ASCE Journal of Hydrologic Engineering, 9(6), 2004, pp. 491-501
doi 10.1061/(ASCE)1084-0699(2004)9:6(491))

M5 Model Trees and Neural Networks: Application to Flood Forecasting in the Upper Reach of the Huai River in China

Dimitri P. Solomatine[1] and Yunpeng Xue[2]

Abstract: The applicability and performance of the so-called M5 model tree machine learning technique is investigated in a flood forecasting problem for the upper reach of the Huai River in China. In one of configurations this technique is compared to multilayer perceptron artificial neural network (ANN). It is shown that model trees, being analogous to piecewise linear functions, have certain advantages compared to ANNs—they are more transparent and hence acceptable by decision makers, are very fast in training and always converge. The accuracy of M5 trees is similar to that of ANNs. The improved accuracy in predicting high floods was achieved by building a modular model (mixture of models); in it the flood samples with special hydrological characteristics are split into groups for which separate M5 and ANN models are built. The hybrid model combining model tree and ANN gives the best prediction result.

DOI: XXXX

CE Database subject headings: Hydrologic models; Hydrologic data; Flood forecasting; Artificial intelligence; China; Neural networks; Multiple regression models; Data analysis.

Introduction

Artificial neural network (ANN) models have become a popular choice among the nonlinear flood forecasting methods (Hsu et al. 1995; Minns and Hall 1996; Solomatine and Torres 1996; Dawson and Wilby 1998; See and Openshaw 1998; Govindaraju and Rao 2000; Dibike and Solomatine 2001; Bhattacharya and Solomatine 2002a; Birikundavyi et al. 2002). Being an accurate predictive tool, the ANN technique has, however, a disadvantage that often limits its acceptance in practice—ANN models are not transparent ("black box") and do not help us to understand the nature of the solution. The arbitrary nature of the internal representation means that there may be dramatic variations between networks of identical architecture trained on the same data (Witten and Frank 2000). Recently some attempts were made to produce the understandable insights from the structure of neural networks, such as saliency analysis (Abrahart et al. 2001) and the methods of recovering rules reported by Setonio et al. (2002). The latter method starts from building an ANN as the "right" tool that further needs a better interpretability.

There are, however, approaches that instead of constructing a single complex model use a number of simpler "local" models specialized in a particular area of input space (called mixtures of experts). Such models were developed already in the 1980s—see, for example, a paper on multilinear models by Becker and Kundzewicz (1987). Another method of such type comes from the "statistics" world—the approach by Friedman (1991) in his multiple adaptive regression splines algorithm. Yet another one, being the subject of this paper, is a M5 model tree (Quinlan 1992; Witten and Frank 2000), a method attributed to the area of machine learning. An earlier method classification and regression tree of Breiman et al. (1984) of regression trees should also be mentioned; it generates, however, zero-order models (constant output values for subsets of input data) rather than the first-order (linear) models.

The M5 algorithm combines the features of classification and regression: trees—structured regression is built on the assumption that the functional dependency is not constant in the whole domain, but can be approximated as such on smaller subdomains (Fig. 1). For the continuous variables, these subdomains are searched for and characterized by the average value (regression trees) or with a linear regression function (model trees) of the dependent variable (on Fig. 1, for example, for the domain $[x2>2, x1>2.5]$ Model 3 is used and its form is $y=a_0+a_1x_1+a_2x_2$) The most attractive advantage is that by dividing the function being induced into linear patches, M5 model trees provide a representation that is reproducible and comprehensible by practitioners.

Still, the M5 model tree is not a very popular method: to our knowledge after the paper of Kompare et al. (1997) in Slovene language the applications of M5 model trees in water-related problems are reported only by Solomatine (2002), by Solomatine and Dulal (2003) (for rainfall-runoff modeling), and by Bhattacharya and Solomatine (2002b) (for modeling the stage–discharge relationship).

In this study that actually took place in 2000–2001, a rather complex catchment area, the upper reach of the Huai River, was considered as the study area, and the performance of various M5 model trees was investigated. In two of the five cases M5 model tree is also compared to ANN.

[1]Associate Professor, UNESCO-IHE Institute for Water Education (IHE Delft), P.O. Box 3015, 2601 DA Delft, The Netherlands. (corresponding author). E-mail: sol@ihe.nl Tel: +31-15-2151815, Fax: +31-15-2122921

[2]Yellow River Conservancy Commission, 11 Jinshui Rd., 450003 Zhengzhou, China. E-mail: ypxue@yellowriver.gov.cn

Note. Discussion open until April 1, 2005. Separate discussions must be submitted for individual papers. To extend the closing date by one month, a written request must be filed with the ASCE Managing Editor. The manuscript for this paper was submitted for review and possible publication on October 29, 2002; approved on February 20, 2004. This paper is part of the ***Journal of Hydrologic Engineering***, Vol. 9, No. 6, November 1, 2004. ©ASCE, ISSN 1084-0699/2004/9(6)/1/0/$18.00.

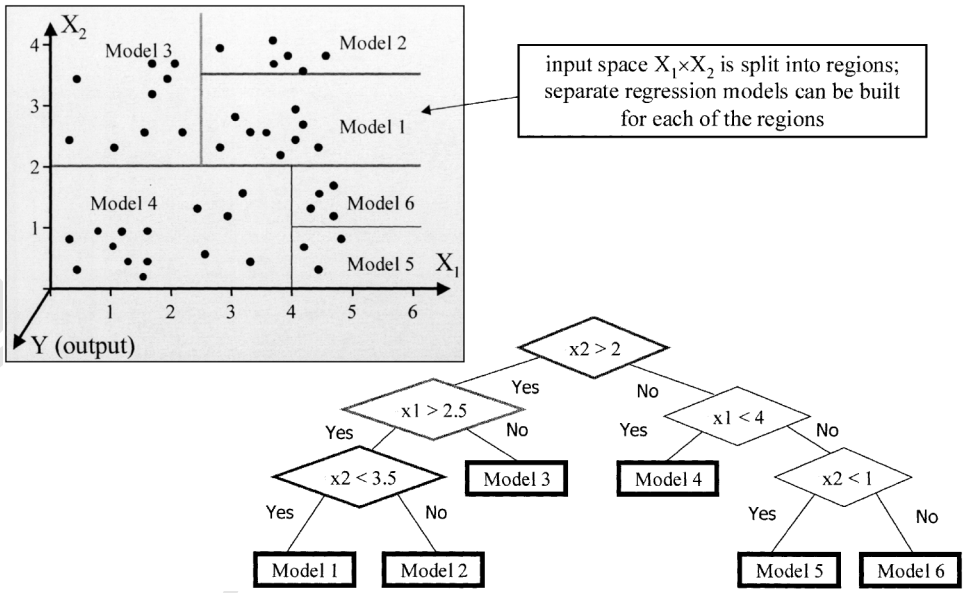


Fig. 1. Example of M5 model tree. Models 1–6 are linear regression models

Introduction to M5 Model Trees and Artificial Neural Network

M5 Model Trees

The M5 model tree algorithm was originally developed by Quinlan (1992); we used the software implementing its variation M5′ provided by Witten and Frank (2000). Model trees combine a conventional decision tree with the possibility of generating linear regression functions at the leaves. This representation is relatively perspicuous because the decision structure is clear and the regression functions do not normally involve many variables. The M5 tree is a piecewise linear model, so it takes an intermediate position between the linear models as ARIMA and truly nonlinear models as ANNs.

The construction of a model tree is similar to that of decision trees. Fig. 1(a) illustrates how the splitting of space is done. First, the initial tree is built and then the initial tree is pruned (reduced) to overcome the overfitting problem (that is a problem when a model is very accurate on the training data set and fails on the test set). Finally, the smoothing process is employed to compensate for the sharp discontinuities between adjacent linear models at the leaves of the pruned tree (this operation is not needed in building the decision tree).

Building Model Trees

Different decision tree inductive algorithms used to solve classification problems employ the divide-and-conquer approach. First, an attribute is selected to be placed at the root node and one branch is made for each possible value; then the example set is split up into subsets; one for every value of the attribute. Now the process can be repeated recursively for each branch using only those samples that actually reach the branch. If at any time all samples at a node have the same classification, the development of that part of the tree is stopped. The attribute, which is chosen to be used for a split for a given set of samples, can be determined by certain statistical property called a splitting criterion. For decision trees the splitting is based on trying to minimize the entropy in the resulting subsets; in other words, trying to filter as many samples from the same class into one subset as possible.

The M5 model tree is a numerical prediction algorithm and its splitting criterion is based on the standard deviation of the values in the subset T of the training data that reaches a particular node (which is an analogue of entropy). It is used as a measure of the error at that node, and the attribute that maximizes the expected error reduction is chosen for splitting at the node. Accordingly, on Fig. 1 the attribute X_2 is selected for the root node with the split value 2.0.

The splitting process terminates when the output values of the samples that reach a node vary slightly, that is, when their standard deviation is just a small fraction (say, less than 5%) of the standard deviation of the original sample set. Splitting also terminates when just a few samples remain in a subset. The linear regression models are then built for each subset of samples associated with the terminating (leaf) nodes.

Pruning and Smoothing Model Trees

Pruning If a generated tree has too many leaves, it may be "too accurate" and hence overfit and be a poor generalizer. It is possible to make a tree more robust by simplifying it, i.e., by *pruning*, that is by merging some of the lower subtrees into one node.

Smoothing This process is used to compensate for the sharp discontinuities that will inevitably occur between adjacent linear models at the leaves of the pruned trees. This is a particular problem for models constructed from a small number of training samples. Smoothing can be accomplished by producing linear models for each internal node, as well as for the leaves at the time the tree is built. Experiments show that smoothing substantially increases the accuracy of prediction.

Fig. 4(c) presents a tree combining seven linear regression models at the leaves. In parenthesis, the first number is the number of samples in the subset sorted to this leave and the second

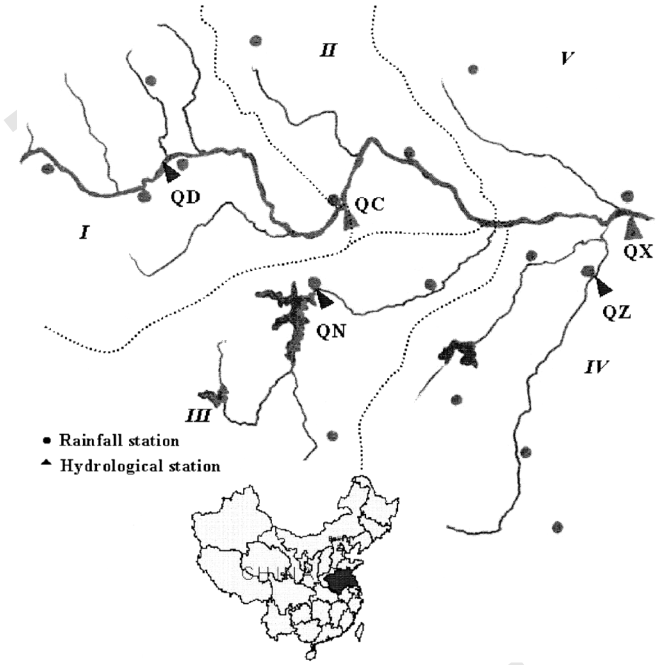


Fig. 2. Sketch map of study area of Huai River (I,II,III,IV,V represent five subcatchment areas)

one—root mean squared error (RMSE) of the corresponding linear model divided by the standard deviation of the samples subset for which it is built (expressed in percent).

Artificial Neural Networks

The ANN is a powerful machine learning method widely used in the problems of numerical prediction and classification. A network is made up of a number of interconnected nodes (processing elements), arranged into three basic layers: input, hidden, and output. The links represent weighted connections between the nodes. A processing element simply multiplies input by a set of weights, and linearly or nonlinearly transforms the result into an output value. By adapting its weights, the neural network works towards generating an output that would be close to the measured (target) output. There is a similarity between ANN and multiple nonlinear regression where coefficients are found as a result of solving an optimization problem. The detailed coverage of ANN can be easily found in many books (e.g., Haykin 1999). The references given in "Introduction" refer to the application of ANN in hydrology especially in rainfall-runoff modeling. A three-layer feed forward multilayer perceptron (MLP) ANN based on the back propagation algorithm is a popular choice in the fields of hydrology in general and runoff analysis in particular.

Study Area

The Huai River is one of the seven largest rivers in China, and also the one that is frequently threatened by floods—approximately once every 5 years, sometimes leading to catastrophes like in 1931 when 75,000 people lost their lives. In order to protect the densely populated areas and the property in the flat flood plain many flood detention and diversion areas have to be built along the river dike to store the excess water. The situation is complicated by the fact that there are about 180,000 people living

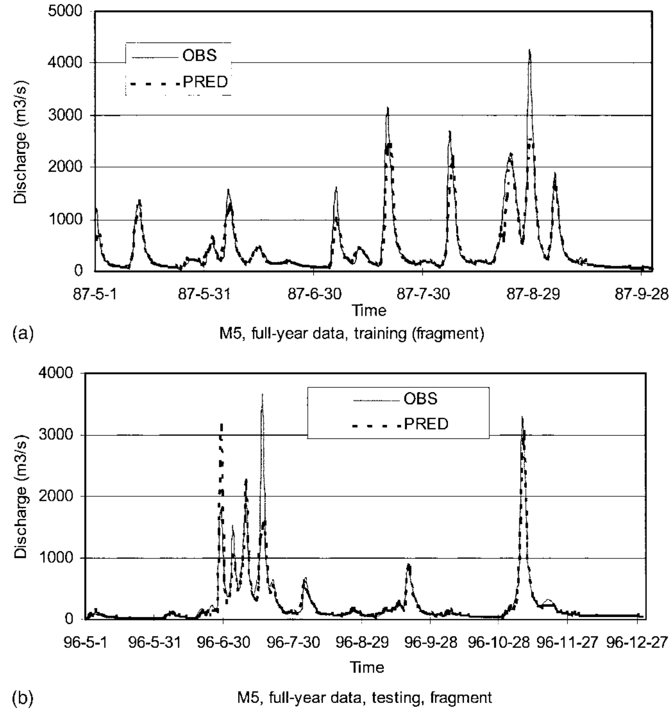


Fig. 3. Performance of M5 model using full-year data in (a) training (fragment) and (b) testing (fragment)

in the flood detention and diversion areas, thus accurate flood forecasting is critical for flood management and optimal control of the flood control projects.

The Xixian subcatchment with the drainage area of 10,190 km^2 is located in the upper reach of the Huai River and is characterized by frequent storms with the highest annual rainfall reaching 1,500 mm. It is a major flood source in Huai River basin. Most of the area is mountainous with the highest peak reaching 1,140 m. The river system and the distribution of the monitoring stations is shown in Fig. 2. The discharge of the main trunk is monitored along the river at three hydrological stations, which are denoted as QC,QD,QX; the QX station (Xixian) is the downstream station. There are nine tributaries that flow into the main trunk of the Huai River but only the two main tributaries, namely, Shihe River (QN) and Zhugan River (QZ) are gaged. Thus the data of the 17 rainfall stations and three evaporation stations in this area can be used for flood forecasting. Since the land use has not significantly changed after the construction of the reservoir on the Shihe River at the end of the 1950s, there is a possibility to apply data-driven modeling to flood forecasting.

Rainfall in the Huai River region is uneven in time and distribution, and also varies from season to season and from year to year. The flood season of the Huai River is from May to October, and the precipitation events during the flood season according to their cause can be classified as low pressure troughs, or as cyclones. Rainfall due to the low pressure troughs covers large areas and has a long duration and amount, which leads to a long duration of flood. On the other hand, the rainfall due to cyclones usually has higher intensities, shorter durations, and is lower compared with the former, and this leads to floods with high peak and short duration. The discharges of Huai River and the tributaries are normally relatively low and even nearly dry out in the dry season; the flood peak is, however, relatively large, reaching 50–100 times its mean discharge. Due to the regulation of the Nanwan Reservoir, the maximum discharge (QN) of the Shihe River

is relatively low (only 415 m^3/s), which does not have much impact on the peaks downstream at Xixian (QX).

Objective and Methodology

In flood control practice of Huai River the traditional hydrological methods (unit hydrograph and gage to gage correlation method) are the main flood forecasting methods. They are complemented by the real time adjustment by experts, in which the prediction accuracy relies mainly on and is limited to the expert's experience. The forecasting performance of the Xinanjiang model (Zhao and Liu 1995), a semidistributed conceptual rainfall-runoff model widely used in China, is not always adequate due to the limitation or the unavailability of the input data and the difficulties of calibration.

The main objective of this study was to build a model for predicting the flood discharge of the Xixian station 1 day ahead (QX_{t+1}) using machine learning methods based on the known hydrological system data (e.g., discharge, rainfall, and evaporation) on the current day and the days before. Another objective was to investigate the applicability of the M5 model tree method and to compare it, at least on some data sets, to ANN. The possibilities of combining M5 and ANN in a hybrid model were also seen as one of the items of interest.

Identification of Data Sets and Variables

There was 21 years of time series data available: (1) The daily discharge time series of 21 years (1976–1996), discharges of the Zhugan River were, however, only for the period of 10 years (1987–1996) (denoted as QZ); (2) the daily rainfall data time series of 21 years (1976–1996) from 17 stations; and (3) the daily evaporation data time series of 14 years (1976–1989) at three stations. The time series of 1976–1989 was used as a training data set, and the remaining data (1990–1996) was used for testing and cross validation. The training set was constructed in such a way that it would cover both low and high flow conditions and include both the maximum and minimum values of discharge.

Any modeling exercise requires accurate identification of input and systems variables, in this case the physical process of rainfall-runoff yield and runoff routing. All the relevant system state parameters should be considered, such as rainfall, evaporation, soil characteristics, upstream discharges at the main trunk, and tributaries. However, due to the data limitations, only the rainfall and discharge data at the main trunk are used to predict the downstream discharge. The reason is that the evaporation data series (14 years) is shorter than the others (21 years), and the discharge of the tributary Zhugan River (QZ) was recorded only for 10 years. In order to reduce the number of input variables the areal average rainfall calculated by the Thiessen method was used instead of the 17 distributed rainfall data sources.

Traditional (physically based) approaches to hydrological modeling require identification of the various types of the rainfall-runoff generation, baseflow separation techniques, etc. However, the data-driven models which work with the total rainfalls and total flows do not require this information. The rainfall losses could be taken into account by assuming that the daily rainfall data can express the rainfall intensity and the moving average of rainfall (the antecedent rainfall) can implicitly express the soil moisture content.

Taking into account the time lags for the input variables is the way to bring the catchment characteristics into a data-driven model. This was done by analyzing the physical properties of the catchment, data analysis, and transformation, and the correlation analysis. The daily areal average rainfall (Pa), moving average of areal average daily rainfall (PaMov), and discharges of predicted station (QX) and upstream station (QC) with different corresponding time lags were used as input variables. Different models required different lags. It was found that the 4 day moving average of the area rainfall with 1 day time lag ($\mathrm{PaMov4}_t$) has the maximum correlation coefficient with the predicted discharge QX_{t+1}. For the flood season data the mostly correlated variable is the 2 day moving average of area rainfall with 1 day time lag ($\mathrm{PaMov2}_t$). In various modeling experiments different combinations of variables were used.

Experiments and Results

In total ten models were built that can be classified into five different types: full-year global (overall) model, high flows global model, flood season global model, flood season modular model, flood season hybrid model, and flood season subarea rainfall model. In two cases both the M5 model tree and ANN model were built and compared using the same input and output variables, and each experiment was designed after the analysis of the results of previous ones. In some cases the results were compared to the naïve ("no-change") model (that is the model $QX_{t+1}=QX_t$, and the three-point linear regression model $QX_{t+1}=a_0+a_1QX_t+a_2QX_{t-1}+a_3QX_{t-2}$.

Training (calibration) of every model was done using the training data set—for example, for the global model it consisted of 5,109 samples, each characterized by 11 measurements of the present and past rainfalls and discharges or their moving averages (considered as input model variables) and one output variable QX_{t+1}. The trained model was tested on another data set (for the global model—2,565 samples). In both training and testing the trained model was run to make the prediction of discharge QX for one time step (1 day) for each of the samples separately, and then the corresponding hydrographs were plotted and the overall model errors calculated. The plots and errors for most of the models built together with the analysis of the results are presented below.

Full-Year Global (Overall) Model

First, the whole set of continuous 21 year data set was used, so the model was called a full-year global model. After a number of experiments aimed at finding the relevant inputs the following 11 input variables were selected:

1. three precipitation values for the current day and the previous 2 days ($\mathrm{Pa}_t, \mathrm{Pa}_{t-1}, \mathrm{Pa}_{t-3}$);
2. 4 day precipitation moving averages calculated for the current and the previous day ($\mathrm{PaMov4}_t$,$\mathrm{PaMov4}_{t-1}$);
3. three values of the upstream discharge for the current and of the previous 2 days (QC_t, QC_{t-1}, QC_{t-2}); and
4. discharge downstream for the current and of the previous 2 days (QX_t, QX_{t-1}, QX_{t-2}).

The M5 model tree with 35 leaf nodes was generated, each leaf node corresponding to a linear equation predicting the discharge of the target station (it is not shown due to its large size). The nodal splitting rules indicate the rainfall and flow condition associated with the predicted discharge, and this gives an indication of the catchment hydrological characteristics. The topmost

PROOF COPY [HE/2002/022579] 001406QHE

Table 1. Comparison of Artificial Neural Network and M5 Model Trees Prediction Results (Full Year and Flood Season)

Performance	Full-year M5 training	Full-year M5 testing	Full-year naïve, testing	Full-year linear, testing	FS-M5 training	FS-M5 testing	FS-ANN training	FS-ANN cross-valid	FS-ANN testing
Years	76–79	90–96	90–96	90–96	76–89	90–96	76–89	90–93	94–96
Number of equations in M5 or hidden nodes in ANN	35	35	n/a	n/a	7	7	8	8	8
Number of samples	5,109	2,565	2,565	2,565	2,625	1,525	2,625	653	872
Root mean square error	69.6	84.5	183.0	160.0	98	87	100	79	96
Mean absolute error	18.7	18.9	37.1	39.9	31.4	24.2	33.0	25.1	24.9
Maximum absolute error	1,695.5	2,208.3	3,009.0	3,008.2	1,766	1,651	1,498	1,130	1,446
Correlation coefficient	0.97	0.95	0.76	0.79	0.97	0.97	0.97	0.98	0.95

splitting attribute is QC_t, the upstream discharge on the current day; it has the maximum correlation with the predicted discharge QX_{t+1}. The attributes at lower levels are QX_t, PaMov4_t and PaMov4_{t-1}, they also appear in the subbranches frequently. The attributes Pa_t, Pa_{t-1}, Pa_{t-2}, and QC_{t-1} are less important and appear only at or near leaf nodes in the trees and are thus indicative of some special situations.

As shown in Figs. 3(a and b) and Table 1 the M5 model tree can predict the low flow correctly, but has higher errors in predicting some of the flood peaks: RMSE was 69 m^3/s in training and 84 m^3/s in testing. Nevertheless, the M5 model tree error was 54% smaller than that of the naïve "no-change" model and 47% smaller than that of the three-point linear regression model.

High error in flood forecasting was attributed to the fact that the number of samples corresponding to high flow was much smaller compared to those of the low flow in the full-year data set. As a result, out of the 35 rules that M5 model generated, there was only one linear model for the samples with $QX_t > 721$ m^3/s corresponding to the flood situation.

Zooming-In: Better Models for Extreme Flows

In order to reproduce the extreme-flow situations better, two other models were built: one for the selected high flows only (that was filtered by the value of QX), and the other one for the data collected during the flood season (filtered by the time constraints).

High-Flows Global Model

A separate model for the flows $QX_{t+1} > 500$ m^3/s was set up, with the 234 samples used for training and 80 for testing. The same 11 inputs were used and the model tree with 11 equations generated. Most of the equations, however, have rather high error with only one rule with the error smaller than 10%. RMSE was 281 m^3/s in training and 411 m^3/s in testing.

Interestingly, in the nodes corresponding to higher discharge values, instead of QX_t the rainfall on the previous day Pa_{t-1} begins to appear at top layers of the generated M5 model tree—this means that this attribute became the most important one for predicting the discharge QX_{t+1}. The physical explanation of this fact is that in flood season the rapid increase in discharge occurs after the intensive rainfall (Pa_{t-1})—this is different from low flow conditions when there is not much influence of rainfall. So, in spite of the errors, the M5 model has correctly suggested that the flood discharge has characteristics different from those of the low flow: this is consistent with the physics of the hydrological processes.

Flood Season Global Model

This model dealt only with the flood season (FS) data from May to October across the 21 years time series, and the 2 day moving average of area rainfall were used instead of the 4 day average (since it has higher correlation with QX_{t+1}). Correlation analysis led to the selection of 16 input attributes (Pa_t, Pa_{t-1}, Pa_{t-2}, Pa_{t-3}, $\text{PaMo}v2_t$, $\text{PaMo}v2_{t-1}$, $\text{PaMo}v2_{t-2}$, $\text{PaMo}v2_{t-3}$, $\text{PaMo}v2_{t-4}$, QC_t, QC_{t-1}, QC_{t-2}, QX_t, QX_{t-1}, QX_{t-2}, and QX_{t-3}). Two versions of model trees were built: with all 16 attributes (the model had 11 regression equations) and the simpler version with seven attributes (Pa_t, Pa_{t-1}, $\text{PaMo}v2_t$, $\text{PaMo}v2_{t-1}$, QC_t, QC_{t-1}, and QX_t) and with seven equations. Accuracy of prediction was very similar.

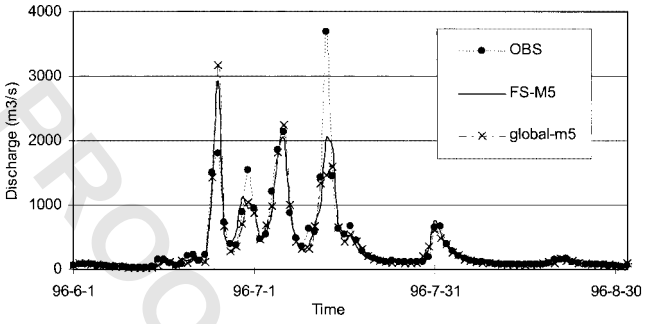
An ANN model with the same input and output variables was built as well. The popular three-layer feed forward ANN topology was employed, and the linear activation functions were used in the output layer since they delivered better performance if compared to sigmoid or tangent ones. The classical backpropagation training method of ANN was adopted. The *Neural Machine* (Neural Machine 2003) and *NeuroSolutions* (NeuroSolutions 2003) software packages were used.

The performance of the models is shown in Figs. 4(a and b) and Table 1, and the induced tree on Fig. 4(c). The ANN prediction overall result is similar to that of the M5 model trees—its RMSE is 10% higher than of M5, mean absolute error (MAE) is the same, and the maximum absolute error (MaxAE) is 12% lower. Fig. 4(b) shows that the prediction of high flows by the M5 model has improved. However, both M5 and ANN still have a high error in predicting some flood events, and the maximum error occurs during the same flood events. This means that the input data have to be processed more efficiently and some new attributes should be added to improve the prediction accuracy.

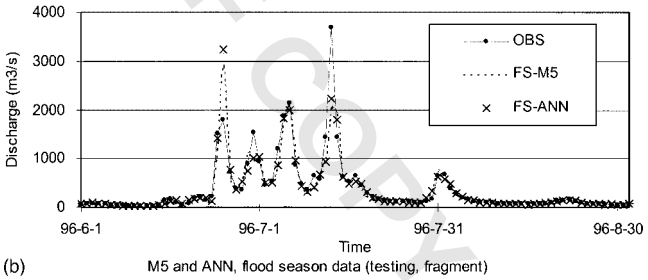
Modular Models (Mixtures of Models): Combining Expert Rules with M5 Trees

More accurate analysis of the hydrological processes in the catchment and the performed error analysis of the models reported above lead us to a conclusion that the conditions used so far were too superficial and this actually did not allow the data-driven models to classify various flood conditions into physically interpretable classes.

In order to improve the effectiveness of the predictive model, the expert-generated rules were used to build modular models. The whole flood season data was split into subsets using domain



(a) M5 models, flood season data (FS-M5) vs full-year data (global-M5) (testing, fragment)



(b) M5 and ANN, flood season data (testing, fragment)

```
QXt <= 154 :
|  PaMov2t <= 4.5 : LM1 (1499/4.86%)
|  PaMov2t >  4.5 :
|  |  PaMov2t <= 18.5 : LM2 (315/15.9%)
|  |  PaMov2t >  18.5 : LM3 (91/86.9%)
QXt >  154 :
|  PaMov2t-1 <= 13.5 :
|  |  PaMov2t <= 4.5 : LM4 (377/15.9%)
|  |  PaMov2t >  4.5 : LM5 (109/89.7%)
|  PaMov2t-1 >  13.5 :
|  |  PaMov2t <= 26.5 : LM6 (135/73.1%)
|  |  PaMov2t >  26.5 : LM7 (49/270%)
Models at the leaves:
   LM1:  QXt+1 = 2.28 + 0.714PaMov2t-1 - 0.21PaMov2t + 1.02Pat-1 + 0.193Pat
               - 0.0085QCt-1 + 0.336QCt + 0.771QXt
   LM2:  QXt+1 = -24.4 - 0.0481PaMov2t-1 - 4.96PaMov2t + 3.91Pat-1 + 4.51Pat
               - 0.363QCt-1 + 0.712QCt + 1.05QXt
   LM3:  QXt+1 = -183 + 10.3PaMov2t-1 + 8.37PaMov2t - 5.32Pat-1 + 1.49Pat
               - 0.0193QCt-1 + 0.106QCt + 2.16QXt
   LM4:  QXt+1 = 47.3 + 1.06PaMov2t-1 - 2.05PaMov2t + 1.91Pat-1 + 4.01Pat
               - 0.3QCt-1 + 1.11QCt + 0.383QXt
   LM5:  QXt+1 = -151 - 0.277PaMov2t-1 - 37.8PaMov2t + 31.1Pat-1 + 30.3Pat
               - 0.672QCt-1 + 0.746QCt + 0.842QXt
   LM6:  QXt+1 = 138 - 5.95PaMov2t-1 - 39.5PaMov2t + 29.6Pat-1 + 35.4Pat
               - 0.303QCt-1 + 0.836QCt + 0.461QXt
   LM7:  QXt+1 = -131 - 27.2PaMov2t-1 + 51.9PaMov2t + 0.125Pat-1 - 5.29Pat
               - 0.0941QCt-1 + 0.557QCt + 0.754QXt
```

(c)

Fig. 4. (a) M5 models, flood season data (FS-M5) versus full-year data (global-M5) (testing, fragment); (b) M5 and artifical neural network models using flood season data (testing, fragment); and (c) M5 model tree (FS-M5) trained on flood season data with nine input variables

knowledge, and then a set of models using the M5 model tree or ANN was built. In total three modules were constructed (Fig. 5).

Module 1 (FS-m1-M5 Model) This model was built for the discharges QX_{t-1} the day before higher than 1,000 m^3/s. Figs. 6(a and b) show that the model is very accurate; Fig. 6(c) presents the M5 model which is very concise and easy to understand. Table 2 shows that the prediction error of this model is much lower than that of the flood season global model FS-M5 calculated only for the samples with $QX_{t-1} > 1{,}000$ m^3/s.

It was found that many samples processed by Module 1 are still associated with low-flow predictions which are not so interesting for flood forecasting and which are already predicted well

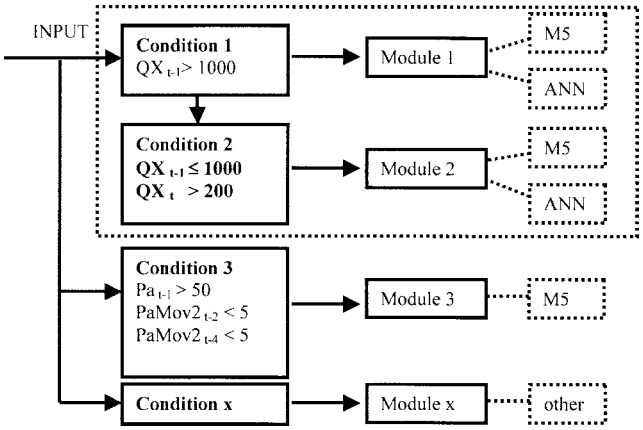
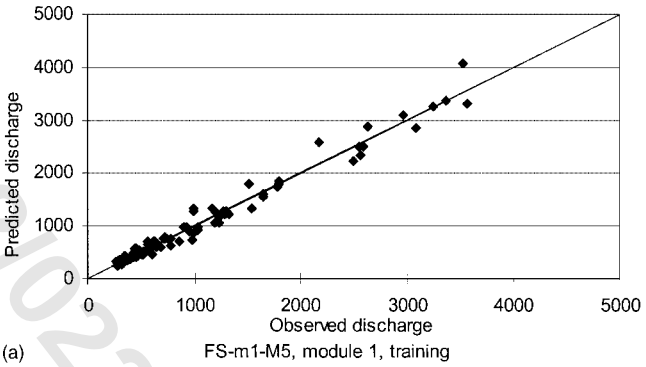


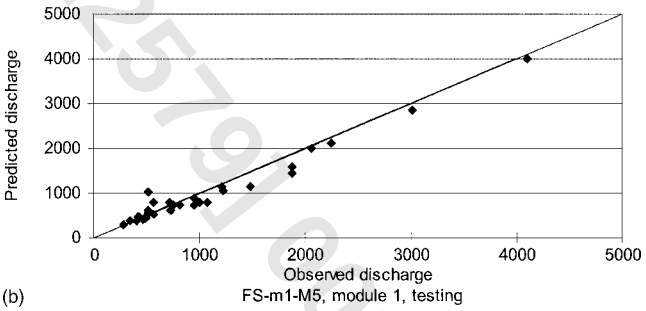
Fig. 5. Modular approach in prediction of flood forecasting

by the global model anyway. This prompted for the further filtering of data and considering the next local model—Module 2.

Module 2 (FS-m2-M5 Model) Data not included in the Module 1 (i.e., data with $QX_{t-1} <= 1{,}000$ m^3/s) was additionally filtered by the rule $QX_t > 200$ m^3/s, and a M5 model was built. As shown in Figs. 7(a and b) and Table 3, there are still some erroneous



(a)



(b)

```
QXt <= 1300 : LM1 (51/8.54%)
QXt >  1300 : LM2 (45/28.1%)
Models at the leaves:
   LM1:  QXt-1 = 139 - 0.844PaMov2t-3 + 1.93Pat-2 + 6.24Pat-1 + 14.2Pat
               + 0.585QCt + 0.0374QXt-2 - 0.0515QXt-1 + 0.283QXt
   LM2:  QXt+1 = 191 - 5.94PaMov2t-3 + 27.7Pat-1 + 16.6Pat + 0.221QCt
               + 0.13QXt-2 + 0.307QXt
```

(c)

Fig. 6. FS-m1-M5 model performance (module 1, samples with $QX_{t-1} > 1{,}000$ m^3/s) in (a) training; (b) testing; and (c) M5 model tree for FS-m1-M5 model (module 1, samples with $QX_{t-1} > 1{,}000$ m^3/s)

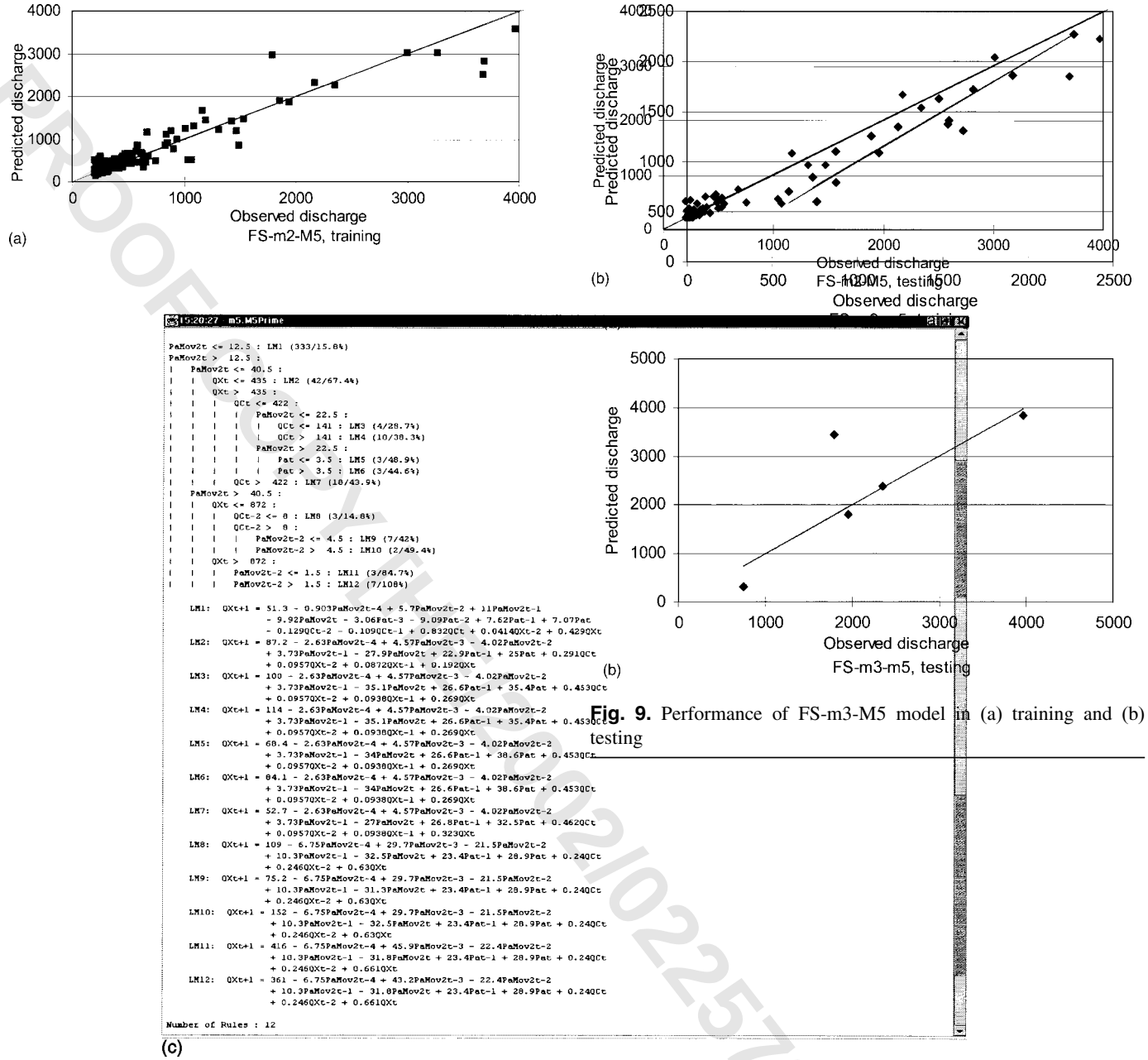


Fig. 7. Performance of FS-m2-M5 (module 2, $QX_{t-1} \leq 1{,}000$ m^3/s and $QX_t > 200$ m^3/s), (a) training; (b) testing; (c) FS-m2-M5 model (module 2, $QX_{t-1} \leq 1{,}000$ and $QX_t > 200$)

Fig. 9. Performance of FS-m3-M5 model in (a) training and (b) testing

predictions, and its prediction performance is close to that of the global model. Fig. 7(c) presents the resulting model tree.

Analysis of Errors for Modules 1 and 2 Figs. 8(a and b) presents the individual flood events hydrographs with the measured and calculated discharges. It can be seen that the data points of Module 1 either lie in the peak and recession part of the flood events, or lie in the rising limb of a flood with long duration. Thus the soil moisture is saturated and the prediction is not affected by flash flood at a tributary, so Module 1 gives a good prediction. However, the situation corresponding to the samples of Module 2 is more complex. If the point lies at the relatively low flow part, the prediction is still good. However the points that are close to peaks of flood events of short duration are not predicted well. This can be explained by the flood effect of the antecedent rainfall, and the heterogeneous distribution of rainfall that is not accounted for due to averaging.

From the generated M5 model trees of Module 2 [Fig. 7(c)], it can be seen that the intensive floods have been classified reasonably well (PaMo$v2_t > 40.5$), and even the distribution of the rainfall duration is modeled correctly. The data filtered into Module 2 does not exhaust the possibilities of building more accurate local models. Consider, for example, equation $LM8$ which is responsible for modeling the short duration of heavy rainfall in the middle reach or downstream part ($QX_t < 870, QC_{t-2} \leq 8$). The boundary of 870 for QX_t that corresponds to the antecedent rainfall (soil moisture) is somehow misleading: the value 870 is too

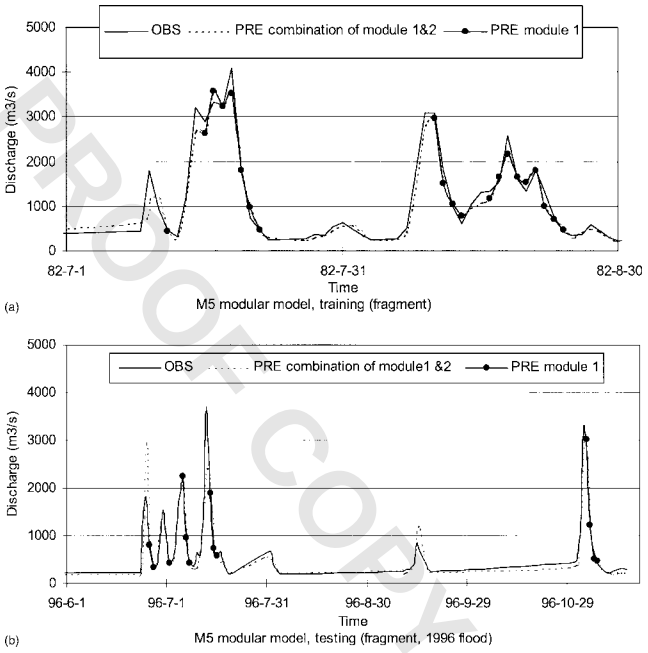


Fig. 8. Perfomance of M5 modular model (a) in training (fragment shown is 1982 flood) and (b) for testing (fragment shown is 1996 flood)

high for that. If, for example, there is small rainfall of long duration the discharge does not rise too much, but the soil becomes saturated. Based on these considerations a new class called Module 3 was constructed.

Module 3 Possibilities for improving accuracy lie in the further analysis of the physical characteristics of the catchment. Module 3 was used to represent the flood data due to the short but intensive rainfall after a period of dry weather, which is mainly included in Module 2. This type of flood is filtered out by the following rule: $\mathrm{Pa}_{t-1} > 50$ AND $\mathrm{PaMo}v2_{t-2} < 5$ AND $\mathrm{PaMo}v3_{t-4} < 5$. There are only 23 samples in the 21 years time series daily data; 18 in training and 5 samples in testing, respectively. The M5 method generates the following single-equation model:

$$LM1: QX_{t+1} = -43.2 + 75.6\mathrm{PaMo}v2_{t-4} + 16.4\mathrm{Pa}_t + 1.06QC_t + 1.52QX_t$$

As shown in Table 4 and Figs. 9(a and b), this formula works quite well both in training and testing. Even in extrapolation, for the sample in the testing data set whose discharge of 3,970 m^3/s is higher than the maximum value of training data, is still predicted correctly. An exception is the June 30 1996 flood sample with $QX_{t+1} = 1{,}800$ m^3/s, which is the only case with the short duration of extreme rainfall ($\mathrm{Pa}_{t-1} = 108, \mathrm{Pa}_t = 0$) in the whole catchment; by adding an additional condition, such as $\mathrm{Pa}_t > 5$, this sample can be easily filtered out.

Hybrid Model: Combining Artificial Neural Network and M5 Model Tree

An insignificant number of samples for Module 2 and the need to validate the results of linear regression models prompted the use of an alternative model—ANN. So, in addition to the M5 model built for Module 2 (FS-m2-M5), an ANN model (FS-m2-ANN) was also built. Such an approach of using different model types for different modules makes it possible to characterize the overall model as a hybrid model. The three models for Module 2 compared in Table 5 are: (1) FS-ANN model trained using the whole flood season data only, but for which the error is calculated only for the samples complying to the constraints of Module 2; (2) ANN trained on the samples of Module 2 (FS-m2-ANN); (3) M5 model tree trained on the samples of module 2 (FS-m2-M5).

From Table 5, it is clear that in comparison with the global ANN model, the hybrid model approach improves the prediction accuracy. The FS-m2-ANN model is slightly better than the modular model FS-m2-M5, and is far better than the ANN flood season model FS-ANN. Another merit of the hybrid approach is that smaller ANN models are easier to train. It is also possible to combine (e.g., by averaging) the predictions of models of various types making predictions for the same subdomain of input space, thus creating a committee machine.

In practice, the presented models constituting the mixture are trained and when the new input data arrives, it is first filtered to an appropriate model and then prediction is made.

Other Experiments: Changes in Discharge and Rainfall, Distributed Rainfall

Some of the previous studies indicated that using the changes in discharge and rainfall (or their derivatives) along with their values as inputs may increase the performance of ANN. Several ANN and M5 models were built for Module 2, and the main conclusion is that the performance of M5 models was getting worse, with the mixed results for ANN (RMSE decreased, but the MAE and MaxAE increased).

Another way to improve the modeling performance would be to use the distributed rainfall as input. Several experiments were conducted, but the lack of detailed data did not allow for drawing reliable conclusions.

Conclusions and Recommendations

1. Data-driven (machine learning) models are capable of performing rainfall-runoff forecasting, even for a rather complex catchment system. The performance of M5 model trees is comparable to that of the widely used MLP ANNs.
2. The advantageous features of M5 model trees if compared to ANN are:
   - the generated tree-like structure of linear models is reproducible and easy to understand for decision makers. It makes it possible for a hydrologist to have a good overview of the relationships between the hydrological characteristics;
   - the M5 algorithm allows one to easily generate a family of interpretable models with different number of component models/leaves and hence different robustness and accuracy;
   - training of M5 model trees is much faster than ANN and always converges; and
   - the knowledge encapsulated in a model tree may also help in parameters selection and assessing their relationships for other models, such as a conceptual hydrological model or an ANN.
3. The general prediction performance of the M5 model trees and ANN are good; the inaccuracies for the peak of some special flood events are mainly due to data-related problems, which include:
   - the unavailability of discharge data in a tributary Zhuguan

Table 2. M5 Model Performance for Module 1 ($QX_{t-1} > 1{,}000$ Using Flood Season Data)

	FS-m1-M5 model		FS-M5 model, extracted samples	
	Training	Testing	Training	Testing
Years	76–89	90–96	76–89	90–96
Number of samples	96	32	96	32
Mean absolute error	83.0	126.8	97.3	114.1
Maximum absolute error	548.0	509.1	1,173.5	1,183.6
Root mean squared error	127.0	176.1	176.6	222.2
Correlation coefficient	0.99	0.98	0.96	0.95

Table 3. M5 Model Performance for Module 2 (Flood Season Data with $QX_{t-1} < 1{,}000$ and $QX_t > 200$)

	FS-m2-M5 model		FS-M5 model, extracted samples	
	Training	Testing	Training	Testing
Years	76–89	90–96	76–89	90–96
Number of samples	433	167	433	167
Mean absolute error	83.9	103.0	81.3	109.3
Maximum absolute error	1,345	2,016	1,125	1,616
Root mean squared error	175.8	251.7	159.7	264.6
Correlation coefficient	0.961	0.938	0.968	0.933

Table 4. M5 Model Performance for Module 3 ($Pa_{t-1} > 50$ and $PaMov2_{t-2} < 5$ and $PaMov2_{t-4} < 5$ Using Flood Season data)

	FS-m3-M5 model		FS-M5 model (extracted samples)	
	Training	Testing	Training	Testing
Years	76–89	90–96	76–89	90–96
Number of samples	17	5	17	5
Mean absolute error	123.6	188.5	193.6	260.5
Maximum absolute error	311.0	437.5	417.6	511.0
Root mean squared error	145.8	241.5	222.7	316.4
Correlation coefficient	0.950	0.995	0.888	0.985

Table 5. M5 Model Trees and Artificial Neural Network (ANN) for Module 2

	Training 1976–1989			Testing 1995–1996		
Performance	FS-ANN, extracted samples	FS-m2-ANN	FS-m2-M5	FS-ANN, extracted samples	FS-m2-ANN	FS-m2-M5
Mean absolute error	125.2	91.7	97.3	121.8	24.5	20.8
Maximum absolute error	1,519	1,135	1,173	1,519	1,258	1,460
Root mean squared error	229.7	154.7	176.6	266.6	253.6	254.8
Correlation coefficient	0.929	0.968	0.96	0.893	0.925	0.91

River that can generate flash flood with a higher peak discharge;
- 24 h daily averaged data are too coarse to capture the rapid changes of rainfalls and discharges; and
- the heterogeneity of rainfall distribution in rather large catchments and the improper accounting for the subcatchment rainfall.

4. The inputs for M5 model trees are mainly selected according to the correlation analysis, which works very well. The prediction can be improved by using hydrological knowledge to refine the selection of inputs further and by a modular model approach that uses the rules offered by a hydrology expert. Such an approach would make it possible to filter out the flood samples with special hydrological characteristics, and then using the M5 model trees to classify the samples into more refined classes. The experiments with the so-called M5felx algorithm are reported by Solomatine and Siek (2003).
5. Using a hybrid model approach combining the M5 model trees and ANN allowed for further accuracy improvements. First the M5 model trees were used to classify the data into different classes and make predictions for most of them, and then ANN was used to forecast using the classified data set as input, to find the nonlinear relation in the data.

The following recommendations could be given:

1. In the situations when data-driven models are used to predict flash floods the data of higher frequency is needed.
2. Since the relation between the flood peak discharge and the

preceding rainfall is highly nonlinear it would be useful to use nonlinear models like ANNs or support vector machines (Dibike et al. 2001) in some of the branches of the M5 trees—this requires the modification of the M5 algorithm. The possibility of a better smoothing algorithm between the linear models of M5, for example using fuzzy methods, should be investigated as well.

3. Machine learning techniques like M5 model trees and ANNs can complement more traditional physically based models and expert judgments, but they cannot be used when a catchment undergoes considerable changes; for example due to urbanization. The combination of both types of models is a recommended approach to flood modeling.

More information on the use of machine learning tools in water-related issues can be found on the Web site on data-driven modeling ("Data-driven modeling" 2003).

Acknowledgments

The writers acknowledge the role of the Huai River Water Resource Commission of the Minister of Water Resource, China for the provision the data in this research, and the Dutch Embassy in China for its financial contribution to the group training of Chinese participants at IHE Delft in 1999–2001. Part of this work was performed in the framework of the project "Data mining, knowledge discovery and data-driven modelling" of the Delft Cluster research program supported by the Dutch government. The writers are also grateful to the anonymous reviewers for the useful comments.

References

Abrahart, R. J., See, L., and Kneale, P. E. (2001). "Applying saliency analysis to neural network rainfall-runoff modelling." *Comput. Geosci.*, 27, 921–928.

Becker, A., and Kundzewicz, Z. W. (1987). "Nonlinear flood routing with multilinear models." *Water Resour. Res.*, 23, 1043–1048.

Bhattacharya, B., and Solomatine, D. P. (2002a). "Application of artificial neural network in reconstructing stage-discharge relationship." *Proc., 4th Int. Conf. on Hydroinformatics*, Iowa.

Bhattacharya, B., and Solomatine, D. P. (2002b). "Application of artificial neural networks and M5 model trees to modelling stage-discharge relationship." *Proc., 2nd Int. Symp. on Flood Defence*, Beijing, China, B. S. Wu, Z. Y. Wang, G. Q. Wang, G. H. Huang, H. W. Fang, and J. C. Huang, eds., Science Press New York.

Birikundavyi, S., Labib, R., Trung, H. T., and Rousselle, J. (2002). "Performance of neural networks in daily streamflow forecasting." *J. Hydrologic Eng.*, 7(5), 392–398.

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1984). *Classification and regression trees*, Wadsworth, Belmont, Calif.

"Data-driven modelling." (2003). ⟨http://datamining.ihe.nl⟩ (22 Aug. 2003).

Dawson, C. W., and Wilby, R. (1998). "An artificial neural network approach to rainfall-runoff modelling." *Hydrol. Sci. J.*, 43(1), 47–66.

Dibike, Y. B., and Solomatine, D. P. (2001). "River flow forecasting using artificial neural network." *Phys. Chem. Earth*, 26(1), 1–7.

Dibike, Y. B., Velickov, S., Solomatine, D. P., and Abbott, M. B. (2001). "Model induction with support vector machines: Introduction and applications." *J. Comput. Civ. Eng.*, 15(3), 208–216.

Friedman, J. H. (1991). "Multivariate adaptive regression splines." *Ann. Stat.*, 19, 1–141.

Govindaraju, R. S., and Rao, A. R., ed. (2000). *Artificial neural networks in hydrology*, Kluwer Academic, Dordrecht, The Netherlands.

Haykin, S. (1999). *Neural networks: a comprehensive foundation*, 2nd Ed., Prentice-Hall, Upper Saddle River, N. J.

Hsu, K., Gupta, H. V., and Sorooshian, S. (1995). "Artificial neural network modeling of the rainfall-runoff process." *Water Resour. Res.*, 31(10), 2517–2530.

Kompare, B., Steinman, F., Cerar, U., and Dzeroski, S. (1997). "Prediction of rainfall runoff from catchment by intelligent data analysis with machine learning tools within the artificial intelligence tools." *Acta Hydrotechnica*, 16, 16 (in Slovene).

Minns, A. W., and Hall, M. J. (1996). "Artificial neural networks as rainfall-runoff models." *Hydrol. Sci. J.*, 41(3), 399–417.

"NeuralMachine." (2003). *Tools for neural networks modeling and global and evolutionary optimization*. ⟨http://www.data-machine.com⟩ (8 Mar. 2004).

"NeuroSolutions." (2003). *Neural Dimension Inc.*, ⟨http://www.nd.com⟩ (8 Mar 2004).

Quinlan, J. R. (1992). "Learning with continuous classes." *Proc., 5th Australian Joint Conf. on Artificial Intelligence*, Adams & Sterling, eds., World Scientific, Singapore, 343–348.

See, L., and Openshaw, S. (1998). "Using soft computing techniques to enhance flood forecasting on the river Ouse." *Proc., 3rd Int. Conf. on Hydroinformatics*, Copenhagen, Balkema Rotterdam, The Netherlands.

Setiono, R., Leow, W. K., and Zurada, J. M. (2002). "Extraction of rules from artificial neural networks for nonlinear regression." *IEEE Trans. Neural Netw.*, 13(3), 564–577.

Solomatine, D. P. (2002). "Applications of data-driven modelling and machine learning in control of water resources." *Computational intelligence in Control*, M. Mohammadian, R. A. Sarker, and X. Yao eds., Idea Group Publishing, 197–217.

Solomatine, D. P., and Dulal, K. (2003). "Model tree as an alternative to neural network in rainfall-runoff modeling." *Hydrol. Sci. J.*, 48(3), 399–411.

Solomatine, D. P., and Siek, M. B. (2003). "Flexibility and optimality in M5 model trees." *Proc., 3rd Int. Conf. on Hybrid Intelligent Systems (HIS'03)*, Melbourne, Australia.

Solomatine, D. P., and Torres, L. A. (1996). "Neural network approximation of a hydrodynamic model in optimizing reservoir operation." *Proc., 2nd Int. Conf. on Hydroinformatics*, Zurich, Switzerland, 201–206.

Witten, I. H., and Frank, E. (2000). *Data mining*, Morgan Kaufmann, San Francisco.

Zhao, R. J., and Liu, X. R. (1995). "The xinanjiang model." *Computer models of watershed hydrology*, V. P. Singh, ed., Water Resources Publications, Colo., 215–232.