

# **Experimental investigation of the predictive capabilities of soft computing techniques in hydrology**

By

A. Elshorbagy<sup>1</sup>, G. Corzo<sup>2</sup>, S. Srinivasulu<sup>1</sup>, and D. Solomatine<sup>2</sup>

<sup>1</sup>Centre for Advanced Numerical Simulation (CANSIM), Department of Civil & Geological Engineering, University of Saskatchewan, Saskatoon, SK, Canada S7N 5A9

<sup>2</sup>Department of Hydroinformatics and Knowledge Management, UNESCO-IHE, Delft, the Netherlands

CANSIM SERIES REPORT NO. CAN-09-01



**Centre for Advanced Numerical Simulation (CANSIM)**  
Department of Civil & Geological Engineering,  
University of Saskatchewan, Saskatoon, SK, CANADA  
June, 2009

## SUMMARY

In this report, an extensive data-driven modeling experiment is proposed. The most important concerns regarding the way soft computing techniques are handled, compared, and evaluated and the basis on which findings and conclusions were drawn are discussed. A concise review of key articles that presented comparisons among various soft computing modeling techniques is summarized. Six soft computing modeling techniques, namely, neural networks, genetic programming, evolutionary polynomial regression, support vector machines, M5 model trees, and K-nearest neighbors are proposed and explained. Multiple linear regression and naïve models are also suggested as baseline for comparison with the various techniques. Five datasets from Canada and Europe representing evapotranspiration, upper and lower soil moisture content, and rainfall-runoff process are described and proposed for the modeling experiment. Twelve different realizations (groups) from each dataset are created by random sampling. Each group contains three subsets; training, cross-validation, and testing. Each modeling technique is proposed to be applied to each of the 12 groups of each dataset. This way, both predictive accuracy and uncertainty of the modeling techniques can be evaluated. The soft computing modeling experiment is implemented. Inputs for the five case studies (half-hourly actual evapotranspiration, daily peat soil moisture, daily till soil moisture, and two daily rainfall-runoff datasets) are identified, either based on previous studies or using the mutual information content. Twelve groups (realizations) were randomly generated from each dataset by randomly sampling without replacement from the original dataset. Neural networks (ANNs), genetic programming (GP), evolutionary polynomial regression (EPR), Support vector machines (SVM), M5 model trees (M5), K nearest neighbors (K-nn), and multiple linear regression (MLR) techniques are implemented and applied to each of the 12 realizations of each case study. The predictive accuracy and uncertainties of the various techniques are assessed using multiple average overall error measures, scatter plots, frequency distribution of model residuals, and the deterioration of prediction performance during the testing phase. Gamma test is used as a guide to assist in selecting the appropriate modeling technique. Unlike the two nonlinear soil moisture case studies, the results of the experiment conducted in this research study show that ANNs were a sub-optimal choice for the actual evapotranspiration and the two rainfall-runoff case studies. GP is the most successful technique due to its ability to adapt the model complexity to the modeled data. EPR performance could be close to GP with datasets that are more linear than nonlinear. SVM is sensitive to the kernel choice and if appropriately selected, the performance of SVM can improve. M5 performs very well with linear and semi linear data, which cover wide range of hydrological situations. In highly nonlinear case studies, ANNs, K-nn, and GP could be more successful than other modeling techniques. K-nn is also successful in linear situations, and it should not be ignored as a potential modeling technique for hydrological applications.

## 1. INTRODUCTION

Data driven modeling techniques in general, and soft computing techniques in particular, have been in use for nearly two decades for hydrological modeling, prediction, and forecast. Many articles reporting the application of various techniques to various hydrological case studies are available in literature. Yet, data driven techniques are still facing some classical opposition because of multiple reasons inherent in such techniques (e.g., lack of transparency and difficulty of reproducing the results). Hydroinformatics researchers started to identify problems of data driven modeling (Maier and Dandy, 2000; Elshorbagy and Parasuraman, 2008) and tried to suggest some solutions or modeling guidelines and frameworks. Cherkassky et al. (2006) have listed the quality of the datasets, choosing robust learning methods that can handle heterogeneous data, and the need for uncertainty estimates associated with predictions as some of the main issues and challenges facing computational intelligence in earth sciences.

There is no doubt that more scientific rigour should have been maintained in the applications and use of data driven techniques in earth sciences. Abrahart et al. (2008) have used the example of neural network applications to highlight the shortcomings of the present approach, and how to build stronger foundations. Apparently, their argument can be easily generalized to apply to other data driven and soft computing techniques. In fact, the modeling shortcomings and ambiguity inherent in soft computing techniques are less than the ones created because of the way such techniques were presented in earth sciences literature. One of the fundamental means to assess a modeling technique is to evaluate it against other modeling techniques, whether conceptual or data driven modeling techniques. One can observe that in the literature of soft computing or data driven hydrology, the modeling comparative studies are usually impaired due to the less-than-comprehensive approach adopted. With few exceptions, the following problems can be noticed: (i) Only one or two modeling techniques have been used at a time in a single study; (ii) if more techniques were employed, then only one or two datasets were used for the applications. This leads to conclusions that are based on the unique characteristics of such dataset (Abrahart et al., 2008); (iii) Datasets were split into two subsets for training and testing, where the testing data were the models were tested iteratively using the testing data subset. This makes the testing data seen, even if not used, during training. In this case, the generalization ability of the developed model is questionable; and (iv) when datasets were correctly split into three subsets for training, cross-validation, and testing, only one random realization of the three subsets was used. Such use of a single realization of the dataset makes it difficult to assess the predictive uncertainty and the effect of the split approach on the adopted models.

The above-mentioned deficiencies, in addition to other requirements identified by Abrahart et al. (2008) including the need for testing the models over a range of conditions, the reasoning behind the data splitting, and the need for designing repeatable experiments and reproducible findings, are the motives behind this study. The aim of this study is to evaluate and test the predictive abilities of six soft computing modeling techniques on five different case studies of rainfall-runoff, evapotranspiration, and soil moisture content. Multiple random realizations of the three subsets of each dataset will be created and used with each and every modeling technique. The techniques will be evaluated against multiple linear regression models and,

when applicable, naïve models. Both predictive accuracy and uncertainty will be evaluated. The authors intend to make all datasets used in this study available for all interested researchers to test the results and conduct further studies. The authors hope and aim that this study could serve as a benchmark study for assessing future proposed modeling, optimization, and input processing methods or techniques.

This study is presented in two companion papers. This first part consists of, after this introduction, a section that briefly summarizes some of the key comparative studies in hydrology literature, followed by a section explaining the study methodology and the experimental set up. The fourth section describes the modeling techniques adopted in this study as well as the implementation tools. The fifth section contains a description of study sites, the collected data, and how five different case studies (datasets) representing various hydrological processes were created from three sites. The last section of this first part is a general summary. The second part begins with an introduction section that explains how the methodology was applied and how inputs for the various case studies were selected. The second section reports on the implementation details and parameter values, when applicable, of each modeling techniques for the various datasets. Results of the various techniques and analysis are presented in the third section. A general discussion and guidelines are presented in section 4, whereas the conclusions and findings of the entire study are presented in the last section.

## **2. Comparative hydrological modeling studies using soft computing techniques**

The number of studies that reported some sort of comparison between various soft computing modeling techniques in hydrology is very large, and it is beyond the possibility of being summarized here. However, some key and representative studies are presented here. Solomatine and Siek (2006) presented an algorithm, which facilitates incorporation of domain knowledge into one particular type of modular model (model tree). They employed the M5flex algorithm to two hourly and daily rainfall-runoff datasets as well as five widely used benchmark datasets—Autompg, Bodyfat, CPU, Friedman, and Housing (Blake & Mertz, 1998). They compared the M5flex method with global ANNs and other local M5 modeling methods (M5o, M5opt). They concluded that M5flex delivered high performance because of the use of additional domain knowledge for determining the best split attributes and values. Solomatine and Xue (2004) showed that both M5 model tree technique and ANNs perform similarly for flood forecasting problem in the upper reach of the Huai River in China, but the model trees have certain advantage in terms of transparency in the model structure over ANNs.

Sivapragasm et al. (2007) found that there is no significant difference in the prediction accuracy between GP and ANNs for forecast of daily flows, but GP has an advantage of identifying the optimum inputs. Makkeasorn et al. (2008) compared between genetic programming (GP) and ANN models for forecasting river discharges. The findings indicated that GP-derived streamflow forecasting models were generally favored for forecasting over ANNs. Further, the most forward looking GP-derived models can even perform a 30-day streamflow forecast ahead of time with a reasonable estimation. Jayawardena et al. (2005)

compared the GP technique in modeling rainfall-runoff process to the traditional modelling approaches. They used the GP technique to predict the runoff from three catchments in Hong Kong and two catchments in southern China, and showed that the GP technique evolved simple models that enabled the quantification of the significance of different input variables for prediction. Parasuraman et al. (2007) used two hourly evapotranspiration (ET) datasets to compare between GP and ANNs for prediction of ET. Not much difference was found, with regard to the prediction accuracy, between the two techniques.

Wu et al. (2007) developed distributed SVR (D-SVR) model with two step Genetic Algorithm parameter optimization method to carry out prediction of water level in a river. The D-SVR method desegregates the couple of subsets from original training set and then generates a local SVR for each subset independently. Wu et al. (2007) evaluated the performance of D-SVR against the predictions from linear regression (LR), nearest neighbor (NN) method, and genetic algorithm-based ANN (ANN-GA) methods. The proposed D-SVR model can predict the water level better in comparison with the other models. However LR model performed better in comparison with NN, ANN-GA models, which was attributed to highly linear mapping relation between input and output variables that restricts the power of NN and ANN. In their study, Lin et al. (2006) employed an SVM model to predict long-term flow discharges in Manwan Hydropower scheme in Tibet. It was found through comparison of results with ARMA and ANN models that the SVM model can provide more accurate predictions of long term flow discharges. Further, Lin et al. (2006) concluded that SVM has its distinct capabilities and advantages in identifying hydrological time series comprising nonlinear characteristics. In their preliminary study, Çimen (2008) applied SVMs for the estimation of suspended sediment concentration/load. The observed streamflow and suspended sediment data of two rivers in the USA, which have been already used in earlier studies using ANNs, were considered. It was found that the negative sediment estimates, which were encountered using ANNs, did not happen during the application of SVMs. Khan and Coulibaly (2006) examined the application of the SVM and successfully demonstrated the mean monthly lake water level prediction up to 12 months ahead. SVM was found to be more advantageous than ANNs, which prescribes more number of controlling parameters. Khan and Coulibaly (2006) deduced that SVM proved to be more competitive and promising compared to the widely used ANNs and conventional seasonal multiplicative autoregressive (SAR) models. Behzad et al. (2008) compared SVM with ANN and ANN-GA models for prediction of daily runoff of Bakhtiyari River watershed in Iran. They considered available climate information as model inputs. They concluded that the prediction accuracy of SVM was at least as good as that of ANN and ANN-GA models in some cases, and better in some other cases. Furthermore, Behzad et al. (2008) found that SVM converges considerably faster compared to other models. Wu et al. (2008) demonstrated the feasibility of SVM for forecasting of soil water content in Purple hilly area located in Southwest University in Chongqing. They compared the predictions from SVM with ANNs, and showed that the results from the SVM predictor significantly outperformed the other baseline predictors such as ANNs.

Giustolisi and Savic (2006) found that EPR was more accurate than GP for extracting a symbolic expression for Chezy resistance coefficient. Elshorbagy and El-Baroudy (2009) differentiated between equation-based GP and program-based GP. They further compared GP with EPR technique using a highly nonlinear dataset (soil moisture content). It was found that

program-based GP outperformed EPR in its prediction accuracy. More importantly, Elshorbagy and El-Baroudy (2009) demonstrated the need for adopting multiple data driven modeling techniques and tools (modeling environments) to obtain reliable predictions. This brief literature review shows that findings and conclusions were sometimes seemingly contradictory. Apparently such findings should be viewed as data-specific, and thus, lacks generality and strong support for cause-effect relationships.

### **3. Methodology and experimental setup**

In order to achieve the objectives of this paper with regard to the comparative predictive performance of various soft computing techniques, first, a set of distinctive modeling techniques were identified. The selected techniques are (i) artificial neural networks (ANNs); (ii) genetic programming (GP); (iii) evolutionary polynomial regression (EPR); (iv) support vector machines (SVM); (v) M5 model trees; and (vi) K-nearest neighbors (K-nn). To facilitate the comparison and allow for performance evaluation in light of easily understandable and widely recognized techniques, multiple linear regression (MLR) models and/or naïve models were employed as base line references.

Second, five different case studies representing different hydrological processes or variables (actual evapotranspiration, soil moisture content, and rainfall-runoff) were selected. The datasets present a wide range of challenges to data driven techniques because of their various levels of complexity, embedded feedback mechanism, and nonlinearity. The datasets will be explained in more details in a later section of this paper. Third, for each dataset, model inputs were either identified in this research or were pre-selected based on previous studies. Even though appropriate model inputs were secured for this study, the identification of the optimum inputs was not given an extraordinary emphasis since the focus of this research is inter-technique comparison. As long as the inputs are the same for the various modeling techniques, an unbiased analysis can be conducted toward achieving the objectives of this study.

Fourth, split samples from each dataset were prepared for the modeling experiment. Each set of the five datasets was randomly sampled 100 times without replacement, such that every time the dataset is split into three distinct subsets: training, which contains one half of the total data instances; cross-validation, which contains one sixth of the data instances, and testing, which contains one third of the data instances. Twelve different groups (three subsets each) out of the 100 groups were selected based on the statistical properties of the output variable (e.g., runoff). The aim was to select the samples where the mean and the standard deviation values of the three subsets (training, cross validation, and testing) are similar or, at least, the differences are minima. The cross-validation subset was used for stopping the model training and selecting the best model, whereas the testing subset is kept completely unseen during the training process. Twelve different models were developed based on the 12 data groups (the best model based on cross-validation was picked every time), and each model was tested using the corresponding testing subset. These procedures were repeated using the six different data driven modeling techniques, applied to each of the five different datasets. The results of

this experiment allows for investigating ensemble outputs from each modeling techniques, average and range of possible prediction accuracy, and predictive uncertainty.

Fifth, the predictive accuracy of the various models and techniques were evaluated using the root mean squared error (RMSE), the mean absolute relative error (MARE), the mean bias (MB), and the correlation coefficient (R). The authors believe that these four error statistics, along with the visual comparison between observed and predicted values, are sufficient to reveal any significant differences among the various modeling techniques with regard to their predictive accuracy. The formulae of the error measures are presented in Equations 1-4 below.

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (O_i - P_i)^2}{N}} \quad (1)$$

$$MARE = \frac{1}{N} \sum_{i=1}^N \left| \frac{O_i - P_i}{O_i} \right| \quad (2)$$

$$MB = \frac{1}{N} \sum_{i=1}^N (O_i - P_i) \quad (3)$$

$$R = \frac{\sum_{i=1}^N (O_i - \bar{O}_i)(P_i - \bar{P}_i)}{\sqrt{\sum_{i=1}^N (O_i - \bar{O}_i)^2 \sum_{i=1}^N (P_i - \bar{P}_i)^2}} \quad (4)$$

Where  $N$  represents the number of instances presented to the model;  $O_i$  and  $P_i$  represent observed and predicted counterparts; and  $\bar{O}$  and  $\bar{P}$  represent the mean of the corresponding variables. However, sometimes conflicting results regarding the performance of various models may arise due to the use of various error measures (Dawson et al., 2007; Elshorbagy et al., 2000). In this study, a supplemental error measure that combines the effects of the four error measures in one indicator is proposed. The new indicator, called the ideal point error (IPE) is based on identifying the ideal point in the four dimensional space that each model aims to reach. The coordinates of the ideal point should be: (RMSE=0.0; MARE=0.0; MB=0.0; R=1.0). The IPE (Equation 5) measures how far the model is from the ideal point. All individual error measures are given equal relative weights, and all are normalized using the maximum error so that the final IPE value for each model ranges between 0.0 for the best model and 1.0 for the worst model.

$$IPE = \left\{ 0.25 \left[ \left( \frac{RMSE_{ij} - 0.0}{\max RMSE_{ij}} \right)^2 + \left( \frac{MARE_{ij} - 0.0}{\max MARE_{ij}} \right)^2 + \left| \frac{MB_{ij} - 0.0}{\max |MB_{ij}|} \right|^2 + \left( \frac{R_{ij} - 1.0}{1/\max R_{ij}} \right)^2 \right] \right\}^{1/2} \quad (5)$$

Where  $i$  denotes model ( $i$ ) and  $j$  denotes technique ( $j$ ).

Sixth, the predictive uncertainty of the models was assessed using the model residuals ( $r$  values), where  $r_i$  is the difference between the observed and the predicted values. For each dataset and each modeling technique, the residuals are computed for all 12 models representing the range of possible residuals. The residuals of the 12 models are merged in one set of presumably random variable, and a probability distribution was fit to this variable.

Seventh, the gamma test was conducted to assist in gaining some insight into the predictability of the output variables using nonlinear smooth functions, and possibly some leads into the process of selecting appropriate modeling techniques for a particular case study. The main idea of the gamma test ( $\Gamma$ -test) is estimating the variance of the noise on the output variable, which could be an estimate of the best mean squared error that a smooth model can achieve for the corresponding output. The test was implemented using *winGamma* (Jones et al., 2001) that assumes that non-determinism in a smooth model from inputs to outputs is due to the presence of statistical noise on the outputs:

$$y = f(X_1, \dots, X_m) + \epsilon \quad (6)$$

Where  $f$  is a smooth function and  $\epsilon$  is noise, and that the variance of the noise  $\text{Var}(\epsilon)$  is bounded. The  $\Gamma$ -test is based on  $L[i, k]$ , which are  $k$  nearest neighbors  $X_{L[i, k]}$  ( $1 \leq k \leq p$ ) for each vector  $X_i$  ( $1 \leq i \leq N$ ) (Stefánsson et al., 1997). Delta ( $\delta$ ) and  $\gamma$  functions can be defined as follows:

$$\delta_N(k) = \frac{1}{N} \sum_{i=1}^N |X_{L(i,k)} - X_i|^2 \quad (1 \leq k \leq p) \quad (7)$$

$$\gamma_N(k) = \frac{1}{2N} \sum_{i=1}^N |y_{L(i,k)} - y_i|^2 \quad (1 \leq k \leq p) \quad (8)$$

Where  $y_{L(i, k)}$  is the corresponding output value for the  $k$  nearest neighbors of  $X_i$  in Equation (7) (Stefánsson et al., 1997). A least squares regression line can be constructed for the  $p$  points ( $\delta_N(k)$ ,  $\gamma_N(k)$ ) where  $\Gamma$  can be computed:

$$\gamma = A\delta + \Gamma \quad (9)$$

The intercept on the vertical axis is the  $\Gamma$  value (Jones et al., 2001). As  $\delta_N(k)$  approaches zero,  $\gamma_N(k)$  approaches  $\text{Var}(\epsilon)$  in probability. In addition to  $\Gamma$ , three other useful statistics can be calculated: (i) the *gradient*, which is the slope of the regression line that indicates the complexity of the system (steeper gradient indicates greater complexity) (Evans and Jones, 2002), (ii) the *V-ratio*, which is a scale invariant noise estimate where  $\Gamma$  is divided by the variance of the output variable. A *V-ratio* close to zero indicates high degree of predictability of the output variable, and (iii) the M-test, which is the size of data that is possibly required to produce a stable asymptote of  $\Gamma$ . The  $\Gamma$  value might be estimated for scaled or unscaled dataset, but it the *gradient* will be more informative if estimated based on scaled dataset. In general, if the inputs have inconsistent units, it is advisable to conduct the  $\Gamma$ -test using the scaled data (Jones et al., 2001).

## 4. The modeling techniques and tools

### 4.1 Artificial neural networks (ANNs)

ANNs are a method of computation and information processing motivated by the functional units of the human brain, namely neurons. Since abundant information on ANNs is available in literature (e.g., Haykin, 1999; ASCE Task Committee on Application of Artificial Neural Networks in Hydrology, 2000), the description of ANNs herein is brief, and limited to the needs of this study. According to Haykin (1999), a neural network is a massively parallel distributed information processing system that is capable of storing the experiential

knowledge gained by the process of learning, and of making it available for future use. Mathematically, ANNs are universal approximators with an ability to solve large-scale complex problems such as time series forecasting, pattern recognition, nonlinear modeling, classification, and control. This is achieved by identifying the relationships among given patterns.

Feedforward neural networks (FFNNs) are the most widely adopted network architecture for the prediction and forecasting of hydrological variables (*Maier and Dandy, 2000*). Typically, FFNNs consist of three layers: input layer, hidden layer, and output layer. The number of nodes in the input layer corresponds to the number of inputs considered for modeling the output. The input layer is connected to the hidden layer with weights that determine the strength of the connections. The number of nodes in the hidden layer(s) indicates the complexity of the problem being modeled. The hidden layer nodes consist of the activation function, which helps in nonlinearly transforming the inputs into an alternative space where the training samples are linearly separable (*Brown and Harris, 1994*). Detailed review of ANNs and their application in hydrology can be found in *Maier and Dandy (2000)* and in *ASCE Task Committee on Application of Artificial Neural Networks in Hydrology (2000)*.

The FFNNs adopted in this study make use of the tan-sigmoidal activation function in the hidden layer and the linear activation function in the output layer. While the tan-sigmoidal activation function squashes the input between -1 and 1, the linear activation function calculates the neurons output by simply returning the value passed to it. One of the important issues in the development of neural networks model is the determination of optimal number of hidden neurons that can satisfactorily capture the nonlinear relationship existing between the input variables and the output. The number of neurons in the hidden layer is usually determined by trial-and-error method with the objective of minimizing the cost function (*ASCE Task Committee on Application of Artificial Neural Networks in Hydrology, 2000*). Levenberg-Marquardt back propagation algorithm is used for training the FFNNs in this study.

#### *4.2 Genetic programming (GP)*

Genetic Programming (GP), introduced by Koza (1992), is an evolutionary algorithm based on the concepts of natural selection and genetics. GP extends the search of genetic algorithms for optimal set of parameters search to include the model space, so that both the model structure and the associated model parameters can be optimized simultaneously. Genetic symbolic regression (GSR) is a special application of GP in the area of symbolic regression, where the objective is to find a mathematical expression in symbolic form, which provides an optimal fit between a finite sample of values of the independent variable and its associated values of the dependent variable (Koza, 1992). GSR can be considered as an extension of numerical regression problems, where the objective is to find the set of numerical coefficients that best fits a predefined model structure (linear, quadratic, or polynomial). Nevertheless, GSR does not require the functional form to be defined a priori, as GSR involves finding the optimal mathematical expression in symbolic form (both the discovery of the correct functional form and the appropriate numerical coefficients) that defines the predictand-predictor relationship. GSR is sometimes referred to as equation-based GP. Another form of GP is program-based GP, where the explicit equation may not be necessarily produced, but

rather a program (code) is the final output. Elshorbagy and El-Baroudy (2009) noted that program-based GP can be more efficient than equation-based GP with regard to its prediction accuracy. GPLAB (Silva, 2005), a GP toolbox for MATLAB that provides the evolved equation in the form of a parse tree is an example of an equation-based GP tool, whereas Discipulus (Francone, 2001), used in this study, is an example of program-based GP tools.

Genetic Programming (GP) is a widely used machine learning (ML) technique; it uses a tree-like structure, as decision trees, to represent its concepts and its interpreter as a computer program. Therefore, it is considered a superset of all other ML representations; this may enable GP to produce any solution that is produced by any other ML system (Banzhaf et al. 1998). It uses different genetic operators such as crossover and mutation, together with beam search to reach candidate solutions from the overall population of solutions. Although GP is computationally intensive, like most soft-computing techniques, and has its own limitations. The major problem is the deterioration of the prediction ability of the developed model with longer prediction horizon, which is a common problem in any modeling method. The adverse consequences of this problem can be mitigated by combining GP technique with knowledge-based techniques that depend on the accumulated knowledge of the process under consideration. This will enhance the quality of the developed models and add to the understanding of the complicated hydrological processes (Babovic and Keijzer, 2002).

Several applications of the GP technique in hydrology exist in the literature. Parasuranam et al. (2007a) explored the utility of GP to develop explicit models for eddy covariance-measured actual evapotranspiration. Babovic and Keijzer (2002) addressed the utility of GP in developing rainfall-runoff models on the basis of hydro-meteorological data, as well as in combination with other conventional models, i.e. conceptual models. It was reported that the GP models gave more insights into the functional relationships between different input variables resulting in more robust models. Parasuraman et al. (2007b) used GP to evolve pedotransfer functions (PTFs) for estimating the saturated hydraulic conductivity ( $K_s$ ) from soil texture (sand, silt, and clay) and the bulk density. Similarly, Jayawardena et al. (2005) compared the GP technique in modeling rainfall-runoff process to the traditional modelling approaches. They used the GP technique to predict the runoff from three catchments in Hong Kong and two catchments in southern China, and showed that the GP technique evolved simple models that enabled the quantification of the significance of different input variables for prediction. In literature, there was an emphasis on GP's ability to produce explicit equations, but in this research program-based GP is employed to utilize the full predictive ability of the technique.

For GP implementation, the first step is to define the functional and terminal sets, along with the objective function and the genetic operators. The functional set and the terminal set are the main building blocks of GP, and hence, their appropriate identification is central in developing a robust GP model. The functional set consists of basic mathematical operators  $\{+, -, *, /, \sin, \exp, \dots\}$  that may be used to form the model. The choice of the operators considered in the functional set depends upon the degree of complexity of the problem to be modeled. The terminal set consists of independent variables and constants. The constants can either be physical constants (e.g. Earth's gravitational acceleration, specific gravity of fluid) or randomly generated constants. Different combinations of functional and terminal sets are

used to construct a population of mathematical models (or programs). Each model (individual) in the population can be considered as a potential solution to the problem. Genetic operators include crossover and mutation, and they are discussed in detail later in this section. Once the functional and terminal sets are defined, the next step is to generate the initial population for a given population size. The initial population can be generated in a multitude of ways, including, the full method, grow method, and ramped half-and-half method. The ramped half-and-half method is a combination of the full and the grow methods. For each depth level considered, half of the individuals are initialized using the full method and the other half using the grow method. The ramped half-and-half method is shown to produce highly diverse trees, both in terms of size and shape (Koza, 1992), and thereby provides a good coverage of the search space. More information on the different methods of generating the initial population can be found in Koza (1992). Once initialized, the fitness of each individual (mathematical model) in the population is evaluated based on the selected objective function. The better the fitness of an individual, the greater is the chance of the individual breeding into the next generation. In this study, root mean squared error is used as the objective function, and a lower value of RMSE indicates better fitness. At each generation, new sets of models are evolved by applying the genetic operators: crossover and mutation (Koza, 1992; Babovic and Keijzer, 2000). These new models are termed offspring, and they form the basis for the next generation.

After the fitness of the individual models in the population is evaluated, the next step is to carry out selection. The objective of the selection process is to create a temporary population called the mating pool, which can be acted upon by genetic operators: crossover and mutation. Selection can be carried out by several methods like truncation selection, tournament selection, and roulette wheel selection. As roulette wheel selection is one of the most commonly used methods including Koza (1992), it has been adopted in this study. Roulette wheel is constructed by proportioning the space in a roulette wheel based on the fitness of each model in the population. The selection process ensures that the models with better fitness have more chance of breeding into the next generation. Crossover is carried out by initially choosing two parent models from the mating pool, and selecting random crossover points for each of the parents. Based on the selected crossover points, the corresponding sub-tree structures are swapped between the parents to produce two different offspring with different characteristics. The number of models undergoing crossover depends upon the chosen probability of crossover ( $P_c$ ). Mutation involves random alteration of the parse tree at the branch or node level. This alteration is done based on the probability of mutation ( $P_m$ ). For an overview of different types of computational mutations, readers are referred to Babovic and Keijzer (2000). While the role of the crossover operator is to generate new models, which did not exist in the old population, the mutation operator guards the search against premature convergence by constantly introducing new genetic material into the population.

#### *4.3 Evolutionary polynomial regression (EPR)*

Evolutionary Polynomial Regression (EPR) is another data driven and soft computing technique that models time series or regression-type data containing information about physical processes (Giustolisi and Savic 2006). EPR combines the power of evolutionary algorithms with numerical regression to develop polynomial models combining the

independent variables together with the user-defined function as follows (Lauccelli et al. 2005):

$$\hat{Y} = \sum_{i=1}^m F(X, f(x), a_i) + a_o \quad (10)$$

where  $\hat{Y}$  is the EPR-estimated dependent variable,  $F(.)$  is the polynomial function constructed by EPR,  $X$  is the independent variables' matrix,  $f(.)$  is a user-defined function,  $a_i$  is the coefficient of the  $i$ -th term in the polynomial,  $a_o$  is the bias and  $m$  is the total number of the polynomial terms. Inclusion of the user-defined function is provided to enhance the characterization of the response (dependant) variable. As the developers of the EPR tool state "EPR is a two-stage technique for constructing symbolic models: (i) structure identification; and (ii) parameter estimation", where it uses genetic algorithm (GA) simple search method to search in the model structure space. EPR uses the least squares (LS) method to estimate the parameters of the selected model structure based on the performed GA search. Applications of EPR are found in Savic et al. (2006), Doglioni et al. (2007), Elshorbagy and el-Baroudy (2009), and Giustolisi et al. (2007). The search proceeds by using the standard GA operators, crossover and mutation; noting that this type of search is not exhaustive as it is practically impossible to conduct such search on an infinite search space (Lauccelli et al. 2005). Even though EPR might be viewed as a subset of GP, its reported good performance while emphasizing the polynomial structure makes it a potential candidate for this study.

This study makes use of the EPR toolbox (Lauccelli et al. 2005), which is based on "homonymous modeling methodology based on a hybrid evolutionary paradigm". It is a multi-objective implementation of EPR in the sense that it produces several models, which are the best trade-off, considering fitness to training data vs. parsimony. The EPR tool performs three types of regression, i.e. dynamic, static, and classification. Dynamic modeling is used to model systems that have memory, or in other words, when the present state of the system depends on the previous states of other input variables. On the other hand, static systems are systems that are not influenced by the previous states of input variables. Classification modeling is a special type of static modeling in which the model output is an integer (Lauccelli et al. 2005). The readers may refer to the user manual for the details of the EPR toolbox and the different components of its simple interface (Lauccelli et al. 2005).

#### 4.4 Support vector machine (SVM)

Support vector (SV) algorithm is a nonlinear generalization of the Generalized Portrait algorithm (Cherkassky and Mulier, 2007). In regression and time series prediction applications, excellent performances were obtained (Muller et al., 1997; Mattera and Haykin, 1999). The goal of  $\epsilon$ -SV regression (Vapnik, 1995) is to find a function  $f(x)$  that has at most  $\epsilon$  deviation from the actually obtained targets  $y_i$  for all the training data, and at the same time, is as flat as possible. In case of linear functions  $f$ ,

$$f(x) = \langle w, x \rangle + b \quad (11)$$

Where  $\langle \cdot, \cdot \rangle$  denotes the dot product in  $X$ . Flatness in this case means seeking small  $w$ , which can be ensured by minimizing the Euclidean norm, i.e.,  $\|w\|^2$ . Sometimes, it is not possible to

approximate all pairs  $(x_i, y_i)$  with  $\varepsilon$  precision. So, it is possible to allow for some errors in the form of slack variables  $\zeta_i, \zeta_i^*$ . The problem can be written as a convex optimization problem: minimize

$$\frac{1}{2}\|w\|^2 + C\sum_{i=1}^l(\zeta_i + \zeta_i^*)$$

subject to

$$\begin{cases} y_i - \langle w, x_i \rangle - b \leq \varepsilon + \zeta_i \\ \langle w, x_i \rangle + b - y_i \leq \varepsilon + \zeta_i^* \\ \zeta_i, \zeta_i^* \geq 0 \end{cases} \quad (12)$$

The constant  $C > 0$  determines the tradeoff between the flatness of  $f$  and the amount up to which deviations larger than  $\varepsilon$  are tolerated. A Lagrange function from both the objective function and the corresponding constraints can be constructed by introducing a dual set of variables (Smola and Schölkopf, 1998):

$$\begin{aligned} L := & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^l(\zeta_i + \zeta_i^*) - \sum_{i=1}^l\alpha_i(\varepsilon + \zeta_i - y_i + \langle w, x_i \rangle + b) \\ & - \sum_{i=1}^l\alpha_i^*(\varepsilon + \zeta_i^* + y_i - \langle w, x_i \rangle - b) - \sum_{i=1}^l(\eta_i\zeta_i + \eta_i^*\zeta_i^*) \end{aligned} \quad (13)$$

where  $\alpha_i, \alpha_i^*, \eta_i, \eta_i^* \geq 0$ . Finally,  $w$  can be written as follows:

$$w = \sum_{i=1}^l(\alpha_i - \alpha_i^*)x_i \text{ and therefore } f(x) = (\alpha_i - \alpha_i^*)\langle x_i, x \rangle + b \quad (14)$$

This is called Support Vector expansion, i.e.  $w$  can be completely described as a linear combination of the training patterns  $x_i$ . The above discussion is based only on linear SVM regression. For nonlinear regression, the SVM has a great advantage that can represent the nonlinear function in an arbitrary number of dimensions efficiently through a defined Kernel. The idea is to map the training input vector  $x_i$  into a higher dimensional space (called feature space) or hyperplane, by the function  $\Phi$ , while the regression for  $x$  remains linear. Thus, the procedure is the same as the linear SVM except changing the dot product  $\langle x_i, x \rangle$  by  $\langle \Phi(x_i), \Phi(x) \rangle$ . The Kernel function:  $K(x_i, x) = \langle \Phi(x_i), \Phi(x) \rangle$  can assume any form. Many Kernels are being proposed by researchers; however, the most common ones are:

Linear Kernel:  $K(x_i, x) = \langle x_i, x \rangle$

Polynomial Kernel:  $K(x_i, x) = (\gamma\langle x_i, x \rangle + \tau)^d, \quad \gamma > 0$

Radial basis function Kernel:  $K(x_i, x) = \exp(-\gamma\|x_i - x\|^2), \quad \gamma > 0$

Sigmoid Kernel:  $K(x_i, x) = \tanh(\gamma\langle x_i, x \rangle + \tau), \quad \gamma > 0$

Where  $\gamma, \tau$ , and  $d$  are Kernel parameters.

In this study, the SVM was implemented within WEKA 3.6.0 Software (Bouckaert et al. 2008; Witten and Frank, 2005). Many machine learning and data driven techniques, including SVM, neural networks, model trees, and instance-based learning, can be implemented using the user friendly WEKA Software.

#### *4.5 Model Trees*

#### *4.6 K-nearest neighbors*

The K-nearest neighbors (K-nn) technique is one of the simplest forms of instance-based learning, which is plain memorization (Witten and Frank, 2005). Once a set of training instances has been memorized, one encountering a new (testing) instance, the memory is searched for the training instance that most closely resembles the testing instance. Instead of creating rules (or continuous function approximation surface), K-nn technique works directly from the examples themselves. Each new instance is compared with existing ones using a distance metric, and the closest existing distance is used to assign the output to the new instance. Usually, more than one nearest neighbors is used. Standard Euclidean distance (or any other distance measure) is used as a metric to represent “resemblance”. When multiple nearest neighbors are employed, the output of the testing instance can be based either on simple average, weighted average, or any more sophisticated function. In this study, the simplest method, which is the average value of the K-nearest neighbors, is used. An apparent drawback to instance-based representation is that it does not make explicit the structures that are learnt. Instances do not really describe the patterns in data. Karlsson and Yakowitz (1987); Parasuraman and Elshorbagy (2007); and Solomatine et al. (2008) presented some hydrological prediction case studies using K-nn technique.

## **5. Datasets**

### *5.1 Actual evapotranspiration*

The eddy covariance (EC)-measured actual evapotranspiration data from the South West Sand Storage (SWSS) facility, located near Ft. McMurray, Alberta, Canada, is considered in this study. The SWSS is currently the largest operational tailings dam in the world, holding approximately 435 million cubic meters of material, covering 25 km<sup>2</sup>, and standing approximately 40 m high with a 20H:1V side-slope ratio. Soils consist of mine tailings sand overlain with 0.4 to 0.8 m of topsoil that is a mixture of peat and secondary mineral soil with a clay loam texture. Both vegetation species and composition vary across the SWSS, with dominant groundcover including horsetail (*Equisetum arvense*), fireweed (*Epilobium angustifolia*), sow thistle (*Sonchus arvensis*), and white and yellow sweet clover (*Melilotus alba*, *Melilotus officinalis*). Tree and shrub species include Siberian larch (*Larix siberica*), hybrid poplar (*Populus sp. hybrid*), trembling aspen (*Populus tremuloides*), white spruce (*Picea glauca*), and willow (*Salix sp.*). For the SWSS facility, the ground-water table is located well below the rooting zone, at a depth between 0.8-1.0 m, and hence do not directly contribute to the evapotranspiration process. Accurate estimation of actual evapotranspiration from the reconstructed watersheds is of vital importance as it plays a major role in the water-balance of the system, which links directly to ecosystem restoration strategies. The weather station located on top of the SWSS facility measured the air temperature (AT) (°C), ground

temperature (GT) ( $^{\circ}\text{C}$ ), net radiation (NR) ( $\text{W}/\text{m}^2$ ), relative humidity (RH), and wind speed (WS) ( $\text{m}/\text{s}$ ). Turbulent fluxes of heat and water vapor were measured using a CSAT3 sonic anemometer and thermometer (Campbell Scientific) and an LI-7500  $\text{CO}_2/\text{H}_2\text{O}$  gas analyzer (Li-Cor). Ground heat flux was measured using a CM3 radiation and energy balance (REBS) ground heat flux plate placed at 0.05 m depth. In EC technique, the covariance of vertical wind speed with temperature and water vapor is used to estimate the sensible heat (H) and latent heat (LE) fluxes (Parasuraman and Elshorbagy, 2008). More information on the EC technique can be found in Drexler et al. (2004). Raw turbulence measurements were made at 10 Hz and fluxes were calculated using 30-minute block averages with a 2-D coordinate rotation.

The half hourly EC-measured LE flux (the product of the latent heat of vaporization and evapotranspiration) at the SWSS facility for two growing seasons (from May 3 to Sept 21, 2005 and from May 27 to Sept 9, 2006) is considered in this study. The total precipitation during the two seasons is 275 mm and 265 mm, respectively and the average day-time reference evaporation rate is 0.27 mm/hr. Nevertheless for modeling purposes, the day time (08:00hrs. – 20:00hrs.) evapotranspiration alone is considered. After eliminating records of missing data, the remaining number of data instances were 5,307 data points. Since evapotranspiration is commonly perceived as being highly dependent on climatic variables, the EC-measured LE flux is modeled as a function of NR, AT, GT, RH, and WS, as well as possible combinations of these variables. The descriptive statistics of the datasets used for training, cross validation, and testing are presented in Table 1. The coefficient of variation (CV) of different variables during training, cross validation, and testing are comparable.

### *5.2 Soil moisture content*

Over the years, several large scale soil cover (reconstructed watersheds) experiments are being conducted to assess the performance of different reclamation strategies in northern Alberta, Canada, by studying the basic mechanisms that control the moisture movement within these covers. In particular, three experimental soil covers (D1, D2, and D3) were established in the year 1999. The experimental covers were constructed over the saline-sodic overburden with thickness of 0.50 m, 0.35 m, and 1.0 m, comprising a thin layer of peat mineral mix over varying thickness of secondary (glacial/till) soil. Cover D1 consists of 20 cm of peat overlying 30 cm of till, and it is considered for this study. The soil cover has an area of 1 ha (approximately 200 m long and 50 m wide), with a 5:1 slope (5 horizontal to 1 vertical). This reconstructed watershed, compared to natural watersheds, is not stable during their initial stages, and hence evolves over time to achieve hydro-sustainability. In order to track the evolution (hydrological changes) of the watershed, intensive instrumentations were installed in the watershed. Each watershed has an individual soil station located at the middle of the slope, which measures the volumetric soil moisture content of the upper peat (SMP) and the lower till (SMT) layers, twice a day. Soil moisture is measured using TDR principles with model CS615 (Boese, 2003). The TDR sensors were installed laterally into the soil profile. Watershed D1 has eight TDR sensors installed over a depth range of 0.05 m to 1.00 m. Hourly values of soil temperature of peat (STP) and till (STT) layers are measured using thermistors buried in the watershed at the depth ranges corresponding to the TDR sensors. Consequently, D1 has eight soil temperature sensors. A weather station located in the mid-slope measures air temperature (AT), and precipitation (P). Similarly, Bowen station located

at the mid-slope measures net-radiation (NR) and energy fluxes. All the meteorological variables are measured in an hourly scale. More details on the field instrumentation program and the data collected can be found in Boese (2003) and Elshorbagy et al. (2007).

Table 1. Descriptive statistics of the AET dataset

	<b>NR</b> W/m <sup>2</sup>	<b>AT</b> °C	<b>GT</b> °C	<b>RH</b>	<b>WS</b> m/s	<b>LE</b> W/m <sup>2</sup>
<b>Training dataset</b>						
<b>Minimum</b>	-189.6	-3.4	4.1	0.14	0.4	-80.2
<b>Maximum</b>	875.4	33.9	27.2	0.96	10.2	503.8
<b>Mean</b>	229.7	18.7	16.7	0.5	2.8	144.9
<b>SD</b>	189.4	5.5	3.8	0.2	1.7	90.0
<b>CV</b>	0.82	0.29	0.23	0.34	0.62	0.62
<b>Cross validation dataset</b>						
<b>Minimum</b>	-119.8	-3.2	3.7	0.16	0.4	-42.2
<b>Maximum</b>	729.5	33.7	26.4	0.95	11	405.6
<b>Mean</b>	224.1	18.7	16.9	0.5	2.8	145.9
<b>SD</b>	181.9	5.6	3.8	0.2	1.7	88.7
<b>CV</b>	0.81	0.30	0.23	0.33	0.60	0.61
<b>Testing dataset</b>						
<b>Minimum</b>	-414.6	-4.3	3.3	0.15	0.4	-56.3
<b>Maximum</b>	801.6	33.8	27.2	0.96	12.3	425.8
<b>Mean</b>	226.9	18.5	16.6	0.5	2.9	143.8
<b>SD</b>	188.9	5.5	3.7	0.2	1.8	89.9
<b>CV</b>	0.83	0.30	0.22	0.34	0.63	0.63

Average daily values of precipitation, air temperature, soil temperature (STP and STT), net radiation (NR), soil moisture (SMP and SMT) as well as possible combinations of them, are considered for modeling purposes. The ground temperature and soil moisture contents are depth averaged for each layer (upper peat and lower till). As the soil stratum is frozen during the winter, only summer (May-September) time data of years 2000 till 2006 are considered. The total number of instances available for modeling purposes was 972 data points. As the reconstructed watersheds evolve over time to achieve hydro-sustainability, the freeze-thaw cycles and decomposition of highly organic peat layer increases the porosity of the soil and consequently increasing infiltration rates (Haigh, 2000). Hence, modeling the moisture dynamics of such evolving watersheds would be adding to the already challenging task of modeling soil moisture. The descriptive statistics of the datasets used for training, cross validation, and testing are presented in Table 2 for the peat and the till layer datasets, respectively. For modeling purposes, two datasets were generated from the site; one for predicting SMP and the other for SMT. The same set of inputs was used in both datasets. The coefficient of variation (CV) of different variables during training, cross validation, and testing are comparable (Table 2).

### 5.3 Rainfall-runoff

The rainfall-runoff dataset used in this study is taken from the Ourthe subcatchment, which is a subcatchment of River Meuse. The greater part of the discharge of the River Meuse is

supplied by its tributaries. Groundwater, precipitation and artificial extractions constitute the discharge to a smaller extent. The Meuse has a great number of tributaries, varying greatly in their sizes. The largest is the Ourthe, with a contributing area of 3,626 km<sup>2</sup>. The Ourthe subcatchment has great discharges rising fast. Through its nature and situation, close to the Dutch border, the Ourthe is also the most important Meuse tributary for flood forecasts. In its upper course, the Ourthe consists of two branches: the Ourthe Occidentale and the Ourthe Orientale, merging near Nisramont. Near Comblain-au-Pont, the Amblève joins the Ourthe and near Angleur the Ourthe also receives the Vesdre. Measuring from the source of the Ourthe Occidentale, the Ourthe is approximately 175 km long. The daily rainfall and runoff data of the Ourthe subcatchment from January 11, 1988 till December 31, 1998 (4,008 data points) were used for modeling purposes in this study. Two distinct datasets were created: (i) the first is a dataset where only rainfall data were used as model inputs to predict the runoff; and (ii) the second is the same dataset but the preceding time step (t-1) runoff, in addition to the rainfall data, were used as inputs to predict the runoff at time t. The descriptive statistics of the variables that are used as model outputs in this study are presented in Table 3.

Table 2. Descriptive statistics of the daily peat and till moisture datasets.

	<b>P</b> mm	<b>AT</b> °C	<b>NR</b> W/m <sup>2</sup>	<b>STP</b> °C	<b>STT</b> °C	<b>SMP</b>	<b>SMT</b>
<b>Training dataset</b>							
<b>Minimum</b>	0.00	-6.30	-10.40	0.50	-0.50	0.304	0.240
<b>Maximum</b>	43.70	25.20	204.40	18.20	16.30	0.539	0.316
<b>Mean</b>	1.54	13.63	90.64	11.71	10.48	0.442	0.288
<b>SD</b>	4.20	6.10	50.22	3.79	3.49	0.055	0.018
<b>CV</b>	2.72	0.45	0.55	0.32	0.33	0.124	0.062
<b>Cross validation dataset</b>							
<b>Minimum</b>	0.00	-3.90	0.00	0.50	-0.70	0.305	0.241
<b>Maximum</b>	27.18	22.90	226.10	18.20	16.10	0.542	0.316
<b>Mean</b>	1.68	13.80	92.96	11.75	10.32	0.440	0.289
<b>SD</b>	3.99	4.96	49.98	4.03	4.17	0.055	0.018
<b>CV</b>	2.38	0.36	0.54	0.34	0.40	0.125	0.062
<b>Testing dataset</b>							
<b>Minimum</b>	0.00	-6.80	0.00	-0.10	-0.60	0.306	0.241
<b>Maximum</b>	23.60	25.80	223.60	18.20	16.10	0.543	0.316
<b>Mean</b>	1.48	14.07	96.94	11.88	10.45	0.440	0.288
<b>SD</b>	3.32	5.96	50.91	3.77	3.56	0.054	0.018
<b>CV</b>	2.25	0.42	0.53	0.32	0.34	0.123	0.061

Table 3. Descriptive statistics of the output variables of all datasets.

	Evapotranspiration	Peat moisture	Till moisture	Runoff
Count	5307	972	972	4008
Minimum	-80.20	0.30	0.24	1.07
Median	133.09	0.45	0.29	11.39
Average	144.52	0.44	0.29	21.91
Maximum	503.77	0.54	0.32	370.63
St. deviation	89.79	0.05	0.02	29.93
CV	0.62	0.12	0.06	1.37
Skew	0.51	-0.72	-1.33	4.06

## 6. Model implementation

The research methodology explained in the first part of this two-companion paper was implemented in the sequence presented earlier. First, inputs of the various models were identified. A mixed approach of input selection was adopted since identification of optimum inputs was not in itself one of the objectives of this study. The two soil moisture datasets (Elshorbagy and Parasuraman, 2008) and a reduced hourly version of the evapotranspiration (AET) dataset (Parasuraman and Elshorbagy, 2008; Parasuraman et al., 2007) were used in earlier studies. This study benefited from the input structure identified in the earlier studies, and sometimes (e.g., the case of the evapotranspiration dataset) enhanced the input structure by considering more inputs identified using the mutual information content. Figure 1 presents the inputs identified for the AET case study using AMI method. For the two rainfall-runoff datasets, the AMI method was used to identify the inputs for predicting the daily runoff (Fig. 2). The inputs-output of the five case studies are presented in Table 4. One should note that in light of the focus of this study, which is the comparative analysis of various data driven techniques, the important criterion is to use the same set of inputs across all adopted models.

After inputs have been identified, each dataset was randomly sampled 100 times; creating 100 realizations of the dataset with three split samples (training, cross-validation, and testing) created from every dataset realization. Figure 3 shows an example of this process for the peat moisture dataset. Similar process was conducted with each one of the five case studies. Based on the similarity of the statistical properties (mean and standard deviation) of the three split samples, the best 12 realizations of each dataset are identified for the modeling exercise in this study.

### 6.1 Artificial neural networks (ANNs)

The Levenberg-Marquardt algorithm was used for training all neural network models using the MATLAB toolbox. For each realization of the 12 dataset realizations of a case study, the ANN was executed 200 times with 200 different random weight initializations. The best model of the 200 runs was identified as the best ANN model. The cross validation sub dataset was used to stop the training process. This process was repeated for each of the 12 dataset realization of each case study. Accordingly, 12 non-dominated ANN models were developed and tested using the corresponding unseen dataset. In all optimum ANN models, the number of input nodes was equivalent to the number of inputs, and all networks had one output node.

The number of hidden nodes ranged from three to 13, with an average number of seven hidden nodes in single hidden layer ANNs.

Table 4 Inputs and outputs of all case studies.

Case study	Inputs	Output
Actual evapotranspiration (half hourly)	$AT_t$ ; $GT_t$ ; $GT_{t-1}$ ; $NR_t$ ; $NR_{t-1}$ ; $Sum(NR_{-4})$ ; $RH_t$ ; $WS_t$	$AET$ ( $W/m^2$ )
Upper layer (peat) soil moisture content (daily)	$P_t$ ; $AT_t$ ; $NR_t$ ; $STP_t$ ; $STT_t$ ; $Sum(P_{-6})$ ; $Sum(AT_{-6})$	$SM_P$ (dimensionless)
Lower layer (till) soil moisture content (daily)	$P_t$ ; $AT_t$ ; $NR_t$ ; $STP_t$ ; $STT_t$ ; $Sum(P_{-6})$ ; $Sum(AT_{-6})$	$SM_T$ (dimensionless)
Rainfall-runoff I (daily)	$P_t$ ; $P_{t-1}$ ; $P_{t-2}$ ; $P_{t-3}$ ; $P_{t-4}$	$Q_{II}$ ( $m^3/s$ )
Rainfall-runoff II (daily)	$P_t$ ; $P_{t-1}$ ; $P_{t-2}$ ; $P_{t-3}$ ; $P_{t-4}$ ; $Q_{t-1}$	$Q_{III}$ ( $m^3/s$ )

$AT$ : air temperature ( $^{\circ}C$ );  $GT$ : ground temperature ( $^{\circ}C$ );  $NR$ : net radiation ( $W/m^2$ );  $Sum(NR_{-4})$ : the cumulative net radiation over the preceding four time steps;  $RH$ : relative humidity;  $WS$ : wind speed ( $m/s$ );  $P$ : precipitation ( $mm$ );  $STP$ : depth averaged soil temperature of the upper peat layer ( $^{\circ}C$ );  $STT$ : depth averaged soil temperature of the lower till layer ( $^{\circ}C$ );  $Sum(P_{-6})$ : the cumulative precipitation over the preceding six time steps ( $mm$ );  $Sum(AT_{-6})$ : the cumulative air temperature over the preceding six time steps ( $^{\circ}C$ );  $SM_P$ : depth averaged soil moisture content of the upper peat layer;  $SM_T$ : depth averaged soil moisture content of the lower till layer; and  $Q_t$ : the runoff ( $m^3/s$ ).

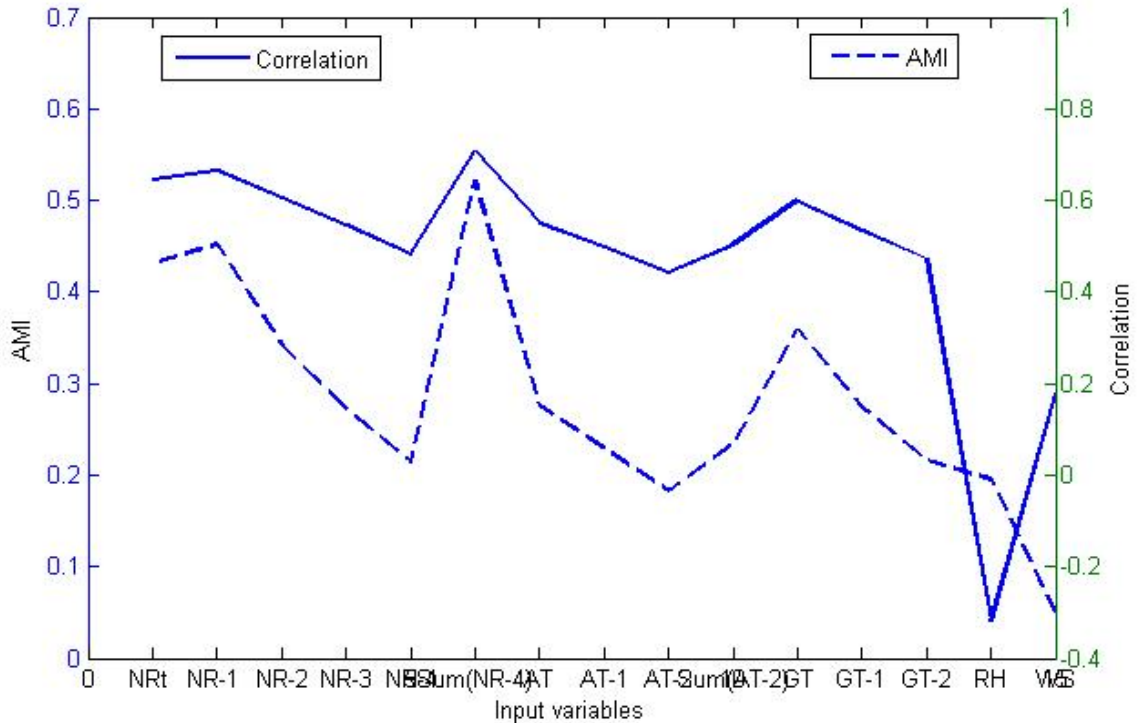


Figure 1 Average mutual information and correlation of inputs-output for the evapotranspiration case study

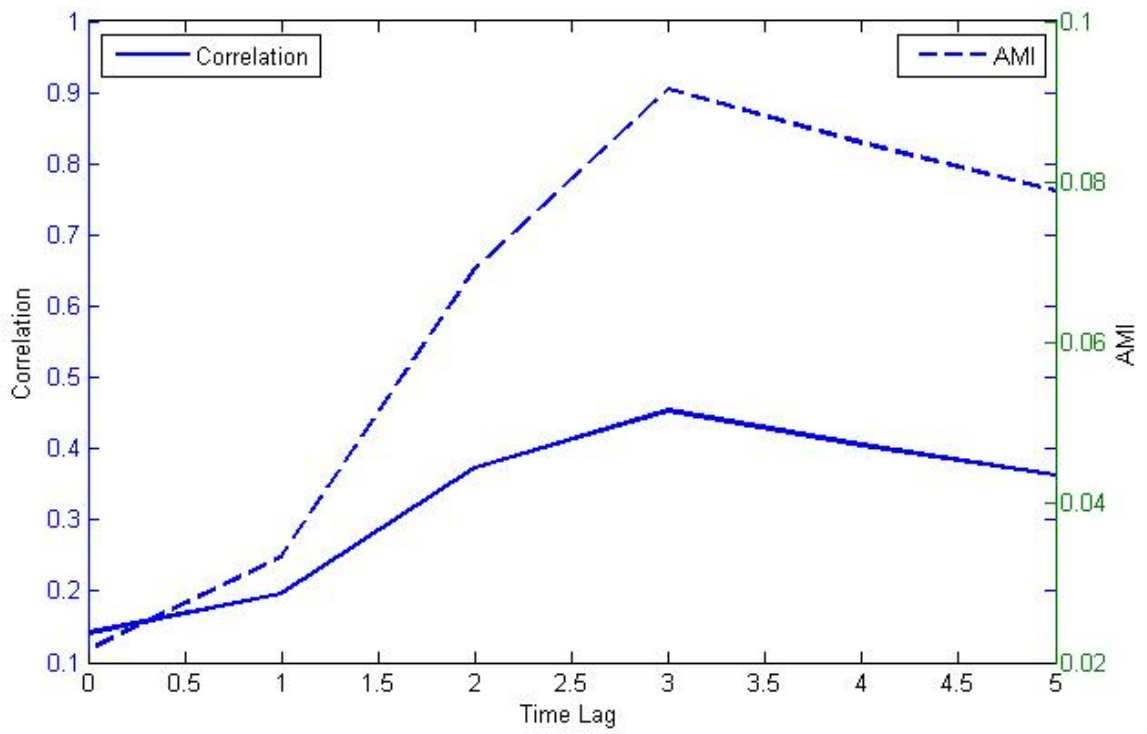


Figure 2 Average mutual information and correlation of inputs-output for the rainfall-runoff case study

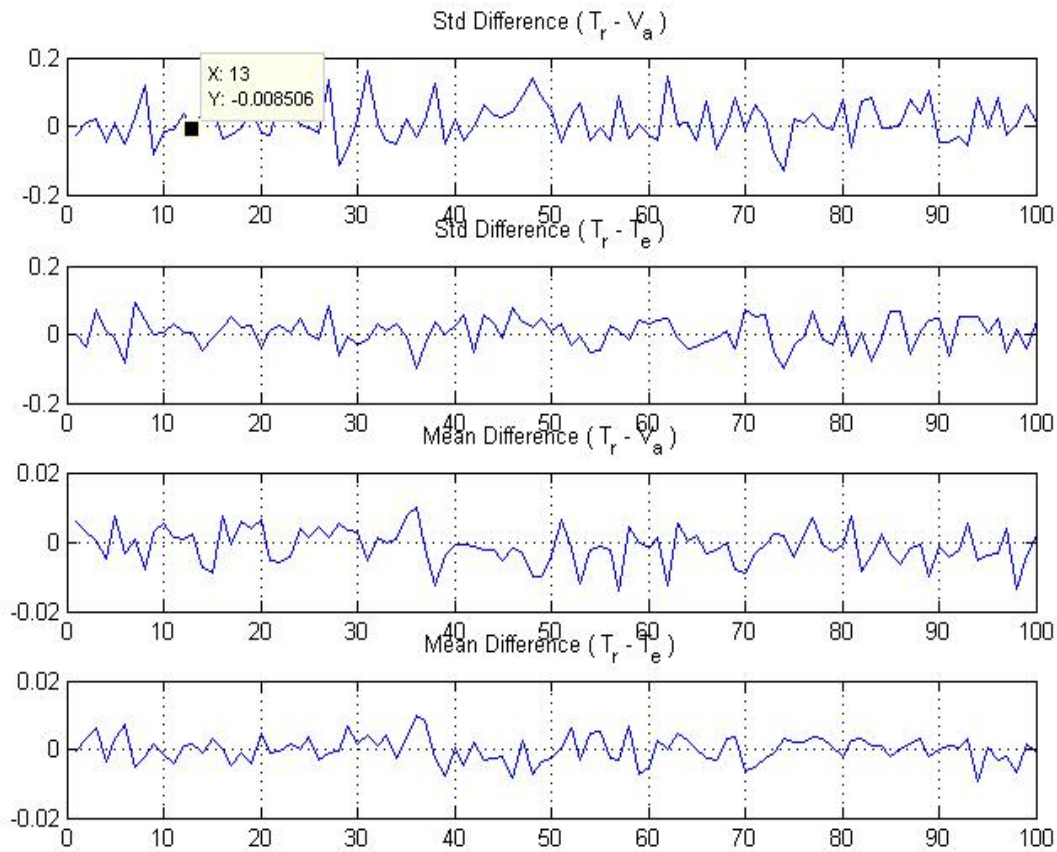


Figure 3 Statistical properties of the training/cross-validation/testing subsets for 100 random realizations

### 6.2 Genetic programming (GP)

Discipulus Software (Francone, 2001) was used to implement the program-based GP to all datasets. GP was applied to the various dataset realizations similar to the way followed with ANNs. The addition, subtraction, multiplication, comparison, conditions, division, and trigonometric operators were allowed. The program size varied from 80-512 bits, with population size of 500 and generations without improvement up to 300. The probabilities of mutation and crossover were 30% and 50%, respectively. The program was allowed to run for at least two hours. The authors experimented with the run time and observed that improvement could be almost negligible beyond two hours. Similar to the case of ANN applications, 12 non-dominated GP models were developed and tested on the corresponding testing set of each case study.

### 6.3 Evolutionary polynomial regression (EPR)

The EPR Toolbox (Laucelli et al., 2005) was used to implement the static EPR technique to all datasets, following the same experimental steps adopted with the ANNs and the GP techniques. The EPR Toolbox allows for many choices in terms of the polynomial types, functions used within the polynomial terms, and the number of terms and exponents. In this study, the default number of terms (up to five) was used whereas a comprehensive search

among the possible combinations of polynomial types and functions was conducted. Accordingly, 12 non-dominated EPR models were developed and tested on the corresponding testing set of each case study. The EPR type and function developed for each case study are presented in Table 5.

Table 5. EPR type and functions of all case studies.

Case study	EPR type	Function (f)
Actual evapotranspiration (half hourly)	Sum $[a_i * X1 * X2 * f(X1 * X2)] + a_o$	No function
Upper layer (peat) soil moisture content (daily)	Sum $[a_i * f(X1 * X2)] + a_o$	Exponential
Lower layer (till) soil moisture content (daily)	Sum $[a_i * f(X1 * X2)] + a_o$	Logarithm
Rainfall-runoff I (daily)	Sum $[a_i * X1 * X2 * f(X1) * f(X2)] + a_o$	No function
Rainfall-runoff II (daily)	Sum $[a_i * X1 * X2 * f(X1) * f(X2)] + a_o$	No function

#### 6.4 Support vector machine (SVM)

WEKA 3.6.0 Software (Bouckaert et al., 2008) was used in this study to implement the SVM to all datasets, following the same experimental steps adopted with the previous techniques. SVM models with linear, polynomial, and radial basis function (RBF) kernels were tested on all datasets. With the exception of the Rainfall –runoff II case study, the RBF kernel was found to provide the best predictive performance. In case of the rainfall-runoff II case study, both linear and RBF kernels were almost on par. Therefore, SVM with RBF kernel was adopted in this study. The constant C (Elshorbagy et al., part I) and the kernel parameter  $\gamma$  were optimized from an exponential range of the following values: 0.0313; 0.0625; 0.125; 0.25; 0.50; 1.00; 2.00; 4.00; 8.00; and 16.00. Non-dominated 12 SVM models were developed and tested on the corresponding testing set of each case study.

#### 6.5 M5 Model Trees

WEKA 3.6.0 Software (Bouckaert et al., 2008) was used in this study to implement the M5 model trees to all datasets, following the same experimental steps adopted with the previous techniques. The tree pruning coefficient was optimized during the execution of the models to minimize the average squared error. A range of values from 3-30 was tested in this study. 12 M5 model tree models were developed and tested on the corresponding testing set of each case study.

#### 6.6 K-nearest neighbors (K-nn)

WEKA 3.6.0 Software (Bouckaert et al., 2008) was used in this study to implement the K-nn technique to all datasets, following the same experimental steps adopted with the previous techniques. The number of the nearest neighbors was optimized during the execution of the models to minimize the average squared error. A range of values from 1-50 neighbors was tested in this study. Accordingly, 12 K-nn models were developed and tested on the corresponding testing set of each case study. The ranges of the optimum numbers of nearest neighbors for each case study are presented in Table 6.

Table 6 The optimum number of nearest neighbors (K-nn) of the 12 models in each case study.

	All 12 values	Min.	Average	Max.
evapotranspiration	17- 28- 10- 21- 21- 34- 22- 18- 26- 9- 15- 40	9	22	40
Upper layer soil moisture	4- 4- 9- 5- 4- 3- 5- 5- 12- 7- 4- 4	3	6	12
Lower layer soil moisture	9- 4- 2- 2- 8- 10- 7- 3- 5- 6- 9- 6	2	6	10
Rainfall-runoff I	19- 33- 9- 11- 3- 18- 8- 24- 44- 12- 6- 13	3	17	44
Rainfall-runoff II	2- 7- 3- 4- 1- 2- 2- 3- 6- 3- 3- 5	1	3	7

## 7. Results and analysis

### 7.1 Evapotranspiration case study

The performance of the various techniques applied to the half-hourly actual evapotranspiration (AET) case study is provided in Table 7. The best, the worst, and the average of the performances of the 12 models of all techniques are shown. It is certainly useful to judge techniques based on the range of performances (difference between the best and the worst models), however, if a single value is needed, then one has to rely on the average performance. Table 7 supports the idea that in most cases, it is not possible to find a technique that dominates others with respect to all error measures. But if a technique is better than the rest with respect to two different error measures (e.g., RMSE and R), this can be a strong indication of the superiority of such a technique. In the AET case study, GP, SVM, M5 model trees, and K-nn techniques can be identified as the best techniques, followed by EPR, in terms of the predictive accuracy. The performance of the ANNs was worse than the linear regression (MLR) technique in this particular case study. This highlights an important fact that the half-hourly AET data were captured reasonably well in a linear relationship considering the provided model inputs. Therefore, a technique that forces highly nonlinear structures on the data (ANNs) may not be favorable in all cases. Certainly, the AET data are not strictly linear; that is why local and/or modular linear models (e.g., M5 and K-nn) could be optimum choices.

Table 7. Testing results of all models applied to the **evapotranspiration** dataset

Models	RMSE			MARE			MB			R		
	Best	Ave	Worst	Best	Ave	Worst	Best	Ave	Worst	Best	Ave	Worst
ANNs	46	57	86	0.52	1.25	2.25	-1.5	5.9	58	0.87	0.84	0.74
GP	42	<b>44</b>	46	0.58	0.69	0.84	-0.1	0.27	1.65	0.88	<b>0.87</b>	0.86
EPR	45	46	48	0.62	0.82	1.07	0.01	0.9	3.1	0.87	0.86	0.85
SVM	42	<b>45</b>	49	0.48	<b>0.54</b>	0.64	-1.26	-2.8	-4.9	0.84	<b>0.87</b>	0.88
M5	43	<b>44</b>	46	0.53	0.63	0.72	0.17	<b>-0.03</b>	1.97	0.86	<b>0.87</b>	0.88
K-nn	43	<b>45</b>	46	0.58	0.69	0.80	0.09	-0.39	-2.16	0.88	<b>0.87</b>	0.86
MLR	47	49	50	0.78	0.93	1.13	-0.15	0.14	2.8	0.85	0.84	0.83
Naïve	-	-	-	-	-	-	-	-	-	-	-	-

Since all 12 models of each technique are non-dominated models and represent possible performances of the technique under consideration, the output of all 12 models are integrated in one set and presented in Figure 4. The figure shows the scatter plots of observed vs. predicted AET data. The scatter around the 45-degree line supports the conclusion made earlier regarding the performances of the various techniques. However, the plots reveal two additional observations; first, all techniques were less successful in predicting high values. The tips of the data plumes were always below the 45-degree line. This might be an indication that the ideal inputs that can describe all dynamics of the process for this case study have not been optimally identified. The SVM (Fig. 4d) was more successful than other techniques in approaching the high values. The M5 model trees and MLR (Fig. 4e; 4g) were the least successful in this regard. Table 8 shows the ideal point error (IPE) measure calculated for all techniques. The IPE statistic, integrating all four error measures in one indicator, lends another support to the conclusions made earlier. Except the ANNs, all other techniques have close performances, with the possibility of identifying the SVM, GP, M5, and K-nn; followed by EPR as better techniques than the rest. The utility of the idea of adopting multiple models (12 in this study) based on different random realizations of the datasets to evaluate various techniques presents itself through Tables 7 and 8. If the modeler picks, for example, the best model of one technique and compares it with the worst model of another technique, a different and perhaps biased conclusion might be made regarding the performance of these techniques. The best ANN model with IPE value of 0.31 is much better than the worst EPR model with IPE value of 0.37 (Table 8).

Based on the outputs of the 12 non-dominated models of each technique, the predictive uncertainty of the various techniques can be easily analyzed. The residuals (predicted value minus observed value) of the 12 models were integrated in one set to conduct probabilistic analysis. Frequency curves were constructed for the residuals of each technique. @RISK Software (Palisade Corporation, 2005) was used to fit the best probability distribution from a selection of more than 15 possible distributions. The best-found probability distributions of the residuals of the various techniques are shown in Figure 5. The Logistic ( $\alpha$ ,  $\beta$ ) distribution was found to fit the residuals of all modeling techniques, with different values of location parameter  $\alpha$  and scale parameter  $\beta$ . Ideally, the best technique is the one that has residuals represented by the narrowest, symmetrical, and tallest (has the highest probability value at zero residuals) probability distribution. Such a distribution implies the smallest level of predictive uncertainty, which could be translated to the highest level of reliability. Figure 5 reveals that, not only in terms of the predictive accuracy, but also the predictive uncertainty of the SVM, GP, K-nn, and M5, followed by EPR is smaller, and thus more reliable, than the other techniques. Clearly, the ANN technique is the most uncertain with the widest range of residuals, whereas the MLR is occupying the middle position.

The Kolmogorov-Smirnov (KS) nonparametric test was conducted on the model residuals of all techniques to test the null hypothesis that the model residuals of any two techniques are

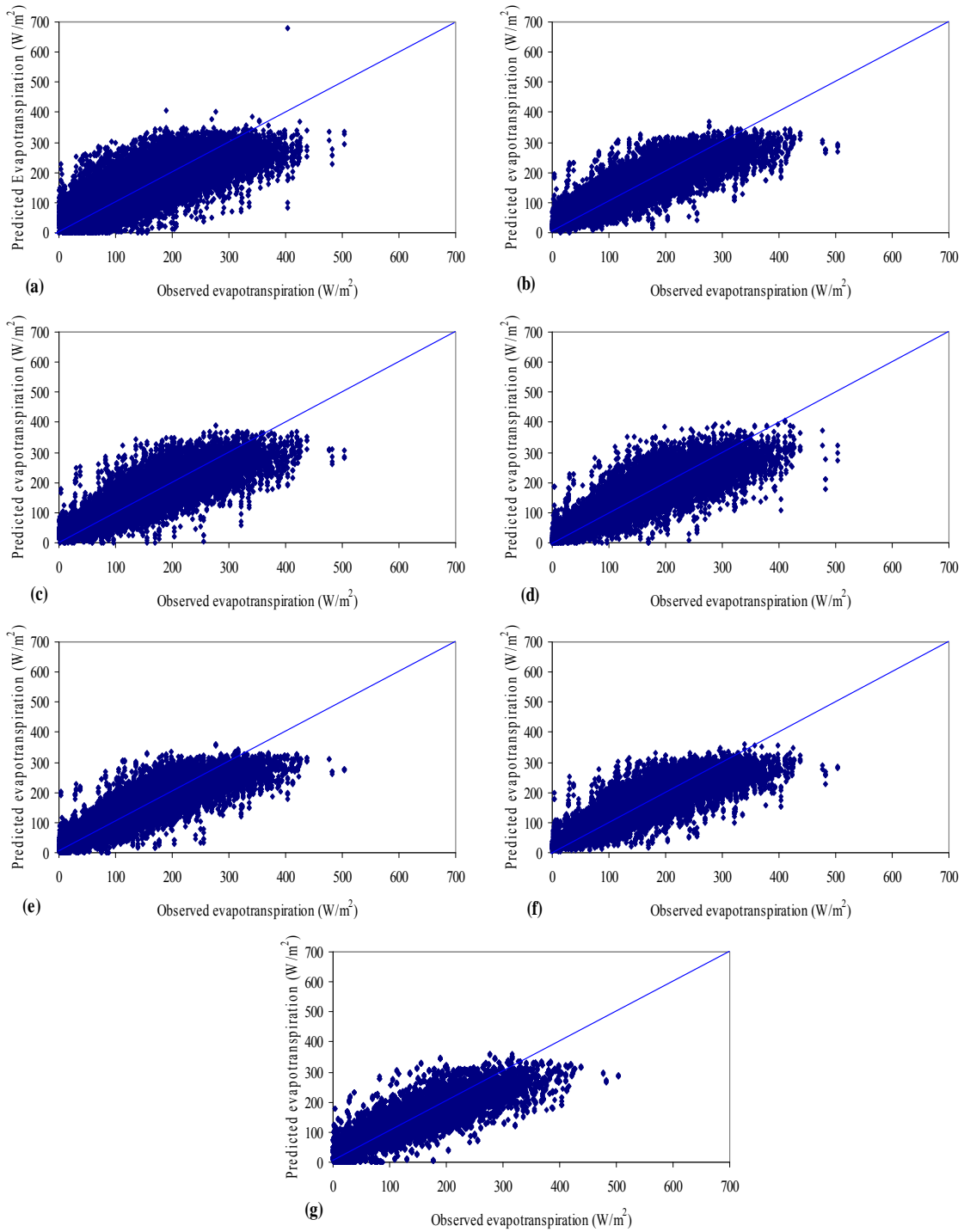


Figure 4 scatter plots of observed and predicted evapotranspiration. (a) ANNs, (b) GP, (c) EPR, (d) SVM, (e) M5, (f) K-nn, and (g) MLR.

sampled from the same distribution. The test was conducted at the default significance level of  $p = 0.05$ . The matrix of the p-values is given as Table 9. With the exception of K-nn and M5 techniques, there is strong statistical evidence that the residuals of the various techniques are stemming from different distributions. There are also no correlations found among the probability distributions of the residuals of the various modeling techniques. Even though the

Table 8. IPE testing results of all models applied to all datasets.

	Evapotranspiration (AET)			Peat moisture (SMP)			Till moisture (SMT)			Rainfall-runoff I (R-R I)			Rainfall-runoff II (R-R II)		
	Best	Ave	Worst	Best	Ave	Worst	Best	Ave	Worst	Best	Ave	Worst	Best	Ave	Worst
ANNs	0.31	0.51	0.79	0.58	0.65	0.71	0.49	0.57	0.92	0.51	0.57	0.69	0.24	0.47	0.78
GP	0.29	0.30	0.33	0.56	0.63	0.72	0.43	0.53	0.67	0.50	0.55	0.58	0.17	0.20	0.22
EPR	0.31	0.33	0.37	0.65	0.68	0.72	0.55	0.58	0.63	0.52	0.56	0.68	0.19	0.22	0.28
SVM	0.28	0.29	0.32	0.65	0.80	0.90	0.55	0.60	0.69	0.52	0.57	0.62	0.24	0.37	0.54
M5	0.29	0.30	0.31	0.57	0.64	0.74	0.49	0.56	0.63	0.50	0.52	0.53	0.18	0.20	0.22
K-nn	0.29	0.31	0.32	0.57	0.65	0.71	0.44	0.51	0.52	0.52	0.54	0.57	0.55	0.59	0.67
MLR	0.34	0.36	0.39	0.72	0.74	0.78	0.57	0.60	0.63	0.51	0.53	0.55	0.44	0.48	0.52
Naïve	-	-	-	-	-	-	-	-	-	-	-	-	0.32	0.35	0.42

Table 9 the p-values of the two samples K-S test on the model residuals (evapotranspiration).

	ANNs	GP	EPR	SVM	M5	<b>K-nn</b>	MLR
ANNs	1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
GP		1	0.0001	0.0000	0.0000	0.0000	0.0000
EPR			1	0.0000	0.0000	0.0000	0.0000
SVM				1	0.0000	0.0000	0.0000
<b>M5</b>					1	<b>0.051</b>	0.0000
K-nn						1	0.0000
MLR							1

visual assessment of Figure 5 shows that the SVM, M5, and K-nn are very similar, the KS test indicates that the SVM performs differently.

### 7.2 Peat (upper layer) soil moisture case study

The performance of the various techniques applied to the daily soil moisture data of the upper peat layer (SMP) case study is provided in Table 10. Unlike the evapotranspiration case study, Table 10 shows that both ANNs and GP techniques can be considered superior to other modeling techniques due to their domination with respect to the four error measures. It has to be noted that in case of soil moisture content, low values of the RMSE and the MARE might be misleading because the entire dataset is limited to a narrow range (0.30-0.55) of values (Table 3). In this case, the R statistic becomes the most important indicator (Elshorbagy and Parasuraman, 2008). For example, if a naïve model is constructed just by assuming that the best predictor is the average soil moisture value of all observations in the training dataset, the predicted value will be always 0.442. In this case, the RMSE and the MARE values are 0.05 and 0.10, respectively, but the R statistic value is almost zero; indicating an extremely poor model. Accordingly, ANNs and GP are the best modeling techniques for this case study (R values of 0.60 and 0.61, respectively), followed by the K-nn and the M5 techniques. The MLR is clearly dominated by other techniques, which points to the possibility that the SMP dataset is a highly nonlinear dataset. The authors believe that this is a major reason for the relative success of ANNs in this case study compared to the previous (AET) case study. The moisture storage effect (Elshorbagy and El-Baroudy, 2009; Elshorbagy and Parasuraman, 2008) attributes to the nonlinearity of the process. Techniques that can handle highly nonlinear data (ANNs and GP) were quite successful, followed closely by local/modular models (M5 and K-nn). Even though the EPR technique was relatively close to the K-nn and M5, the performance of the SVM technique was the poorest with an R value of 0.44; slightly higher than the MLR.

The scatter plots (Figure 6) show clearly that the error measures, including the IPE (Table 8), reflect only the average overall performance of the models, and favor models that produce scatter with less dispersion (e.g., GP and EPR). However, the plots reveal that ANNs outperforms other techniques where, at least, the trend of the higher range of peat moisture values was captured better than other techniques. Similar to the AET case study, frequency curves were constructed for the residuals of each technique (Figure 7). Interestingly, the best-found probability distributions of the residuals of the various techniques differed. The LogLogistic ( $\gamma, \beta, \alpha$ ) probability distribution was found to fit the residuals of SVM and K-nn, and M5 modeling techniques, Logistic ( $\alpha, \beta$ ) for ANNs, Lognormal ( $\mu, \sigma$ ) for GP, Beta ( $\alpha, 1$ ,

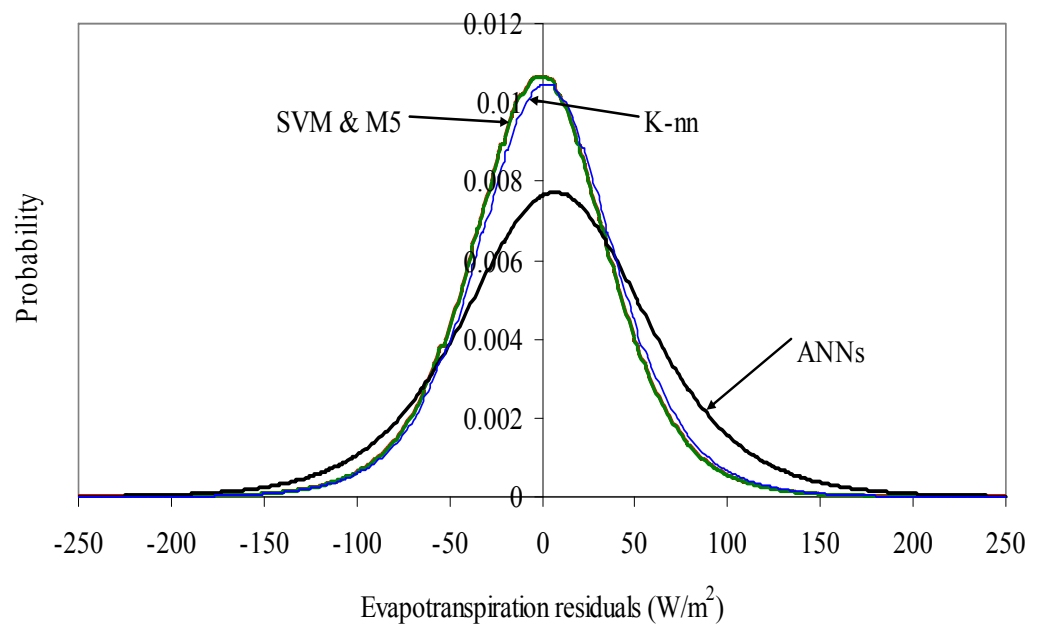
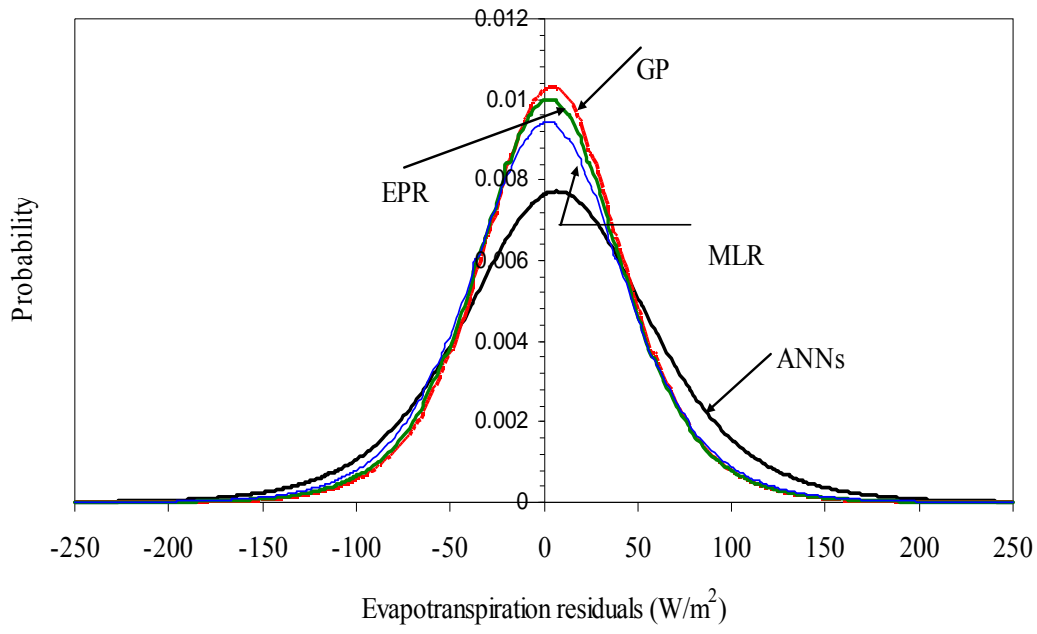


Figure 5. Probability distribution of the 12 model residuals of all techniques evapotranspiration case study).

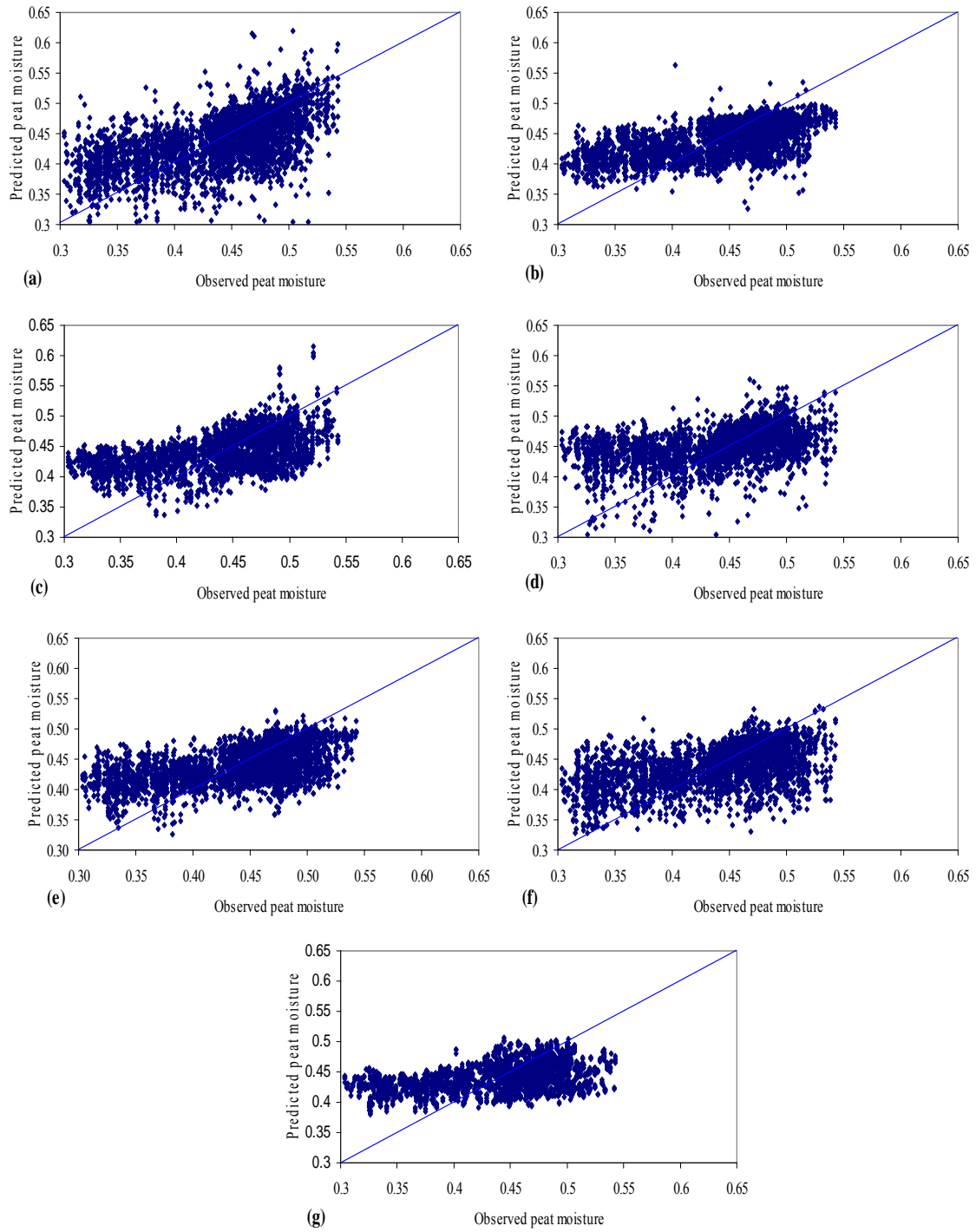


Figure 6 scatter plots of observed and predicted peat moisture content, (a) ANNs, (b) GP, (c) EPR, (d) SVM, (e) M5, (f) K-nn, and (g) MLR.

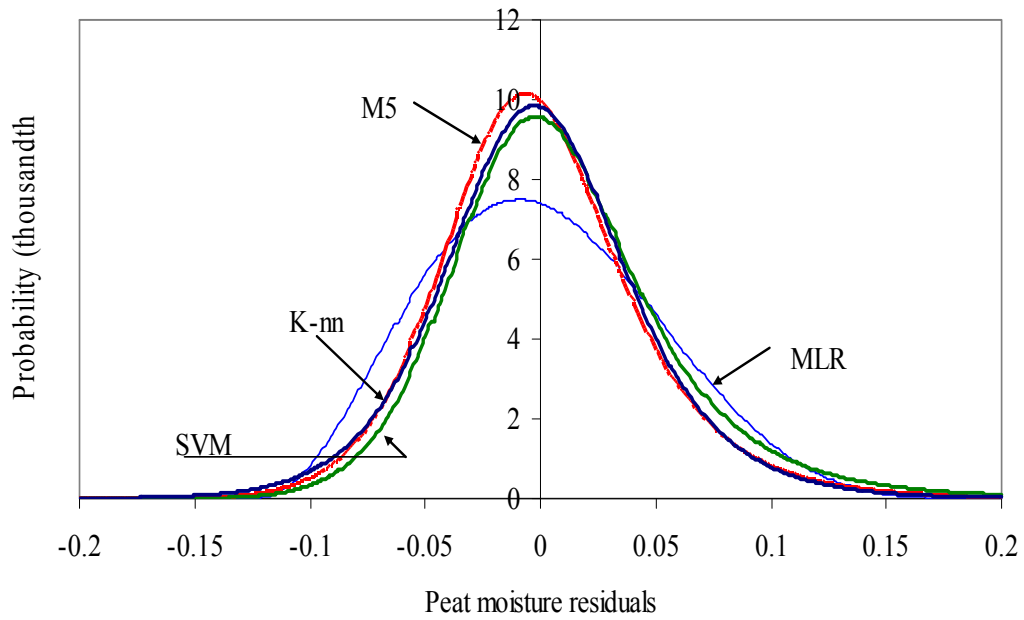
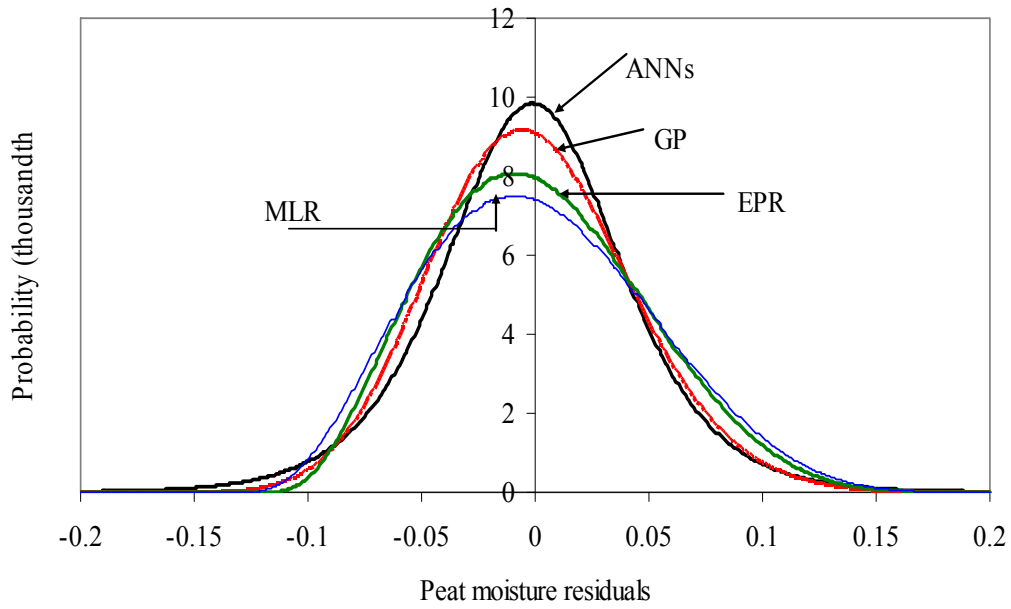


Figure 7 Probability distribution of the 12 model residuals of all techniques (peat moisture case study).

α2) for EPR and MLR techniques. This reflects the fact that the adopted modeling techniques are different in the way that they predict the output and minimize the errors, even if their average overall error values are close. The frequency curves reflect the considerable outperformance of the ANNs, K-nn, M5, and SVM over other more uncertain and biased techniques, such as MLR and the EPR techniques. An important observation here is the smaller uncertainty of the poor SVM technique. The small uncertainty of the SVM technique reflected by the probability distribution is affected by the narrow range of residuals and small overall RMSE, however, the SVM models failed to capture the trend (lower R value) of the SMP data.

Table 10. Testing results of all models applied to the **Peat moisture** dataset

Models	RMSE			MARE			MB			R		
	Best	Ave	Worst	Best	Ave	Worst	Best	Ave	Worst	Best	Ave	Worst
<b>ANNs</b>	0.04	<b>0.04</b>	0.05	0.08	<b>0.08</b>	0.09	0.00	<b>0.00</b>	-.009	0.66	<b>0.60</b>	0.53
<b>GP</b>	0.04	<b>0.04</b>	0.05	0.08	<b>0.08</b>	0.09	0.00	<b>0.00</b>	-.007	0.70	<b>0.61</b>	0.47
<b>EPR</b>	0.05	0.05	0.05	0.09	0.09	0.10	0.00	<b>0.00</b>	0.006	0.56	0.52	0.46
<b>SVM</b>	0.05	0.05	0.05	0.08	0.09	0.10	-.004	0.011	0.016	0.57	0.44	0.35
<b>M5</b>	0.04	<b>0.04</b>	0.05	0.07	<b>0.08</b>	0.10	0.001	<b>0.00</b>	0.004	0.66	<b>0.57</b>	0.37
<b>K-nn</b>	0.04	0.05	0.05	0.07	<b>0.08</b>	0.09	0.00	<b>0.00</b>	0.005	0.62	0.53	0.43
<b>MLR</b>	0.05	0.05	0.05	0.10	0.10	0.10	0.00	0.001	0.004	0.43	0.40	0.33
<b>Naïve</b>	-	-	-	-	-	-	-	-	-	-	-	-

The Kolmogorov-Smirnov nonparametric test was conducted on the model residuals of all techniques to test the null hypothesis that the model residuals of any two techniques are sampled from the same distribution. The matrix of the p-values is given as Table 11. There is strong statistical evidence that the residuals of the various techniques are stemming from different populations. There are also no correlations found among the probability distributions of the residuals of the various modeling techniques.

Table 11 the p-values of the two samples K-S test on the model residuals (peat moisture).

	ANNs	GP	EPR	SVM	M5	K-nn	MLR
ANNs	1	0.0000	0.0021	0.0000	0.0000	0.0156	0.0000
GP		1	0.0001	0.0000	0.0015	0.0000	0.0000
EPR			1	0.0000	0.0003	0.0000	0.0069
SVM				1	0.0000	0.0000	0.0000
M5					1	0.0000	0.0000
K-nn						1	0.0000
MLR							1

### 7.3 Till (lower layer) moisture case study

The till moisture case study (SMT) is similar to the previous case study with regard to the small variability in the dataset, and the highly nonlinear response to the climatic variables due to the large storage effect. Table 3 shows that the variability (CV) of the till moisture data is half that of the peat moisture data, whereas the skew in the till moisture dataset is nearly double that of the peat moisture. The error measures shown in Table 12 (and in particular the R statistic) reveal that K-nn, GP, and ANNs are better candidates than other modeling

techniques based on the same argument mentioned earlier regarding the R statistic. Similar to the previous case study, SVM and MLR techniques were the lowest in the rank with regard to the prediction accuracy. The small variability, combined with the high nonlinearity, of the SMT dataset contributed to the relative success of the K-nn technique in this particular case study. The failure of the MLR is an indicator of the potential utility of the ANNs for modeling the SMT.

Table 12. Testing results of all models applied to the **Till moisture** dataset

Models	RMSE			MARE			MB			R		
	Best	Ave.	Worst	Best	Ave.	Worst	Best	Ave.	Worst	Best	Ave.	Worst
<b>ANNs</b>	0.01	0.02	0.02	0.04	<b>0.04</b>	0.06	0.00	-.002	-.006	0.63	<b>0.55</b>	0.21
<b>GP</b>	0.01	<b>0.01</b>	0.02	0.03	<b>0.04</b>	0.05	0.00	-.001	0.002	0.72	<b>0.57</b>	0.38
<b>EPR</b>	0.02	0.02	0.02	0.04	<b>0.04</b>	0.05	0.00	<b>0.00</b>	0.002	0.52	0.44	0.32
<b>SVM</b>	0.01	0.02	0.02	0.04	<b>0.04</b>	0.04	0.001	.003	0.005	0.57	0.48	0.32
<b>M5</b>	0.01	0.02	0.02	0.04	<b>0.04</b>	0.05	0.00	<b>0.00</b>	0.002	0.59	0.46	0.30
<b>K-nn</b>	0.01	<b>0.01</b>	0.02	0.03	<b>0.04</b>	0.04	0.00	<b>0.00</b>	0.002	0.70	<b>0.57</b>	0.49
<b>MLR</b>	0.02	0.02	0.02	0.04	<b>0.04</b>	0.05	0.00	<b>0.00</b>	0.002	0.50	0.41	0.32
<b>Naïve</b>	-	-	-	-	-	-	-	-	-	-	-	-

Frequency curves were constructed for the residuals of each technique (Figure 8) to investigate the predictive uncertainty. The graph in this case provides useful and more insightful view of the predictive reliability of the various techniques. The K-nn, GP, ANNs, and the SVM are clearly less uncertain and less skewed than EPR and other linear techniques (M5 and MLR) in this case study. The best-found probability distributions of the residuals of the various techniques differed across techniques. The LogLogistic ( $\gamma, \beta, \alpha$ ) distribution was found to fit the residuals of SVM and K-nn, and ANNs modeling techniques, Logistic ( $\alpha, \beta$ ) for GP, Lognormal ( $\mu, \sigma$ ) for EPR and MLR, and ExtremeValue (a, b) for M5. This reflects the fact that some of the adopted modeling techniques are really different in the way that they predict the output and minimize the errors, whereas some similarity is identified among the ANNs, K-nn, and SVM techniques. This similarity is only in terms of approaching the optimum solution, and leaving model residuals to be similarly distributed, but not necessarily in the distribution parameters. Similar to the SMP case study, less uncertainty with the use of the SVM is due to model residuals that stay around the mean, and thus, reduce the variability and the average error. This should not be confused with the poor accuracy of capturing trends in the data (low R value in Table 12 and even high IPE value in Table 8).

The Kolmogorov-Smirnov nonparametric test was conducted on the model residuals of all techniques to test the null hypothesis that the model residuals of any two techniques are sampled from the same population. The matrix of the p-values is given as Table 13. The KS test reveals that there is no evidence to reject the hypothesis in the case of the EPR and M5, and also GP and M5. The visual analysis of Figure 8 confirms the finding regarding EPR and M5; however, M5 and GP are visually different. The reason is that the graph presents the best-fit distributions that should be used to make conclusions regarding the potential of the techniques and their possible performance on untested cases in the future. The KS is a nonparametric test that relies on the cumulative frequency of the sample itself. For the rest of the adopted techniques, there is strong statistical evidence that the residuals of the various

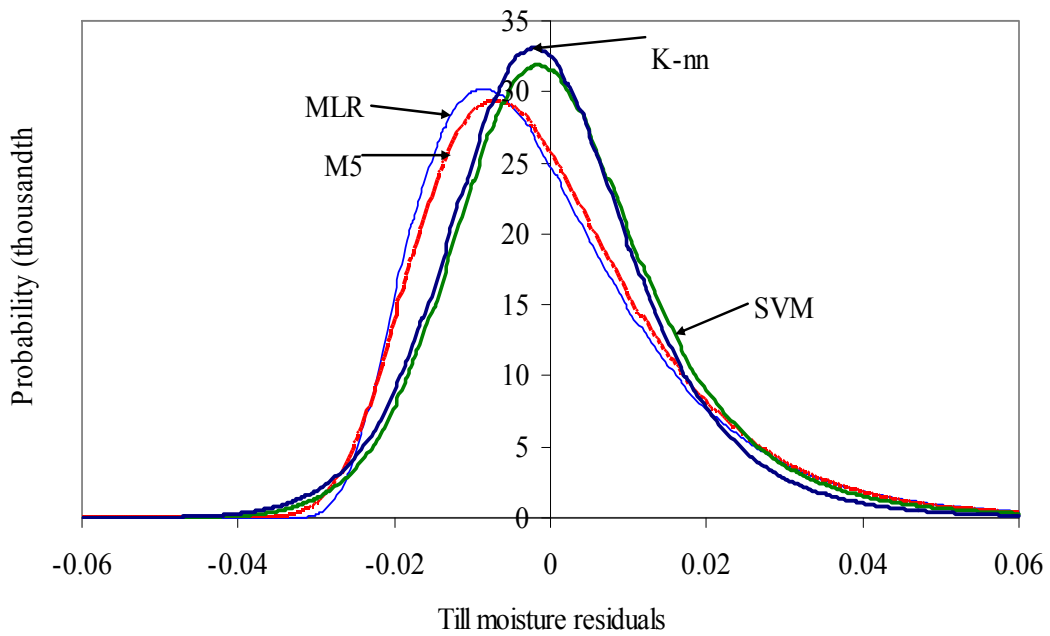
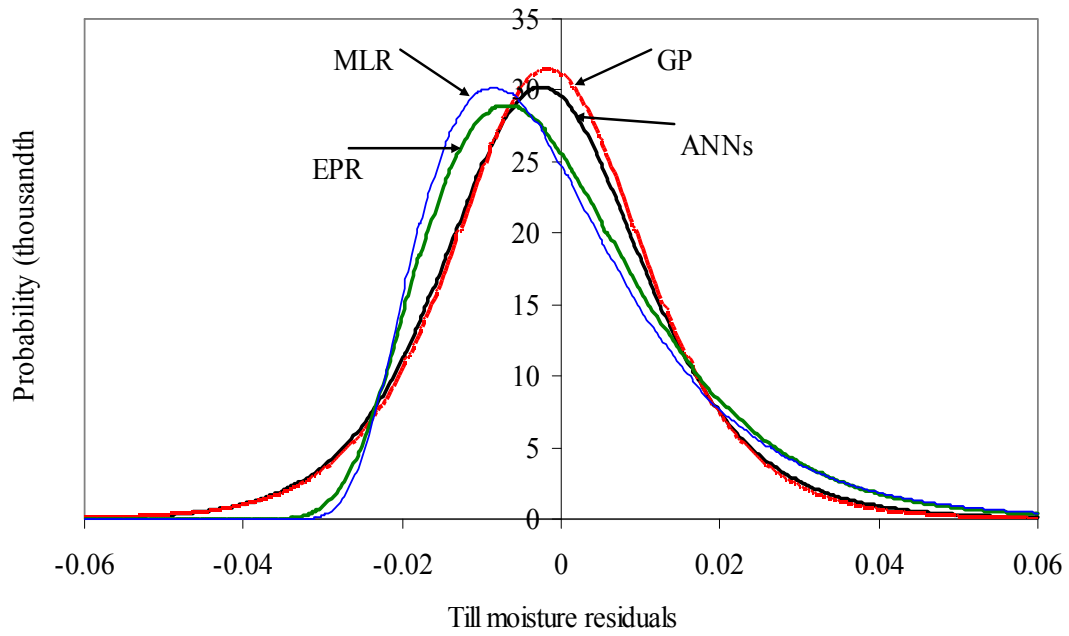


Figure 8 Probability distribution of the 12 model residuals of all techniques (till moisture case study).

techniques are stemming from different populations. There are also no correlations found among the probability distributions of the residuals of the various modeling techniques.

Table 13 the p-values of the two samples K-S test on the model residuals (till moisture).

	ANNs	GP	EPR	SVM	<b>M5</b>	K-nn	MLR
ANNs	1	0.0040	0.0043	0.0000	0.0094	0.0000	0.0000
<b>GP</b>		1	0.0400	0.0000	<b>0.1843</b>	0.0000	0.0006
<b>EPR</b>			1	0.0000	<b>0.1667</b>	0.0000	0.0101
SVM				1	0.0000	0.0001	0.0000
M5					1	0.0000	0.0007
K-nn						1	0.0000
MLR							1

#### 7.4 Rainfall-runoff case study I

The performance of the various techniques applied to the daily rainfall-runoff I (R-R I) case study is provided in Table 14. In this case study, the preceding runoff was not used as an input for the models, therefore, the information content can be considered limited (only rainfall of the current and preceding three days were used). The performances of all techniques were almost on par as shown by close values of average RMSE and R (Table 14) as well as close values of the IPE indicator (Table 8). Nonetheless, one can observe that M5, GP, and MLR were slightly better and less biased (lower MB values) than the other techniques. In a situation like this R-R I case study, where the information content itself is limited; it may not be practical to differentiate among the various modeling techniques. The limiting factor for the prediction accuracy becomes the information content rather than the predictive capability of the various techniques. A linear (e.g., MLR) or a modular linear (M5) technique is sufficient for such dataset.

Table 14. Testing results of all models applied to the **Rainfall-Runoff I**

Models	RMSE			MARE			MB			R		
	Best	Ave	Worst	Best	Ave	Worst	Best	Ave	Worst	Best	Ave	Worst
<b>ANNs</b>	25	26	28	1.04	1.47	2.03	0.59	-2.3	-8.8	0.59	0.53	0.40
<b>GP</b>	23	<b>25</b>	28	1.61	1.71	1.83	0.52	1.05	1.84	0.66	<b>0.57</b>	0.52
<b>EPR</b>	24	27	40	1.55	1.69	1.81	-0.05	<b>0.05</b>	1.66	0.61	0.54	0.49
<b>SVM</b>	25	26	27	1.01	<b>1.11</b>	1.18	-5.0	-6.1	-7.75	0.60	0.54	0.47
<b>M5</b>	24	<b>25</b>	26	1.48	1.60	1.65	0.08	<b>-0.17</b>	-1.82	0.62	<b>0.58</b>	0.54
<b>K-nn</b>	25	26	27	1.45	1.58	1.70	-0.74	-1.55	-3.28	0.58	0.52	0.44
<b>MLR</b>	24	<b>25</b>	26	1.5	1.61	1.71	0.01	<b>0.12</b>	-1.55	0.60	<b>0.56</b>	0.53
<b>Naïve</b>	-	-	-	-	-	-	-	-	-	-	-	-

The best-found probability distributions of the residuals of the various techniques did not differ. The Logistic ( $\alpha$ ,  $\beta$ ) probability distribution, with different parameter values for each technique, was found to fit the residuals of all modeling techniques. This reflects the fact that the adopted modeling techniques produce residuals that have similar nature, and that all techniques were similar in the way that they predict the output and minimize the errors (Figure 9). Even though the visual analysis of Figure 9 shows almost no practical differences

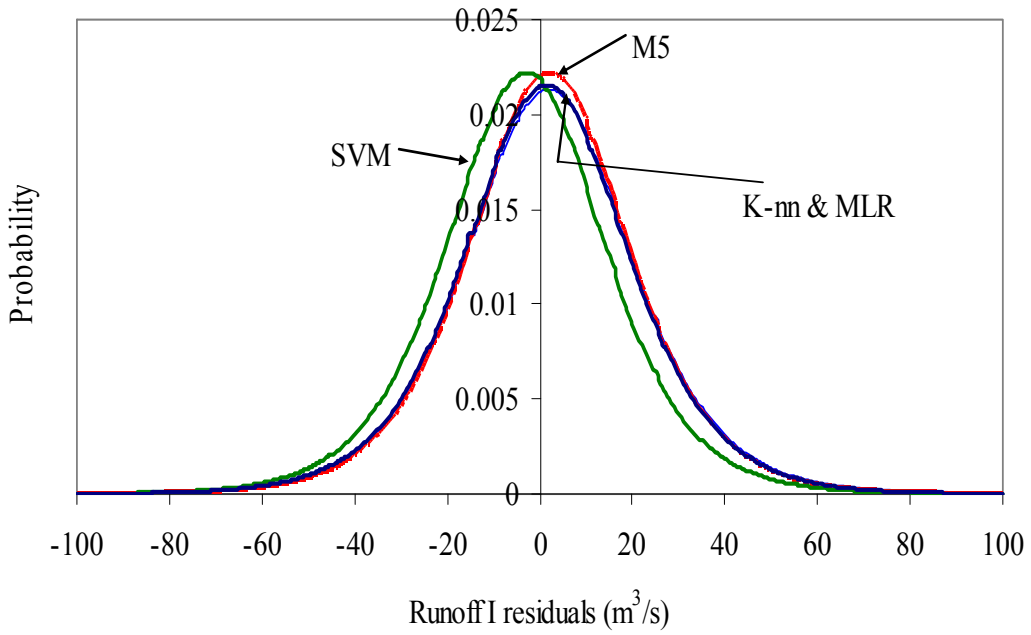
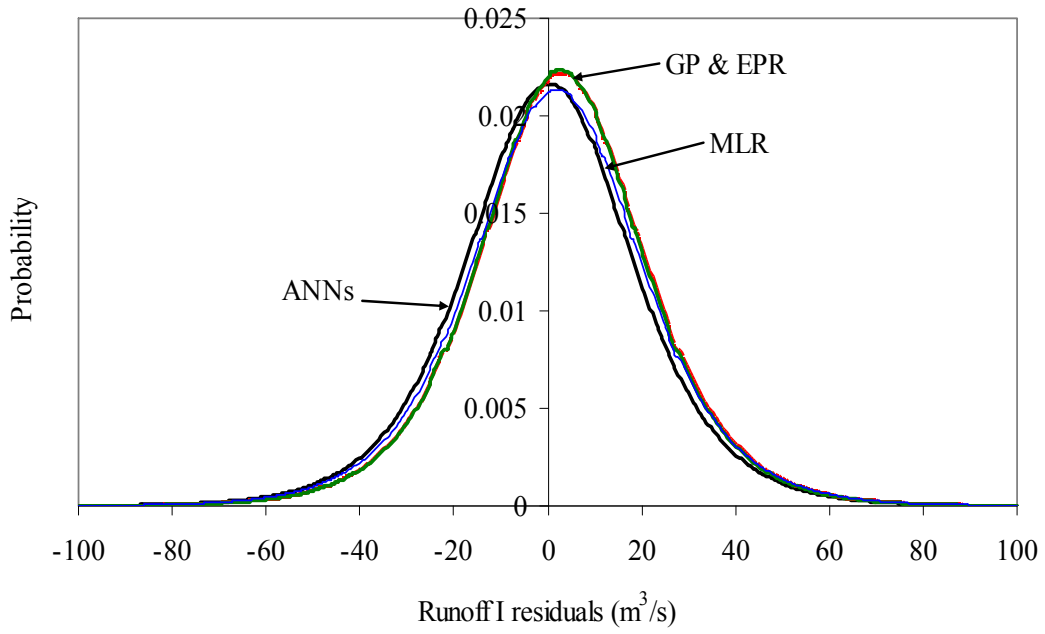


Figure 9 Probability distribution of the 12 model residuals of all techniques (rainfall-runoff I case study).

among the various probability distributions, the p-values of the K-S test (Table 15) indicate that there is strong evidence to reject the null hypothesis. Based on the K-S test, the model residuals of the various techniques could represent different distributions. There is no contradiction between the K-S test results and the visual test because a slight shift on the graph might be translated to a statistically significant difference.

Table 15 the p-values of the two samples K-S test on the model residuals (rainfall-runoff I).

	ANNs	GP	EPR	SVM	M5	K-nn	MLR
ANNs	1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
GP		1	0.0000	0.0000	0.0000	0.0000	0.0000
EPR			1	0.0000	0.0000	0.0000	0.0000
SVM				1	0.0000	0.0000	0.0000
M5					1	0.0000	0.0000
K-nn						1	0.0000
MLR							1

### 7.5 Rainfall-runoff case study II

This rainfall-runoff II (R-R II) case study is the same as the previous R-R I dataset with one difference; that is the preceding runoff was used as an additional input. In such a strongly autocorrelated series as the daily runoff, providing the preceding runoff as an input to predict the current runoff make strong information content at the disposal of the predictive models. Even though the MLR technique may not be suitable for this case study because one of the inputs (preceding runoff) is autocorrelated, it is used to show how much information can be captured by a global linear model. In addition to this, a naïve model for predicting the daily runoff was developed just by using the preceding runoff value as an estimate of the current runoff. The performance of the various techniques applied to the daily R-R II case study is provided in Table 16. GP, M5, and EPR, followed by the MLR, techniques are better choices than the other techniques for this case studies. They provide the lowest RMSE, MARE, MB, and the highest R values. The IPE indicator in Table 8 also mostly supports this finding. Expectedly, the presence of the preceding runoff as an input in this case study makes the input-output relationship more globally linear than nonlinear. The superiority of the MLR over the ANNs supports this idea. Instance-based learning techniques that use simple average of the nearest neighbors (K-nn) may not be a good choice. K-nn found almost most of the information within a range of very small number of neighbors (average of 3 neighbors, Table 6), but the failure to regress the information weakens the input-output relationship. The information capture in linear models could be even enhanced by local/modular techniques, such as the M5 model trees.

Figure 10 shows the scatter plots of observed vs. predicted runoff II data. The scatter around the 45-degree line supports the conclusion made earlier regarding the superiority of the GP, M5, and EPR, and the inferiority of K-nn, ANNs, and SVM techniques. The success of GP, EPR, and M5 across all ranges of the dataset is noticeable (Fig. 10b; 10c; 10e). With the exception of the SVM and naïve models, the best-found probability distributions of the residuals of the various techniques did not differ. The Logistic ( $\alpha$ ,  $\beta$ ) probability distribution, with different parameter values for each technique, was found to fit the residuals of ANNs,

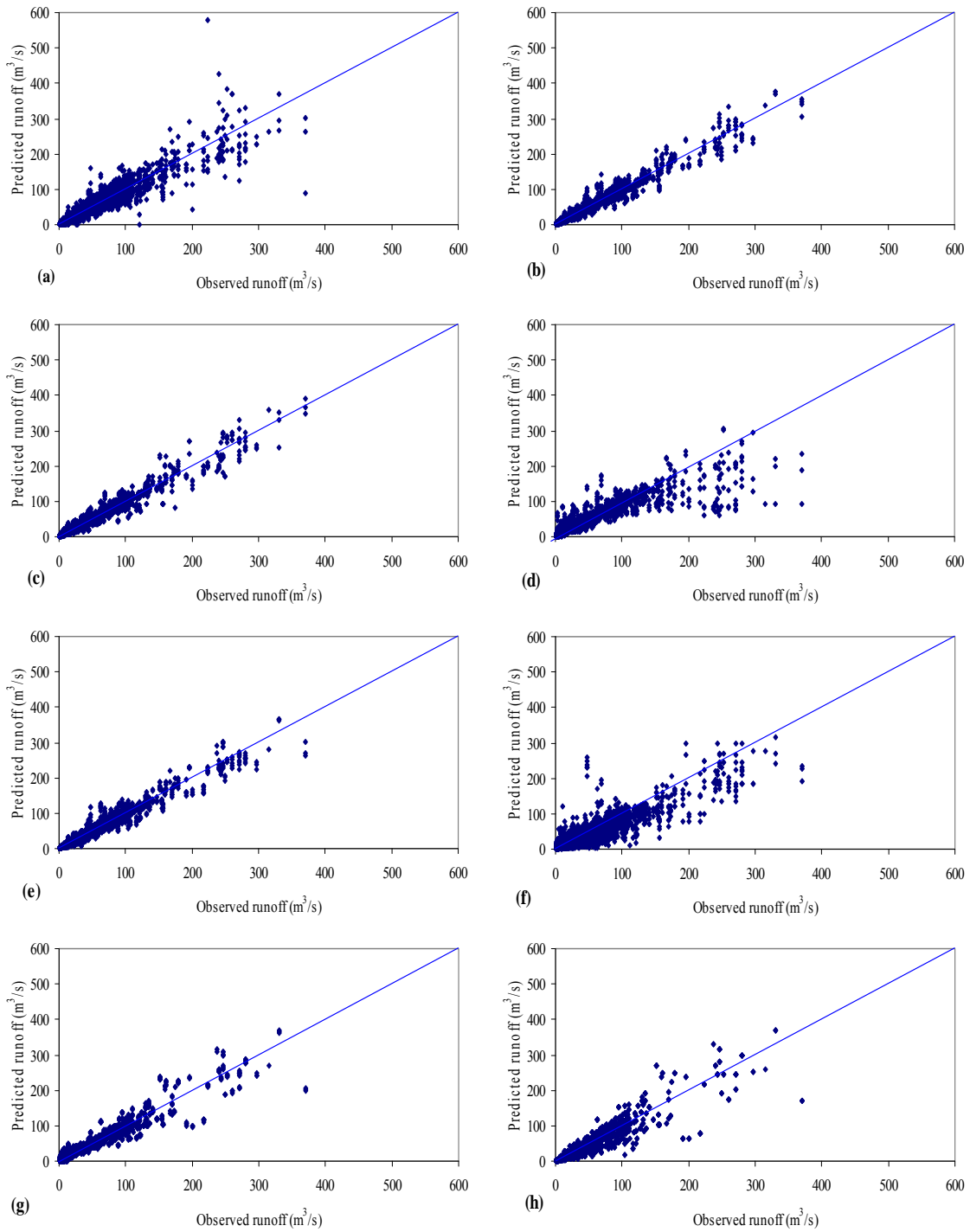


Figure 10 scatter plots of observed and predicted runoff II, (a) ANNs, (b) GP, (c) EPR, (d) SVM, (e) M5, (f) K-nn, (g) MLR, and (h) naive.

GP, EPR, M5, K-nn, and MLR techniques, whereas Normal ( $\mu$ ,  $\sigma$ ) was found to fit the residuals of the SVM and the naïve models. In spite of the similarity in the best-fit distribution, the parameters were completely different even visually (Figure 11). All modeling techniques produced symmetrical distributions of model residuals, but GP, EPR, and M5 possess the smallest predictive uncertainty. The p-values of the K-S test (Table 17) indicate that there is strong evidence to reject the null hypothesis. Based on the K-S test, the model residuals of the various techniques could represent different distributions.

Table 16. Testing results of all models applied to the **Rainfall-Runoff II**

Models	RMSE			MARE			MB			R		
	Best	Ave	Worst	Best	Ave	Worst	Best	Ave.	Worst	Best	Ave	Worst
<b>ANNs</b>	5.6	9.1	14.8	0.10	0.21	0.43	-0.27	-0.69	7.54	0.99	0.97	0.91
<b>GP</b>	4.3	<b>4.9</b>	6.0	0.09	<b>0.11</b>	0.14	0.03	0.06	0.62	0.99	<b>0.99</b>	0.98
<b>EPR</b>	4.7	<b>5.5</b>	7.0	0.10	<b>0.11</b>	0.15	0.02	<b>0.01</b>	-0.34	0.99	0.98	0.97
<b>SVM</b>	6.5	10.1	15.6	0.09	0.12	0.15	-0.02	-0.59	-1.53	0.98	0.94	0.87
<b>M5</b>	4.4	<b>5.2</b>	6.0	0.09	<b>0.09</b>	0.10	0.00	<b>0.00</b>	0.44	0.99	<b>0.99</b>	0.98
<b>K-nn</b>	10.4	11.8	13.8	0.33	0.37	0.42	-1.26	-1.86	-2.63	0.96	0.93	0.89
<b>MLR</b>	6.8	7.8	9.4	0.31	0.35	0.41	-0.06	0.07	0.48	0.97	0.97	0.95
<b>Naïve</b>	8.8	10.1	12.1	0.12	0.12	0.12	-0.01	0.04	-0.44	0.96	0.94	0.92

Table 17 the p-values of the two samples K-S test on the model residuals (rainfall-runoff II).

	ANNs	GP	EPR	SVM	M5	K-nn	MLR
ANNs	1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
GP		1	0.0003	0.0000	0.0000	0.0000	0.0000
EPR			1	0.0000	0.0000	0.0000	0.0000
SVM				1	0.0000	0.0000	0.0000
M5					1	0.0000	0.0000
K-nn						1	0.0000
MLR							1

## 8. Discussion

After evaluating the various soft computing techniques from both perspectives of prediction accuracy and uncertainty, one of the means to gain further insight into their modeling capabilities is by assessing the performance deterioration in the testing phase compared to the training phase. Less deterioration may indicate a higher level of reliability and less uncertainty about the technique performance in future and untested applications. The percent deterioration is calculated for each technique by dividing the difference between training and testing performance by the training performance. A negative percent means that the performance of the technique during the testing phase was better than that during the training phase. Table 18 presents the percent deterioration in both RMSE and MARE for all techniques and case studies. For each technique, the average values of RMSE and MARE of the 12 models were used. A few observations can be noted from Table 18: (i) ANNs had the highest level of performance deterioration in all case studies, which is an intricate

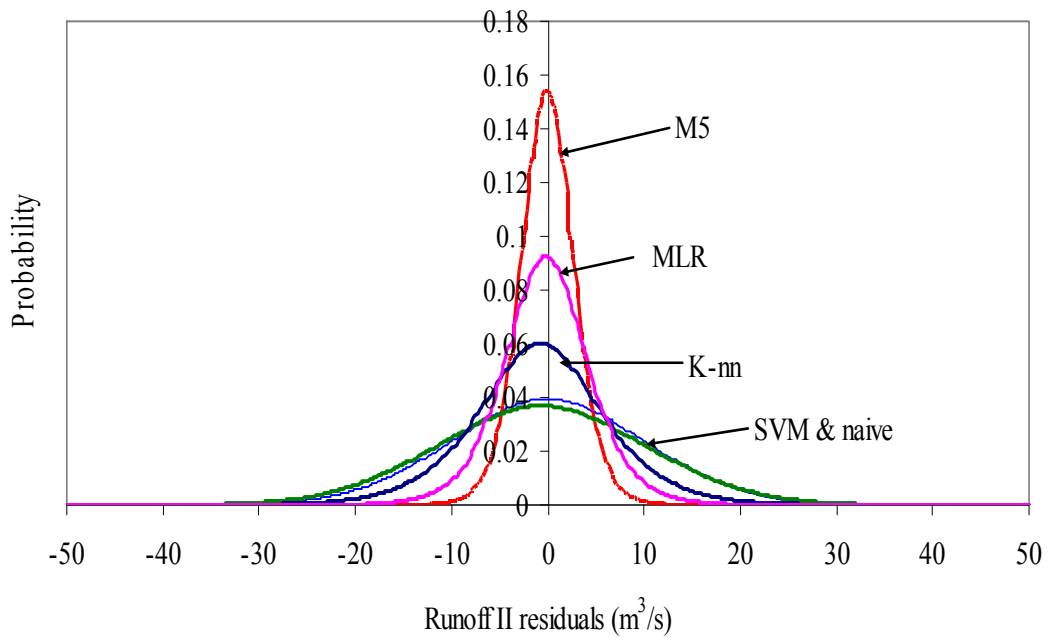
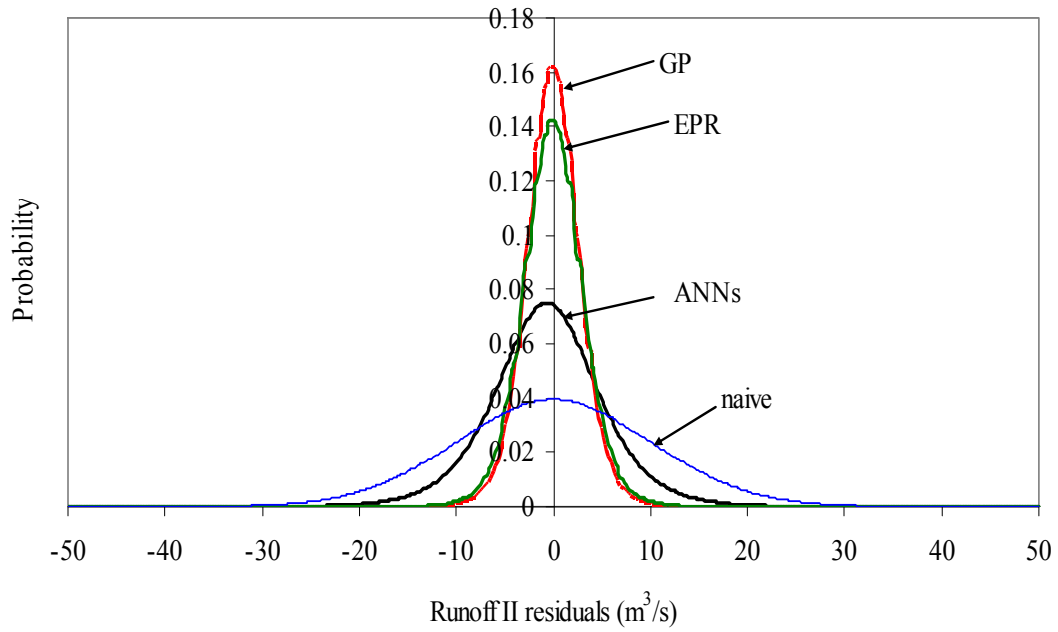


Figure 11 Probability distribution of the 12 model residuals of all techniques (rainfall-runoff II case study).

characteristic of the technique and perhaps any highly nonlinear technique. ANNs seem to go after some individual and local patterns even when training is stopped by a third sample of cross-validation; (ii) similar to ANNs, SVM suffered from similar phenomenon in four out of the five case studies. This might be counter intuitive and requires further investigation because a technique that employs the concept of error tolerance and flatness of the approximation function should do better in this regard. Users of SVM are encouraged to study further the effect of the error tolerance and the flatness coefficient C on the technique performance; (iii) in highly nonlinear case studies (e.g., peat and till soil moisture), the compromise between improving the prediction accuracy while reducing the deterioration might be difficult. The deterioration of the K-nn technique in both case studies was the highest, while performing relatively better than other techniques in terms of the prediction accuracy and uncertainty; (iv) EPR, almost similar to MLR, was excellent in its generalization ability. The deterioration of performance during the testing phase was very small in all case studies; highlighting a great potential of this technique; and (v) in most cases GP and M5 model trees were not far from the EPR regarding the performance deterioration. Therefore, whenever EPR, GP, and M5 are comparable to other techniques in terms of prediction accuracy and uncertainty, they deserve to be given preference as candidate modeling techniques.

Table 18 the percent deterioration of model performance during testing compared to training

	AET		SMP		SMT		R-R I		R-R II	
	RMSE	MARE	RMSE	MARE	RMSE	MARE	RMSE	MARE	RMSE	MARE
ANNs	<b><u>29</u></b>	<b><u>118</u></b>	<b><u>27</u></b>	<b><u>26</u></b>	<b><u>23</u></b>	<b><u>26</u></b>	<b><u>18</u></b>	-8	<b><u>127</u></b>	<b><u>65</u></b>
GP	0	10	11	10	12	9	13	0	11	2
EPR	1	12	4	4	2	2	16	-1	7	-1
SVM	<b><u>22</u></b>	<b><u>47</u></b>	11	19	<b><u>17</u></b>	<b><u>29</u></b>	<b><u>20</u></b>	<b><u>12</u></b>	<b><u>140</u></b>	<b><u>73</u></b>
M5	1	12	12	12	8	7	8	1	15	6
K-nn	4	16	<b><u>26</u></b>	<b><u>29</u></b>	<b><u>24</u></b>	<b><u>26</u></b>	12	9	45	46
MLR	-1	11	1	2	0	1	1	0	0	-1

One of the fundamental questions of this research study is whether there are real differences among the techniques under consideration with regard to their predictive capabilities. The results and analysis show that serious evaluation of the various techniques has to rely on multiple ways, such as the average overall error represented by multiple error measures, scatter plots of the observed vs. predicted outputs, probabilistic analysis of the model residuals, and statistical tests of the significance of the differences among the residuals of various models/techniques. As an example, the SVM technique performed well on the peat moisture case study in terms of the overall average error measures and the probability distribution of the residuals, however, the scatter plots reveal that the models were not behavioral; i.e., could not capture the trend of the phenomenon at all. On the other hand, the superiority of the ANNs over other techniques on the same dataset was revealed by the scatter plots. The analysis presented in the previous section show that SVM, M5, K-nn, and GP techniques were the best candidates for modeling the evapotranspiration case study. In the peat moisture case study, ANNs, GP, and followed by K-nn, M5, and EPR provided the best performances, whereas ANNs, GP, and K-nn were the best for modeling the till moisture dataset. Even though the K-S test show that the difference between the residuals of GP and

M5 was insignificant, this should be treated with caution. The test compares the residuals but fail to assess the difference in the R statistic, which is the key indicator in this particular case study. M5 was not successful in this highly nonlinear dataset. For the rainfall-runoff I dataset, all techniques were on par, and perhaps there is no need for a sophisticated nonlinear model. In the last, and most linear, case study (rainfall-runoff II), GP, M5, and EPR were obviously better than the other techniques.

Neural networks could be one of the optimum modeling choices for highly nonlinear case studies (e.g., peat and till soil moisture), but could be completely dominated by other techniques as it was the case for the AET and the rainfall-runoff II case study, depending on the level of linearity in the dataset. M5 is an excellent choice for linear and some nonlinear dataset; it performed poorly only in the till moisture dataset. EPR, though it was not a top choice except in the rainfall-runoff II case study, it was never completely dominated by others, and sometimes it was among the best technique. The excellent generalization ability (minimum performance deterioration during the testing phase) of the EPR adds to its potential for hydrological applications. However, in highly nonlinear datasets, EPR was always less successful than GP. GP was the only technique that was always either the top model or, at least, among the best models regarding both prediction accuracy and uncertainty. The ability of GP to adapt the structural complexity of the generated model/program to the dataset could be one of the main reasons of its superb predictive capability. The SVM seems to be significantly affected by the choice of kernels. In this study, the RBF kernel was chosen based on its performance on the cross validation sample of most case studies (four out of five cases). In the linear rainfall-runoff II case study, when a linear kernel was tested, the prediction accuracy, represented by RMSE, MARE, and R, improved by 20-25%.

Two limitations of this study are noted here: (i) the effect of the model inputs on the predictive capabilities was not investigated. More comprehensive inputs or lack of important inputs affect the degree of linearity/nonlinearity of the input-output relationship, and thus, the model performance. Such an effect may differ from one technique to the other; (ii) some capabilities of the various techniques and tools were not, and perhaps cannot be, thoroughly covered. The Discipulus software for GP was run for almost two hours each time. It was observed that allowing from 24-48 hours of run could slightly improve the results. The EPR tool allows for multiobjective optimization, rather than just minimizing the squared error, which was not tried in this study. Also instance-based techniques (K-nn) could be further improved using weighted average or regression of the nearest neighbors. ANNs could be trained using Bayesian regularization algorithm (Demuth and Beale, 2001), which could improve the generalization ability. In this study, multiobjective cost functions were avoided as much as possible. However, future research by the authors and/or other researchers could add to this experiment and build on it.

The non-parametric Gamma test ( $\Gamma$ -test) (Chuzhanova et al., 1998; Evans and Jones, 2002, and applied in hydrology by Remesan et al., 2008) was conducted to gain insight into the predictability and the complexities of the modeled processes, and possible leads into selection of suitable modeling techniques. The  $\Gamma$  statistic was calculated for every dataset using the original training and cross-validation subsets as one integrated subset (all unique points). The V-ratio, gradient, and the M-test were all calculated using the scaled data (zero mean and 0.5

standard deviation). The  $\Gamma$  statistic calculated using the unscaled data to facilitate the comparison with the error variance of the various modeling techniques (Table 19). The following observations can be made: (i) for the AET case study, the error variance of the linear regression technique (2302) was already lower than the  $\Gamma$  statistic; indicating that complex nonlinear model (e.g., ANNs) may not be necessary. The low gradient value of 0.041 shows that a noncomplex smooth function can be used for modeling the AET process, whereas the reasonably low V-ratio indicates that there is high predictability in the output variable. GP, shown to perform well on all case studies, achieved the lowest error variance. Even though it is lower than the estimated  $\Gamma$ , but when it is divided by the AET variance (Table 1), the ratio is 0.23; similar to the V-ratio.; (ii) for the R-R I case study, similar to the AET, there is no need for nonlinear complex model, especially in light of the high V-ratio that indicates low level of predictability. The low level of predictability is attributed to the lack of appropriate inputs, which was rectified in the R-R II case study. All techniques were found to perform on par. The slight superiority of the M5 (ratio of error variance to output variance is 0.44), which is a modular linear technique can be attributed to the fact that it does not produce a smooth function. This is something that the  $\Gamma$ -test may not capture well; (iii) similar conclusions can be made for the R-R II case study. Nonlinear techniques, such as ANNs, will not perform well. The very low V-ratio that indicates very high predictability might be achieved by techniques that can outperform MLR, yet have the ability to adapt to linear situations. As expected GP, EPR, and M5 performed extremely well in this case; (iv) both SMP and SMT case studies, the MLR technique failed to achieve the estimated  $\Gamma$  value, and actually produced ratios of error variance to output variance of 1.0 and 0.8, respectively. This finding points to the possibility that more complex nonlinear models are needed. As the results of this study show, in addition to GP, the ANNs and K-nn were relatively more successful in the SMP and SMT case studies. However, it should be noted that  $\Gamma$ -test relates well to the model performance with regard to the squared error, but in cases where the criterion of performance is the R statistic, the test may not be the optimum tool; (v) the M-test indicates the number of data points that is perhaps needed to achieve the accuracy indicated by the V-ratio. It can be noticed from Table 19 that the size of the datasets needed for developing nonlinear models for the peat and till soil moisture are slightly more than what was used in this study. For the other three case studies, the size of the training datasets exceeded the M-test.

The  $\Gamma$ -test may assist in the selection of the appropriate modeling techniques by applying first multiple linear regression models and evaluating the residuals against the  $\Gamma$ -test values. Decision can be made regarding the need for a complex nonlinear technique. If there is a need for such technique, then ANNs and K-nn (in addition to GP for example) should be seriously considered. If it is assessed that complex nonlinear techniques are not needed, then improvement of results can be sought using GP, EPR, and M5. When complex nonlinear techniques are not needed, and the predictability is low (i.e., high V-ratio) significant improvement may not be at all possible.

Table 19 the gamma test results on all case studies.

Case study	Error variance MLR technique	$\Gamma$ statistic	Error variance best technique	V ratio	Gradient	M statistic
AET	2302	2778	1928 (GP)	0.207	0.0414	1200
SMP	0.0025	0.0018	0.002 (ANNs)	0.410	0.2970	500
SMT	0.0003	0.0002	0.0002 (K-nn)	0.273	0.4140	520
R-R I	459	495	397 (M5)	0.560	0.1040	1300
R-R II	26	27	7 (GP)	0.013	0.1100	1100

## 9. Summary and conclusions

Data driven modeling techniques, and in particular soft computing techniques, have addressed and solved many issues in hydrological modeling but also caused questions and concerns to be raised. The most important concerns are regarding the way soft computing techniques are handled, compared, and evaluated and the basis on which findings and conclusions were drawn. The sub-optimal approach in designing modeling experiments, the use and the split of datasets, the exclusive use of techniques and case studies, and writing research articles from the standpoint of advocating certain techniques have contributed to the problem. In this first part of two-part paper, a concise but comprehensive review of key articles that presented comparisons among various soft computing modeling techniques was summarized. It was concluded that findings were usually dataset-specific, contradictory, and thus, difficult to generalize. A comprehensive data driven modeling experiment was proposed and explained. Six soft computing modeling techniques, namely, neural networks, genetic programming, evolutionary polynomial regression, support vector machines, M5 model trees, and K-nearest neighbors were proposed and briefly explained. Multiple linear regression and naïve models were also suggested as baseline for comparison with the various techniques.

Five different case studies representing three different hydrological processes or variables (evapotranspiration, soil moisture, and rainfall-runoff) from Canada and Europe were described and proposed for the modeling experiment. The central step of the methodology is creating 12 different realizations (groups) from each dataset by random sampling. Each group contains three subsets; training, cross-validation, and testing. Each technique was proposed to be applied to each of the 12 groups of each dataset. This methodology was designed to evaluate both predictive accuracy and uncertainty of the various techniques on a wide range of possibilities that allow for comprehensive testing the modeling capabilities of these techniques. The second part of this paper addresses the application of the proposed methodology through the input selection and the implementation of the various techniques. Results, analysis, and discussion of the findings of this study are presented in the second part.

Neural networks (ANNs) that have hidden nodes with nonlinear transfer functions may impose on the data a model with complexity level that is higher than that needed by many hydrological data. The results of the experiment conducted in this research study show that ANNs were a sub-optimal choice for the actual evapotranspiration (AET) and the two rainfall-runoff case studies. In the highly nonlinear case studies (peat and till soil moisture), ANN models were the most successful ones. In general, genetic programming (GP) was the most

successful technique due to its ability to adapt the model complexity to the modeled data. Evolutionary polynomial regression (EPR) performance could be close to the GP with datasets that are more linear than nonlinear. Support vector machines (SVM) are sensitive to the kernel choice and if appropriately selected, the performance of SVM can improve. M5 model trees performs very well with linear and semi linear data, which cover wide range of hydrological situations. In highly nonlinear case studies, ANNs, K nearest neighbors (K-nn), and GP could be more successful than other modeling techniques. K-nn was also successful in linear situations, and it deserves more attention as a potential modeling technique for hydrological applications.

Soft computing modeling techniques should be applied in ensemble fashion. Multiple groups (realizations) of each dataset should be randomly generated, by sampling without replacement, and should be divided into three split samples of training, cross-validation for stopping the training phase, and testing for applying the model once. Developing multiple non-dominated models of each technique, based on the multiple realizations of the dataset, allows for evaluating the predictive accuracy and uncertainty in a comprehensive way. Multiple overall average error measures, frequency distributions of model residuals, and scatter plots of observed vs. predicted data should be all used as one package to evaluate the predictive capabilities of the modeling techniques. Gamma test can be used as a guide to assist in the selection of the appropriate modeling technique for a particular dataset. Further studies can build on the experiment presented in this research to evaluate other data driven techniques and to study the impact of input selection and input pr-processing on the relative predictive capabilities of the techniques.

## References

Abrahart, R., See, L., and Dawson, C. Neural Network hydroinformatics: Maintaining scientific rigour. In: Abrahart, R., See, L., and Solomatine, D., (eds) Practical hydroinformatics. Computational intelligence and technological developments in water applications. Springer-Verlag, Berlin Heidelberg, pp. 33-47.

ASCE Task Committee on Application of Artificial Neural Networks in Hydrology, 2000. Artificial neural networks in hydrology. I: Preliminary concepts, *J. Hydrol. Eng.*, 5(2), 115-123.

Babovic, V., and Keijzer, M. (2002). "Rainfall-runoff modelling based on genetic programming." *Nordic Hydrology J.*, 33(5), 331-346.

Babovic, V. and Keijzer, M. (2000). "Genetic Programming as Model Induction Engine." *Journal of Hydroinformatics*, 2 (1), 35-60.

Banzhaf, W., Nordin, P., Keller, R. E., and Francone, F. D. (1998). "Genetic programming-an introduction: on the automatic evolution of computer programs and its applications." *Morgan Kaufmann Publishers, Inc.*

Behzad, M., Asghari, K., Eazi, M., Palhang, M., 2008. Generalization performance of Support Vector Machines and Neural Networks in Runoff Modeling. *Expert Systems with Applications*, doi:10.1016/j.eswa.2008.09.053 (In press).

Boese, K., 2003. The design and installation of a field instrumentation program for the evaluation of soil-atmosphere water fluxes in a vegetated cover over saline/sodic shale overburden, M.Sc. thesis, University of Saskatchewan, Saskatoon, Sask.

Bouckaert, R. R., Frank, E., Hall, M., Kirkby, R., Reutemann, P., Seewald, A. and Scuse, D. (2008). WEKA Manual for version 3.6.0. University of Waikato, Hamilton, New Zealand.

Brown, M., and Harris, C. (1994), *Neurofuzzy adaptive modeling and control*, Prentice Hall: New York.

Cherkassky, V., Krasnopolsky, V., Solomatine, D. and Valdes, J. 2006. Computational intelligence in earth sciences and environmental applications: Issues and challenges. *Neural Networks*, 19: 113-121.

Cherkassky, V. S. and Mulier, F. (2007). Learning from data: concepts, theory, and methods, 2<sup>nd</sup> Edition, John Wiley & Sons, Inc., Hoboken, New Jersey.

Chuzhanova, N. A., Jones, A. J., and Margett, S. 1998. Feature selection for genetic sequence classification. *Bioinformatics*, 14(2): 139-143.

Çimen, M. (2008). Estimation of Daily Suspended Sediments using Support Vector Machines. *Hydrol. Sci. J.*, 53(3), 656 – 666.

Demuth, H., Beale, M., 2001. Neural Network Toolbox Learning For Use with MATLAB. The Math Works Inc, Natick, Mass.

Dogliani, A., Giustolisi, O., Savic, D. A., and Webb, B.W. (2007). “An evolutionary approach to stream temperature analysis.” *Hydrological Processes J.*, doi: 10.1002/hyp.6607

Drexler, J. Z., R. L. Snyder, D. Spano, and K. T. Paw (2004), A review of models and micrometeorological methods used to estimate wetland evapotranspiration, *Hydrol. Processes*, 18, 2071-2101.

Elshorbagy, A. and El-Baroudy, I. 2009. Investigating the capabilities of evolutionary data-driven techniques using the challenging estimation of soil moisture content. *J. Hydroinfo.* (in press)

Elshorbagy, A. and Parasuraman, K. 2008. On the relevance of using artificial neural networks for estimating soil moisture content. *J. Hydrol.*, 362(1-2): 1-18.

Elshorbagy, A. and Parasuraman, K. 2008. Toward bridging the gap between data-driven and mechanistic models: cluster-based neural networks for hydrologic processes. In: Abrahart, R.,

See, L., and Solomatine, D., (eds) Practical hydroinformatics. Computational intelligence and technological developments in water applications. Springer-Verlag, Berlin Heidelberg, pp. 389-403.

Elshorbagy, A., Jutla, A., and Kells, J., 2007. Simulation of the hydrological processes on reconstructed watersheds using system dynamics, *Hydrol. Sci. J.*, 52, 538-562.

Giustolisi, O., Doglioni, A., Savic, D. A. and Webb, B. W. (2007). "A multi-model approach to analysis of environmental phenomena." *Environmental Modelling & Software*, 22(5), 674 - 682.

Evans, D., Jones, A. J. 2002. A proof of the gamma test. *Proceedings of Royal Society. Series A*, 458: 2759-2799.

Francone, F. D. 2001. Discipulus: Owner's Manual. Register Machine Learning Technologies, Inc., Littleton, CO.

Giustolisi, O., Savic, D. A. (2006). "A symbolic data-driven technique based on evolutionary polynomial regression." *J. Hydroinformatics*, 8(3), 207 - 222. doi: 10.2166/hydro.2006.020

Haigh, M. J., 2000. The aims of land reclamation, *Land Reconstruction and Management*, A.A. Balkema Publishers, Rotterdam, The Netherlands, 1: pp1-20.

Haykin, S., 1999. *Neural Networks: A Comprehensive Foundation*, 2<sup>nd</sup> ed. MacMillan, New York.

A. W. Jayawardena, N. Muttill, and J. H. W. Lee., 2006. Comparative Analysis of Data-Driven and GIS-Based Conceptual Rainfall-Runoff Model, *J. Hydrologic. Engg.*, 11(1), 1-11.

Jayawardena, A. W., Muttill, N., and Fernando, T. M. K. G. (2005). "Rainfall-runoff modelling using genetic programming." *MODSIM 2005 International Congress on Modelling and Simulation. Modelling and Simulation Society of Australia and New Zealand*, In Zenger, A. and Argent, R.M. (eds), December 2005, 1841-1847. ISBN: 0-9758400-2-9.

Jones, A. J., Margetts, S., and Durrant, P. 2001. *The winGamma<sup>TM</sup> User Guide*. University of Wales, Cardiff.

Khan, M. S. and Coulibaly, P. (2006). Application of Support Vector Machine in Lake Water Level Prediction. *J. Hydrol. Engg.*, 11( 3), 199-205.

Koza, J.R. (1992). "Genetic programming: On the programming of computers by means of natural selection." *The MIT Press*, Cambridge, MA.

Laucelli, D., Berardi, L., Doglioni, A. (2005). "Evolutionary polynomial regression toolbox: version 1.SA." Department of Civil and Environmental Engineering, Technical University of Bari, Bari, Italy. Available from: <http://www.hydroinformatics.it/prod02.htm>. (accessed March 2008).

- Maier, H., and Dandy, G. (2000), Neural networks for the prediction and forecasting of water resources variables: A review of modeling issues and applications, *Environ. Modell. Software*, 15(1), 101-124.
- Makkeasorn, A., Chang, N. B. and Zhou, X. 2008. Short-term streamflow forecasting with global climate change implications – A comparative study between genetic programming and neural network models. *J. Hydrol.* 352, 336-354.
- Mattera, D. and Haykin, S. (1999). Support vector machines for dynamic reconstruction of a chaotic system. In B. Schölkopf, C. J. C. Burges, and A. J. Smola, editors, *Advances in Kernel Methods – Support Vector Learning*, pages 211-242, Cambridge, MA, MIT Press.
- Müller, K. R., Smola, A., Rätsch, G., Schölkopf, B., Kohlmorgen, J., and Vapnik, V. (1997). Predicting time series with support vector machines. In W. Gerstner, A. Germond, M. Hasler, and J. D. Nicoud, editors, *Artificial Neural Networks – ICANN’97*, pages 999-1004, Berlin. Springer Lecture Notes in Computer Science, Vol. 1327.
- Karlsson M, Yakowitz S. 1987. Nearest neighbour methods for nonparametric rainfall–runoff forecasting. *Water Resources Research* 23(7): 1300–1308.
- Palisade Corporation Inc. 2005. Guide to using @RISK. Advanced risk analysis for spreadsheets. Palisade Corporation, NY, U.S.A
- Parasuraman, K. and Elshorbagy, A. 2008. Model Structure Uncertainty and its Quantification Using Ensemble-Based Genetic Programming Framework. *Water Resources Research*, 44, W12406, doi:10.1029/2007WR006451.
- Parasuraman, K. and Elshorbagy, A. 2007. Cluster-based hydrologic prediction using genetic algorithm-trained neural networks. *J. Hydrol. Engng., ASCE*, 12(1): 52-62.
- Parasuraman, K., Elshorbagy, A., Carey, S.K. (2007a). “Modelling dynamics of the evapotranspiration process using genetic programming.” *Hydrological Science J.*, 53(3), 563-578.
- Parasuraman, K., Elshorbagy, A., and Si, B. C. (2007b). “Estimating saturated hydraulic conductivity using genetic programming.” *Soil Science Society of America J.*, 71, 1676–1684.
- Parasuraman, K. and Elshorbagy, A. 2008. Model Structure Uncertainty and its Quantification Using Ensemble-Based Genetic Programming Framework. *Water Resources Research*, 44, W12406, doi:10.1029/2007WR006451.
- Rabun˜al, J. R., Puertas, J., Su´arez, J. and Rivero, D. 2007. Determination of the unit hydrograph of a typical urban basin using genetic programming and artificial neural Networks *Hydrol. Process.* 21, 476–485.
- Remesan, R., Shamim, M. A., and Han, D. 2008. Model data selection using gamma test for daily solar radiation estimation. *Hydrol. Proc.*, 22: 4301-4309.

- Savic, D.A., Giustolisi, O., Berardi, L., Shepherd, W., Djordjevic, S. and Saul, A. (2006). "Sewers failure analysis using evolutionary computing." *Water Management J.*, 159(2), 111 - 118. doi: 10.1680/wama.2006.159.2.111.
- Silva, S. (2005), GPLAB – a genetic programming toolbox for MATLAB, <http://gplab.sourceforge.net>
- Sivapragasam, C., Vincent, P. and Vasudevan, G. 2007. Genetic programming model for forecast of short and noisy data. *Hydrol. Process.* **21**, 266–272.
- Smola, A. J. and Schölkopf, B. (1998). A Tutorial on support vector regression. NeuroCOLT2 Technical Report Series, NC2-TR-1998-030, ESPRIT Working Group, <http://www.neurocolt.com>.
- Solomatine, D., Maskey, M. and Shrestha, D. L. 2008. Instance-based learning compared to other data-driven methods in hydrological forecasting. *Hydrol. Proc.*, 22: 275-287.
- Solomatine, D. P. and Siek, M. B. (2006). Modular learning models in forecasting natural phenomena. *Neural Networks*, 19, 225–235.
- Solomatine, D. P. and Xue, Y. (2004). M5 Model Trees and Neural Networks: Application to Flood Forecasting in the Upper Reach of the Huai River in China. *J. Hydrol. Engng.*, 9(6), 491–501.
- [Stefánsson, A, Končar, N., and Jones, A. J. 1997. A note on the Gamma test. \*Neural Computing & Applications\*, 5:131-133.](#)
- Vapnik, V. (1995). *The Nature of statistical learning theory*. Springer, N.Y.
- Witten, I. H. and Frank, E. (2005) "Data Mining: Practical machine learning tools and techniques", 2<sup>nd</sup> Edition, Morgan Kaufmann, San Francisco.
- Wu, C. L., Chau, K. W. and Li, Y. S. (2008). River Stage Prediction based on a Distributed Support Vector Regression" *Journal of Hydrology*, 358, 96–111.
- Wu, W., Wang, X., Xie, D., and Liu, H. (2008). Soil Water Content Forecasting by Support Vector Machine in Purple Hilly Region, *Computer and Computing Technologies in Agriculture*, 1, 223–230.
- Zhang, B., and Govindaraju, S., 2000. Prediction of watershed runoff using Bayesian concepts and modular neural networks, *Water Resour. Res.*, 36(3), 753-762.