), prediction of Cd(II) removal by biosorption in Iasi city, Romania (Hlihor et al. 2015), numerical modeling for algal blooms of freshwater in in Macau Main Storage Reservoir located at south of China (Lou et al. 2017), prediction of five-day biochemical oxygen demand (BOD<sub>5</sub>) parameter in the Sefidrood River basin, Iran (Noori et al. 2015), lake management to prevent eutrophication in in Chaohu Lake located in southeast China (Xu et al. 2015), predicting the sorption capacity of lead (II) ions in India (Parveen et al. 2016), and eutrophication (enrichment of a water body with nutrients) classification in Dez reservoir located in Iran (Bashiri et al. 2017) reporting the advantages and generalization ability of the SVM method based over multilayer perceptron (MLP) or neuro-fuzzy network.

## **SVM-Based Applications for Air Quality/Pollution Control/Forecasting**

It has been reported that air quality is essential to people's health and the environment, and accurate forecasting of the concentration of air pollutants is

crucial to the effective monitoring of air quality (Lin et al. 2011). From this point of view, the accurate models for air pollutant prediction are needed because such models would allow forecasting and diagnosing potential compliance or noncompliance in both short- and long-term aspects (Lu and Wang 2005). In recent years, based on the emission and meteorological data collected from air-monitoring stations in different parts of the world, SVM paradigm has become popular and gained importance in forecasting problems related to air quality (Yeganeh et al. 2012).

Lu and Wang (2005) examined the feasibility of applying SVM to predict air pollutant levels in advancing time series based on the monitored air pollutant database in Hong Kong downtown area. The experimental comparisons between the SVM model and the classical radial basis function (RBF) network demonstrated that the SVM was superior to the conventional RBF network in predicting air quality parameters with different time series and of better generalization performance than the RBF model. The study concluded that SVM model provided a promising alternative and advantage in time series forecast and offered several advantages (i.e., it contains fewer free parameters (or small number of learning data), and eliminates the typical drawbacks, such as over-fitting training and local minima, of conventional neural network) over the conventional feed-forward RBF neural networks.

Oowski and Garanty (2007) used SVM and wavelet decomposition for daily air pollution forecasting based on the observed data of  $\text{NO}_2$ , CO,  $\text{SO}_2$ , and dust in the northern region of Poland. The authors decomposed the measured time series data into wavelet representation and predicted the wavelet coefficients in order to obtain the acceptable accuracy of predictions. The study concluded that application of SVM instead of classical MLP had enabled to obtain much better accuracy of forecast of the wavelet coefficients and the whole pollutant concentration at all stations.

Lin et al. (2011) proposed a support vector regression with logarithm pre-processing procedure and immune algorithms (SVRLIA) model to forecast concentrations of air pollutants, namely, particulate matter ( $\text{PM}_{10}$ ), nitrogen oxide, ( $\text{NO}_x$ ), and nitrogen dioxide ( $\text{NO}_2$ ), in Taiwan. Experimental results of the study indicated that the proposed SVRLIA model provided more accurate forecasting results than the other models such as general regression neural networks (GRNN), seasonal autoregressive integrated moving average model (SARIMA), and backpropagation neural networks (BPNN).

Yeganeh et al. (2012) conducted studies on an innovative method of daily air pollution prediction using combination of SVM as predictor and Partial Least Square (PLS) as a data selection tool in the forecasting of CO concentrations. The authors aimed to examine the feasibility of applying SVM and hybrid PLS-SVM models to predict air pollutant levels in short- and long-term periods based on the measured air pollutant database in Tehran. The study concluded that the proposed hybrid PLS-SVM model required lower computational time than SVM model and had better performance (more accurate and faster prediction ability) to predict air pollution in different time intervals.

In Rio de Janeiro City (Brazil), Luna et al. (2014) analyzed the behavior of the variables (nitrogen dioxide (NO<sub>2</sub>), nitrogen monoxide (NO), nitrogen oxides (NO<sub>x</sub>), carbon monoxide (CO), ozone (O<sub>3</sub>), scalar wind speed, global solar radiation, temperature, and moisture content in the air), using the method of PCA for exploratory data analysis, and proposed forecasts of O<sub>3</sub> levels from primary pollutants and meteorological factors, using nonlinear regression methods like ANN and SVM, from primary pollutants and meteorological factors. The study concluded that the models' predictions and the actual observations were consistent, and PCA-ANN-SVM demonstrated their robustness as useful tools for modeling and analysis of O<sub>3</sub> concentrations in tropospheric levels.

In recent study, Moazami et al. (2016) proposed a modeling approach to analyze the uncertainty of support vector regression (SVR) and FS-SVR models for the prediction of the next day CO concentration in Tehran metropolitan. They compared the results of the present study and another research on uncertainty determination of ANFIS and ANN. The results showed that the SVR had less uncertainty in CO prediction than the ANN and ANFIS models. On the other hand, they reported that the running time for uncertainty determination of SVR and FS-SVR models were more than one day, and high computational time was one of the most limitations of the implemented methodology. For this reason, the authors suggested using the faster optimization techniques for tuning the SVR parameters and applying the stop training algorithm instead of cross validation technique in order to reduce the running time for uncertainty determination of SVR model.

---

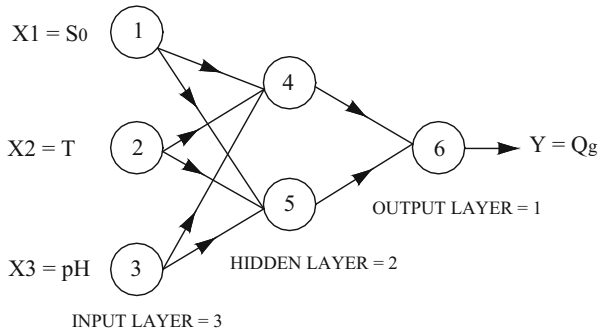
## Illustrative Soft Computing Examples for Environmental Engineers

In this section, some special illustrative soft computing examples on ANN and FL modeling and the respective MATLAB<sup>®</sup>-based solutions are presented for environmental engineers.

### Example 1

A three-layer feed-forward back propagated (FFBP) artificial neural network (ANN) model is proposed to predict the daily biogas production from a laboratory-scale anaerobic sludge bed reactor (ASBR) (Fig. 7). The input variables of the proposed ANN model are selected as follows: Total chemical oxygen demand (TCOD =  $X_1 = S_0$ : kg/m<sup>3</sup>), daily operating temperature ( $X_2 = T$ : °C), and pH of the feeding slurry ( $X_3 = \text{pH}$ ). In the model structure, logarithmic sigmoid function (*logsig*) is used for both hidden layer and the output layer as an activation and transfer function, respectively. The learning rate is selected as  $\eta = 0.90$ . The model variables ( $X_1$ ,  $X_2$ ,  $X_3$ , and  $Y$ ) will be normalized for the scale factors of  $a = 0.60$  and  $b = 0.20$  by

**Fig. 7** A three-layer feed-forward back propagated (FFBP) artificial neural network (ANN) proposed to predict the daily biogas production from a laboratory-scale anaerobic sludge bed reactor (ASBR)



**Table 1** Ranges of model variables

Model variables	Symbols and units	Ranges
Total chemical oxygen demand	$TCOD = X_1 = S_0$ (kg/m <sup>3</sup> )	11.2–19.6
Daily operating temperature	$X_2 = T$ (°C)	26–37
pH of the feeding slurry	$X_3 = pH$	6.2–8.5
Daily biogas production	$Y = Q_g$ (L/day)	8.4–14.6

**Table 2** Initial values of weights ( $w$ ) and bias ( $\theta$ ) terms for the first iteration

Weights ( $w_{ij}$ and $w_{jk}$ )			Bias terms ( $\theta$ )
$w_{14} = + 0.20$	$w_{15} = -0.30$	$w_{46} = -0.30$	$\theta_4 = -0.40$
$w_{24} = + 0.40$	$w_{25} = + 0.10$	$w_{56} = -0.20$	$\theta_5 = + 0.20$
$w_{34} = -0.50$	$w_{35} = + 0.20$		$\theta_6 = + 0.10$

using the min-max rule based on the following formulation:  $(O_i = a \frac{X_i - X_{min}}{X_{max} - X_{min}} + b)$ .

The ranges of model variables are given in Table 1. For the first iteration, the initial values of weights ( $w_{ij}$  and  $w_{jk}$ ) and bias terms ( $\theta_j$  and  $\theta_k$ ) are given in Table 2.

Based on the above-noted facts, write a MATLAB® script to determine the following questions:

- (a) At the steady-state conditions, an experimental data is given as follows: The daily biogas production is measured as  $Q_g = 13.2$  L/day for the values of  $S_0 = 14.8$  kg TCOD/m<sup>3</sup>,  $T = 32$  °C, and  $pH = 7.1$ . According to this data, evaluate the performance of the proposed ANN model in prediction of the daily biogas production by denormalizing the predicted value.
- (b) Update the initial values of weights and bias terms by performing a back-propagation (BP) process. After the BP operation, perform a new feed-forward process to observe the recovery in prediction of the daily biogas production (Table 3).

## Solution of Example 1

```

MATLAB® script

%Artificial Neural Network Model
%Programmed by Assoc.Prof.Dr. Kaan Yetilmezsoy
clear, clc

%Scale factors
a = 0.60; b = 0.20;

%Learning rate
L = 0.90;

%Ranges of variables
X1min = 11.2; X1max = 19.6; X2min = 26; X2max = 37;
X3min = 6.2; X3max = 8.5; Ymin = 8.4; Ymax = 14.6;

%Experimental data
X1 = 14.8; X2 = 32; X3 = 7.1; Y = 13.2

%Normalization of input variables
O1 = a*((X1-X1min)/(X1max-X1min))+b;
O2 = a*((X2-X2min)/(X2max-X2min))+b;
O3 = a*((X3-X3min)/(X3max-X3min))+b;
Yn = a*((Y-Ymin)/(Ymax-Ymin))+b

%Weight values
w14 = 0.20; w24 = 0.40; w34 = -0.50; w15 = -0.30;
w25 = 0.10; w35 = 0.20; w46 = -0.30; w56 = -0.20;

%Bias values
t4 = -0.40; t5 = 0.20; t6 = 0.10;

%ITERATION #1

%Computation of hidden neurons 4 and 5
I4 = (O1*w14 + O2*w24 + O3*w34) + t4;
I5 = (O1*w15 + O2*w25 + O3*w35) + t5;
%Computation of activation functions
O4 = 1/(1+exp(-I4)); O5 = 1/(1+exp(-I5));
%Computation of hidden neuron 6
I6 = (O4*w46 + O5*w56) + t6;
%Computation of transfer function
O6 = 1/(1+exp(-I6))
%Denormalization of predicted value
Yp1 = (((O6-b)/a)*(Ymax-Ymin))+Ymin

%Error at node 6
E6 = O6*(1-O6)*(Yn-O6);
E5 = O5*(1-O5)*E6*w56;
E4 = O4*(1-O4)*E6*w46;

%Update of weights
w14_1 = w14 + L*E4*O1; w24_1 = w24 + L*E4*O2;
w34_1 = w34 + L*E4*O3; w15_1 = w15 + L*E5*O1;
w25_1 = w25 + L*E5*O2; w35_1 = w35 + L*E5*O3;
w46_1 = w46 + L*E6*O4; w56_1 = w56 + L*E6*O5;

```



```

%Update of bias terms
t4_1 = t4 + L*E4;  t5_1 = t5 + L*E5;  t6_1 = t6 + L*E6;

%ITERATION #2

%Computation of hidden neurons 4 and 5
I4_1 = (O1*w14_1 + O2*w24_1 + O3*w34_1) + t4_1;
I5_1 = (O1*w15_1 + O2*w25_1 + O3*w35_1) + t5_1;
%Computation of activation functions
O4_1 = 1/(1+exp(-I4_1));
O5_1 = 1/(1+exp(-I5_1));
%Computation of hidden neuron 6
I6_1 = (O4_1*w46_1 + O5_1*w56_1) + t6_1;
%Computation of transfer function
O6_1 = 1/(1+exp(-I6_1))
%Denormalization of predicted value
Yp2 = (((O6_1-b)/a)*(Ymax-Ymin))+Ymin

%Error at node 6
E6_1 = O6_1*(1-O6_1)*(Yn-O6_1);
E5_1 = O5_1*(1-O5_1)*E6_1*w56_1;
E4_1 = O4_1*(1-O4_1)*E6_1*w46_1;

%Update of weights
w14_2 = w14_1 + L*E4_1*O1;  w24_2 = w24_1 + L*E4_1*O2;
w34_2 = w34_1 + L*E4_1*O3;  w15_2 = w15_1 + L*E5_1*O1;
w25_2 = w25_1 + L*E5_1*O2;  w35_2 = w35_1 + L*E5_1*O3;
w46_2 = w46_1 + L*E6_1*O4_1;  w56_2 = w56_1 + L*E6_1*O5_1;

%Update of bias terms
t4_2 = t4_1 + L*E4_1;
t5_2 = t5_1 + L*E5_1;
t6_2 = t6_1 + L*E6_1;

%ITERATION #3 (with no updates for weights and bias terms)

%Computation of hidden neurons 4 and 5
I4_2 = (O1*w14_2 + O2*w24_2 + O3*w34_2) + t4_2;
I5_2 = (O1*w15_2 + O2*w25_2 + O3*w35_2) + t5_2;
%Computation of activation functions
O4_2 = 1/(1+exp(-I4_2));  O5_2 = 1/(1+exp(-I5_2));
%Computation of hidden neuron 6
I6_2 = (O4_2*w46_2 + O5_2*w56_2) + t6_2;
%Computation of transfer function
O6_2 = 1/(1+exp(-I6_2))
%Denormalization of predicted value
Yp3 = (((O6_2-b)/a)*(Ymax-Ymin))+Ymin
    
```

<b>Answer</b>			
Y = 13.2000	Yn = 0.6645	O6 = 0.4659	Yp1 = 11.1475
O6_1 = 0.4824	Yp2 = 11.3186	O6_2 = 0.4977	Yp3 = 11.4759

**Table 3** Mathematical expressions for the proposed ANN model

Explanation	Mathematical expression
Net value transferred from input layer to hidden layer (or net value transferred from hidden layer to output layer)	$I_j = \sum w_{ij}O_i + \theta_j$ $I_k = \sum w_{jk}O_j + \theta_k$
Output of <i>logsig</i> activation or transfer function	$O_j = 1/[1 + \exp(-I_j)]$ $O_k = 1/[1 + \exp(-I_k)]$
Error value at the output layer (only for <i>logsig</i> function)	$\delta_k = O_k(1 - O_k)$ $(T_k - O_k)$
Error value at the hidden layer (only for <i>logsig</i> function)	$\delta_j = O_j(1 - O_j) \sum \delta_k w_{jk}$
Updated weight terms ( $w_{ij}$ and $w_{jk}$ )	$w_{ij}^n = w_{ij} + \eta \delta_j O_i$ $w_{jk}^n = w_{jk} + \eta \delta_k O_j$
Updated bias terms ( $\theta_j$ and $\theta_k$ )	$\theta_j^n = \theta_j + \eta \delta_j$ $\theta_k^n = \theta_k + \eta \delta_k$

**Table 4** Ranges of model variables

Model variables	Symbols and units	Ranges
Organic loading rate	$X_1 = \text{OLR (kg COD/m}^3/\text{day)}$	3–12
Daily operating temperature	$X_2 = T (^{\circ}\text{C})$	10–40
Daily biogas production	$Y = Q_g (\text{L/day})$	20–100

## Example 2

A fuzzy logic (FL) model is introduced to estimate the daily biogas production obtained from an experimental study. The properties of the proposed FL model are summarized below.

- The proposed FL model is a MISO (Multiple Input Single Output) type model which is consisted of 2 inputs ( $X_1 = \text{OLR} = \text{Organic loading rate (kg COD/m}^3/\text{day)}$ ,  $X_2 = T = \text{Temperature (}^{\circ}\text{C)}$ ) and 1 output ( $Y = Q_g = \text{Daily biogas production (L/day)}$ ).
- Trapezoidal membership functions (*trapmf*) with two and three levels are used, respectively, for input and output variables, and the functions are categorized as LOW, MOD, and HIGH for processing of the fuzzy rules.
- The ranges of model variables are given in Table 4. The ranks of the membership functions for input and output variables considered in the fuzzy sets are presented in Table 5.

**Table 5** Ranks of the trapezoidal membership functions selected for the model variables

Model variables	Low	Mod	High
$X_1 = \text{OLR} = \text{organic loading rate (kg COD/m}^3/\text{day)}$	[-4 2 4 10]	-	[4 10 14 20]
$X_2 = T = \text{temperature (}^\circ\text{C)}$	[-15 5 15 35]	-	[15 35 45 65]
$Y = Q_g = \text{daily biogas production (L/day)}$	[-10 0 40 50]	[40 50 80 90]	[80 90 110 120]

**Table 6** Rule sets for the proposed FL-based model

$X_1 = \text{OLR} = \text{organic loading rate (kg COD/m}^3/\text{day)}$	$X_2 = T = \text{temperature (}^\circ\text{C)}$	$Y = Q_g = \text{daily biogas production (L/day)}$
Low	Low	Low
High	High	High
High	Low	Mod

**Table 7** Steady-state data obtained from experimental studies

#	OLR (kg COD/m <sup>3</sup> /day)	T (°C)	Q <sub>g</sub> (L/day)
1	5	16	41
2	9	32	87
3	8	17	60

- (d) The rule base is developed by taking into account the experimental results and the suggestions of the experts (Table 6). The weight factors are taken as equal (1) for each fuzzy rule.
- (e) The fuzzy inference system (FIS) is proposed as the Mamdani’s type, and the *prod* (product) method is used by the AND operator for each fuzzy rule in the FIS. The other methods implemented for implication, aggregation, and defuzzification processes are *prod*, *max*, and *centroid* (COG), respectively.
- (f) The steady-state data obtained from experimental studies are given in Table 7.

According to the foregoing points, write a MATLAB<sup>®</sup> script to estimate the outputs of the FL-based model for each experimental data given in Table 7 and to calculate the value of the determination coefficient ( $R^2$ ) associated with these predictions.

## Solution of Example 2

### MATLAB<sup>®</sup> script (development of the fuzzy inference system, FIS)

```
[System]
Name='fuzzy1'
Type='mamdani'
NumInputs=2
NumOutputs=1
NumRules=3
AndMethod='prod'
OrMethod='max'
ImpMethod='prod'
AggMethod='max'
DefuzzMethod='centroid'

[Input1]
Name='OLR'
Range=[3 12]
NumMFs=2
MF1='LOW': 'trapmf', [-4 2 4 10]
MF2='HIGH': 'trapmf', [4 10 14 20]

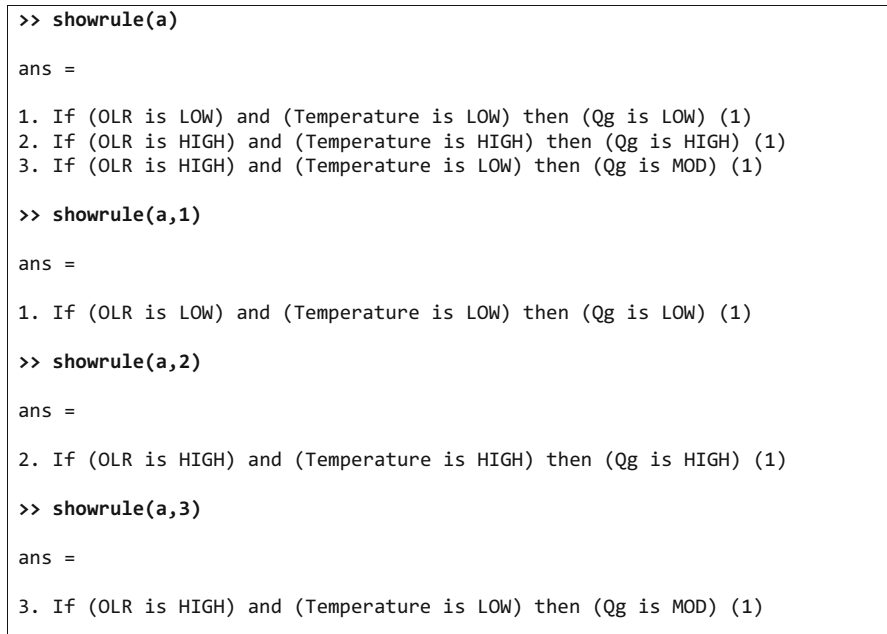
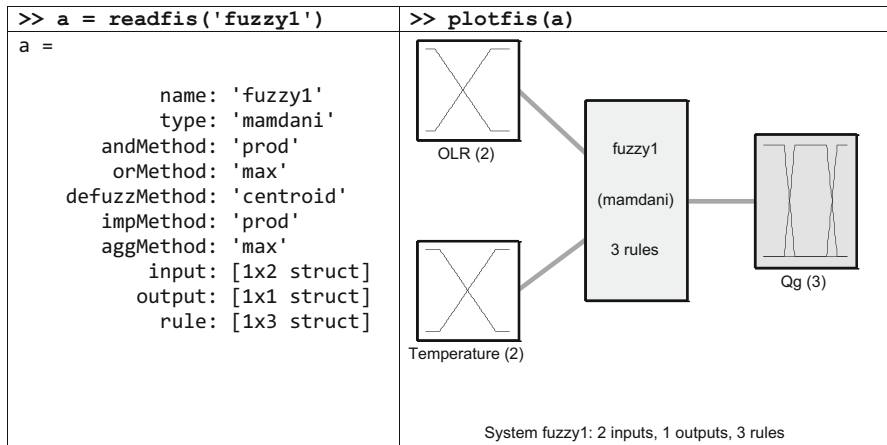
[Input2]
Name='Temperature'
Range=[10 40]
NumMFs=2
MF1='LOW': 'trapmf', [-15 5 15 35]
MF2='HIGH': 'trapmf', [15 35 45 65]

[Output1]
Name='Qg'
Range=[20 100]
NumMFs=3
MF1='LOW': 'trapmf', [-10 0 40 50]
MF2='MOD': 'trapmf', [40 50 80 90]
MF3='HIGH': 'trapmf', [80 90 110 120]

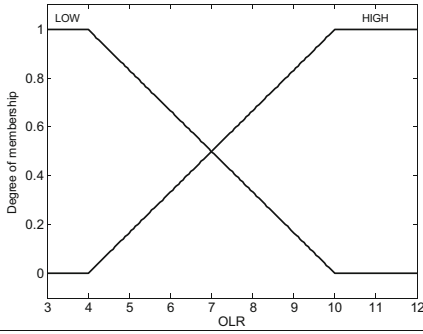
[Rules]
1 1, 1 (1) : 1
2 2, 3 (1) : 1
2 1, 2 (1) : 1
```

The numbers in the parentheses represent weights that can be applied to each rule if desired. You can specify the weights by typing in a desired number between zero and one. If you do not specify them, the weights are assumed to be unity (1).

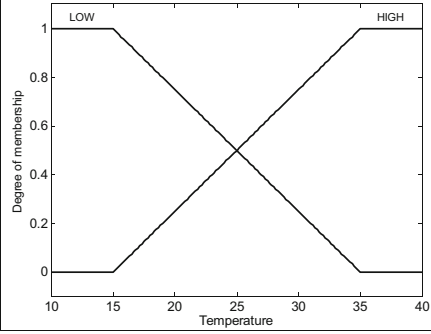
**Save the file as fuzzy1.fis by selecting File > Save (or use CTRL + S).**



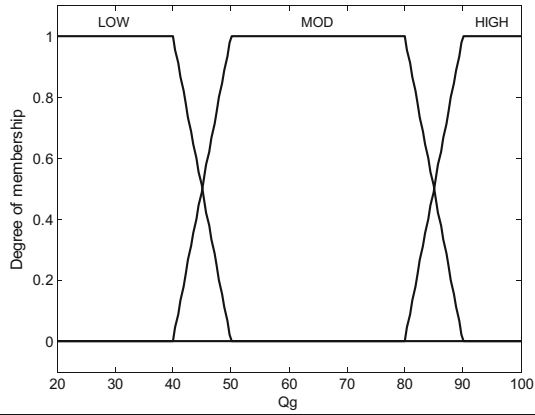
```
>> plotmf(a, 'input', 1)
```



```
>> plotmf(a, 'input', 2)
```



```
>> plotmf(a, 'output', 1)
```



**Show overall results (measured and predicted) in the form of a table.**

**Calculation of determination coefficient ( $R^2$ )**

**MATLAB® script**

```
clear, clc
a = readfis('fuzzy1');
data = [5 16; 9 32; 8 17];
predictions = evalfis(data, a);
measured = [41; 87; 60];

%Create a table using "measured" and "predictions" vectors.
%Since these vectors are in vertical format, use transpose of them.
%Add a semicolon operator between them, and insert this table
%in fprintf function.

table = [measured';predictions'];

%Location of the title words can be adjusted
%as desired by pressing the SPACE key on the keyboard.
%i.e., 4 Spaces for "Observed" word
%and 3 Spaces for "Predicted" word.

disp('   Observed   Predicted')
fprintf('%10.2f %11.2f \n',table)

O = measured;
P = predictions;
Om = mean(O);
Pm = mean(P);
n = numel(O);

%Determination of linear regression coefficient (b)
%and constant term (a)
%Microsoft Excel uses this formulation
b = (n*sum(O.*P)-sum(O)*sum(P))/((n*sum(O.^2))-((sum(O))^2));
a = (sum(P)-b*sum(O))/n;
%Determination coefficient (Rsqr)
Rsqr = ((b^2)*sum((O-Om).^2))/(sum((P-Pm).^2))

%Determination coefficient (Rsqr):
%Same result by using the compact form
Rsqr1 = ((sum((O-Om).*(P-Pm)))^2)/((sum((O-Om).^2))*(sum((P-Pm).^2)));
```

**Answer**

Observed	Predicted	Rsqr = 0.9994	$R^2 = \frac{\left( \sum_{i=1}^n (O_i - O_m)(P_i - P_m) \right)^2}{\sum_{i=1}^n (O_i - O_m)^2 \sum_{i=1}^n (P_i - P_m)^2}$
41.00	40.30		
87.00	82.02		
60.00	58.44		

## Conclusion

In this chapter, important applications of the soft computing-based prediction models, such as ANN, FL, ANFIS, and SVM, are specifically explored for the real-life problems of environmental engineering field. It is apparent from the literature that soft computing methods can be successfully implemented as complementary technologies in various applications of water/wastewater treatment and air quality/pollution control/forecasting. Modeling of environmental processes is very difficult, since they include biological, chemical, and physical phenomena, together. At this point, soft computing techniques serve as a modern paradigm for computing and simulating complex natural processes with basic principles of the prediction modeling using environmental data sets obtained from various real applications. Additionally, the applicability of these models is very simple, posing no need to identify nonlinear relationships between multiple variables and define the complex reactions in the environmental problems.

It is worth mentioning that many investigators have compared the performance of soft computing-based techniques with conventional methods. Based on the literature review, it can be concluded that soft computing-based models have provided better results compared to traditional linear/nonlinear regression methods due to their ability to precisely discriminate the arbitrary nonlinear functional relationship between input and output data sets. Furthermore, the literature findings clearly corroborate that the soft computing methodology could describe the behavior of the complex reaction system with the range of experimental conditions adopted. Simulation based on these models can estimate the behavior of the system under different conditions. To conclude, a simulation on the basis of the soft computing model can deliver further contribution in developing a better understanding of the dynamic behavior of the environmental processes where still some phenomena cannot be clarified in all details.

The encouraging results obtained from the application of the described soft computing-based approaches in modeling of water and air pollution-related problems indicate that these techniques are worth for further research and extension to other similar real-life problems from the environmental engineering field. Considering the predictive capability and robustness, of the soft computing-based methodology, these prognostic models may be integrated into full-scale water/wastewater treatment plants and mobile pollution air monitoring stations as advanced control, early warning, and decision support systems using different on-line and off-line control strategies in a cost-effective manner by means of energy and environment.

---

## References

- Abdul-Wahab SA, Al-Alawi SM (2002) Assessment and prediction of tropospheric ozone concentration levels using artificial neural networks. *Environ Model Softw* 17(3):219–228
- Acaroglu O, Ozdemir L, Asbury B (2008) A fuzzy logic model to predict specific energy requirement for TBM performance prediction. *Tunn Undergr Space Technol* 23(5):600–608
- Adriaenssens V, Goethals PL, De Pauw N (2006) Fuzzy knowledge-based models for prediction of *Asellus* and *Gammarus* in watercourses in Flanders (Belgium). *Ecol Model* 195(1):3–10



- Agirre-Basurko E, Ibarra-Berastegi G, Madariaga I (2006) Regression and multilayer perceptron-based models to forecast hourly  $O_3$  and  $NO_2$  levels in the Bilbao area. *Environ Model Softw* 21(4):430–446
- Akkoyunlu A, Yetilmmezsoy K, Erturk F, Oztemel E (2010) A neural network-based approach for the prediction of urban  $SO_2$  concentrations in the Istanbul metropolitan area. *Int J Environ Pollut* 40(4):301–321
- Akkurt S, Tayfur G, Can S (2004) Fuzzy logic model for the prediction of cement compressive strength. *Cem Concr Res* 34(8):1429–1433
- Al-Alawi SM, Abdul-Wahab SA, Bakheit CS (2008) Combining principal component regression and artificial neural networks for more accurate predictions of ground-level ozone. *Environ Model Softw* 23(4):396–403
- Al-Daoud E (2009) A comparison between three neural network models for classification problems. *J Artif Intell* 2(2):56–64
- Altunkaynak A, Özger M, Çakmakci M (2005) Water consumption prediction of Istanbul city by using fuzzy logic approach. *Water Resour Manag* 19(5):641–654
- Antwi P, Li J, Boadi PO, Meng J, Shi E, Deng K, Bondinuba FK (2017) Estimation of biogas and methane yields in an UASB treating potato starch processing wastewater with backpropagation artificial neural network. *Bioresour Technol* 228:106–115
- Assimakopoulos MN, Dounis A, Spanou A, Santamouris M (2013) Indoor air quality in a metropolitan area metro using fuzzy logic assessment system. *Sci Total Environ* 449:461–469
- Atmaca H, Cetisli B, Yavuz HS (2001) The comparison of fuzzy inference systems and neural network approaches with ANFIS method for fuel consumption data. In: Second international conference on electrical and electronics engineering papers ELECO'2001, Bursa, Turkey
- Ausati S, Amanollahi J (2016) Assessing the accuracy of ANFIS, EEMD-GRNN, PCR, and MLR models in predicting  $PM_{2.5}$ . *Atmos Environ* 142:465–474
- Bai Y, Li Y, Wang X, Xie J, Li C (2016) Air pollutants concentrations forecasting using back propagation neural network based on wavelet decomposition with meteorological conditions. *Atmos Pollut Res* 7(3):557–566
- Bashiri S, Akbarzadeh A, Zarrabi M, Yetilmmezsoy K, Fingas M, Moosakhaani M (2017) Using PCA combined SVM in the classification of eutrophication in Dez Reservoir (Iran). *Environ Eng Manag J*. in press
- Biancofiore F, Busilacchio M, Verdecchia M, Tomassetti B, Aruffo E, Bianco S, Tommaso SD, Colangeli C, Rosatelli G, Di Carlo P (2017) Recursive neural network model for analysis and forecast of  $PM_{10}$  and  $PM_{2.5}$ . *Atmos Pollut Res* 8:652–659
- Biyikoglu A, Akcayol MA, Özdemir V, Sivrioglu M (2005) Temperature prediction in a coal fired boiler with a fixed bed by fuzzy logic based on numerical solution. *Energy Convers Manag* 46(1):151–166
- Cai M, Yin Y, Xie M (2009) Prediction of hourly air pollutant concentrations near urban arterials using artificial neural network approach. *Transp Res Part D: Transp Environ* 14(1):32–41
- Çakmakci M (2007) Adaptive neuro-fuzzy modelling of anaerobic digestion of primary sedimentation sludge. *Bioprocess Biosyst Eng* 30(5):349–357
- Çakmakci M, Kınacı C (2008) Adaptive neuro-fuzzy modeling of head loss in iron removal with rapid sand filtration. *Water Environ Res* 80(12):2268–2275
- Çakmakci M, Kinaci C, Bayramoğlu M, Yildirim Y (2010) A modeling approach for iron concentration in sand filtration effluent using adaptive neuro-fuzzy model. *Expert Syst Appl* 37(2):1369–1373
- Carbajal-Hernández JJ, Sánchez-Fernández LP, Carrasco-Ochoa JA, Martínez-Trinidad JF (2012) Assessment and prediction of air quality using fuzzy logic and autoregressive models. *Atmos Environ* 60:37–50
- Chaloulakou A, Grivas G, Spyrellis N (2003) Neural network and multiple regression models for  $PM_{10}$  prediction in Athens: a comparative assessment. *J Air Waste Manage Assoc* 53(10):1183–1190
- Chelani AB (2005) Predicting chaotic time series of  $PM_{10}$  concentration using artificial neural network. *Int J Environ Stud* 62(2):181–191
- Civelekoglu G, Perendeci A, Yigit NO, Kitis M (2007) Modeling carbon and nitrogen removal in an industrial wastewater treatment plant using an adaptive network-based fuzzy inference system. *Clean-Soil, Air, Water* 35(6):617–625

- Daneshvar N, Khataee AR, Djafarzadeh N (2006) The use of artificial neural networks (ANN) for modeling of decolorization of textile dye solution containing CI Basic Yellow 28 by electro-coagulation process. *J Hazard Mater* 137(3):1788–1795
- Dwarakish GS, Nithyapriya B (2016) Application of soft computing techniques in coastal study – a review. *J Ocean Eng Sci* 1(4):247–255
- Elangasinghe MA, Singhal N, Dirks KN, Salmond JA (2014) Development of an ANN-based air pollution forecasting system with explicit knowledge through sensitivity analysis. *Atmos Pollut Res* 5(4):696–708
- Erdirencelebi D, Yalpir S (2011) Adaptive network fuzzy inference system modeling for the input selection and prediction of anaerobic digestion effluent quality. *Appl Math Model* 35(8):3821–3832
- Esmaelzadeh R, Dariane AB (2014) Long-term streamflow forecasting by adaptive Neuro-Fuzzy Inference System using K-fold cross-validation: (case study: Taleghan Basin, Iran). *J Water Sci Res* 6(1):71–83
- Feng X, Li Q, Zhu Y, Hou J, Jin L, Wang J (2015) Artificial neural networks forecasting of PM 2.5 pollution using air mass trajectory based geographic model and wavelet transformation. *Atmos Environ* 107:118–128
- Firat M, Gungor M (2007) River flow estimation using adaptive neuro fuzzy inference system. *Math Comput Simul* 75(3):87–96
- Firat M, Turan ME, Yurdusev MA (2009) Comparative analysis of fuzzy inference systems for water consumption time series prediction. *J Hydrol* 374(3):235–241
- Garcia Nieto PG, Alonso Fernández J, de Cos Juez FJ, Sánchez Lasheras F, Diaz Muñoz C (2013) Hybrid modelling based on support vector regression with genetic algorithms in forecasting the cyanotoxins presence in the Trasona reservoir (Northern Spain). *Environ Res* 122:1–10
- Ghaedi AM, Vafaei A (2017) Applications of artificial neural networks for adsorption removal of dyes from aqueous solution: a review. *Adv Colloid Interf Sci* 245:20–39
- Gocić M, Motamedi S, Shamshirband S, Petković D, Ch S, Hashim R, Arif M (2015) Soft computing approaches for forecasting reference evapotranspiration. *Comput Electron Agric* 113:164–173
- Goodarzi M, Olivieri AC, Freitas MP (2009) Principal component analysis-adaptive neuro-fuzzy inference systems (ANFISs) for the simultaneous spectrophotometric determination of three metals in water samples. *Spectrochim Acta A Mol Biomol Spectrosc* 73(4):608–614
- Grivas G, Chaloulakou A (2006) Artificial neural network models for prediction of PM<sub>10</sub> hourly concentrations, in the Greater Area of Athens, Greece. *Atmos Environ* 40(7):1216–1229
- Guo H, Jeong K, Lim J, Jo J, Kim YM, Park J-P, Kim JH, Cho KH (2015) Prediction of effluent concentration in a wastewater treatment plant using machine learning models. *J Environ Sci* 32:90–101
- Hamed MM, Khalafallah MG, Hassanien EA (2004) Prediction of wastewater treatment plant performance using artificial neural networks. *Environ Model Softw* 19(10):919–928
- Hlihor RM, Diaconu M, Leon F, Curteanu S, Tavares T, Gavrilescu M (2015) Experimental analysis and mathematical prediction of Cd (II) removal by biosorption using support vector machines and genetic algorithms. *New Biotechnol* 32(3):358–368
- Hu YF, Yang CZ, Dan JF, Pu WH, Yang JK (2017) Modeling of expanded granular sludge bed reactor using artificial neural network. *J Environ Chem Eng* 5(3):2142–2150
- Huang Y, Lan Y, Thomson SJ, Fang A, Hoffmann WC, Lacey RE (2010) Development of soft computing and applications in agricultural and biological engineering. *Comput Electron Agric* 71(2):107–127
- Iliadis LS, Spartalís SI, Paschalidou AK, Kassomenos P (2007) Artificial neural network modelling of the surface ozone concentration. *International. J Comput Appl Math* 2(2):125–138
- Jang H, Topal E (2014) A review of soft computing technology applications in several mining problems. *Appl Soft Comput* 22:638–651
- Jang JS (1993) ANFIS: adaptive-network-based fuzzy inference system. *IEEE Trans Syst Man Cyber* 23(3):665–685
- Jang JSR, Sun CT, Mizutani E (1997) Neuro-fuzzy and soft computing. A computational approach to learning and machine intelligence. Prentice-Hall, Englewood Cliffs
- Jantzen J (1999) Design of fuzzy controllers. Technical University of Denmark: technical report (No: 98–E864) Department of Automation, Hoboken

- Kotti IP, Sylaios GK, Tsihrintzis VA (2013) Fuzzy logic models for BOD removal prediction in free-water surface constructed wetlands. *Ecol Eng* 51:66–74
- Kurt A, Gulbagci B, Karaca F, Alagha O (2008) An online air pollution forecasting system using neural networks. *Environ Int* 34(5):592–598
- Kusan H, Aytekin O, Özdemir İ (2010) The use of fuzzy logic in predicting house selling price. *Expert Syst Appl* 37(3):1808–1813
- Lin KP, Pai PF, Yang SL (2011) Forecasting concentrations of air pollutants by logarithm support vector regression with immune algorithms. *Appl Math Comput* 217(12):5318–5327
- Liu Z, Meng X (2009) Integration of improved BPNN algorithm and multistage dynamic fuzzy judgement and its application on ESMP evaluation. *J. Computers* 4(1):69–76
- Liu S, Tai H, Ding Q, Li D, Xu L, Wei Y (2013) A hybrid approach of support vector regression with genetic algorithm optimization for aquaculture water quality prediction. *Math Comput Model* 58(3):458–465
- Lou I, Xie Z, Ung WK, Mok KM (2017) Integrating support vector regression with particle swarm optimization for numerical modeling for algal blooms of freshwater. In *Advances in monitoring and modelling algal blooms in freshwater reservoirs*, Springer Netherlands, pp. 125–141
- Lu WZ, Wan WJ (2005) Potential assessment of the “support vector machine” method in forecasting ambient air pollutant trends. *Chemosphere* 59(5):693–701
- Luna AS, Paredes MLL, de Oliveira GCG, Corrêa SM (2014) Prediction of ozone concentration in tropospheric levels using artificial neural networks and support vector machine at Rio de Janeiro, Brazil. *Atmos Environ* 98:98–104
- Mamdani EH, Assilian S (1975) An experiment in linguistic synthesis with a fuzzy logic controller. *Int J Man Mach Stud* 7(1):1–13
- Mandal S, Mahapatra SS, Patel RK (2015) Neuro fuzzy approach for arsenic (III) and chromium (VI) removal from water. *J Water Process Eng* 5:58–75
- McCulloch WS, Pitts W (1943) A logical calculus of the ideas immanent in nervous activity. *Bull Math Biophys* 5(4):115–133
- Metternicht G, Gonzalez S (2005) FUERO: foundations of a fuzzy exploratory model for soil erosion hazard prediction. *Environ Model Softw* 20(6):715–728
- Mingzhi H, Ma Y, Jinquan W, Yan W (2009) Simulation of a paper mill wastewater treatment using a fuzzy neural network. *Expert Syst Appl* 36(3):5064–5070
- Mishra D, Goyal P (2016) Neuro-Fuzzy approach to forecasting Ozone Episodes over the urban area of Delhi, India. *Environ Technol Innov* 5:83–94
- Moazami S, Noori R, Amiri BJ, Yeganeh B, Partani S, Safavi S (2016) Reliable prediction of carbon monoxide using developed support vector machine. *Atmos Pollut Res* 7(3):412–418
- Molga E, Cherbański R, Szpyrkowicz L (2006) Modeling of an industrial full-scale plant for biological treatment of textile wastewaters: application of neural networks. *Ind Eng Chem Res* 45(3):1039–1046
- Mullai P, Arulselvi S, Ngo HH, Sabarathinam PL (2011) Experiments and ANFIS modelling for the biodegradation of penicillin–G wastewater using anaerobic hybrid reactor. *Bioresour Technol* 102(9):5492–5497
- Murnleitner E, Becker TM, Delgado A (2002) State detection and control of overloads in the anaerobic wastewater treatment using fuzzy logic. *Water Res* 36(1):201–211
- Nasiri F, Huang G (2008) A fuzzy decision aid model for environmental performance assessment in waste recycling. *Environ Model Softw* 23(6):677–689
- Noori R, Hoshyaripour G, Ashrafi K, Araabi BN (2010) Uncertainty analysis of developed ANN and ANFIS models in prediction of carbon monoxide daily concentration. *Atmos Environ* 44(4):476–482
- Noori R, Yeh HD, Abbasi M, Kachoosangi FT, Moazami S (2015) Uncertainty analysis of support vector machine for online prediction of five-day biochemical oxygen demand. *J Hydrol* 527:833–843
- Nunnari G, Dorling S, Schlink U, Cawley G, Foxall R, Chatterton T (2004) Modelling SO<sub>2</sub> concentration at a point with statistical approaches. *Environ Model Softw* 19(10):887–905

- Oliveira-Esquerre KP, Mori M, Bruns RE (2002) Simulation of an industrial wastewater treatment plant using artificial neural networks and principal components analysis. *Braz J Chem Eng* 19(4):365–370
- Olvera-García MÁ, Carbajal-Hernández JJ, Sánchez-Fernández LP, Hernández-Bautista I (2016) Air quality assessment using a weighted Fuzzy Inference System. *Eco Inform* 33:57–74
- Oswowski S, Garanty K (2007) Forecasting of the daily meteorological pollution using wavelets and support vector machine. *Eng Appl Artif Intell* 20(6):745–755
- Ozcan F, Atiş CD, Karahan O, Uncuoğlu E, Tanyildizi H (2009) Comparison of artificial neural network and fuzzy logic models for prediction of long-term compressive strength of silica fume concrete. *Adv Eng Softw* 40(9):856–863
- Ozdemir H, Demir G, Altay G, Albayrak S, Bayat C (2008) Prediction of tropospheric ozone concentration by employing artificial neural networks. *Environ Eng Sci* 25(9):1249–1254
- Ozger M, Sen Z (2007) Prediction of wave parameters by using fuzzy logic approach. *Ocean Eng* 34(3):460–469
- Ozkaya B, Demir A, Bilgili MS (2007) Neural network prediction model for the methane fraction in biogas from field-scale landfill bioreactors. *Environ Model Softw* 22(6):815–822
- Ozkaya B, Sahinkaya E, Nurmi P, Kaksonen AH, Puhakka JA (2008) Biologically  $\text{Fe}^{2+}$  oxidizing fluidized bed reactor performance and controlling of  $\text{Fe}^{3+}$  recycle during heap bioleaching: an artificial neural network-based model. *Bioprocess Biosyst Eng* 31(2):111–117
- Pai TY, Wan TJ, Hsu ST, Chang TC, Tsai YP, Lin CY, Su HC, Yu LF (2009) Using fuzzy inference system to improve neural network for predicting hospital wastewater treatment plant effluent. *Comput Chem Eng* 33(7):1272–1278
- Pai TY, Yang PY, Wang SC, Lo MH, Chiang CF, Kuo JL, Chu HH, Su HC, Yu LF, Hu HC, Chang YH (2011) Predicting effluent from the wastewater treatment plant of industrial park based on fuzzy network and influent quality. *Appl Math Model* 35(8):3674–3684
- Park S, Kim M, Kim M, Namgung HG, Kim KT, Cho KH, Kwon SB (2018) Predicting  $\text{PM}_{10}$  concentration in Seoul metropolitan subway stations using artificial neural network (ANN). *J Hazard Mater* 341:75–82
- Parveen N, Zaidi S, Danish M (2016) Support vector regression model for predicting the sorption capacity of lead (II). *Perspect Sci* 8:629–631
- Pendashteh AR, Fakhru'l-Razi A, Chaibakhsh N, Abdullah LC, Madaeni SS, Abidin ZZ (2011) Modeling of membrane bioreactor treating hypersaline oily wastewater by artificial neural network. *J Hazard Mater* 192(2):568–575
- Perendeci A, Arslan S, Çelebi SS, Tanyolaç A (2008) Prediction of effluent quality of an anaerobic treatment plant under unsteady state through ANFIS modeling with on-line input variables. *Chem Eng J* 145(1):78–85
- Platt L (1998) Fast training of SVM using sequential optimization. In: Scholkopf B, Burges B, Smola A (eds) *Advances in kernel methods – support vector learning*. MIT Press, Cambridge, pp 185–208
- Podder MS, Majumder CB (2016) Phycoremediation potential of *Botryococcus braunii*. *Water Conserv Sci Eng* 1(1):49–68
- Prasad K, Gorai AK, Goyal P (2016) Development of ANFIS models for air quality forecasting and input optimization for reducing the computational cost and time. *Atmos Environ* 128:246–262
- Qaderi F, Babanejad E (2017) Prediction of the groundwater remediation costs for drinking use based on quality of water resource, using artificial neural network. *J Clean Prod* 161:840–849
- Rangasamy P, Iyer PVR, Ganesan S (2007) Anaerobic tapered fluidized bed reactor for starch wastewater treatment and modeling using multilayer perceptron neural network. *J Environ Sci* 19(12):1416–1423
- Raduly B, Gernaey KV, Capodaglio AG, Mikkelsen PS, Henze M (2007) Artificial neural networks for rapid WWTP performance evaluation: methodology and case study. *Environ Model Softw* 22(8):1208–1216

- Rahimzadeh A, Ashtiani FZ, Okhovat A (2016) Application of adaptive neuro-fuzzy inference system as a reliable approach for prediction of oily wastewater microfiltration permeate volume. *J Environ Chem Eng* 4(1):576–584
- Rihani R, Bensmaili A, Legrand J (2009) Fuzzy logic modelling tracer response in milli torus reactor under aerated and non-aerated conditions. *Chem Eng J* 152(2):566–574
- Rubens N (2006) The application of fuzzy logic to the construction of the ranking function of information retrieval systems. *Comput Model New Technol* 10:20–27
- Sadiq R, Al-Zahrani MA, Sheikh AK, Husain T, Farooq S (2004) Performance evaluation of slow sand filters using fuzzy rule-based modelling. *Environ Model Softw* 19(5):507–515
- Sadrzadeh M, Ghadimi A, Mohammadi T (2009) Coupling a mathematical and a fuzzy logic-based model for prediction of zinc ions separation from wastewater using electro dialysis. *Chem Eng J* 151(1):262–274
- Sahinkaya E, Özkaya B, Kaksonen AH, Puhakka JA (2007) Neural network prediction of thermophilic (65°C) sulfidogenic fluidized-bed reactor performance for the treatment of metal-containing wastewater. *Biotechnol Bioeng* 97(4):780–787
- Sahinkaya E (2009) Biotreatment of zinc-containing wastewater in a sulfidogenic CSTR: performance and artificial neural network (ANN) modelling studies. *J Hazard Mater* 164(1):105–113
- Saral A, Ertürk F (2003) Prediction of ground level SO<sub>2</sub> concentration using artificial neural networks. *Water Air Soil Pollut Focus* 3(5):307–316
- Sari H, Yetilmezsoy K, İlhan F, Yazici S, Kurt U, Apaydin O (2013) Fuzzy-logic modeling of Fenton's strong chemical oxidation process treating three types of landfill leachates. *Environ Sci Pollut Res* 20(6):4235–4253
- Schölkopf B, Smola AJ (2002) *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT Press, Cambridge, MA
- Shahraiyni HT, Sodoudi S, Kerschbaumer A, Cubasch U (2015) A new structure identification scheme for ANFIS and its application for the simulation of virtual air pollution monitoring stations in urban areas. *Eng Appl Artif Intell* 41:175–182
- Singh KP, Basant N, Gupta S (2011) Support vector machines in water quality management. *Anal Chim Acta* 703(2):152–162
- Sofuoglu SC, Sofuoglu A, Birgili S, Tayfur G (2006) Forecasting ambient air SO<sub>2</sub> concentrations using artificial neural networks. *Energy Sources Part B* 1(2):127–136
- Sowlat MH, Gharibi H, Yunesian M, Mahmoudi MT, Lotfi S (2011) A novel, fuzzy-based air quality index (FAQI) for air quality assessment. *Atmos Environ* 45(12):2050–2059
- Sozen A, Kurt M, Akçayol MA, Özalp M (2004) Performance prediction of a solar driven ejector-absorption cycle using fuzzy logic. *Renew Energy* 29(1):53–71
- Takagi T, Sugeno M (1985) Fuzzy identification of systems and its applications to modeling and control. *IEEE Trans Syst Man Cybern* 1:116–132
- Tay JH, Zhang X (2000) A fast predicting neural fuzzy model for high-rate anaerobic wastewater treatment systems. *Water Res* 34(11):2849–2860
- Taylan O (2017) Modelling and analysis of ozone concentration by artificial intelligent techniques for estimating air quality. *Atmos Environ* 150:356–365
- Topcu İB, Saridemir M (2008) Prediction of rubberized concrete properties using artificial neural network and fuzzy logic. *Constr Build Mater* 22(4):532–540
- Traore A, Grieu S, Puig S, Corominas L, Thiéry F, Polit M, Colprim J (2005) Fuzzy control of dissolved oxygen in a sequencing batch reactor pilot plant. *Chem Eng J* 111(1):13–19
- Turkdogan-Aydinol FI, Yetilmezsoy K (2010) A fuzzy-logic-based model to predict biogas and methane production rates in a pilot-scale mesophilic UASB reactor treating molasses wastewater. *J Hazard Mater* 182(1):460–471
- Vakili M, Sabbagh-Yazdi SR, Kalhor K, Khosrojerdi S (2015) Using artificial neural networks for prediction of global solar radiation in Tehran considering particulate matter air pollution. *Energy Procedia* 74:1205–1212
- Vapnik V (1995) *Nature of statistical learning theory*. Springer, New York
- Vapnik V (1998) *Statistical learning theory*. Wiley, New York

- Wan J, Huang M, Ma Y, Guo W, Wang Y, Zhang H, Li W, Sun X (2011) Prediction of effluent quality of a paper mill wastewater treatment using an adaptive network-based fuzzy inference system. *Appl Soft Comput* 11(3):3238–3246
- Wieland D, Wotawa F, Wotawa G (2002) From neural networks to qualitative models in environmental engineering. *Comput Aided Civ Infrastruct Eng* 17(2):104–118
- Wotawa F, Wotawa G (2001) Deriving qualitative rules from neural networks – a case study for ozone forecasting. *AI Commun* 14(1):23–33
- Wu GD, Lo SL (2008) Predicting real-time coagulant dosage in water treatment by artificial neural networks and adaptive network-based fuzzy inference system. *Eng Appl Artif Intell* 21(8):1189–1195
- Xie Q, Ni JQ, Su Z (2017) A prediction model of ammonia emission from a fattening pig room based on the indoor concentration using adaptive neuro fuzzy inference system. *J Hazard Mater* 325:301–309
- Xu Y, Ma C, Liu Q, Xi B, Qian G, Zhang D, Huo S (2015) Method to predict key factors affecting lake eutrophication – a new approach based on Support Vector Regression model. *Int Biodeterior Biodegrad* 102:308–315
- Yeganeh B, Motlagh MSP, Rashidi Y, Kamalan H (2012) Prediction of CO concentrations based on a hybrid Partial Least Square and Support Vector Machine model. *Atmos Environ* 55:357–365
- Yetilmezsoy K (2006) Determination of optimum body diameter of air cyclones using a new empirical model and a neural network approach. *Environ Eng Sci* 23(4):680–690
- Yetilmezsoy K (2012) Fuzzy-logic modeling of Fenton's oxidation of anaerobically pretreated poultry manure wastewater. *Environ Sci Pollut Res* 19(6):2227–2237
- Yetilmezsoy K, Abdul-Wahab SA (2012) A prognostic approach based on fuzzy-logic methodology to forecast PM<sub>10</sub> levels in Khaldiya residential area, Kuwait. *Aerosol Air Qual Res* 12(6):1217–1236
- Yetilmezsoy K, Demirel S (2008) Artificial neural network (ANN) approach for modeling of Pb (II) adsorption from aqueous solution by Antep pistachio (*Pistacia Vera* L.) shells. *J Hazard Mater* 153(3):1288–1300
- Yetilmezsoy K, Sapci-Zengin Z (2009) Stochastic modeling applications for the prediction of COD removal efficiency of UASB reactors treating diluted real cotton textile wastewater. *Stoch Environ Res Risk Assess* 23(1): 13–26.
- Yetilmezsoy K, Saral A (2007) Stochastic modeling approaches based on neural network and linear–nonlinear regression techniques for the determination of single droplet collection efficiency of countercurrent spray towers. *Environ Model Assess* 12(1):13–26
- Yetilmezsoy K (2010) Modeling studies for the determination of completely mixed activated sludge reactor volume: Steady-state, empirical and ANN applications. *Neural Netw World* 20(5): 559–589.
- Yetilmezsoy K, Ozkaya B, Cakmakci M (2011a) Artificial intelligence-based prediction models for environmental engineering. *Neural Netw World* 21(3):193–218
- Yetilmezsoy K, Fingas M, Fieldhouse B (2011b) An adaptive neuro-fuzzy approach for modeling of water-in-oil emulsion formation. *Colloids Surf A Physicochem Eng Asp* 389(1):50–62
- Yetilmezsoy K, Fingas M, Fieldhouse B (2012) Modeling water-in-oil emulsion formation using fuzzy logic. *J Mult Valued Log Soft Comput* 18:329–353
- Yetilmezsoy K, Turkdogan FI, Temizel I, Gunay A (2013) Development of ann-based models to predict biogas and methane productions in anaerobic treatment of molasses wastewater. *Int J Green Energy* 10(9):885–907
- Yetilmezsoy K, Ozgun H, Dereli RK, Ersahin ME, Ozturk I (2015) Adaptive neuro-fuzzy inference-based modeling of a full-scale expanded granular sludge bed reactor treating corn processing wastewater. *J Intell Fuzzy Syst* 28(4):1601–1616
- Yildirim Y, Bayramoglu M (2006) Adaptive neuro-fuzzy based modelling for prediction of air pollution daily levels in city of Zonguldak. *Chemosphere* 63(9):1575–1582
- Zadeh LA (1965) Fuzzy sets. *Inf Control* 8(3):338–353
- Zakaria Z, Isa NAM, Suandi SA (2010) A study on neural network training algorithm for multiface detection in static images. *World Acad Sci Eng Techn* 4(2):345–348