Hydrol. Earth Syst. Sci., 17, 2669–2684, 2013
www.hydrol-earth-syst-sci.net/17/2669/2013/
doi:10.5194/hess-17-2669-2013

Hydrology and Earth System Sciences
Open Access

Assessing the predictive capability of randomized tree-based ensembles in streamflow modelling

S. Galelli[1] **and A. Castelletti**[2,3]

[1]Singapore-Delft Water Alliance, National University of Singapore 2 Engineering Drive 2, 117577, Singapore
[2]Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano Piazza L. da Vinci, 32, 20133 Milano, Italy
[3]Centre for Water Research, University of Western Australia, Crawley, Western Australia, Australia

Correspondence to: A. Castelletti (andrea.castelletti@polimi.it)

Received: 11 January 2013 – Published in Hydrol. Earth Syst. Sci. Discuss.: 1 February 2013
Revised: 5 June 2013 – Accepted: 5 June 2013 – Published: 11 July 2013

Abstract. Combining randomization methods with ensemble prediction is emerging as an effective option to balance accuracy and computational efficiency in data-driven modelling. In this paper, we investigate the prediction capability of extremely randomized trees (Extra-Trees), in terms of accuracy, explanation ability and computational efficiency, in a streamflow modelling exercise. Extra-Trees are a totally randomized tree-based ensemble method that (i) alleviates the poor generalisation property and tendency to overfitting of traditional standalone decision trees (e.g. CART); (ii) is computationally efficient; and, (iii) allows to infer the relative importance of the input variables, which might help in the ex-post physical interpretation of the model. The Extra-Trees potential is analysed on two real-world case studies – Marina catchment (Singapore) and Canning River (Western Australia) – representing two different morphoclimatic contexts. The evaluation is performed against other tree-based methods (CART and M5) and parametric data-driven approaches (ANNs and multiple linear regression). Results show that Extra-Trees perform comparatively well to the best of the benchmarks (i.e. M5) in both the watersheds, while outperforming the other approaches in terms of computational requirement when adopted on large datasets. In addition, the ranking of the input variable provided can be given a physically meaningful interpretation.

1 Introduction

Streamflow processes are complex nonlinear hydrological phenomena exhibiting a high degree of spatial and temporal variability. Their accurate characterisation plays an important role in any decision-making process concerned with water availability, such as water reservoirs planning and management, operation of hydropower plants and irrigation systems, management of urban water supply systems, and many others. Two main approaches to streamflow modelling and prediction can be discerned in the hydrological literature (e.g. Beck, 1991; Wheater et al., 1993; Young, 2003): the hypothetico-deductive (or bottom-up) approach, according to which the physical mechanisms that contribute to streamflow formation in the hydrological cycle are conceptualised in a simplified lumped or semi-distributed representation (process-based models); and the inductive (or top-down) approach, in which the mapping from the space of predictor variables (e.g. precipitation, temperature) to that of the response variables (i.e. streamflow) is inferred totally and directly from observational data to a more general class of models (data-driven or metric models).

Depending on the objective of the modelling exercise, one approach can be more appropriate than the other. The the complexity of process-based models is key to improve our understanding of the hydrological process and has a clear advantage in "what-if" or scenario analyses. However, the high number of parameters and states these models include, particularly to characterise spatial variability, often result in mis-calibration and over-parameterisation (e.g. Jakeman and Hornberger, 1993; Beven, 2001), ultimately limiting the model predictive capability and operational value. Data-driven models combine high predictive potential and a more compact representation, with generally considerably less parameters and state variables, which well combines with

the computational burden of optimization-based decision-making (e.g. Castelletti et al., 2010). Yet, their effective identification requires long data records and their normal black-box nature, revealing very little of the internal structure, is often a deterrent to the systematic use in operational hydrology, though some successful attempts have been made to produce understandable insights from these model structures (e.g. Young and Beven, 1994; Babovic and Keijzer, 2002; See et al., 2008; Young, 2013).

Data-driven type of models applied to streamflow modelling includes traditional ARMA (e.g. Rasmussen et al., 1996, and references therein) and all its extensions, transfer function models (e.g. Young, 2006), and databased mechanistic (DBM) models (Young, 2003; Romanowicz et al., 2008). Methods from data mining, machine learning and artificial intelligence have also gained a good reputation in operational hydrology (Solomatine and Ostfeld, 2008). Among them, artificial neural networks, firstly used for streamflow modelling by Hsu et al. (1995), are the most popular choice (see the reviews by Maier and Dandy, 2000; Shamseldin et al., 2002). Other data-driven approaches largely experienced in hydrological modelling (e.g. see the comparative analysis by Elshorbagy et al., 2010a,b) include Fuzzy rule-based systems (e.g. Hundecha et al., 2001) and support vector machines (e.g. Lin et al., 2006). All these data-driven model families are based on the parameterisation of the input-output relationship and are built by a two-stage identification process: first, the model structure is selected (including, when relevant, model input selection), then the parameters are estimated with appropriate automatic algorithms. The wrong selection of the model structure might have a significant impact on the predicting capability of the identified model, even when the parameters can be estimated optimally within the selected family of functions.

A less traditional data-driven approach that is receiving increasing attention in the hydrological literature (e.g. Laaha and Blöschl, 2006; Sauquet and Catalogne, 2011; Bachmair and Weiler, 2012) is represented by decision trees, in particular Classification And Regression Trees (CART, Breiman et al., 1984), which are the simplest form of a decision tree. CART are non-parametric regressors with tree-like structures obtained by recursively partitioning the input space into mutually exclusive regions. The most internal regions (leaves) are associated with a constant output value obtained as the average of the output data falling in each leaf. CART have two advantages over most of the above mentioned data-driven approaches. First, they avoid the need to find potentially complicated parametric functions, thus reducing the potential for a model structural component to the prediction error (Iorgulescu and Beven, 2004). Second, the tree structure can readily be interpreted as a cascade of "if-then" rules between combinations of inputs and the output, and so CART can give better insight on the model internal structure and underlying physical processes (Iorgulescu and Beven, 2004; Wei and Watkins Jr., 2011). CART have been shown to perform comparatively well than other data-driven models in a number of applications (Dawson et al., 2000; Iorgulescu and Beven, 2004; Vezza et al., 2010). Yet they suffer from a double drawback: (i) the predicted output is composed of discrete values and the streamflow is reconstructed as a piecewise constant function. To ensure a good predicting accuracy, the number of output classes (tree leaves) must be very high, but this increases the risk of overfitting the observed data and reduces the model generalisation ability (Ho, 1995; Breiman, 1996). (ii) The partitioning process is deterministically performed by exhaustively comparing all the possible combinations of input values to select the best performing partition. This makes computation requirements growing rapidly with the input space dimensionality and indeed the optimal training of a decisions tree is NP-hard (Hyafil and Rivest, 1976).

The first weakness can be resolved in two ways. One idea is to replace averaging in the tree leaves by fitting a linear regression function to the data and obtaining a continuous representation of the output. This approach, mostly known as M5 tree-modelling, was first introduced by Quinlan (1992) and applied to hydrological problems by Solomatine and Dulal (2003); Solomatine and Xue (2004); Bhattacharya and Solomatine (2005); Stravs and Brilly (2007); Jothiprakash and Kote (2011). Another idea is to use an ensemble method (e.g. bagging, Breiman, 1996 or boosting, Freund and Schapire, 1996) to build a forest of regression trees. The underlying concept of ensembles is that multiple model predictions aggregated in one ensemble output allow to obtain better predictive performance than any of the constituent models (Dietterich, 2000). The adoption of tree ensembles for hydrological modelling has been reported by Snelder et al. (2009); Erdal and Karakurt (2013). Both the contributions show that trees ensembles remarkably advance the prediction capability of CART and generally compare favorably to other data-driven approaches.

Unfortunately, neither M5 or CART ensembles help in reducing the computational burden associated with the optimal deterministic tree building process they incorporate. Rather, model identification is made even more computationally intensive by generally increasing the number of operations to be performed by the training algorithm. Recently, randomization methods have been shown to be an effective companion of ensemble tree methods (e.g. Geurts, 2002, and references therein). In fact, ensemble methods highly benefit from the diversity in the constituent models (Kuncheva and Whitaker, 2003) and injection of randomness is a way of producing more or less diversified ensembles (Ho, 1995). In particular, the direct randomization of the individual tree growing method seems to be more productive for the ensemble in terms of both accuracy and computational requirements than the optimality of traditional induction algorithms, such those in M5 and CART (Geurts, 2002).

Several approaches have been developed based on the direct randomization of the tree growing method (e.g. Bagging

predictors, Breiman, 1996; Random Subspace, Ho, 1998; Random forests, Breiman, 2001; PERT, Cutler and Guohua, 2001). Lately, the extremely randomized trees developed by Geurts et al. (2006) (Extra-Trees in short) have been empirically demonstrated to outperform most of the other randomized and deterministic methods in terms of both prediction accuracy (more specifically, variance and bias reduction) and computational efficiency. Extra-Trees are ensembles of totally randomized trees in that they randomize both the input variables and the splitting values considered in creating a partition, in the process of building a tree, and create a forest of trees to compensate for the randomization, via averaging of the constituent tree outcomes. The combination of averaging and randomization ensures (i) modelling flexibility/accuracy (i.e. ability of characterising strong nonlinear relationships), (ii) computational efficiency, and (iii) scalability with respect to input dimensionality. In addition, (iv) Extra-Trees, like several other tree-based ensemble methods (Jong et al., 2004), can be exploited to infer the relative importance of the input variables and to order them accordingly (Wehenkel, 1998; Fonteneau et al., 2008). This allows to provide an ex-post interpretation of the model and makes the model more understandable and credible to the users than other data-driven approaches.

In this paper we explore the applicability of Extra-Trees to streamflow modelling and comprehensively analyse their advantages and disadvantages in terms of predicting accuracy, explanation ability and computational efficiency. Specifically, we adopt a four-step assessment procedure including (i) random sampling of the observational dataset to ensure a robust evaluation of the model performance (Elshorbagy et al., 2010a); (ii) multi-criteria assessment of the model performance (Hwang et al., 2012, and references therein) to consistently validate the model behaviour under different flow conditions; (iii) comparative assessment of predicting accuracy and computational efficiency against tree-based methods (M5 and CART) already experimented in water-related applications and other traditional data-driven approaches (ANNs and multiple linear regression); (iv) uncertainty analysis on the model residual.

The numerical analysis is conducted on two streamflow modelling problems with different spatial domains, hydrometeorological features, and temporal dynamics. Marina catchment, Singapore, is a relatively small urban catchment with a very short time of concentration, considerably altered by human intervention and subject to a tropical climate; the Cunning River, Western Australia, is a large river basin, predominantly natural, characterised by a Mediterranean climate and modelled with a daily time step.

2 Extremely randomized trees (Extra-Trees)

Tree-based regressors are structured as a hierarchical cascade of rules able to predict numerical values of the output (Breiman et al., 1984). The process of building the nodes and branches forming a tree is based on the partitioning of the input space into mutually exclusive regions according to a predefined splitting criterion, progressively narrowing down the size of the regions. Eventually, when the number of instances in a region becomes smaller than a specific a preassigned value (or their values vary just slightly), the partitioning of that region stops and a leaf is created. Whenever a new instance is fed into the tree, a specific path is followed according to the splitting rules defined in the tree-building procedure, and the predicted output is then obtained from the aggregation of the values stored in the leaf. The splitting criterion, the termination test, the number of trees grown, and the rule adopted to associate a numerical value to each leaf are the key-features differentiating the many tree-based methods available in the literature. On one extreme CART are a fully deterministic single-tree method, on the other, Extra-Trees are a totally randomized ensemble method as explained next.

2.1 Model building

Extra-Trees substantially differ from traditional deterministic and randomized methods in two particular aspects. First, in the process of building a tree, the selection of the input and splitting value to split a node are randomized, i.e. they occur independently of the output variable. Second, an ensemble of M trees is created in order to compensate for the effect of randomization, and the outcome of the ensemble is the average of each tree output. Nodes are split using the following rule: K alternative inputs (cut-directions) are randomly selected and, for each one, a random splitting value (cut-point) is chosen; a score is then associated to each cut-direction and the one maximising the variance reduction following the adopted splitting criterion is adopted to split the node. The termination test that determines when to stop partitioning a node is based on the number of instances within the node. When this number is smaller than a user-defined value $n_{\min}$, the algorithm stops partitioning a node and a leaf is created (Geurts et al., 2006). To each leaf a value is eventually assigned, obtained as the average of the target values associated to the inputs falling in that leaf. The estimates produced by the M trees are finally aggregated by arithmetic average (see Table 1 for a tabular version of the Extra-Trees building algorithm). The rationale behind the approach is that the use of the original training dataset (instead of a bootstrap replica, as in the Bagging method, Breiman, 1996) is motivated to minimise bias, while the combined use of randomization and ensemble averaging is aimed at reducing the variance of the model output (Geurts et al., 2006).

2.2 Hyperparameters

The three hyper-parameters M, K, and $n_{\min}$ characterising the model building algorithm diversely affect the ensemble performance and overall method efficiency. Increasingly

Table 1. Tabular version of the Extra-Trees building algorithm.

Input:	an output variable y, n inputs $\{x^1, x^2, \ldots, x^n\}$ and a training dataset $\mathcal{D}$ composed of $\lvert\mathcal{D}\rvert$ input-output observations.
Output:	a single Extremely Randomized Tree. The algorithm is repeated M times to produce an ensemble.
Step 1.	Randomly select without replacement K inputs $\{x^1, x^2, \ldots, x^K\}$ among the n available (non-constant in $\mathcal{D}$).
Step 2.	For each selected input variable x^i (with $i = 1, \ldots, K$):
Step 2a.	Compute the minimum and maximum value of x^i in $\mathcal{D}$, denoted as $x_{\mathcal{D}}^{i,\min}$ and $x_{\mathcal{D}}^{i,\max}$.
Step 2b.	Randomly select a cut-point s^i in the interval $\left[x_{\mathcal{D}}^{i,\min}, x_{\mathcal{D}}^{i,\max}\right]$.
Step 2c.	Return the split $\left[x^i < s^i\right]$.
Step 3.	Among the K splits $\{s^1, s^2, \ldots, s^K\}$, select the split s^* such that $s^* = \arg \max_{i=1,\ldots,K} \Delta_{\text{var}}(s^i, \mathcal{D})$ where: – $\Delta_{\text{var}}(s^i, \mathcal{D})$ is the variance reduction defined as $\text{var}\{y\lvert\mathcal{D}\} - \frac{\lvert\mathcal{D}^l(x^i)\rvert}{\lvert\mathcal{D}\rvert}\text{var}\{y\lvert\mathcal{D}^l(x^i)\} - \frac{\lvert\mathcal{D}^r(x^i)\rvert}{\lvert\mathcal{D}\rvert}\text{var}\{y\lvert\mathcal{D}^r(x^i)\}$. – $\mathcal{D}^l(x^i)$ and $\mathcal{D}^r(x^i)$ are the two subsets of $\mathcal{D}$ satisfying the conditions $x^i < s^i$ and $x^i \geqslant s^i$, – $\lvert\mathcal{D}\rvert$ is the number of samples in $\mathcal{D}$, $\lvert\mathcal{D}^l(x^i)\rvert$ and $\lvert\mathcal{D}^r(x^i)\rvert$ are the number of samples in $\mathcal{D}^l(x^i)$ and $\mathcal{D}^r x^i)$.
Step 4.	According to s^*, split the set $\mathcal{D}$ into the subsets $\mathcal{D}^l(x^i)$ and $\mathcal{D}^r(x^i)$, and return the (non-terminal) node v^j.
Step 5.	For the subset $\mathcal{D}^l(x^i)$ (and $\mathcal{D}^r(x^i)$), verify the following conditions: – $\lvert\mathcal{D}^l(x^i)\rvert$ (or $\lvert\mathcal{D}^r(x^i)\rvert$) is lower than $n_{\min}$ (minimum cardinality). – All input variables $\{x^1, x^2, \ldots, x^n\}$ are constant in $\mathcal{D}^l(x^i)$ (or $\mathcal{D}^r(x^i)$). – The output variable is constant in $\mathcal{D}^l(x^i)$ (or $\mathcal{D}^r(x^i)$).
Step 6.	If one of the conditions in Step 5 is satisfied, the subset is is leaf (labelled with the average of the output variables values). Alternatively, Steps 1–5 are repeated by replacing $\mathcal{D}$ with $\mathcal{D}^l(x^i)$ (or $\mathcal{D}^r(x^i)$).

high values of M reduce the variance of the final estimate (Breiman, 2001), but also considerably add to the computational requirements of the building algorithm, so the final choice depends on a trade-off between the desired model accuracy and available computing power. K can be chosen in the interval $[1, \ldots, n]$, with n being the number of input variables, and controls the level of randomness in the tree building process. The smaller K, the stronger the randomization of the trees and the weaker the dependence of their structure on the values of the output variable in the training dataset. In the extreme case, when K is equal to 1, the splits (cut-directions and cut-points) are chosen in a totally independent way of the output variable and the method builds totally randomized trees. As empirically demonstrated by Geurts et al. (2006), the optimal value of K for regression problems is equal to the number n of inputs, and so the number of cut-directions randomly selected. Finally, the threshold $n_{\min}$ is used to balance bias and variance reduction. Large values of $n_{\min}$ lead to small trees, with high bias and small variance; conversely, low values of $n_{\min}$ lead to fully-grown trees, which may overfit the data. The optimal tuning of $n_{\min}$ can depend on the level of noise in the training dataset: the noisier the outputs, the higher the optimal value of $n_{\min}$ should be. Although this tuning might require some experiments, Geurts et al. (2006) have shown that a value of $n_{\min}$ between 5 and 50 is a robust choice in a broad range of typical conditions.

2.3 Computational requirements

From the computational point of view, the complexity of the Extra-Trees building procedure is in the order of $\lvert\mathcal{D}\rvert \cdot \log(\lvert\mathcal{D}\rvert)$, with $\lvert\mathcal{D}\rvert$ being the number of input-output observations in the training dataset $\mathcal{D}$. The computational time linearly increases with M and K, and logarithmically decreases for increasing values of $n_{\min}$, meaning that the approach still remains computationally efficient, though based on the construction of a tree ensemble. This is because the splitting rule is very simple compared to other splitting rules that locally optimise the cut-points, as, for example, those adopted by CART and M5.

2.4 Input ranking

The particular structure of Extra-Trees can be exploited to rank the importance of the n input variables in explaining the selected output behaviour. This approach, as originally proposed by Wehenkel (1998), is based on the idea of scoring each input variable by estimating the relative variance reduction it can be associated with by propagating the training dataset $\mathcal{D}$ over the M different trees composing the ensemble.

Table 2. Tabular version of the Extra-Trees input ranking algorithm.

Input:	an output variable y, n inputs $\{x^1, x^2, \ldots, x^n\}$ and a training dataset $\mathcal{D}$ composed of $\lvert\mathcal{D}\rvert$ input-output observations.
Output:	ranking of the input variables (sorted by decreasing values of their relevance), and an ensemble of M Extra-Trees.
Step 1.	Assign to each input variable x^i (with $i = 1, \ldots, K$) a score $G(x^i)$ equal to 0.
Step 2.	Define suitable values for M, K and $n_{\min}$ and build an ensemble of Extra-Trees (as described in Table 1). At each split node v^j update the score corresponding to the selected input variable x^i according to the following equation: $G(x^i, j) = G(x^i, j-1) + \Delta_{\text{var}}(v^j)$
Step 3.	Normalise the score $G(x^i)$ of each input variable, and sort these values in decreasing order.

More precisely, the relevance $G(x^i)$ of the i-th input variable x^i in explaining the output y can be evaluated as follows

$$G\left(x^i\right) = \frac{\sum_{\tau=1}^{M}\sum_{j=1}^{\Omega}\delta\left(v^j, x^i\right)\cdot\Delta_{\text{var}}\left(v^j\right)\lvert\mathcal{D}\rvert}{\sum_{\tau=1}^{M}\sum_{v^j=1}^{\Omega}\Delta_{\text{var}}\left(v^j\right)\lvert\mathcal{D}\rvert} \quad (1)$$

where v^j is the j-th non-terminal node in the τ-th tree, Ω is the number of non-terminal nodes in the tree, $\delta(v^j, x^i)$ is equal to 1 if the variable x^i is used to split the node v^j (and 0 otherwise), and $\Delta_{\text{var}}(v^j)$ (or $\Delta_{\text{var}}(s^i, \mathcal{D})$) is the variance reduction associated to node v^j (see Table 1). Finally, the input variables $\{x^1, x^2, \ldots, x^n\}$ are sorted by decreasing values of their relevance (see Table 2 for a tabular version of the input ranking algorithm).

3 Experimental setup

3.1 Datasets

The Extra-Trees capabilities are tested on two streamflow modelling problems with different spatial domains and hydro-meteorological features: Marina catchment is a relatively small urban catchment, considerably altered by human intervention and subject to a tropical climate; the Cunning River watershed is a large basin, predominantly natural, characterised by a Mediterranean climate.

3.1.1 Marina catchment

Marina catchment feeds the homonymous reservoir located in the heart of Singapore. The reservoir, created in late 2008 with the construction of a tidal barrier, has a surface area of 2.45 km^2 and an active storage of about 3.2×10^6 m^3 operated for floods control and drinking water supply (Galelli et al., 2013). Five main tributaries discharge water into the reservoir, draining a catchment of ≈ 100 km^2 (almost 15 % of the land area of Singapore) and producing a mean annual inflow of about 150×10^6 m^3 with a typical tropical pattern. The catchment includes one of the most densely populated and urbanised regions in Singapore and Southeast Asia (Xie, 2006), and its drainage system consists of concrete lined canals, which make the time of concentration extremely short (≈ 1 h) and the base flow almost null. Because of the high-intensity rainfall events characterising the region (Selvalingam et al., 1987), discharges occur in high peaks over short periods of few hours (see Fig. 1, upper panel).

Table 3. Descriptive statistics of the output variable for Marina catchment and Canning River datasets.

	Marina streamflow [m^3 s-1]	Canning streamflow [m^3 s^{-1}]
Number of samples	24 120	4017
Minimum	0.00	0.00
Maximum	845.21	16.77
Mean	5.92	0.31
Std. dev.	25.32	1.08
Coefficient of variation	4.28	3.43
Skewness	14.13	7.38

The available dataset consists of hourly rainfall and inflow measurements over the period 1 April 2009–31 December 2011, for a total of 24 120 data points (see Table 3 for the descriptive statistics of the output variable). The selection of the most significant time-lags is performed by means of the Mutual Information (MI) criterion (e.g. Hejazi and Cai, 2009, and references therein), which singled out an input set composed of three time-lags for each variable, namely $[y_{t-1}, y_{t-2}, y_{t-3}, r_{t-1}, r_{t-2}, r_{t-3}]$, with y_{t-1} and r_{t-1} denoting the inflow and rainfall in the time interval $[t-1, t]$. The streamflow modelling exercise is then performed over a prediction horizon of 1 h.

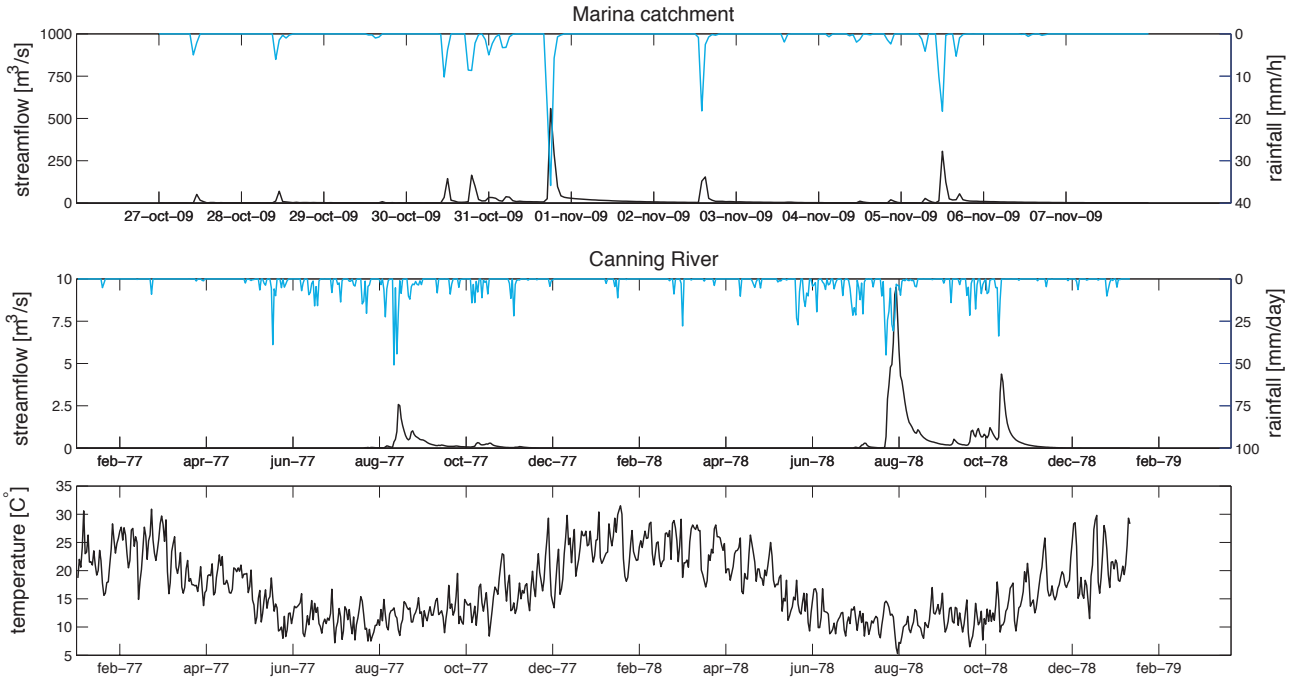

Fig. 1. Specimen of the hydrograph generated in Marina catchment and Canning River under different rainfall (and temperature) scenarios.

3.1.2 Canning River

The second dataset is taken from the Canning River basin, a major tributary of the Swan River in Western Australia. The river drains a catchment area of $\approx 850\,km^2$, where woodland is the predominant land use. The climate shows a Mediterranean pattern, characterised by warm and dry summers and cool, wet winters. The long-term average annual rainfall for the catchment is ≈ 900 mm mostly falling between May and September. The combination of this rainfall pattern and land use gives the river an ephemeral nature (Young, 2002) with practically no flow during the summer period. As discussed in Young et al. (1997), a data analysis shows indeed a strong nonlinear correlation between the rainfall and the river flow (Fig. 1).

For the present analysis the dataset consists of daily rainfall, temperature and flow measurements available for the period 1 January 1977–31 December 1987, for a total of 4017 data points (Table 3). As for the former dataset, the most significant input variables are selected with the MI criterion. According to this criterion two time-lags for each variables, namely $[y_{t-1}, y_{t-2}, r_{t-1}, r_{t-2}, T_{t-1}, T_{t-2}]$ (with T_{t-1} denoting the average temperature in the time interval $[t-1, t]$), are selected to predict the flow one-day-ahead.

3.2 Setting the experiments

The quantitative assessment of Extra-Trees is performed using a four-step procedure:

– *Random sampling*: to ensure a robust evaluation of the model performance (Elshorbagy et al., 2010a), the two datasets are randomly sampled (without replacement) 100 times, in order to create at each sampling exercise a training/cross-validation and testing subsets, respectively containing two thirds and one third of the available data. Ten different groups (each composed of training/cross-validation and testing subsets) are then selected based on their statistical properties, namely mean and standard deviation of the output variable. Ten different models are identified on the 10 data groups, with each model finally evaluated on the corresponding testing subset.

– *Model evaluation*: the Extra-Trees evaluation is based on multi-assessment criteria (Hwang et al., 2012), aimed at describing the model behaviour under different flow conditions. The criteria considered are (i) the Nash–Sutcliffe (NS) criterion and (ii) the Relative Root Mean Squared Error (RRMSE), which are normalised statistics providing a description of the models behaviour over the whole range of flow conditions; (iii) the Root Mean Squared Error (RMSE), which measures the goodness of fit relevant to high flows (iv) the Mean Absolute Error (MAE), which indicates the goodness of fit at moderate flow values. This assessment is completed by a graphical analysis of the scatter plots and hydrographs.

– *Comparative assessment*: the best Extra-Trees ensemble so identified is compared against several machine

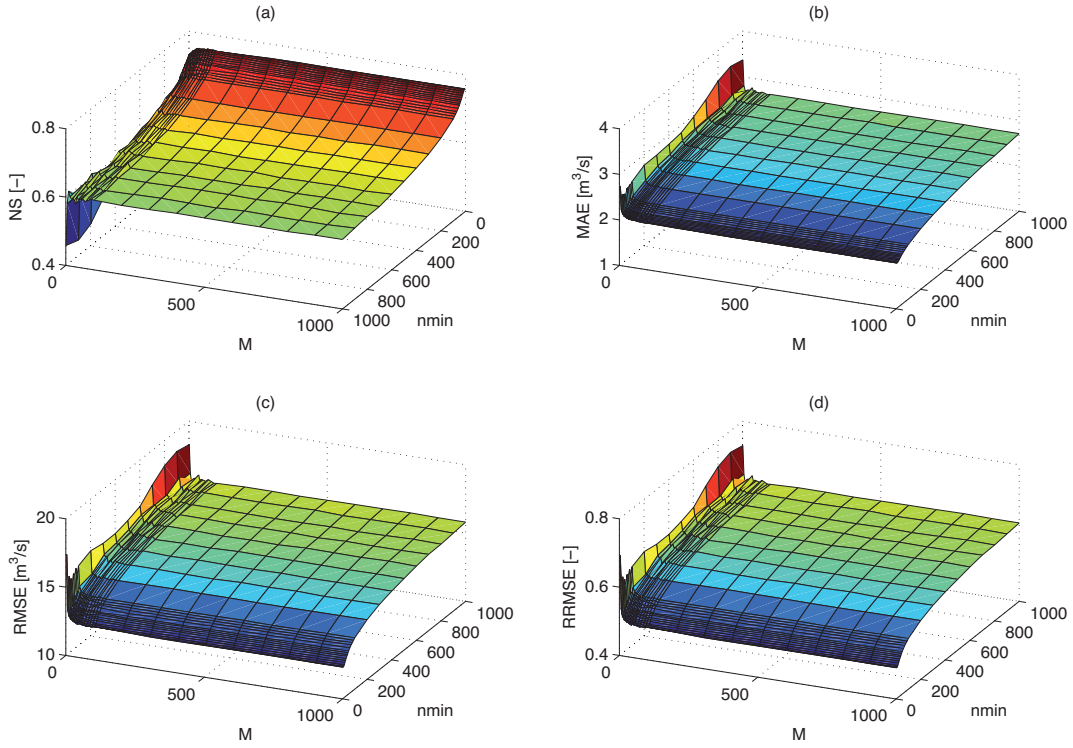

Fig. 2. Values of NS **(a)**, MAE **(b)**, RMSE **(c)** and RRMSE **(d)** as a function of $n_{\min}$ and M over the testing subsets for Marina Catchment (average over 10 data groups).

learning modelling methods, including tree-based methods (M5 model trees and CART) and ANNs. To facilitate the comparison, Multiple Linear Regression (MLR) models are employed as base line references.

– *Uncertainty analysis*: to estimate the uncertainty associated with model predictions, the residuals of the 10 testing subsets are computed and aggregated in a single dataset, for which both the empirical and probability distributions are fit. Each probability distribution is selected using a trial-and-error analysis on several candidate distributions (e.g. Beta, Gamma, Logistic, Normal, *t* location-scale, etc.), whose parameters are calibrated by means of maximum likelihood estimation. Among the different candidate distributions, the most performing one with respect to the Bayesian information criterion is then adopted. In the benchmarking exercise, a two-sample Kolmogorov–Smirnov test is also performed to compare the distributions of model residuals. In particular, residuals are tested under the null hypothesis that they are from the same distribution: two residuals are considered significantly different if the null hypothesis is rejected at the 5 % confidence level (p value $\leqslant 0.05$). This means that, if the null hypothesis is rejected, the residuals generated by two models on the 10 testing subsets are likely to belong (with 95 % confidence) to different probability distributions.

4 Extra-Trees application results

4.1 Prediction

Extra-Trees' predicting potential is assessed for different values of M, K, and $n_{\min}$. The sensitivity analysis is performed by running an extensive number of training/cross-validation and testing experiments on the selected 10 data groups of each dataset. As explained in Sect. 2.1, the value of K is fixed equal to the number n of input variables, which is 6 for both Marina and Canning dataset. 25 values for M and $n_{\min}$ are sampled in the domains [1, 1000] and [2, 1000], leading to 625 different parameterisations. The extreme cases are: (i) a single Extra-Tree with large leaves (i.e. $M = 1$, $n_{\min} = 1000$) or a fully-grown tree (i.e. $M = 1$, $n_{\min} = 2$), (ii) a large forest composed of small or fully-grown trees ($M = 1000$ with $n_{\min} = 1000$ or 2, respectively).

The values of the multi-assessment criteria as a function of M and $n_{\min}$ are illustrated in Figs. 2 and 3, while a graphical analysis of the parameters' effect on the NS criterion is given in Fig. 4. For both Marina and Canning dataset the larger the number M of trees in the forest, the higher the variance reduction. The reduction in the variance has a positive effect on the Extra-Trees estimation error and reflects in the abatement of the distance between observed and predicted values for M growing from 1 to 100. Since the computation time linearly increases with M, a balance must be found between accuracy

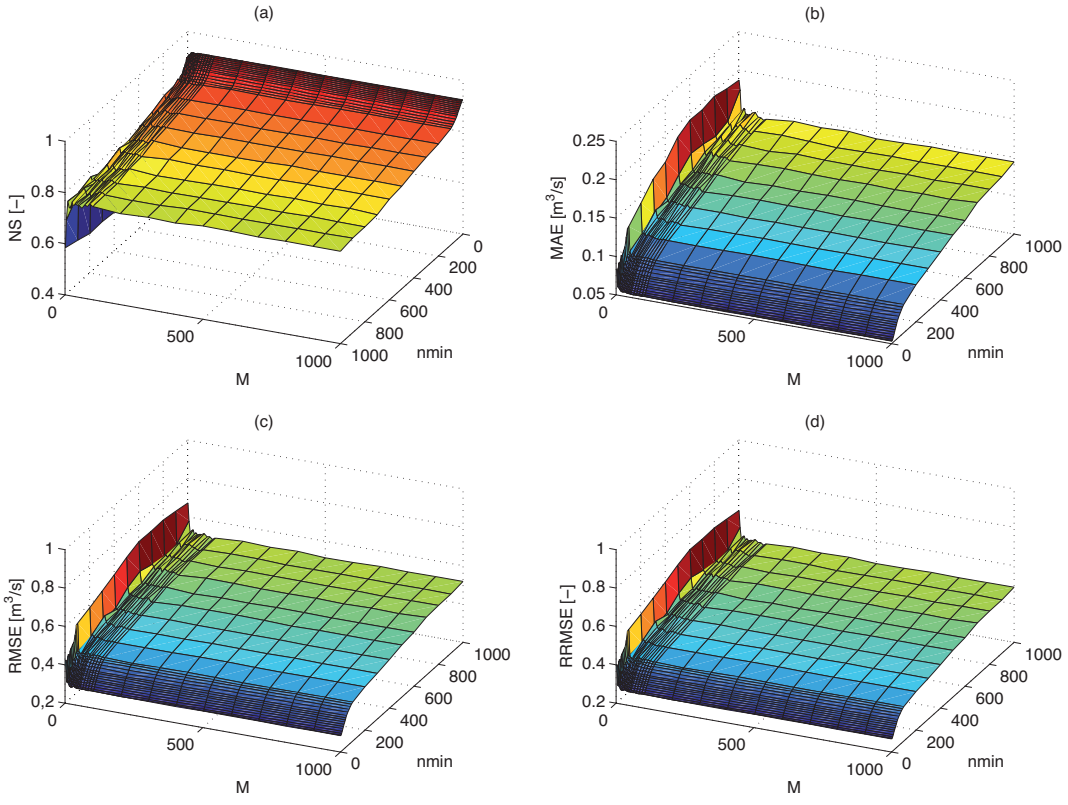

Fig. 3. Values of NS **(a)**, MAE **(b)**, RMSE **(c)** and RRMSE **(d)** as a function of $n_{\min}$ and M over the testing subsets for Canning River (average over 10 data groups).

and time requirements. The saturation effect (Fig. 4c and d) might help in deciding a proper value (see also Castelletti et al., 2010): the performance improvement from values of M greater than 200–300 is distinctively negligible. The value of $n_{\min}$ determines the number of leaves in a tree and, thus, the ensemble's overall trade-off between bias and variance. As shown in Figs. 2 and 3, reducing $n_{\min}$ has a positive effect on all the assessment criteria. This effect is consistent up to a value of $n_{\min}$ equal to about 5. Indeed, when this threshold is reached, the model building algorithm produces fully grown trees, with the consequent risk of over-fitting the data (i.e. lower bias but higher variance in the model output).

In synthesis, sensitivity analysis shows that Extra-Trees provide reasonably good performance over a broad range of parameter values: the value of M must indeed be as large as possible, though a saturation effect is reached for M greater than 200–300, while $n_{\min}$, as already discussed by Geurts et al. (2006), should be comprehended between 5 and 15. For the subsequent analysis (i.e. input ranking and benchmarking) a parameterisation with M and $n_{\min}$ equal to 500 and 5, respectively, is finally chosen.

Table 4. Input Ranking results for the Marina catchment dataset (average over 10 data groups). The initial variance is 10 421 100.

Ranking	x^i	$G(x^i)$ (%)
1	r_{t-1}	66.89
2	y_{t-1}	15.92
3	r_{t-2}	5.16
4	y_{t-2}	4.49
5	y_{t-3}	4.09
6	r_{t-3}	3.45

4.2 Explanation

As anticipated, the Extra-Trees model building algorithm implicitly allows to rank the model inputs in terms of their relevance in explaining the output. This is useful for the ex-post physical interpretation of the cause-effect relationships captured by the model. The ranking is run on the ensemble selected at the end of the model building process. In particular, an ensemble is cross-validated on the selected 10 data groups of each dataset, and the inputs are sorted in decreasing order according to the ranking algorithm described in Sect. 2.4. The results obtained as the average relative contribution (over 10 data groups) are reported in Tables 4 and 5.

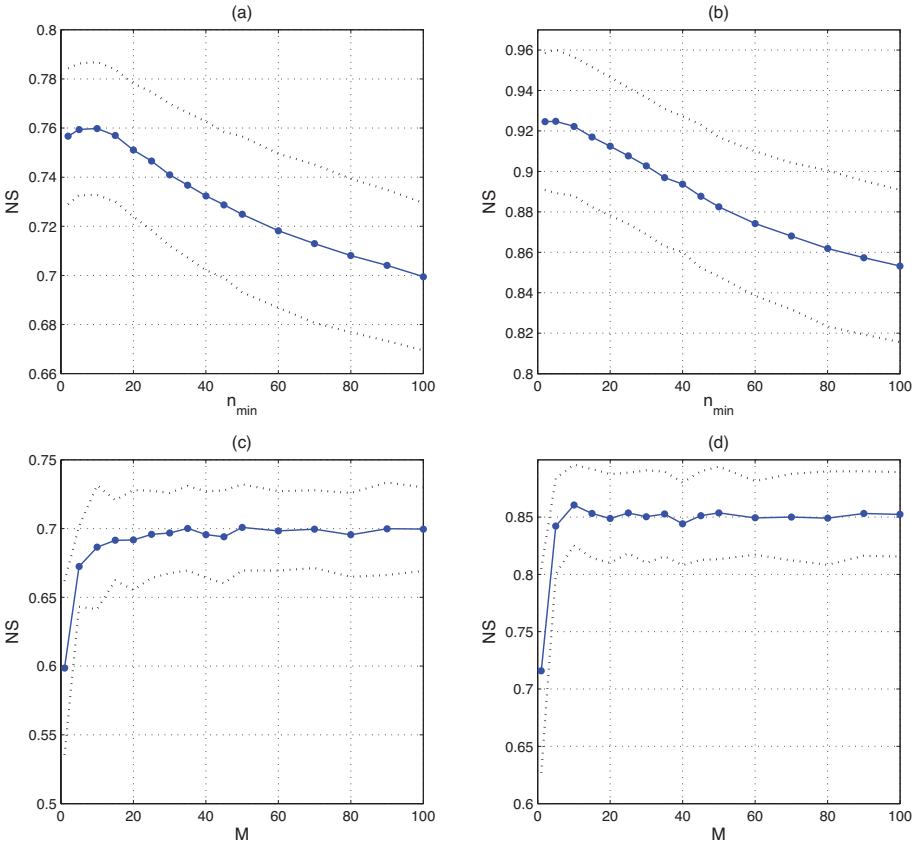

Fig. 4. Values of NS as a function of $n_{\min}$ (with $M = 500$) and of M (with $n_{\min} = 100$) over the testing subsets for Marina (**a** and **c**) and Canning River (**b** and **d**). Dotted lines represent the standard deviation calculated over the the selected 10 data groups of each dataset.

As for Marina Catchment, the measured rainfall r_{t-1} and antecedent flow y_{t-1} are the most important variables, contributing for about 80 % of the ensemble total variance. The measured rainfall r_{t-1} is ranked in the first position, with a relative score of almost 67 %. This high relevance is due to the hydraulic characteristics of Marina catchment, which is drained by concrete lined canals with an almost null base flow: high flow peaks are mainly driven by rainfall, so the cumulated precipitation in the previous hour becomes the most relevant information to the model output. Because of the short time of concentration (approximately one hour), the measured precipitation and antecedent flow with 2 and 3 time-lags are less important.

The Canning River drains a large, natural catchment forced by a Mediterranean climate. As illustrated in Table 5, the antecedent flow with 1 and 2 time lags is the most relevant variable (87 % of the ensemble output), followed by rainfall and temperature.

Table 5. Input ranking results for the Canning River dataset (average over 10 data groups). The initial variance is 2958.69.

Ranking	x^i	$G(x^i)$ (%)
1	y_{t-1}	63.56
2	y_{t-2}	22.90
3	r_{t-1}	5.28
4	r_{t-2}	3.26
5	T_{t-1}	2.56
6	T_{t-2}	2.44

5 Benchmarking

The best Extra-Trees ensemble identified in the model building process is compared against M5 model trees, CART, ANNs and MLR. The same experimental setting and datasets used for the Extra-Trees are adopted in this benchmarking exercise in order to guarantee a rigorous and unbiased comparison.

5.1 Models implementation

The MatLab toolbox M5PrimeLab (Jekabsons, 2010) is used to implement the M5 model trees in the different case studies and relative data groups. Pruning and smoothing are accounted for as suggested in Jothiprakash and Kote (2011); in particular, the smoothing coefficient is optimised via

Table 6. k-fold cross-validation (with $k = 10$) and testing results of Extra-Trees and benchmarking models for Marina Catchment dataset.

	k fold cross-validation				Testing			
Model	NS [−]	RMSE [$m^3 s^{-1}$]	RRMSE [−]	MAE [$m^3 s^{-1}$]	NS [−]	RMSE [$m^3 s^{-1}$]	RRMSE [−]	MAE [$m^3 s^{-1}$]
Extra-Trees	0.76	12.39	0.49	2.01	0.76	12.29	0.49	1.99
M5	0.77	11.89	0.48	2.01	0.78	11.77	0.47	1.99
CART	0.69	13.68	0.55	2.31	0.71	13.61	0.54	2.26
ANNs	0.65	14.45	0.57	3.99	0.69	13.92	0.55	4.06
MLR	0.74	12.66	0.51	3.84	0.74	12.82	0.51	3.84

trial-and-error in the range [0, 20] (Wang and Witten, 1997). The other parameters requiring a manual tuning are the split threshold and the minimum number of training samples one node may represent. The former is explored in the range [0.05, 0.20], the latter in the range [2, 1000].

CART are implemented with the MatLab Statistics Toolbox, which relies on the original algorithm proposed by Breiman et al. (1984). Similarly to the other tree-based methods adopted in this study (i.e. Extra-Trees and M5), the minimum number of training samples one node may represent is heuristically optimised in the range [2, 1000]. Pruning is adopted to compute the full tree and the optimal sequence of pruned subtrees, thus minimising the risk of over-fitting the cross-validation data.

The MatLab Neural Network Toolbox is adopted to set up the ANNs, whose parameters are optimised by means of the Levenberg–Marquardt algorithm. For each of the 10 data groups (of each case study), the ANNs cross-validation process is repeated 100 times with 100 different initialisation of the random weights. The best performing parameterisation in terms of RMSE is then selected as representative of a data group. As for the ANNs architecture, the number of input nodes corresponds to the number of input variables (thus 6 for both Marina and Canning River case study), while the number of hidden nodes is heuristically optimised in the range [1, 10].

MLR models are also implemented in MatLab, and calibrated using least-squares.

For each machine learning method considered in this study, this implementation eventually leads to 10 models (for each case study) developed and tested using the corresponding unseen data groups.

5.2 Results and analysis

As discussed in Sects. 3.1 and 4.2, the Marina catchment dataset is characterised by a weak autocorrelation in the hourly inflow to the reservoir. This is the reason why providing the antecedent flow as an input to predict future discharges does not strongly increase the information available to the different models. Rather, the limiting factor for the model performance seems to be the capability of exploring the correlation between the future inflows and the measured rainfall and flow. This is confirmed by the results reported in Table 6. Extra-Trees and M5 outperform the other models with respect to all the multi-assessment criteria. In this specific comparison, Extra-Trees and M5 are, de-facto, comparable over the whole range of flows, as shown by the NS and RRMSE values. Extra-Trees and M5 are also comparable in terms of MAE, which indicates the goodness of fit at moderate flow values. Yet, M5 stands out as the best performing model when accounting for the RMSE, which measures the model performance relevant to high flows. This behaviour can probably be explained by considering the different models architectures: M5 have linear models in the final (pruned) leaves, and this allows them to extrapolate over unseen events; the Extra-Trees prediction corresponds to the average of the output values associated to the inputs falling in a specific leaf, and this can limit their extrapolation capabilities. The third model family in order of performance is ANNs, while the worst results are attributable to CART and MLR. The low CART performance can again be explained by accounting for the model architecture: the CART model building algorithm provides an optimal partitioning of the input space (with respect to the standard deviation reduction of the output variables; see Breiman et al., 1984), but the prediction associated to each leaf is simply the average of the output values associated to the inputs falling in a specific leaf. As a consequence, a CART structure can be seen as a classification of the different flow regimes registered in the training/cross-validation data group, and this can limit the overall model predictive capabilities. This does not occur with Extra-Trees since the model building algorithm improves the performance of a single model by ensemble averaging. Unlike Marina catchment, the Canning River dataset shows a stronger autocorrelation in the flow process, and this enhances the information content at the disposal of the different models. As shown in Table 7, models are characterised by more comparable performance, although Extra-Trees and M5 stand out as the best performing models.

In order to better comment on the ability of Extra-Trees in reproducing different flow regimes, both Extra-Trees and the benchmarking models are evaluated on four specific regimes, i.e. base, low, intermediate and high flow. The identification

Table 7. k fold cross-validation (with $k = 10$) and testing results of Extra-Trees and benchmarking models for Canning River dataset.

	k-fold cross-validation				Testing			
Model	NS [−]	RMSE [$m^3 s^{-1}$]	RRMSE [−]	MAE [$m^3 s^{-1}$]	NS [−]	RMSE [$m^3 s^{-1}$]	RRMSE [−]	MAE [$m^3 s^{-1}$]
Extra-Trees	0.92	0.30	0.28	0.05	0.93	0.28	0.27	0.05
M5	0.94	0.25	0.24	0.06	0.94	0.26	0.24	0.06
CART	0.87	0.36	0.35	0.07	0.88	0.37	0.35	0.07
ANNs	0.88	0.35	0.34	0.12	0.90	0.34	0.33	0.10
MLR	0.92	0.29	0.28	0.08	0.92	0.30	0.29	0.08

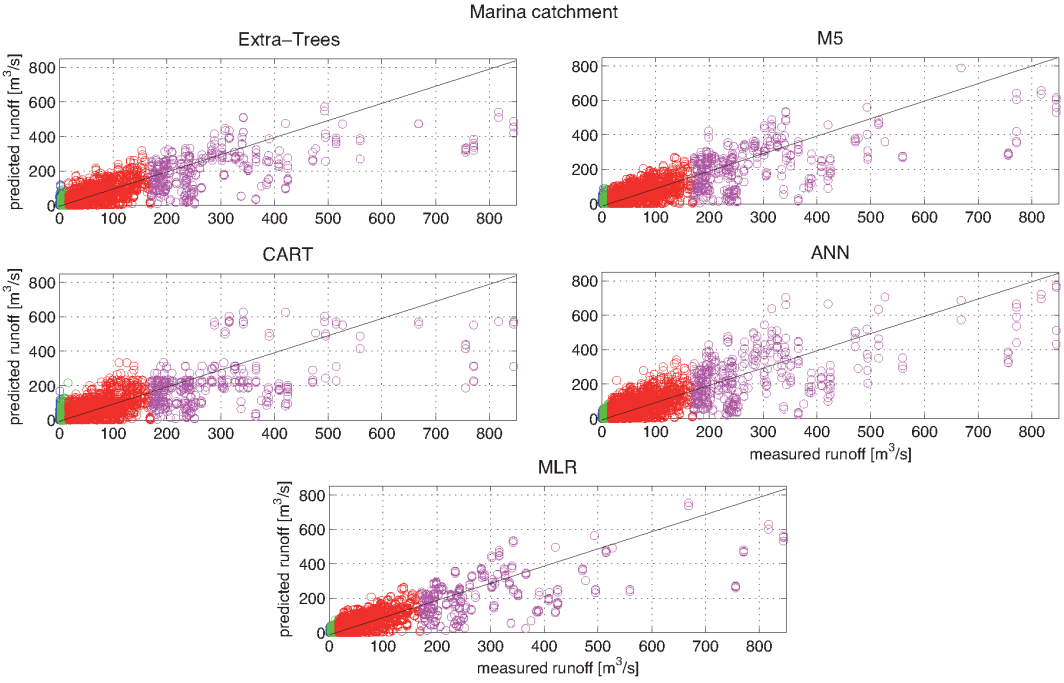

Fig. 5. Scatter plots of predicted (y-axis) and measured (x-axis) streamflow [$m^3 s^{-1}$] in Marina catchment for the different models on the testing subsets. Different colours are used to represent the flow regimes: blue, green, red and purple correspond to base, low, intermediate and high flow, respectively.

of these flow regimes is based on the calculation of specific percentile values on the testing dataset. For both Marina catchment and Canning River, the flow regimes are categorised by using the 75th, 95th and 99.5th percentiles, as shown in Table 8. These percentile values are related to Marina catchment and Canning River dynamics, which are characterised by prolonged periods of low or null flow. In the former case this is due to the presence of large paved areas that reduce the infiltration, while in the latter case the null flow is due to the ephemeral nature of the river during the summer period. Because of this prolonged periods of no flow, 75 % of the observations falls below the base flow threshold. The other extreme of the flow regimes, i.e. high flow, also corresponds to a high percentile value (99.5th). The models performance for the different flow regimes are graphically represented in two scatter plots (Figs. 5 and 6) and measured in terms of RMSE (Table 9). The table shows that M5 and Extra-Trees are the best performing models, with M5 having better performance on base and high flow. Indeed, as explained above, the tendency of Extra-Trees in overestimating and underestimating base and high flows, respectively, is due to the model architecture. Unsurprisingly, other models that do not make use of a regression system, such as CART, also show underestimation problems for high flows (and overestimation for base flows). On the other hand, Extra-Trees predictive capabilities are either comparable or better than M5 one on low and intermediate flow conditions. A comparison between the measured and predicted streamflow with two best performing models, i.e. Extra-Trees and M5, is given in Fig. 7, where 95 % confidence bounds associated to each prediction are also reported. These latter are computed as twice the error standard deviation assuming the

Table 8. Percentile-based categorised flow regimes for Marina catchment and Canning River.

Flow regime	Percentile	Marina catchment Flow limit [$m^3 s^{-1}$]	Canning River Flow limit [$m^3 s^{-1}$]
Base flow	< 75th	3.17	0.16
Low flow	75th–95th	3.17–17.10	0.16–1.44
Intermediate flow	95th–99.5th	17.10–169.48	1.44–7.73
High flow	> 99.5th	169.48	7.73

Table 9. Testing results, in terms of RMSE [$m^3 s^{-1}$], of Extra-Trees and benchmarking models for Marina catchment and Canning River datasets on four different flow regimes.

Model	Marina catchment				Canning River			
	Base fl.	Low fl.	Int. fl.	High fl.	Base fl.	Low fl.	Int. fl.	High fl.
Extra-Trees	2.14	4.56	31.64	141.27	0.01	0.15	0.86	3.07
M5	1.73	4.25	31.70	139.75	0.01	0.17	0.89	2.39
CART	2.87	6.57	35.36	152.09	0.02	0.16	1.28	3.41
ANNs	2.70	8.82	40.66	142.13	0.16	0.21	0.98	3.21
MLR	2.23	7.61	33.63	141.08	0.10	0.15	0.83	3.17

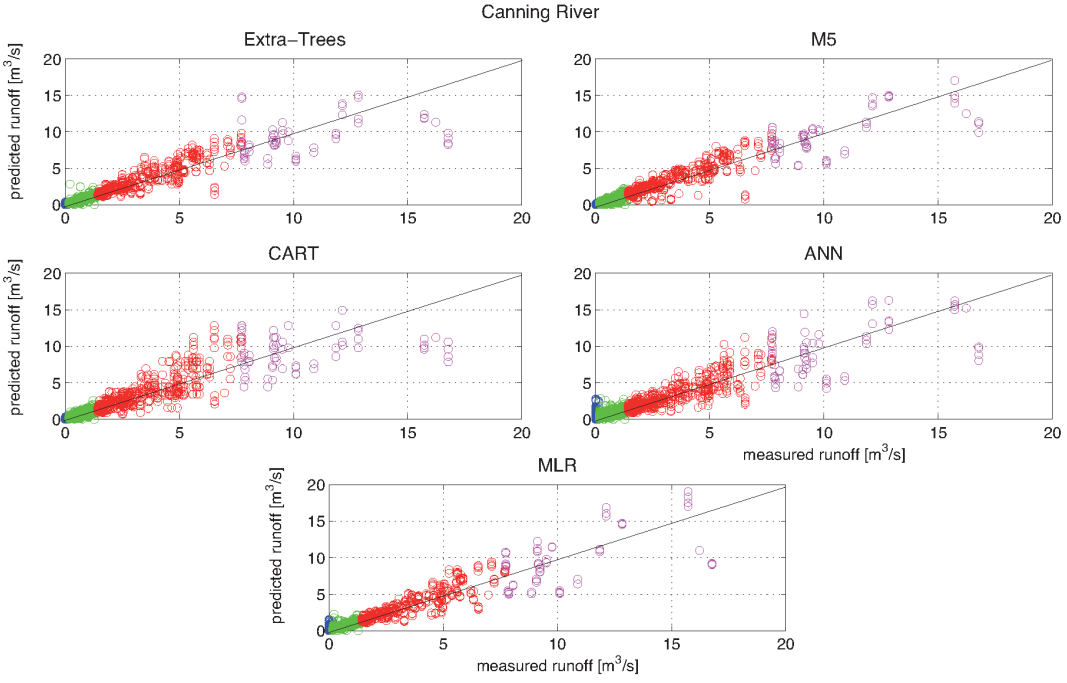

Fig. 6. Scatter plots of predicted (y-axis) and measured (x-axis) streamflow [$m^3 s^{-1}$] in Canning River for the different models on the testing subsets. Different colours are used to represent the flow regimes: blue, green, red and purple correspond to base, low, intermediate and high flow, respectively.

logistic distribution as underlying probability distribution (as explained in the next section).

5.3 Residuals analysis

The Logistic probability distribution, with different parameters α and β, is found to best fit the residuals of the different models in both case studies (on the testing subsets). Although being characterised by the same distribution, a

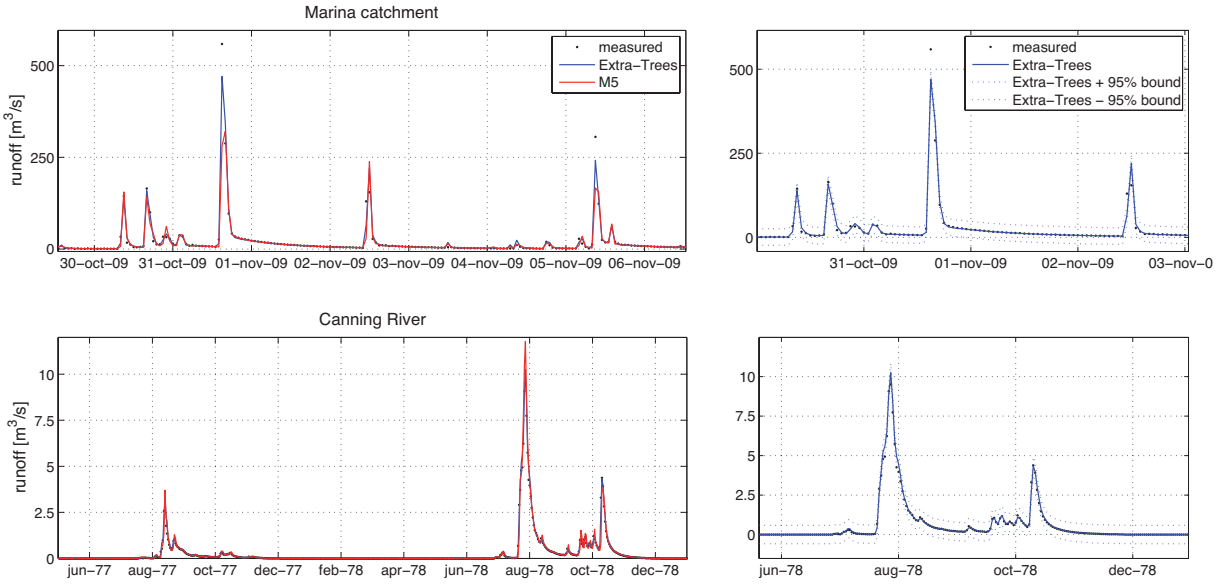

Fig. 7. Comparison between the measured and predicted streamflow (with Extra-Trees and M5) for Marina catchment and Canning River (left panels), and comparison between measured and predicted streamflow with Extra-Trees and 95 % confidence bounds (right panels).

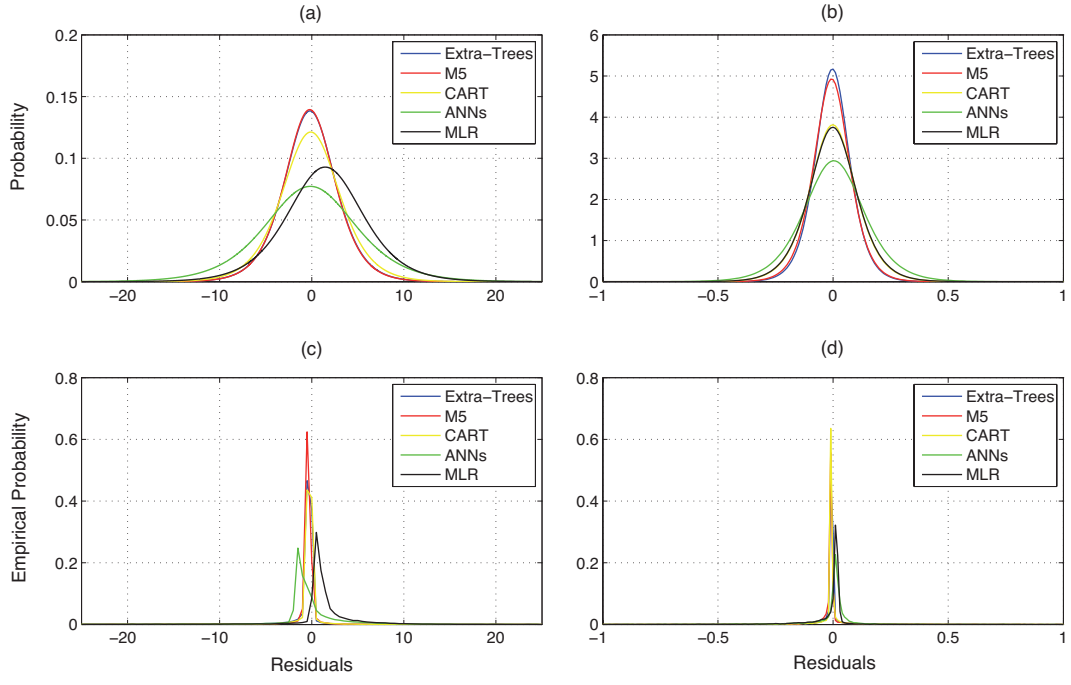

Fig. 8. Fitted logistic and empirical probability distribution of the models residuals for Marina catchment **(a, c)** and Canning River **(b, d)** on the testing subsets.

graphical analysis shows a substantial difference in the estimated parameters (Fig. 8a and b). Apart from MLR, all the models residuals are characterised by a symmetrical distribution, with Extra-Trees and M5 having the smallest predictive uncertainty. These two models are followed by CART and ANNs, which show lower probability of null residuals and a more prominent kurtosis. The linear model residuals are statistically comparable to CART residuals for Canning River case study, while for Marina catchment dataset they show an asymmetrical distribution with higher probability of positive residuals. These findings are confirmed by the empirical distributions (Fig. 8c and d). However, there is not a full resemblance between these latter and the probability distributions, and this could be due to the streamflow generation

Table 10. Comparison of k fold cross-validation (with $k = 10$) and testing CPU time for Extra-Trees, M5, CART, ANNs and MLR for Marina and Canning River dataset. The estimates are with respect to a single (of the 10) data group composing each dataset.

	Marina catchment		Canning River	
Model	k fold cross-valid. [s]	Testing [s]	k fold cross-valid. [s]	Testing [s]
Extra-Trees	1008.876	20.898	78.404	1.202
M5	1788.300	2.045	32.211	0.255
CART	9.891	0.037	1.580	0.011
ANNs	16.691	0.084	8.240	0.079
MLR	0.225	0.019	0.136	0.011

process and the associated model error. Indeed, both Marina catchment and Canning river are characterised by prolonged periods of null (or very little) flow, during which the associated streamflow prediction error is around zero, and shorter periods of high flow, during which the prediction error can raise to larger values. This is reflected by the empirical distributions, which show that the highest frequency of models errors is concentrated in the intervals -2.5 to $2.5\,m^3\,s^{-1}$ and -0.5 to $0.5\,m^3\,s^{-1}$ for Marina catchment and Canning river, respectively. Such very high kurtosis is not fully captured by most of the fitted probability distributions we compared, as they are not capable of concentrating the models errors in a little interval. The comparison between empirical and probability distributions also shows that the fitted logistic for Canning River assumes values larger than one. This may appear misleading, but the probability density function can actually be larger than one, especially if the deviation is relatively low (Box et al., 2005). The overall difference in the pdf parameterisations is also confirmed by the two-sample Kolmogorov–Smirnov test: the p-value is null for all the combinations of models residuals, and it thus indicates that the models residuals may represent different distributions.

5.4 Computational requests

All the cross-validation and testing experiments for M5, CART, ANNs and MLR are carried out in MatLab 7.10.0 (R2010a) environment running on a 2.4 GHz Intel Core 2 Duo with 4 GB Ram. The experiments for Extra-Trees are carried out using a compiled C++ package running on the same machine. From Table 10 it can be noticed that when the different models are applied to the Canning River case study, the computational requests are quite limited, with Extra-Trees and M5 requiring for example 78.40 and 32.21 s, respectively for the cross-validation process of a single data group consisting of 2560 samples (1280 in testing). The computational requests of ANNs are smaller, but it is here necessary to account for the 100 random initialisations (for a single initialisation the computational request is equal to 8.24 s).

On the other hand, the application of these models to Marina catchment problem, characterised by a much larger number of samples (16 080 in cross-validation and 8030 in testing), shows a different picture. The Extra-Trees CPU time to cross-validate an ensemble of 500 Extra-Trees (with $n_{min} = 5$) increases to 1008.88 s, while the amount of time spent on M5 is 1788.30 s. The Extra-Trees model building algorithm is roughly 45 % faster than the M5 one. Apart from the specific model implementation (the C++ executable may be faster than Matlab environment), the reason for this important difference stands in the rule adopted when splitting a node during the building process. The M5 building procedure examines all possible splits by exhaustive search (and then chooses the one that maximises the standard deviation reduction of the output variable), while the Extra-Trees model building algorithm explores only K cut-directions (with K equal to the number of input variables) with corresponding splitting values. Although building an ensemble of trees, the overall computational burden remains limited because of the simple splitting rule.

6 Conclusions

Extra-Trees have been evaluated in their predicting accuracy, explanation ability and computational performance comparatively to other very popular data-driven methods in a streamflow modelling exercise. The analysis was numerically conducted on two hydrological datasets. Results show that (i) Extra-Trees provide good performance on both datasets, in terms of different assessment criteria. In particular, their performance is numerically equivalent to that of the best performing model identified during the benchmarking exercise (i.e. M5) on low and intermediate flows, while it slightly decreases on base and high flow conditions; (ii) despite their ensemble nature, Extra-Trees outperform the other best performing methods in terms of computational efficiency when adopted on large datasets (good scalability), such as Marina catchment; finally, (iii) Extra-Trees provide a physically interpretable ranking of the input variables in terms of relevance in explaining the output. This result on two case studies of "known behaviour" suggests that Extra-Trees could then be adopted in more complex domains, for example, under varying meteorological conditions. In synthesis, it can be argued that Extra-Trees represent a good compromise between predicting accuracy and computational requirements, and they ensure further benefits in terms of explanation ability.

It can also be observed that being a non-parametric method, Extra-Trees do not require any parameter optimisation whereas they provide good performance over a broad range of hyper-parameters. In addition, the combined use of randomization and ensemble averaging is aimed at minimising the output variance without the need for any a-posteriori processing, such as pruning and smoothing (adopted for M5). This has two advantages in that it further simplifies the model identification and it adds to Extra-Trees computational efficiency.

In conclusion, Extra-Trees are a valid alternative to traditional parametric data-driven methods, such as ANNs, and to other non-ensemble tree-based approaches. They can be adopted for any hydrological problem (as they provide performance equivalent to those achievable with parametric methods), and should be recommended for computational intensive problems. These include modelling of large datasets and input selection: large datasets are becoming more frequent in several hydrological applications, such as the modelling of urban hydrological processes, where the short time of concentration of urban catchments requires adopting a very short sampling/modelling time (e.g. one hour in Marina catchment), thus largely adding to the dimensionality of the training and testing datasets.

Acknowledgements. The research presented in this work was carried out as part of the Singapore-Delft Water Alliance (SDWA) Multi-Objective Multiple-Reservoir Management research programme (R-303-001-005-272). This paper forms CWR reference 2649-AC.

Edited by: D. Solomatine

References

Babovic, V. and Keijzer, M.: Rainfall-runoff modeling based on genetic programming, Nord. Hydrol., 5, 331–346, 2002.

Bachmair, S. and Weiler, M.: Hillslope characteristics as controls of subsurface flow variability, Hydrol. Earth Syst. Sci., 16, 3699–3715, doi:10.5194/hess-16-3699-2012, 2012.

Beck, M.: Forecasting environmental change, J. Forecast., 10, 3–19, 1991.

Beven, K.: How far can we go in distributed hydrological modelling?, Hydrol. Earth Syst. Sci., 5, 1–12, doi:10.5194/hess-5-1-2001, 2001.

Bhattacharya, B. and Solomatine, D.: Neural networks and M5 model trees in modelling water level-discharge relationship, Neurocomputing, 63, 381–396, 2005.

Box, G. E. P., Hunter, J. S., and Hunter, W. G.: Statistics for Experimenters: Design, Innovation, and Discovery, 2nd Edition, Wiley, New York, USA, 2005.

Breiman, L.: Bagging predictors, Mach. Learn., 24, 123–140, 1996.

Breiman, L.: Random forests, Mach. Learn., 45, 5–32, 2001.

Breiman, L., Friedman, J., Olsen, R., and Stone, C.: Classification and Regression Trees, Wadsworth & Brooks, Pacific Grove, CA, 1984.

Castelletti, A., Galelli, S., Restelli, M., and Soncini-Sessa, R.: Tree-based reinforcement learning for optimal water reservoir operation, Water Resour. Res., 46, W09507, doi:10.1029/2009WR008898, 2010.

Cutler, A. and Guohua, Z.: PERT – Perfect Random Trees Ensembles, Comp. Sci. Stat., 33, 490–497, 2001.

Dawson, C., Brown, M., and Wilby, R.: Inductive learning approaches to rainfall-runoff modelling, Int. J. Neural Syst., 10, 43–57, 2000.

Dietterich, T.: Ensemble methods in machine learning, Lect. Notes Comput Sc., 1857, 1–15, 2000.

Elshorbagy, A., Corzo, G., Srinivasulu, S., and Solomatine, D. P.: Experimental investigation of the predictive capabilities of data driven modeling techniques in hydrology –Part 1: Concepts and methodology, Hydrol. Earth Syst. Sci., 14, 1931–1941, doi:10.5194/hess-14-1931-2010, 2010a.

Elshorbagy, A., Corzo, G., Srinivasulu, S., and Solomatine, D. P.: Experimental investigation of the predictive capabilities of data driven modeling techniques in hydrology –Part 2: Application, Hydrol. Earth Syst. Sci., 14, 1943-1961, doi:10.5194/hess-14-1943-2010, 201b.

Erdal, H. and Karakurt, O.: Advancing monthly streamflow prediction accuracy of CART models using ensemble learning paradigms, J. Hydrol., 477, 119–128, 2013.

Fonteneau, R., Wehenkel, L., and Ernst, D.: Variable selection for dynamic treatment regimes: a reinforcement learning approach, in: Proceedings of the European Workshop on Reinforcement Learning, 30 June–4 July 2008, Villeneuve d'Ascq, France, 2008.

Freund, Y. and Schapire, R.: Experiments with a new boosting algorithm, in: Proceedings of 13th International Conference on Mach. Learn., 3–6 July 1996, Bari, Italy, 148–146, 1996.

Galelli, S., Goedbloed, A., Schwanenberg, D., and van Overloop, D.: Optimal real-time operation of multi-purpose urban reservoirs: a case study in Singapore, J. Water Res. Pl.-ASCE, doi:10.1061/(ASCE)WR.1943-5452.0000342, in press, 2013.

Geurts, P.: Contributions to Decision Tree Induction: Bias/Variance Tradeoff and Time Series Classification, Ph.D. thesis, University of Liège, Liège, Belgium, 2002.

Geurts, P., Ernst, D., and Wehenkel, L.: Extremely randomized trees, Mach. Learn., 63, 3–42, 2006.

Hejazi, M. and Cai, X.: Input variable selection for water resources systems using a modified minimum redundancy maximum relevance (mMRMR) algorithm, Adv. Water Resour., 32, 582–593, 2009.

Ho, T.: Random decision forests, in: Proceedings of the Third International Conference on Document Analysis and Recognition, Vol. 1, ontreal, Quebec, 278–282, 1995.

Ho, T.: The random subspace method for constructing decision forests, IEEE T. Pattern Anal., 20, 832–844, 1998.

Hsu, K., Gupta, H., and Sorooshian, S.: Artificial neural-network modeling of the rainfall-runoff process, Water Resour. Res., 31, 2517–2530, 1995.

Hundecha, Y., Bardossy, A., and Theisen, H.: Development of a fuzzy logic-based rainfall-runoff model, Hydrolog. Sci. J., 46, 363–376, 2001.

Hwang, S., Ham, D., and Kim, J.: A new measure for assessing the efficiency of hydrological data-driven forecasting models, Hydrolog. Sci. J., 57, 1–18, 2012.

Hyafil, L. and Rivest, R.: Constructing optimal binary decision trees in NP-Complete, Inform. Process. Lett., 5, 15–17, 1976.

Iorgulescu, I. and Beven, K.: Nonparametric direct mapping of rainfall-runoff relationships: an alternative approach to data analysis and modeling?, Water Resour. Res., 40, W08403, doi:10.1029/2004WR003094, 2004.

Jakeman, A. and Hornberger, G.: How much complexity is warranted in a rainfall-runoff model, Water Resour. Res., 29, 2637–2649, 1993.

Jekabsons, G.: M5PrimeLab – M5' Regression Tree and Model Tree Toolbox for Matlab/Octave, Technical Report ver. 1.0.1, Faculty of Computer Science and Information Technology – Riga Technical University, Riga, Latvia, 2010.

Jong, K., Mary, J., Cornuéjols, A., Marchiori, E., and Sebag, M.: Ensemble feature ranking, in: Knowledge Discovery in Databases, Springer, 267–278, 2004.

Jothiprakash, V. and Kote, A.: Effect of pruning and smoothing while using M5 model tree technique for reservoir inflow prediction, J. Hydrol. Eng., 16, 563–574, 2011.

Kuncheva, L. and Whitaker, C.: Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy, Mach. Learn., 51, 181–207, 2003.

Laaha, G. and Blöschl, G.: A comparison of low flow regionalisation methods – catchment grouping, J. Hydrol., 323, 193–214, 2006.

Lin, J.-Y., Cheng, C.-T., and Chau, K.-W.: Using support vector machines for long-term discharge prediction, Hydrolog. Sci. J., 51, 599–612, 2006.

Maier, H. and Dandy, G.: Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications, Environ. Model. Softw., 15, 101–124, 2000.

Quinlan, J.: Learning with continuous classes, in: Proceedings of the 5th Australian Joint Conference on Artificial Intelligence, 16–18 November, Hobart, Australia, 343–348, 1992.

Rasmussen, P., Salas, J., Fagherazzi, L., Rassam, J., and Bobee, B.: Estimation and validation of contemporaneous PARMA models for streamflow simulation, Water Resour. Res., 32, 3151–3160, 1996.

Romanowicz, R., Young, P., Beven, K., and Pappenberger, F.: A data based mechanistic approach to nonlinear flood routing and adaptive flood level forecasting, Adv. Water Resour., 31, 1048–1056, 2008.

Sauquet, E. and Catalogne, C.: Comparison of catchment grouping methods for flow duration curve estimation at ungauged sites in France, Hydrol. Earth Syst. Sci., 15, 2421–2435, doi:10.5194/hess-15-2421-2011, 2011.

See, L., Jain, A., Dawson, C., and Abrahart, R.: Visualisation of hidden neuron behaviour in a neural network rainfall-runoff model, in: Practical Hydroinformatics, vol. 68 of Water Science and Technology Library, edited by: Abrahart, R., See, L., and Solomatine, D., Springer, Berlin, Heidelberg, 87–99, 2008.

Selvalingam, S., Liong, S., and Manoharan, P.: Use of RORB and SWMM models to an urban catchment in Singapore, Adv. Water Resour., 10, 78–86, 1987.

Shamseldin, A. Y., Nasr, A. E., and O'Connor, K. M.: Comparison of different forms of the Multi-layer Feed-Forward Neural Network method used for river flow forecasting, Hydrol. Earth Syst. Sci., 6, 671–684, doi:10.5194/hess-6-671-2002, 2002.

Snelder, T., Lamouroux, N., Leathwick, J., Pella, H., Sauquet, E., and Shankar, U.: Predictive mapping of the natural flow regimes of France, J. Hydrol., 373, 57–67, 2009.

Solomatine, D. and Dulal, K.: Model trees as an alternative to neural networks in rainfall-runoff modelling, Hydrolog. Sci. J., 48, 399–411, 2003.

Solomatine, D. and Ostfeld, A.: Data-driven modelling: some past experiences and new approaches, J. Hydroinform., 10, 3–22, 2008.

Solomatine, D. and Xue, Y.: M5 model trees compared to neural networks: application to flood forecasting in the upper reach of the Huai River in China, J. Hydrol. Eng., 9, 491–501, 2004.

Stravs, L. and Brilly, M.: Development of a low-flow forecasting model using the M5 machine learning method, Hydrolog. Sci. J., 52, 466–477, 2007.

Vezza, P., Comoglio, C., Rosso, M., and Viglione, A.: low flows regionalization in North-Western Italy, Water Resour. Manage., 24, 4049–4074, 2010.

Wang, Y. and Witten, I.: Induction of model trees for predicting continuous classes, in: Proceedings of the European Conference on Mach. Learn., Prague, Czech Republic, 128–137, 1997.

Wehenkel, L.: Automatic Learning Techniques in Power Systems, Kluwer Academic, Boston, USA, 1998.

Wei, W. and Watkins Jr., D.: Data mining methods for hydroclimatic forecasting, Adv. Water Resour., 34, 1390–1400, 2011.

Wheater, H., Jakeman, A., and Beven, K.: Progress and Directions in Rainfall-Runoff Modelling, John Wiley, Chichester, 101–132, 1993.

Xie, J.: Dealing with Water Scarcity in Singapore: Institutions, Strategies, and Enforcement, China: Addressing Water Scarcity Background Paper No. 4, The World Bank – Environment and Social Development Department – East Asia and Pacific Region, Washington, DC, available at: http://www.worldbank.org/eapenvironment/ChinaWaterAAA (last access: 15 January 2013), 2006.

Young, P.: Data-based mechanistic and top-down modelling, in: Proceedings of the First Biennial Meeting of the International Environmental Modelling & Software Society, Vol. I, Lugano, Suisse, 363–374, 2002.

Young, P.: Top-down and data-based mechanistic modelling of rainfall–flow dynamics at the catchment scale, Hydrol. Process., 17, 2195–2217, 2003.

Young, P.: Rainfall-Runoff Modeling: Transfer Function Models, John Wiley & Sons, Ltd, 2006.

Young, P.: Hypothetico-inductive data-based mechanistic modeling of hydrological systems, Water Resour. Res., 49, 915–935, 2013.

Young, P. and Beven, K.: Data-based mechanistic modeling and the rainfall-flow nonlinearity, Environmetrics, 3, 335–363, 1994.

Young, P., Jakeman, A., and Post, D.: Recent advances in the data-based modelling and analysis of hydrological systems, Water Sci. Technol., 36, 99–116, 1997.