

Hydrological Neural Modeling aided by Support Vector Machines

L. S. Iliadis¹, S.I. Spartalis²

¹*Department of Forestry & Management of the Environment & Natural Resources liliadis@fmenr.duth.gr*

²*Department of Production Engineering & Management, School of Engineering sspart@pme.duth.gr Democritus University of Thrace, Greece*

Abstract: This paper aims in the construction of an Artificial Neural Network model that performs successful estimation of the Maximum Water Supply (m^3/sec) and of the Special Water Flow ($\text{m}^3/\text{sec} \cdot \text{Km}^2$) using several topographic, meteorological and morphometric parameters as independent variables. Support Vector Machines applying specific Kernel algorithms [9] are used to determine the Error or Loss of the Neural Network model and to enhance its ability to generalize. Data come from the Greek island of Thasos, which is located in the North-Eastern part of the Aegean sea. As a matter of fact, this manuscript can be considered as a specific case study, but its modeling design principles and its Error minimization methods can be applied in a wide range of research studies and applications

Keywords: Artificial Neural Networks, Support Vector Machines, Error Minimization, Water Resources management

1. Introduction

Effective water resources management is one of the most crucial environmental challenges of our times. Extreme flood incidents happen more and more often all over the world and they cause immense damages in infrastructure and in human lives.

This research effort has two distinct orientations of equal importance. First it deals with the development of Artificial Neural Network (ANN) modeling of both average annual maximum water supply (m^3/sec) (AAMWS) and special water flow (SWF) ($\text{m}^3/\text{sec} \cdot \text{Km}^2$) for the torrential watersheds of a wide mountainous area. It also deals with the supportive role of *Soft Computing* methods like e-Regression Support Vector Machines towards the evaluation of neural network models. From this point of view this specific contribution can be considered as having an innovative role towards ANN development and evaluation.

Another key aspect is the fact that it is performed by using structural data (remaining unchanged over-time) and only few actual real time measurements. Under this perspective this modeling effort can be considered cost and time effective and it can provide invaluable assistance towards the design of flood protection and prevention policy. Existing methods like the one of Gavrilovic [6] [23] concentrate mostly in the load of sediments and secondly in the influence of various structural

and dynamic factors. The equation of Gavrilovic is very useful and widely used but it cannot respond to sudden changes in the morphometric background of a torrential stream. For example if the percentage of forest cover is suddenly reduced dramatically due to a major forest fire, the Gavrilovic equation absorbs this change during the first years and it requires a longer period of time to show a significant difference.

1.1. Literature Review

Estimation of both AAMWS and SWF is a very important process, as it can serve for the rough calculation of existing water supplies and for the evaluation of the potential Torrential Risk for each mountainous watershed.

ANN have been used effectively in various research projects concerning water protection and water management in Europe, in USA and in many other countries. An ANN using Fuzzy Logic has been developed in the Netherlands for the control of water levels in polder areas [14]. Another ANN that performs river flow forecasting has been developed also in the Netherlands [3]. Also in USA (US Department of Energy) and in Europe, ANN have been developed for the prediction of stream-water quality [1],[15] or towards water-management in general. Wastewater and water management and protection models using ANN have also been developed recently [7],[20]. UNESCO has also funded several research projects using ANN and fuzzy logic for urban water management and flood risk evaluation [25][10]. Finally neural networks have been used for the prediction and forecasting of water resources variables [15].

2. The Environmental Problem of floods

2.1. Data and research area

Actual data corresponding to the eight input parameters described above were gathered from the *twenty* most important mountainous streams of the Thasos island. The *average annual rain height* corresponds to a period of *thirty* years. Thus, thirty data records were structured for each torrential stream, raising the total number of existing data vectors to *six hundred*. Except of the rain height and forest cover, the rest of the input parameters are structural and they do not change significantly overtime. A small sample of the available data that were used in this research is presented in the following table 1.

In the case of water resources modeling the input data are points in a space R^n and the output represents points in the well-known plain R^k where $k=2$ in this case, since the output parameters are two.

According to [18] there are eight main parameters that influence the AAMWS and the SWF and for which there are available data in the Greek repositories. So the input space can be considered as vectors of the R^8 space. The eight input data are the *average altitude*, the *average slope*, the *average annual rain height*, the *percentage of forest cover*, the *percentage of compact geological forms*, the *area*, the *total number of pipes* and the *total length of the pipes* of the torrential watersheds of Thasos island.

Table 1: A small sample of the selected data records

Torrential Stream	Average Watershed Altitude	Average Watershed Slope	Average watershed annual rain-height	Percentage of forest cover of watershed	Percentage of watershed compact geological forms
	<i>m</i>	%	<i>Mm</i>	%	%
Thasos	433.21	35.61	1059.40	81.47	87.81
Panagia	261.14	28.50	919.03	56.68	78.97
Potamia	574.74	49.93	1151.40	88.83	84.89
Skala Potamias	535.44	53.31	1126.10	56.14	76.55
Mavrou Lakou	378.19	50.84	1023.20	88.08	85.15
Kleisidiou	330.64	35.67	975.11	36.49	100.02
Thimonias	325.41	35.10	971.88	93.74	96.83

The two output parameters of the model are the AAMWS and the corresponding SWF which is estimated [13] by the following formula 1 where F is the area of the mountainous watershed in Km^2 and a is a parameter determined by water management specialists. The parameter a takes values in the closed interval $[0.6,1]$. SWF is a normalized version (per km^2) of the mean water supply (MWS).

$$q_{\max} = a \frac{32}{0.5 + \sqrt{F}} \quad (1)$$

3. Theoretical Background

3.1. ANN

Modern ANNs are rooted in many disciplines, like neurosciences, mathematics, statistics, physics and engineering. They find many successful applications in such diverse fields as modeling, time series analysis, pattern recognition and signal processing, due to their ability to learn from input data with or without a teacher.

Their computing power is achieved through their massively parallel distributed structure and their ability to learn and therefore generalize [8]. Generalization refers to their ability to produce reasonable output for inputs not encountered during the training process [8]. ANN consist of units called neurons whose computing ability is typically restricted to a rule for combining input signals and an activation rule that takes the combined input to calculate the output signal [2].

The proper design of ANN requires the development of various topologies using numerous optimization algorithms and transfer functions, before determining the optimal one. We have experimented thoroughly towards this direction. Sixty percent of the actual data records corresponding to the eight independent parameters,

have been used as input in the *training* phase and the other 40% were used as input in the *testing* phase of the ANN in order to evaluate the generalization ability.

3.2. Support Vector Machines

The *support vector machines* (SVM) or optimal margin classifiers are new type of learning algorithms based on statistical learning theory proposed by Vapnick [27]. SVM are used not only for classification but for regression (functional approximation) problems as well [4],[28]. When used for classification, a SVM algorithm creates a *Hyperplane* that separates the data into two classes. Given training examples labeled either "yes" or "no", a maximum-margin hyperplane is identified which splits the "yes" from the "no" training examples, such that the distance between the hyperplane and the closest examples (the margin) is maximized. It is a fact that SVM can handle multiple continuous and categorical variables. To construct an optimal hyperplane, SVM employees an iterative training algorithm, which is used to minimize an error function. According to the form of the error function, SVM models can be classified into various distinct groups:

- Classification SVM Type 1 (also known as C-SVM classification)
- Classification SVM Type 2 (also known as nu-SVM classification)
- Regression SVM (also known as epsilon-SVM regression)

In this project, an *epsilon-regression* SVM (ERSVM) was applied. In an ERSVM one has to estimate the functional dependence of the dependent variable y on a set of independent variables x . Here an **ε-tube** is constructed that determines the loss degree of the regression. It assumes, like other regression problems, that the relationship between the independent and dependent variables is given by a deterministic function f plus the addition of some additive noise: $y = f(x) + \text{noise}$ (2)

The task is then to find a functional form for f that can correctly predict new cases that the SVM has not been presented with before. This can be achieved by training the SVM model on a sample set. In our case the ANN plays the role of the predicting function. In fact the learning machine is given I training data from which it attempts to learn the input-output relationship (which may have the form of dependency, mapping or function) $f(x)$. In this case the training data set has the form $D = \{(X_i, Y_i) \in \mathbb{R}^n \times \mathbb{R} \mid i \in \{1, \dots, I\}\}$ which means that it contains one pair of values (X_i, Y_i) . It should be specified that the inputs X are n -dimensional Vectors that belong to \mathbb{R}^n and the output Y of the system are continuous values [12]. The SVM

uses approximating functions of the form $f(x, w) = \sum_{i=1}^N w_i \phi_i(x)$ (3) where the

functions $\phi_i(x)$ are called features [12]. In regression problems like the one we are facing in this study, typically some measures of error approximation are used.

3.3. Applications of RSVM to ANN evaluation

Let's suppose that we have an ANN with an input vector \mathbf{X} , a bias weight vector \mathbf{b} , a *hidden layer* weights matrix \mathbf{V} , and an output weight vector \mathbf{w}^T . More specifically:

$$\mathbf{X} = [X_1, X_2, X_3, \dots, X_n]^T \quad (4) \quad \mathbf{b} = [b_1, b_2, b_3, \dots, b_j]^T \quad (5)$$

$$\mathbf{w} = [w_1, w_2, w_3, \dots, w_j, w_{j+1}]^T \quad (6)$$

$$V = \begin{bmatrix} u_{11} & \dots & u_{1j} & \dots & u_{1n} \\ \dots & \dots & \dots & \dots & \dots \\ u_{j1} & \dots & u_{jj} & \dots & u_{jn} \\ \dots & \dots & \dots & \dots & \dots \\ u_{j1} & \dots & u_{ji} & \dots & u_{jn} \end{bmatrix} \quad (7)$$

The output layer neurons may have linear activation functions (for regression problems) or *sigmoid* for classification or pattern recognition tasks. The approximation scheme for such an ANN is shown in formula 8 [12]:

$$\alpha(X, V, w, b) = F(X, V, w, b) = \sum_{j=1}^J w_j \sigma_j(v_j^T x + b_j) \quad (8)$$

The construction of a SVM algorithm for regression is actually a problem of minimizing the empirical risk R_{emp}^e and $\|W\|^2$ at the same time. Actually the problem is the estimation of a linear regression hyperplane $f(x, w) = W^T x + b$. This can be achieved by minimizing the quantity R given by formula 9 [12]:

$$R = \frac{1}{2} \|w\|^2 + C \left(\sum_{i=1}^I |y_i - f(x_i, w)| \right) \quad (9) \quad \text{Here Vapnick's } \square$$

insensitivity loss function replaces squared error and $C \sim 1/\square$.

In fact Vapnick introduced a general type of error (loss) function the linear loss function with \square -insensitivity zone which is given by the following formula 10 [12].

$$|y - f(x, w)|_{\varepsilon} = \begin{cases} 0 \dots \text{if } |y - f(x, w)| \leq \varepsilon \\ |y - f(x, w)| - \varepsilon \dots \text{otherwise} \end{cases} \quad (10)$$

According to this algorithm the loss equals to 0 if the difference between the predicted by the ANN value $f(x, w)$ and the measured value is less than \square . In this way function 10 defines a \square -tube. In the cases with the ANN predicted value within the tube, the loss (or error) is zero. In all other cases the loss equals to the magnitude of the difference between the ANN output value and the radius \square of the tube. If \square and \square^* are given by the following formulas 11 and 12, then formula 13 presents the final value of risk R that has to be minimized [27],[12].

$$\text{The following equations 11 and 12 } |y - f(x, w)| - \varepsilon = \xi \quad (11)$$

$|y - f(x, w)| - \varepsilon = \xi^*$ (12) are valid for all of the data that are located above the tube.

For all of the data that are located below the tube equation 13 is

$$\text{true } R_{w, \xi, \xi^*} = \left[\frac{1}{2} \|w\|^2 + C \left(\sum_{i=1}^I \xi + \sum_{i=1}^I \xi^* \right) \right] \quad (13)$$

In the above formula 13 the following constrains should stand.

$$y_i - w^T x_i - b \leq \varepsilon + \xi \quad (14) \quad w^T x_i + b - y_i \leq \varepsilon + \xi^* \quad (14)$$

$$\xi \geq 0 \quad i = 1, I \quad (15) \quad \xi^* \geq 0 \quad i = 1, I \quad (16)$$

The following figure 1 clearly shows the case of an \square -tube and the parameters involved.

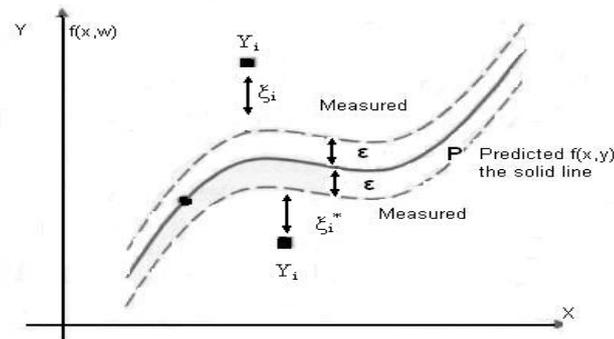


Figure 1: The parameters used in one dimensional ERSVM

In the previous formulae the \square and \square^* are slack variables for measurements “above” and “below” an \square -tube. Both slack variables are positive values. Finally it should be clarified that the constant C which influences a trade-off between an approximation error and the weights vector norm $\|W\|$ is a design parameter whose value is chosen by the user. Also the norm $\|W\| = \sqrt{w_1^2 + \dots + w_n^2}$ where W is the weight vector. An increase in C penalizes larger errors (large \square and \square^*) and so it leads to a decrease in approximation error [12].

4. Hydrological Neural Model

Training is the process of ANN development, where the connection weights are adapted or modified in response to stimuli being presented at the input buffer and optionally to the output buffer. The *hidden layer* is the place where the data is being processed and it may consist of one or two sub-layers.

The *feed forward* network structure with input, output and HSL varying from 1 to 2, applying various optimization algorithms were tried in the training phase. We have experimented with the *back propagation* (BP) optimization algorithm (introduced by Rummelhart et al.) [21],[22][24][26], the *general regression* ANN architecture, the *modular ANN* topology [5] and the *radial basis function (RBF)* model. Our experimentations included the employment of various *transfer functions*.

Modular ANNs were proposed by Jacobs, Jordan, Nowlan and Hinton [19]. They consist of a group of BP networks referred to as “*local experts*” competing to learn different aspects of a problem. A “*gating network*” controls the competition and learns to assign different parts of the data space to different networks. When only one ANN is appropriate for a given problem, the *gating network* tends to favor it [19]. RBF are ANN having an internal representation of hidden neurons which are “*radially symmetric*”. For a neuron to be *radially symmetric* it needs to have the following three constituents: a) A center which is a vector in the input space that is typically stored in the weight vector from the input layer to the pattern unit. b) A

distance measure to determine how far an input vector stands from the center. In this case standard Euclidean distance is used. c) A transfer function (using a single variable) which determines the output of the Neuron by mapping the output of the distance function.

The “*root mean square error*” (RMS Error) and the Confusion Matrix (CM) were used to check the validity of the ANN. The CM is roughly a kin to a scatter diagram, with the x-axis representing the desired output and the y-axis representing the actual output. The main difference from a scatter diagram is that the CM breaks the diagram into a grid. Each grid square is called a bin. Each output from the probe point produces a count within one of the bins [19]. For example if the probe point produces an output of 0.6 and the desired output was 0.5 then the bin around the intersection of 0.6 from the y-axis and 0.5 from the x-axis receives a count. These counts are displayed by a bar within the bin and the bar grows as counts are received. The bin receiving the most counts is shown at full height, while all of the other bins are scaled in relation to it [19]. The network with the optimal configuration must have the bins (the cells in each matrix) on the diagonal from the lower left to the upper right. Also more sophisticated Soft Computing tools were also applied to examine and determine the validity of the optimal ANN. An important aspect of the CM is that the value of the vertical axis of the produced histogram is the *common mean correlation (CMC) coefficient* of the desired (d) and the predicted output (y) across the Epoch. The CMC is calculated by the following formula (2):

$$CMC = \frac{\sum (d_i - \bar{d})(y_i - \bar{y})}{\sqrt{\sum (d_i - \bar{d})^2 \sum (y_i - \bar{y})^2}}, \quad \bar{d} = \frac{1}{E} \sum_1^n d_i \quad \text{and} \quad \bar{y} = \frac{1}{E} \sum_1^n y_i \quad (17)$$

It should be clarified that i ranges from 1 to n (the number of cases in the data training set) and E is the Epoch size which is the number of sets of training data presented to the ANN learning cycles between weight updates.

5. Experimental Results

The first and the most important thing that had to be determined was the topology of the ANN and its optimization and transfer functions. Thus various experimentations were performed using various numbers of hidden sub-layers with various numbers of neurons used in each sub-layer.

The six hundred data records were divided in two subsets, the training and the testing one. The division was performed in a totally random manner. The training set included the 60% of the data and the testing the rest 40%. Training has shown that the optimal Artificial Neural Network (OANN) was the one that used BP optimization algorithm with the “*tangent hyperbolic*” activation function (mapping into the range -1.0 to 1.0) and with the “*extended delta bar delta*” (ExtDBD) training rule [11], [17] as the transfer function. It consisted of eight neurons in the input layer, only one hidden sub-layer with nine neurons and two neurons in the output layer. The RMS Error in the training phase was 0.0045 and the $R^2=0.9997$. It should be clarified here that the ExtDBD is a *heuristic* technique that has been

successful in a number of application areas and it uses termed momentum. A term is added to the standard weight change, which is proportional to the previous weight change. In this way good general trends are reinforced and oscillations are damped [19]. The random number seed was kept constant before each training round and the learning coefficient ratio was kept at 1. The following figure 2, shows the architecture of the OANN.

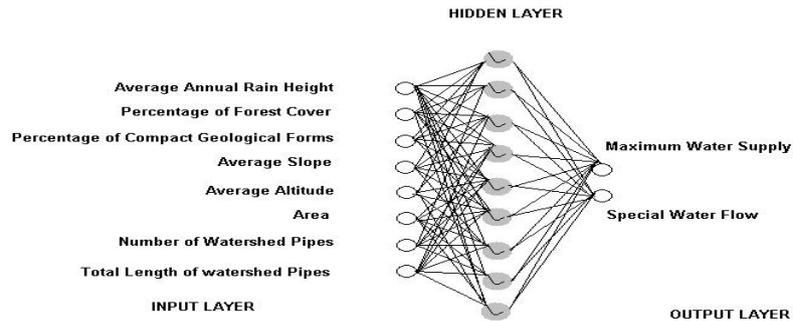


Figure 2: Architecture of the optimal ANN

Every time an ANN model is developed we need to ensure that the data are compatible to the learning algorithms and to the transfer functions applied. When a value coming into a processing element (PE) is beyond the PEs transfer function range, that PE is said to be saturated and of course the function produces dummy values [19]. The tangent hyperbolic function that is used in this specific research is mapping into the range -1.0 to 1.0 but it accepts values only between -3 and $+3$. Saturation occurs when a PE's summation value (the sum of the inputs times the weights) exceeds this range. So the input data were normalized by dividing them properly (by 1000) so that they will not exceed the acceptable range. This provided a good performance of the transfer function.

After the determination of the ANN structure and topology, the testing process was performed to prove the potential ability of the ANN to generalize with first time seen data. The results of the testing phase are the following. The RMS Error was 0.1404 and the $R^2 = 0.9763$. Two confusion matrices were developed for the two output parameters. Both confusion matrices had all of their cells located very close to the diagonal from the lower left to the upper right. The CMC in the case of the *maximum water flow* was equal to 0.9654 whereas in the case of the Special Water flow it was equal to 0.9872. The following figure 3 shows the two confusion matrices corresponding to the two output parameters. It is clear that the ANN shows a good performance with unseen data and also it retains a good and simple structure which is also very important.

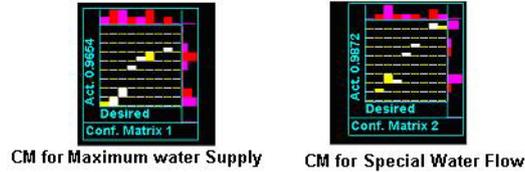


Figure 3: The Confusion Matrices in the Testing Phase

A complicated ANN with many hidden sub-layers or with a large number of hidden neurons does not guarantee good level of generalization capacity. The specific ANN that was developed here has a simple architecture and this is an important benefit. Though the generalization ability of the ANN has been demonstrated an ERSVM kernel was used to determine the risk of the ANN model.

5.1. Using the SVM to determine the ANN with the minimum Loss

The output of the ANN (with first time seen data) and the actual experimental values (coming from direct measurements in the most important torrential streams of the island) were used as input to the RSVM. Also the weights of the output Layer of the BP ANN were input to the SVM to determine the actual risk according to formula 13, [27],[29].

In both cases the actual measured values were subtracted from the ANN output ones in order to estimate the maximum value. For the case of AAMWS ϵ was defined to be equal to 4.5 almost equal to the maximum difference between the predicted and the measured values. This value of ϵ defines the width of a specific tube. Based on formulae 11 and 12 and on the values of the weight vector W of the developed ANN, $\sum_{i=1}^I \xi = 4.6608$ whereas $\sum_{i=1}^I \xi^* = 4.924$ where constant C was defined to have a low value of 0.1.

Based on these values and using formula 13, the final estimation for the value of risk R for AAMWS was equal to 0.204.

The following estimations were performed for the SWF. According to formulae 11 and 12 and using the values of the weight vector W of the developed ANN, the

$\sum_{i=1}^I \xi = 3.289$ whereas $\sum_{i=1}^I \xi^* = 0.935$ where $C=0.1$ and ϵ was defined to be equal to 1.4 almost equal to the maximum difference between the predicted and the measured values.

Finally the risk R for the Special Water Flow was estimated using formula 13 to have a value of 0.1998. In both of the cases the risk was quite low showing that the ANN prediction values include quite low risk. This means that the degree of loss is quite low for the ANN predicted values, compared to the actual ones.

Table 2: The BP ANN output compared to actual measurements in Testing

Special Water Flow			Average Annual Maximum Water Supply		
Actual Ex- perimental Measurements y	ANN Output f(X,w) for SWF	Tube width value \square for SWF	Actual Ex- perimental Measurements y	ANN Output f(X,w) for AAMWS	Tube width value \square for AAMWS
4.97	3.754	1.4	180.94	196.01	4.5
11.07	11.562		81.69	80.84	
5.11	4.11		176.36	191.628	
9.11	10.588		98.84	94.279	
4.64	3.355		190.77	205.605	
10.45	11.745		86.44	83.275	
14.28	13.206		63.28	66.413	
7	6.349		128.9	140.593	
11.27	11.117		80.07	80.146	
6.59	5.458		136.54	154.573	

6. Conclusion

The developed OANN for the case of Thasos island, has proven its ability to estimate the AAMWS and the SWF successfully and reliably and most of all its ability to generalize with first time seen data. This has been proven not only by the instruments used in the Training and Testing phases but also from the low risk estimation that was done by an ERSVM. Water management experts should consider the estimations of the ANN for each watershed in order to design specific measures and the proper actions towards an orthological management of the existing water supplies in the mountainous watersheds. In this way the Torrential Risk and the Risk of water lack can be reduced.

Thus not only a useful and reliable tool has been developed for the water management of Thasos but also the potential use of ANN on a wider scale has been proven. More similar applications can be developed for other areas with available data. The modeling methodology is innovative and it uses a lot of modern instruments and algorithms that evaluate its performance from various perspectives.

Of course we can not claim that this model has covered all of the parameters determining the AAMWS and its corresponding SWF. With this volume of data (twenty torrential streams concerning a period of thirty years) using the eight mentioned input parameters this study can be considered as a successful preliminary one. The results of this research effort have a limited scope for the area of Thasos, but the modeling methodology can be applied in any place. However it can only be considered a preliminary one. A future effort with more data series concerning

perhaps more variables, will determine beyond any doubt the final structures of the proper ANN.

References

1. Bowers J.A., Shedrow C.B., (2000) Predicting Stream Water Quality using Artificial Neural Networks. Westinghouse Savannah River Company, SRS Ecology Environmental Information Document, MS-2000-00112. Savannah River Site, Aiken, SC 29808, US Department of Energy
2. Callan R., (1999) The Essence of Neural Networks. Prentice Hall, UK
3. Dibike Y., Solomatine D., (1999) River Flow Forecasting using Artificial Neural Networks European Geophysical Society (EGS) XXIV General Assembly, The Hague, The Netherlands
4. Drucker, H., C.J.C. Burges, L. Kaufman, A Smola, and V.N. Vapnik, (1997) Support Vector Regression Machines Advances in neural information processing systems Volume 9, 155-161. Cambridge, MA:MIT Press
5. Gaupe, D., (1997) Principles of artificial neural networks World Scientific. Singapore
6. Gavrilovic Sl. (1972) Inzenjering o bujicnim tovoklima i eroziji. Beograd
7. Hamed M. M., Khalafallah M. G., and Hassanien E. A., 2004. Prediction of wastewater treatment plant performance using artificial neural networks. Environmental Modelling and Software, Elsevier Science, October 2004, Vol. 19, Issue 10, pp. 919-928.
8. Haykin S., (1999) Neural Networks A Comprehensive Foundation. Prentice Hall,USA.
9. Herbrich R., (2002) Learning Kernel Classifiers: Theory and Algorithms. MIT Press, Cambridge, MA.
10. Iliadis L., Maris F., Marinos D., (2004) A Decision Support System using Fuzzy relations for the estimation of long-term torrential risk of mountainous watersheds: The case of river Evros. Proceedings ICNAAM 2004 Conference, Chalkis, Greece
11. Jacobs, R.A. (1988) Increased rates of convergence through learning rate adaption. Neural Networks 1:295-307.
12. Kecman V., (2001) Learning and Soft Computing. MIT Press. London England.
13. Kotoulas D., (2001) Hydrology and Hydraulics of natural Environment, pp. 223. Aristoteles University of Thessaloniki, Greece.
14. Lobbrecht A.H., Solomatine D.P., (1999) Control of water levels in polder areas using neural networks and fuzzy adaptive systems. Water Industry Systems: Modelling and Optimization Applications, Vol. 1. pp. 509-518. Research Studies Press. Baldock, UK
15. Maier H. R., Dandy G. C., (2000) Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications

- Environmental Modelling and Software, Volume 15, Issue 1, January 2000, pp. 101-124.
16. Maier H. R., Morgan N., Chow C. W. K., (2004) Use of ANN for predicting optimal alum doses and treated water quality parameters Environmental Modelling & Software, Volume 19, Issue 5, May 2004, pp. 485-494.
 17. Minai, A.A. and R.D. Williams. (1990) Acceleration of Back-Propagation through Learning Rate and Momentum Adaption. International Joint Conference on Neural Networks. Volume I, pp. 676- 679.
 18. Maris, L. Iliadis, (2007) "A computer system using two membership functions and T-Norms for the calculation of mountainous watersheds torrential risk: The case of lakes Trixonida and Lisimaxia" Book Chapter Book Series: "[Developments in Plant and Soil Sciences](#)" Book Title: "[Eco-and Ground Bio-Engineering: The Use of Vegetation to Improve Slope Stability](#)" Volume 103 pp. 247-254 Springer Netherlands
 19. Neuralware, (2001) Getting started. A tutorial for Neuralworks Professional II/PLUS. Carnegie, PA, USA
 20. Partalas I., Tsoumakas G., Hatzikos E., Vlahavas I., 2008. Greedy Regression Ensemble Selection: Theory and an Application to water Quality. Information Sciences 178 (20) pp. 3867-3879
 21. Rummelhart D.E., Hinton G.E., Williams, R.J., (1985) Learning Internal Representations by Error Propagation. Institute for Cognitive Science Report 8506. San Diego, University of California
 22. Rummelhart D.E., Hinton G.E., Williams R.J., (1986) Learning Internal Representations by Error Propagation. Parallel Distributed Processing, Explorations in the Microstructure of Cognition. Vol 1, Foundations. Cambridge, MA: MIT Press.
 23. A. Tazioli, (2009) Evaluation of erosion in equipped basins preliminary results of a comparison between the Gavrilovic model and direct measurements of sediment transport. Environmental Geology. Volume 56, number 5, pp. 825-831, Springer Berlin
 24. Thamarai S., Malmathanraj R., (2005) Missile Defense and Interceptor Allocation by Modified Bionet Neural Network. Proceedings of the 9th International Conference of EANN, Lille France pp. 299-306.
 25. UNESCO (1998-2000). <http://www.unesco-ihe.org/hi/projects.htm>
 26. Van Looy S., Meeus J., Wyns B., Vander Cruyssen B., Boullart L., Keyser F. (2005) Comparison of Machine Learning models for prediction of dose increase in patients with rheumatoid arthritis. Proceedings of the 9th International Conference of Engineering Applications of Neural Networks, Lille France pp. 189-196.
 27. Vapnik V.N., (1995) The nature of statistical learning theory. New York: Springer Verlag
 28. Vapnik V.N., S. Golowich, and A. Smola, (1997) Support Vector method for function approximation regression estimation and signal processing. In Advances in neural information processing systems. Vol.9. Cambridge, MA:MIT Press.
 29. Vapnik V.N., (1998) Statistical learning Theory. New York Wiley.