

## Chapter 2

# Data-Driven Modelling: Concepts, Approaches and Experiences

D. Solomatine, L.M. See and R.J. Abrahart

**Abstract** Data-driven modelling is the area of hydroinformatics undergoing fast development. This chapter reviews the main concepts and approaches of data-driven modelling, which is based on computational intelligence and machine-learning methods. A brief overview of the main methods – neural networks, fuzzy rule-based systems and genetic algorithms, and their combination via committee approaches – is provided along with hydrological examples and references to the rest of the book.

**Keywords** Data-driven modelling · data mining · computational intelligence · fuzzy rule-based systems · genetic algorithms · committee approaches · hydrology

### 2.1 Introduction

Hydrological models can be characterised as physical, mathematical (including lumped conceptual and distributed physically based models) and empirical. The latter class of models, in contrast to the first two, involves mathematical equations that are not derived from physical processes in the catchment but from analysis of time series data. Examples include the unit hydrograph method, linear regression and ARIMA models. Recent developments in computational intelligence, in the area of machine learning in particular, have greatly expanded the capabilities of empirical modelling. The field which encompasses these new approaches is called data-driven modelling (DDM). As the name suggests, DDM is based on analysing the data about a system, in particular finding connections between the system state variables (input, internal and output variables) without explicit knowledge of the physical behaviour

---

D. Solomatine

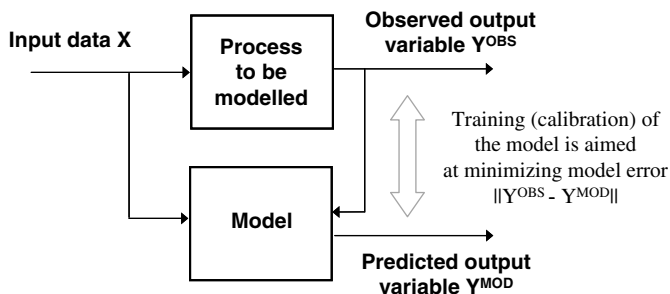
UNESCO-IHE Institute for Water Education, P.O. Box 3015, 2601 DA Delft, The Netherlands

L.M. See

School of Geography, University of Leeds, Woodhouse Lane, Leeds, LS2 9JT, UK

R.J. Abrahart

School of Geography, University of Nottingham, Nottingham, NG7 2RD, UK



**Fig. 2.1** General approach to modelling

of the system. These methods represent large advances on conventional empirical modelling and include contributions from the following overlapping fields:

- artificial intelligence (AI), which is the overarching study of how human intelligence can be incorporated into computers.
- computational intelligence (CI), which includes neural networks, fuzzy systems and evolutionary computing as well as other areas within AI and machine learning.
- soft computing (SC), which is close to CI, but with special emphasis on fuzzy rule-based systems induced from data.
- machine learning (ML), which was once a sub-area of AI that concentrates on the theoretical foundations used by CI and SC.
- data mining (DM) and knowledge discovery in databases (KDD) are focused often at very large databases and are associated with applications in banking, financial services and customer resources management. DM is seen as a part of a wider KDD. Methods used are mainly from statistics and ML.
- intelligent data analysis (IDA), which tends to focus on data analysis in medicine and research and incorporates methods from statistics and ML.

Data-driven modelling is therefore focused on CI and ML methods that can be used to build models for complementing or replacing physically based models. A machine-learning algorithm is used to determine the relationship between a system's inputs and outputs using a training data set that is representative of all the behaviour found in the system (Fig. 2.1).

Once the model is trained, it can be tested using an independent data set to determine how well it can generalise to unseen data. In the next section, the main DDM techniques are discussed.

## 2.2 An Overview of Data-Driven Modelling Techniques

This section describes the most popular computational intelligence techniques used in hydrological modelling, including neural networks, fuzzy rule-based systems, genetic algorithms, as well as approaches to model integration.

### **2.2.1 Neural Networks**

Neural networks are a biologically inspired computational model, which is based on the way in which the human brain functions. There has been a great deal written on this subject; see, e.g. Beale and Jackson (1990) and Bishop (1995). Neural network models are developed by training the network to represent the relationships and processes that are inherent within the data. Being essentially non-linear regression models, they perform an input–output mapping using a set of interconnected simple processing nodes or neurons. Each neuron takes in inputs either externally or from other neurons and passes it through an activation or transfer function such as a logistic or sigmoid curve. Data enter the network through the input units arranged in what is called an input layer. These data are then fed forward through successive layers including the hidden layer in the middle to emerge from the output layer on the right. The inputs can be any combination of variables that are thought to be important for predicting the output; therefore, some knowledge of the hydrological system is important.

The hidden layer is the essential component that allows the neural network to learn the relationships in the data as shown originally in Rumelhart et al. (1986); these authors popularised also the backpropagation algorithm for training a feed-forward neural network but the principle was first developed by Werbos in 1974 (see Werbos, 1994). This configuration is also referred to as a multilayer perceptron (MLP) and it represents one of the most commonly used neural networks (Kasabov, 1996). The backpropagation algorithm is a variation of a gradient descent optimisation algorithm that minimises the error between the predicted and actual output values. The weighted connections between neurons are adjusted after each training cycle until the error in the validation data set begins to rise. The validation data set is a second data set that is given to the network to evaluate during training. If this approach is not used, the network will represent the training data set too well and will then be unable to generalise to an unseen data set or a testing data set. Once the networks are trained to satisfaction, it can be put to operation when the new input data are passed through the trained network in its non-training mode to produce the desired model outputs. In order to validate the performance of the trained network before it is put into real operation, however, the operation mode is usually imitated by using the test data set. An important way to help promote generalisation to unseen data is to ensure that the training data set contains a representative sample of all the behaviour in the data. This could be achieved by ensuring that all three data sets – training, validation and test – have similar statistical properties. Note that the test set cannot be used to change the properties of the trained model.

The use of ANNs has many successful applications in hydrology, in modelling rainfall-runoff processes: Hsu et al. (1995); Minns and Hall (1996); Dawson and Wilby (1998); Dibike et al. (1999); Abrahart and See (2000); Govindaraju and Ramachandra Rao (2001); replicating the behaviour of hydrodynamic/hydrological models of a river basin where ANNs are used to provide optimal control of a reservoir (Solomatine and Torres, 1996); building an ANN-based intelligent controller

for real-time control of water levels in a polder (Lobbrecht and Solomatine, 1999); and modelling stage-discharge relationships (Sudheer and Jain, 2003; Bhattacharya and Solomatine, 2005).

Section two of the book specifically deals with neural network applications in different areas of hydrology. In Chap. 3, Abrahart and See provide a guide to neural network modelling in hydrology. The next six chapters are applications of neural networks to rainfall-runoff modelling. Dawson demonstrates how neural networks can be used to estimate floods at ungauged catchments (Chap. 4). Jain decomposes the flood hydrograph into subsets and trains a neural network on each individual subset (Chap. 5). Coulibaly considers the behaviour of neural networks on nonstationary time series (Chap. 6). See et al. examine the behaviour of the hidden neurons of a neural network as a way of providing physical interpretation of a neural network rainfall-runoff model (Chap. 7). De Vos and Rientjes look at the effects of modifying the objective function used by neural network rainfall-runoff models (Chap. 8), while Toth considers the influence on flow forecasts when changing the temporal resolution of the input and output variables (Chap. 9). The final two chapters are applications of neural networks in groundwater (Mohammadi, Chap. 10) and sediment modelling (White et al., Chap. 11).

### ***2.2.2 Fuzzy Rule-Based Systems (FRBS)***

Fuzzy rule-based systems use fuzzy logic for inference. Fuzzy logic is based on fuzzy set theory in which binary set membership has been extended to include partial membership ranging between 0 and 1 (Zadeh, 1965). Fuzzy sets, in contrast to their crisp counterparts, have gradual transitions between defined sets, which allow for the uncertainty associated with these concepts to be modelled directly. After defining each model variable with a series of overlapping fuzzy sets, the mapping of inputs to outputs can be expressed as a set of IF-THEN rules, which can be entirely specified from expert knowledge, or from data. However, unlike neural networks, fuzzy models are prone to a rule explosion, i.e. as the number of variables or fuzzy sets per variable increases, there is an exponential increase in the number of rules, which makes it difficult to specify the entire model from expert knowledge alone (Kosko, 1997). Different automated methods for optimising fuzzy models are now available (Wang, 1994), including neural networks and genetic algorithms.

The fuzzy sets and rules are referred to as the fuzzy model knowledgebase. Crisp inputs to the model are first fuzzified via this knowledgebase, and a fuzzy inference engine is then used to process the rules in parallel via a fuzzy inference procedure such as max-min or max-product operations (Jang et al., 1997). The fuzzy solution surface resulting from the execution of the rulebase is defuzzified to produce the system output(s). Fuzzy IF-THEN rules can also be comprised of functional consequents, usually of a linear or polynomial form, in a formulation referred to as a TSK model (Takagi and Sugeno, 1985; Sugeno and Kang, 1988). The crisp inputs are fuzzified according to the fuzzy set definitions, combined via the inference engine,

and the functional consequents are weighted by the memberships that result from the execution of the rules. The overall result is a weighted average of the equations as more than one rule can fire positively during a single pass of the rulebase.

Fuzzy logic has found multiple successful applications, mainly in control theory (see, e.g. Kosko, 1997). As mentioned previously, fuzzy rule-based systems can be built by interviewing human experts, or by processing historical data and thus forming a data-driven model. The basics of the latter approach and its use in a number of water-related applications can be found in Bárdossy and Duckstein (1995). FRBS were effectively used for drought assessment (Pesti et al., 1996), prediction of precipitation events (Abebe et al., 2000), analysis of groundwater model uncertainty (Abebe et al., 2000), control of water levels in polder areas (Lobrecht and Solomatine, 1999) and modelling rainfall-discharge dynamics (Vernieuwe et al., 2005).

Part III of the book deals specifically with fuzzy systems applications in hydrology. Mujumdar provides an overview of fuzzy logic-based approaches in water resource systems (Chap. 12). Examples of fuzzy rule-based flood forecasting models are then presented by Bárdossy (Chap. 13) and Jacquin and Shamseldin (Chap. 14), while Cluckie et al. (Chap. 15) consider the use of an adaptive neuro-fuzzy inference system in the development of a real-time flood forecasting expert system. Finally, the section ends with Chap. 16 by Makropoulos et al. who examine the use of fuzzy inference for building hydrological decision support systems.

### ***2.2.3 Genetic Algorithms (GAs) in Model Optimisation***

Genetic algorithms (GAs) (or, more widely, evolutionary algorithms) are non-linear search and optimisation methods inspired by the biological processes of natural selection and survival of the fittest (Goldberg, 1989). They do not belong to the class of data-driven models, but since they are widely used in optimising models, we consider them here as well. Genetic (evolutionary) algorithms are typically attributed to the area of computational intelligence.

Unlike other methods such as hillclimbing and simulated annealing, a GA, like other randomised search algorithms such as Adaptive Cluster Covering (Solomatine, 1999), exhibits implicit parallelism, considering many points at once during the search process and thereby reduces the chance of converging to a local optimum. GAs also use probabilistic rules in the search process, and they can generally outperform conventional optimisation techniques on difficult, discontinuous and multimodal functions. Despite their unique and adaptive search capabilities, there is no guarantee that GAs will find the global solution; however, they can often find an acceptable one quite quickly. A detailed introductory survey can be found in Reeves and Rowe (2003).

The basic unit of a GA is the gene, which in biological terms represents a given characteristic of an individual, such as eye colour. In a GA, a gene represents a parameter that is being optimised. An individual or chromosome is simply the combined set of all the genes, i.e. all the parameters needed to generate the solution. To

start the search, a population of these individuals or strings is randomly generated. Each string is then evaluated by a fitness or objective function according to some measure of performance. This represents the success of the solution and is analogous to the survival ability of an individual within the population. In order to evolve better performing solutions, the fittest members of the population are selected and exposed to a series of genetic operators, which produce offspring for the next generation. The least fit solutions, on the other hand, will die out through natural selection as they are replaced by new, recombined individuals.

The main genetic operator is crossover in which a position along the bit string is randomly chosen that cuts two parent chromosomes into two segments, which are then swapped. The new offspring are comprised of a different segment from each parent and thereby inherit bits from both. The occurrence of crossover is determined probabilistically; when crossover is not applied, offspring are simply duplicates of the parents, thereby giving each individual a chance of passing on a pure copy of its genes into the gene pool. The second main genetic operator is mutation, which is applied to each of the offspring individually after crossover. Mutation can alter the bits in a string, but with an extremely low probability. Crossover allows the genetic algorithm to explore new areas in the search space and gives the GA the majority of its searching power while mutation exploits existing areas to find a near optimal solution and essentially provides a small amount of random search to ensure that no point in the search space has a zero probability of being examined. The newly generated offspring are then placed back into the population, and the exercise is repeated for many generations until a set of user-specified termination criteria are satisfied, such as exceeding a preset number of generations or if no improved solution is found after a given period of time. Over many generations, a whole new population of possible solutions, which possess a higher proportion of the characteristics found in the fitter members of the previous generation, is produced.

GAs are a very useful tool for handling difficult problems where conventional techniques cannot cope, or alternatively, they can be used to improve existing methods through hybridisation. For example, fuzzy logic rule-based models can be entirely optimised by a GA in a completely inductive approach, or expert knowledge can be used to specify the rules or membership functions, leaving the GA to optimise only the unknown parts of the model. See Cordón and Herrera (1995) and Karr (1991) for more details of fuzzy logic model optimisation using a GA. GAs can be also used to optimise other data-driven models like neural networks (Yao and Liu, 1997); this approach was also used by Parasuraman and Elshorbagy (Chap. 28 in this volume).

Part IV of the book is devoted to examples of hydrological optimisation by genetic algorithms. Savic, in Chap. 17, provides an overview of global and evolutionary optimisation in hydrology and water management problems. Tsai considers the use of a GA in a groundwater problem (Chap. 18). Efstratiadis and Koutsoyiannis (Chap. 19) and Khu et al. (Chap. 20) look at two different multi-objective versions of evolutionary optimisation algorithms for model calibration (in the latter chapter, in order to reduce the number of model runs, a meta-model for the error surface approximation is used). Jain in Chap. 21 considers the calibration of a hydrological

model using real-coded GAs. In Chap. 22, Solomatine and Vojinovic compare a series of different global optimisation algorithms for model calibration. In the final chapter of this section by Heppenstall et al., a cooperative co-evolutionary approach is demonstrated that evolves neural network rainfall-runoff models.

### 2.2.4 Other Approaches

In addition to neural networks and fuzzy rule-based systems, there are other data-driven methods that have been used successfully to solve hydrological problems. These methods are still less known if compared to ANN and FRBS, and the chapters covering these are in Part V entitled “Emerging Technologies”. Methods that are currently finding a lot of researchers’ attention are listed below:

- Genetic Programming and evolutionary regression
- Chaos theory and non-linear dynamics
- Support vector machines

Additionally, machine-learning methods for clustering and classification are often used to support regression methods considered above, as well as methods of instance-based learning (IBL), used for both classification and regression.

*Genetic programming* (GP) is a method for evolving equations by taking various mathematical building blocks such as functions, constants and arithmetic operations and combining them into a single expression and was originally developed by Koza (1992). Evolutionary regression is similar to GP but the goal is to find a regression equation, typically a polynomial regression, where the coefficients are determined through an evolutionary approach such as a GA. Examples of hydrological applications include the work by Khu et al. (2001), who applied GP to real-time runoff forecasting for a catchment in France, and Giustolisi and Savic (2006) who used evolutionary regression for ground water and river temperature modelling.

*Classification* is a method for partitioning data into classes and then attributing data vectors to these classes. The output of a classification model is a class label, rather than a real number like in regression models. The classes are typically created such that they are far from one another in attribute space but the points within a class are as tightly clustered around the centre point as possible. Examples of classification techniques include  $k$ -nearest neighbour, Bayesian classification, decision tree classification (Witten and Frank, 2000) and support vector machines (SVM) (Vapnik, 1998).

There are many examples of applying classification methods in hydrology. For example, Frapporti et al. (1993) used fuzzy c-means clustering to classify shallow Dutch groundwater sites, Hall and Minns (1999) used a SOFM to classify catchments into subsets on which ANNs were then applied to model regional flood frequency. Hannah et al. (2000) used clustering for finding groups of hydrographs on the basis of their shape and magnitude; clusters are then used for classification by experts. Harris et al. (2000) applied clustering to identify the classes of river regimes.

Velickov et al. (2000) used self-organising feature maps (Kohonen networks) as clustering methods, and SVM as a classification method in aerial photograph interpretation with the purpose of subsequent construction of flood severity maps. Solomatine et al. (2007) used decision trees and  $k$ -NN in classification of river flow levels according to their severity in a flood forecasting problem in Nepal. Zhang and Song (2006) used a combination of SOF and ART networks for special pattern identification of soil moisture. In this volume, Parasuraman and Elshorbagy (Chap. 28) used clustering before applying ANNs to forecasting streamflow.

In *instance-based learning* (IBL), classification or prediction is made by combining observations from the training data set that are close to the new vector of inputs (Mitchell, 1997). This is a local approximation and works well in the immediate neighbourhood of the current prediction instance. The nearest neighbour classifier approach classifies a given unknown pattern by choosing the class of the nearest example in the training set as measured by some distance metric, typically Euclidean. Generalisation of this method is the  $k$ -nearest neighbour ( $k$ -NN) method. For a discrete valued target function, the estimate will just be the most common value among  $k$  training examples nearest to  $\mathbf{x}_q$ . For real-valued target functions, the estimate is the mean value of the  $k$ -nearest neighbouring examples. Locally weighted regression (LWR) is a further extension in which a regression model is built on  $k$ -nearest instances.

Applications of IBL in water-related problems mainly refer to the simplest method, viz  $k$ -NN. Karlsson and Yakowitz (1987) showed the use of this method in hydrology, focusing however only on (single-variate) time series forecasts. Galeati (1990) demonstrated the applicability of the  $k$ -NN method (with the vectors composed of the lagged rainfall and flow values) for daily discharge forecasting and favourably compared it to the statistical ARX model. Shamseldin and O'Connor (1996) used the  $k$ -NN method for adjusting the parameters of the linear perturbation model for river flow forecasting. Toth et al. (2000) compared the  $k$ -NN approach to other time series prediction methods in a problem of short-term rainfall forecasting. Solomatine et al. (2007) considered IBL in a wider context of machine learning and tested their applicability in short-term hydrologic forecasting.

*Chaos theory* and non-linear dynamics can be used for time series prediction when the time series data are of sufficient length and carry enough information about the behaviour of the system (Abarbanel 1996). The main idea is to represent the state of the system at time  $t$  by a vector in  $m$ -dimensional state space. If the original time series exhibits chaotic properties, then its equivalent trajectory in phase space has properties allowing for accurate prediction of future values of the independent variable. Hydrological examples include the work by Solomatine et al. (2000) and Velickov et al. (2003), who used chaos theory to predict the surge water level in the North Sea close to Hook of Holland. For two-hourly predictions, the error was as low as 10 cm and was at least on par with the accuracy of hydrodynamic models. Babovic et al. (2000) used a chaos theory-based approach for predicting water levels at the Venice lagoon, and Phoon et al. (2002) for forecasting hydrologic time series.



*Support vector machines* (SVM) is a relatively new important method based on the extension of the idea of identifying a line (or a plane or some surface) that separates two classes in classification. It is based on statistical learning theory initiated by V. Vapnik in the 1970s (Vapnik, 1998). This classification method has also been extended to solving prediction problems, and in this capacity was used in hydrology-related tasks. Dibike et al. (2001) and Liong and Sivapragasam (2002) reported using SVMs for flood management and in prediction of water flows and stages. Chapter 26 by Yu et al. provides a recent example of flood stage forecasting using SVM.

## 2.3 Combination and Integration of Models

### 2.3.1 Modular Models

Since natural processes are complex, it is sometimes not possible to build a single global model that adequately captures the system behaviour. Instead the training data can be split into a number of subsets, and separate specialised models can be built on each subset. These models are called local or expert models, and this type of modular model is sometimes called a committee machine (CM) (Haykin, 1999). Two key decisions must be made when building a CM. The first is how to split the data and the second is how to combine the individual models to produce a final output.

The group of statistically driven approaches with “soft” splits of input space is represented by *mixtures of experts* (Jordan and Jacobs, 1995), *bagging* (Breiman, 1996) and *boosting* (Freund and Schapire, 1997).

Another quite popular approach is to build an *ensemble* of models and to combine the model results by some averaging scheme; this approach is widely used in meteorology.

Yet another group of methods does not combine the outputs of different models but explicitly uses only one of them, i.e. the most appropriate one (a particular case when the weights of other expert models are zero). Such methods use “hard” splits of input space into regions. Each individual local model is trained individually on subsets of instances contained in these regions, and finally the output of only one specialised expert is taken into consideration. This can be done manually by experts on the basis of domain knowledge. Another way is to use information theory to perform such splits and to perform splitting progressively; examples are decision trees, regression trees, MARS (Breiman et al., 1984) and M5 model trees (Quinlan, 1992).

These machine-learning techniques use the following idea: split the parameter space into areas (subspaces) and build a separate regression model in each of them. Tree-based models are constructed by a divide-and-conquer method. The set  $T$  is either associated with a leaf, or some test is chosen that splits  $T$  into subsets corresponding to the test outcomes and the same process is applied recursively to the

subsets. If models in the leaves are of zero order (numeric constants) then this model is called a regression tree (Breiman et al., 1984); if the models are of first order (linear regression models) then the model is referred to as an M5 model tree (Quinlan 1992; “M5” stands for “Model trees, version 5”). The splitting criterion in both algorithms is based on treating the standard deviation of the output values that reach a node as a measure of the error at that node, and calculating the expected reduction in this error as a result of testing each attribute at that node. Solomatine and Dulal (2003) used M5 model trees in rainfall-runoff modelling of a catchment in Italy.

Note that to denote a combination of models (or modular models), various authors use different terms: in machine learning these are typically mixtures of experts and committee machines; when other models are combined the term “data fusion” is often used – see, for example, an earlier chapter by Abrahart and See (2002) where six alternative methods to combine data-driven and physically based hydrologic models were compared.

Two chapters in Part V of the book deal with such modular approaches that lately are becoming more and more popular. Solomatine starts off the part in Chap. 24 with an overview of modular models. Stravs et al. then provide an example of precipitation interception modelling using M5 model trees (Chap. 25).

It is also possible to use a combination of models in a given solution. If these models work together to create a single solution they are referred to as *hybrid models*. If, on the other hand, this combination of models is not used to model the same process but instead they work with each other, then this combination is referred to as a *complementary model*. Examples of hybrid models include a study by See and Openshaw (2000) where several types of models were combined using an averaging scheme, a Bayesian approach and two fuzzy logic models; the combination of physically based models using a fuzzy model (Xiong et al., 2001); and the combination of data-driven models of various types trained on subsets of the original data set (Solomatine and Xue, 2004). Examples of complementary models include updating a physically based model using a neural network (Shamseldin and O'Connor, 2001; Lekkas et al., 2001; Abebe and Price, 2004). Solomatine et al. (2007) built an ANN-based rainfall-runoff model where its outputs were corrected by an instance-based model.

### 2.3.2 *Integration of Models*

The final section of the book deals specifically with model integration and different hydrological examples. The focus here is on the technological developments in the area of model integration, i.e. the integration of models and a variety of data sources. The chapters by Fortune and Gijssbers and by Werner describe the architectures of modern model integration frameworks (OpenMI and Delft-FEWS, respectively). The chapter by Xuan and Cluckie addresses the issue of uncertainty propagation in the integrated model including numerical weather prediction and

hydrologic components. Betts et al. describe an integrated modelling framework implemented for the Yangtze River basin in China. Finally, the chapter by O’Kane addresses the issue of incorporation of data into models and of “social calibration” of models – involving stakeholders with the best knowledge of the aquatic system in question, rather than purely numerical calibration without an insight – which is especially important when models are to be used in education and in real-life-decision-making frameworks. An extensive study of flooding in the polder landscape of the Lower Feale catchment in Ireland is used as illustration of the principle.

## 2.4 Conclusions

Data-driven modelling and computational intelligence in general have proven their applicability to various water-related problems: modelling, short-term forecasting, data classification, reservoir optimisation, building flood severity maps based on aerial or satellite photos, etc. Data-driven models would be useful in solving a practical problem or modelling a particular system or process if (1) a considerable amount of data describing this problem is available; (2) there are no considerable changes to the modelled system during the period covered by the model. Such models are especially effective if it is difficult to build knowledge-driven simulation models (e.g. due to lack of understanding of the underlying processes), or the available models are not adequate enough. It is of course always useful to have modelling alternatives and to validate the simulation results of physically based models with data-driven ones, or vice versa.

The developers and users of data-driven models should realise that such models typically do not really represent the physics of a modelled process; they are just devices used to capture relationships between the relevant input and output variables. However, such devices could be more accurate than process models since they are based on objective information (i.e. the data), and the latter may often suffer from incompleteness in representing the modelled process.

A contemporary trend is to combine data-driven models, i.e. combining models of different types and which follow different modelling paradigms (thus constituting *hybrid* models), including the combination with physically based models in an optimal way. One of the challenges for hydroinformatitians in this respect is to ensure that data-driven models are properly incorporated into the existing modelling and decision support frameworks.

## References

- Abarbanel HDI (1996) Analysis of Observed Chaotic Data. Springer-Verlag: New York.  
 Abebe AJ, Solomatine DP, Venneker R (2000) Application of adaptive fuzzy rule-based models for reconstruction of missing precipitation events. Hydrological Sciences Journal 45(3): 425–436.

- Abebe AJ, Guinot V, Solomatine DP (2000) Fuzzy alpha-cut vs. Monte Carlo techniques in assessing uncertainty in model parameters. *Proc. 4th Int. Conference on Hydroinformatics*, Cedar Rapids.
- Abebe AJ, Price RK (2004) Information theory and neural networks for managing uncertainty in flood routing. *ASCE Journal of Computing in Civil Engineering* 18(4): 373–380.
- Abrahart RJ, See L (2000) Comparing neural network and autoregressive moving average techniques for the provision of continuous river flow forecast in two contrasting catchments. *Hydrological Processes* 14: 2157–2172.
- Abrahart RJ, See L (2002) Multi-model data fusion for river flow forecasting: an evaluation of six alternative methods based on two contrasting catchments. *Hydrology and Earth System Sciences* 6(4): 655–670.
- Babovic V, Keijzer M, Stefansson M (2000) Optimal embedding using evolutionary algorithms. *Proc. 4th Int. Conference on Hydroinformatics*, Cedar Rapids.
- Bárdossy A, Duckstein L (1995) *Fuzzy Rule-Based Modeling with Applications to Geophysical, Biological and Engineering Systems*. CRC press Inc: Boca Raton, Florida, USA.
- Beale R, Jackson T (1990) *Neural Computing: An Introduction*, Adam Hilger: Bristol.
- Bhattacharya B, Solomatine DP (2005) Neural networks and M5 model trees in modelling water level – discharge relationship. *Neurocomputing* 63: 381–396.
- Bishop CM (1995) *Neural Networks for Pattern Recognition*. Clarendon Press: Oxford.
- Breiman L, Friedman JH, Olshen RA, Stone CJ (1984) *Classification and regression trees*. Wadsworth International: Belmont.
- Breiman L (1996) Stacked regressor. *Machine Learning* 24(1): 49–64.
- Cordón O, Herrera F (1995) A general study on genetic fuzzy systems. In: Winter G, Périaux J, Gálan M, Cuesta P (eds) *Genetic Algorithms in Engineering and Computer Science*. John Wiley & Sons, Chichester, pp. 33–57.
- Dawson CW, Wilby R (1998) An artificial neural network approach to rainfall-runoff modelling. *Hydrological Sciences Journal* 43(1): 47–66.
- Dibike Y, Solomatine DP, Abbott MB (1999) On the encapsulation of numerical-hydraulic models in artificial neural network. *Journal of Hydraulic Research* 37(2): 147–161.
- Dibike YB, Velickov S, Solomatine DP, Abbott MB (2001) Model induction with support vector machines: introduction and applications. *ASCE Journal of Computing in Civil Engineering* 15(3): 208–216.
- Frapporti G, Vriend SP, Van Gaans PFM (1993) Hydrogeochemistry of the shallow Dutch groundwater: classification of the national groundwater quality monitoring network. *Water Resources Research*, 29(9): 2993–3004.
- Freund Y, Schapire R (1997) A decision-theoretic generalisation of on-line learning and an application of boosting. *Journal of Computer and System Science* 55(1): 119–139.
- Galeati G (1990) A comparison of parametric and non-parametric methods for runoff forecasting. *Hydrology Sciences Journal* 35(1): 79–94.
- Giustolisi O, Savic DA (2006) A symbolic data-driven technique based on evolutionary polynomial regression. *Journal of Hydroinformatics* 8(3): 202–207.
- Goldberg DE (1989) *Genetic Algorithms in Search Optimisation and Machine Learning*. Addison-Wesley: USA.
- Govindaraju RS, Ramachandra Rao A (eds) (2001) *Artificial Neural Networks in Hydrology*. Kluwer: Dordrecht.
- Hall MJ, Minns AW (1999) The classification of hydrologically homogeneous regions. *Hydrological Sciences Journal* 44: 693–704.
- Hannah DM, Smith BPG, Gurnell AM, McGregor GR (2000) An approach to hydrograph classification. *Hydrological Processes* 14: 317–338.
- Harris NM, Gurnell AM, Hannah DM, Petts GE (2000) Classification of river regimes: a context for hydrogeology. *Hydrological Processes* 14: 2831–2848.
- Haykin S (1999) *Neural Networks: A Comprehensive Foundation*. McMillan: New York.
- Hsu KL, Gupta HV, Sorooshian S (1995) Artificial neural network modelling of the rainfall-runoff process. *Water Resources Research* 31(10): 2517–2530.

- Jang J-S, Sun C-T, Mizutani E (1997) *Neuro-Fuzzy and Soft Computing*. Prentice Hall.
- Jordan MI, Jacobs RA (1995) Modular and hierarchical learning systems. In: Arbib M (ed) *The Handbook of Brain Theory and Neural Networks*. MIT Press, Cambridge.
- Karr CL (1991) Genetic algorithms for fuzzy logic controllers. *AI Expert* 6: 26–33.
- Karlsson M, Yakowitz S (1987) Nearest neighbour methods for non-parametric rainfall runoff forecasting. *Water Resources Research* 23(7): 1300–1308.
- Kasabov K (1996) *Foundations of Neural Networks, Fuzzy Systems and Knowledge Engineering*. MIT Press: Cambridge.
- Khu S-T, Liong S-Y, Babovic V, Madsen H, Muttil N (2001) Genetic programming and its application in real-time runoff forecasting. *Journal of the American Water Resources Association* 37(2): 439–451.
- Kosko B (1997) *Fuzzy engineering*. Prentice-Hall: Upper Saddle River.
- Koza JR (1992) *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press: Cambridge, MA
- Lekkas DF, Imrie CE, Lees MJ (2001) Improved non-linear transfer function and neural network methods of flow routing for real-time forecasting. *Journal of Hydroinformatics* 3(3): 153–164.
- Liong SY, Sivapragasam C (2002) Flood stage forecasting with SVM. *Journal of American Water Resources Association* 38(1): 173–186.
- Lobrecht AH, Solomatine DP (1999) Control of water levels in polder areas using neural networks and fuzzy adaptive systems. In: Savic D, Walters G (eds) *Water Industry Systems: Modelling and Optimization Applications*. Research Studies Press Ltd., Baldock, pp. 509–518.
- Minns AW, Hall MJ (1996) Artificial neural network as rainfall-runoff model. *Hydrological Sciences Journal* 41(3): 399–417.
- Mitchell TM (1997) *Machine Learning*. McGraw-Hill: New York.
- Pesti G, Shrestha BP, Duckstein L, Bogárdi I (1996) A fuzzy rule-based approach to drought assessment. *Water Resources Research* 32(6): 1741–1747.
- Phoon KK, Islam MN, Liaw CY, Liong SY (2002) A practical inverse approach for forecasting nonlinear hydrological time series. *ASCE Journal of Hydrologic Engineering*, 7(2): 116–128.
- Quinlan JR (1992) Learning with continuous classes. In: Adams A, Sterling L (eds) *Proc. AI'92, 5th Australian Joint Conference on Artificial Intelligence*. World Scientific, Singapore, pp. 343–348.
- Reeves CR, Rowe JE (2003) *Genetic Algorithms – Principles and Perspectives. A Guide to GA Theory*, Kluwer Academic Publishers Group.
- Rumelhart D, Hinton G, Williams R (1986) Learning internal representations by error propagation. In: Rumelhart D, McClelland J, (eds) *Parallel Distributed Processing: Explorations in the microstructure of cognition. Volume 1: Foundations*. MIT Press, Cambridge, MA, pp. 318–363.
- See LM, Openshaw S (2000) A hybrid multi-model approach to river level forecasting. *Hydrological Sciences Journal* 45: 523–536.
- Shamseldin AY, O'Connor KM (1996) A nearest neighbour linear perturbation model for river flow forecasting. *Journal of Hydrology* 179: 353–375.
- Shamseldin AY, O'Connor KM (2001) A non-linear neural network technique for updating of river flow forecasts. *Hydrology and Earth System Sciences* 5 (4): 557–597.
- Solomatine DP, Torres LA (1996) Neural network approximation of a hydrodynamic model in optimizing reservoir operation. *Proc. 2nd Int. Conference on Hydroinformatics, Balkema: Rotterdam*, 201–206.
- Solomatine DP (1999) Two strategies of adaptive cluster covering with descent and their comparison to other algorithms. *Journal of Global Optimization* 14(1): 55–78.
- Solomatine DP, Rojas C, Velickov S, Wust H (2000) Chaos theory in predicting surge water levels in the North Sea. *Proc. 4th Int. Conference on Hydroinformatics, Cedar-Rapids*.
- Solomatine DP, Dulal KN (2003) Model tree as an alternative to neural network in rainfall-runoff modelling. *Hydrological Sciences J.* 48(3): 399–411.
- Solomatine DP, Xue Y (2004) M5 model trees and neural networks: application to flood forecasting in the upper reach of the Huai River in China. *ASCE J. Hydrologic Engineering* 9(6): 491–501.

- Solomatine DP, Maskey M, Shrestha DL (2007) Instance-based learning compared to other data-driven methods in hydrologic forecasting. *Hydrological Processes*, 21 (DOI: 10.1002/hyp.6592).
- Sudheer KP, Jain SK (2003) Radial basis function neural network for modeling rating curves. *ASCE Journal of Hydrologic Engineering* 8(3): 161–164.
- Sugeno M, Kang GT (1988) Structure identification of fuzzy model, *Fuzzy Sets and Systems* 28(1): 15–33.
- Takagi T, Sugeno M (1985) Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetic* SMC-15: 116–132.
- Toth E, Brath A, Montanari A (2000) Comparison of short-term rainfall prediction models for real-time flood forecasting. *Journal of Hydrology* 239: 132–147.
- Vapnik VN (1998) *Statistical Learning Theory*. Wiley & Sons: New York.
- Velickov S, Solomatine DP, Yu X, Price RK (2000) Application of data mining techniques for remote sensing image analysis. *Proc. 4th Int. Conference on Hydroinformatics, USA*.
- Velickov S, Solomatine DP, Price RK (2003) Prediction of nonlinear dynamical systems based on time series analysis: issues of entropy, complexity and predictability. *Proc. of the XXX IAHR Congress, Thessaloniki, Greece*.
- Vernieuwe H, Georgieva O, De Baets B, Pauwels VRN, Verhoest NEC, De Troch FP (2005) Comparison of data-driven Takagi–Sugeno models of rainfall–discharge dynamics. *Journal of Hydrology* 302(1–4): 173–186.
- Wang LX (1994) *Adaptive Fuzzy Systems and Control: Design and Stability Analysis*. PTR Prentice Hall Inc.: Englewood Cliffs, NJ.
- Werbos PJ (1994) *The Roots of Backpropagation*. New York: John Wiley & Sons (includes Werbos's 1974 Ph.D. thesis, *Beyond Regression*).
- Witten IH, Frank E (2000) *Data Mining*. Morgan Kaufmann: San Francisco.
- Xiong LH, Shamseldin AY, O'Connor KM (2001) A non-linear combination of the forecasts of rainfall–runoff models by the first-order Takagi-Sugeno fuzzy system. *Journal of Hydrology* 245(1–4): 196–217.
- Yao X, Liu Y (1997) A new evolutionary system for evolving artificial neural networks. *IEEE Transactions on Neural Networks* 8(3): 694–713.
- Zadeh LA (1965) Fuzzy sets. *Information and Control* 8: 338–353.
- Zhang X, Song X (2006) Spatial pattern identification of soil moisture based on self-organizing neural networks. *Proc. 7th Intern. Conf on Hydroinformatics, Nice, September*.

Practical Hydroinformatics

Computational Intelligence and Technological

Developments in Water Applications

Abrahart, R.J.; See, L.M.; Solomatine, D.P. (Eds.)

2008, XVI, 506 p. 243 illus., 5 illus. in color., Hardcover

ISBN: 978-3-540-79880-4