Available online at www.sciencedirect.com
SCIENCE DIRECT
Advances in Water Resources 29 (2006) 987–999
Advances in Water Resources
www.elsevier.com/locate/advwatres

Conditional resampling of hydrologic time series using multiple predictor variables: A *K*-nearest neighbour approach

R. Mehrotra *, Ashish Sharma

School of Civil and Environmental Engineering, The University of New South Wales, Kensington, Sydney NSW 2032, Australia

Received 24 May 2005; accepted 26 August 2005
Available online 12 October 2005

Abstract

Unlike parametric alternatives for time series generation, non-parametric approaches generate new values by conditionally resampling past observations using a probability rationale. Observations lying 'close' to the conditioning vector are resampled with higher probability, 'closeness' is defined using a Euclidean or Mahalanobis distance formulation. A common problem with these approaches is the difficulty in distinguishing the importance of each predictor in the estimation of the distance. As a consequence, the conditional probability and hence the resampled series, can offer a biased representation of the true population it aims to simulate. This paper presents a variation of the *K*-nearest neighbour resampler designed for use with multiple predictor variables. In the modification proposed, an influence weight is assigned to each predictor in the conditioning set with the aim of identifying nearest neighbours that represent the conditional dependence in an improved manner. The workability of the proposed modification is tested using synthetic data from known linear and non-linear models and its applicability is illustrated through an example where daily rainfall is downscaled over 15 stations near Sydney, Australia using a predictor set consisting of selected large-scale atmospheric circulation variables.

Keywords: *K*-nearest neighbour resampling; Predictor; Time series; Conditional probability distribution; Downscaling; Non-linear dynamics

1. Introduction

Hydrologic time series models aim to reproduce relevant statistical characteristics of the observed series to aid in the planning, design, operation and management of water resources. As the observed sequence is just one realization amongst an infinite number that may have occurred, reliance on only the historical series underrepresents the uncertainty associated with the resulting design or management plan. This uncertainty is often incorporated through the use of stochastic time series approaches. These modeling approaches are broadly grouped into parametric and non-parametric, the former being expressed via selected parameters (often estimated as functions of sample moments), while the latter, being formulated based on the observed (sampled) variability through an empirically specified conditional probability distribution. Non-parametric stochastic approaches offer an efficient alterative when sufficient data are available. Their intuitive simplicity and transparency have made them attractive and popular for use in hydrology and other sciences.

Two non-parametric methods used commonly for stochastic generation are kernel density estimation and nearest neighbour resampling. Both have been used extensively for a range of hydro-climatological applications [37,14,15,32,27,21,6,28,36,3,10,34,19]. Similarly, considerable use of non-parametric nearest neighbour methods has been made characterizing hydro-climatological systems from a non-linear dynamical systems

* Corresponding author. Address: National Institute of Hydrology, Roorkee 247 667, India. Tel.: +61 2 9326 7071; fax: +61 2 9385 6139.
E-mail address: raj@civeng.unsw.edu.au (R. Mehrotra).

0309-1708/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.advwatres.2005.08.007

perspective (e.g. [1,29]). Readers are referred to a good review paper by Sivakumar [30], on the use of these methods in a non-linear dynamical context.

The rationale behind the above mentioned non-parametric approaches is to generate new realisations using a conditional cumulative distribution function (CDF) that is estimated using the observed data. This conditional CDF is broadly expressed as:

$$P(\mathbf{R}_t|\mathbf{X}_t) = \sum_i p_i \tag{1a}$$

$$p_i = \frac{\psi(\mathbf{X}_t - \mathbf{X}_i)}{\sum_j \psi(\mathbf{X}_j - \mathbf{X}_i)} \tag{1b}$$

where, $\mathbf{R}_t$ is a vector of predictands, $\mathbf{X}_t$, $\mathbf{X}_j$ and $\mathbf{X}_i$, respectively, are multi-variate vectors (also called as feature vectors) containing predictor variables at times t, j and i, p_i is a weight or probability associated with observation i, $\psi(\cdot)$ is a measure of proximity of $\mathbf{X}_t$ to $\mathbf{X}_i$ or $(\mathbf{X}_j)$ (indicative of the probability of selecting $\mathbf{R}_i$ as the basis for simulating the new realisation $\mathbf{R}_t$, this measure being different depending on the non-parametric method used) and $P(\mathbf{R}_t|\mathbf{X}_t)$ is the conditional CDF based on which $\mathbf{R}_t$ is simulated. Note that the subscript i or j is used to denote an observation, while the subscript t is used to denote the time step at which the conditional probability distribution is formulated. Note also that if $\mathbf{X}_t$ is expressed as $\mathbf{R}_{t-1}$, then the above formulation becomes analogous to a classical multi-variate time series model [23]. In the notations used above, one can think of $\mathbf{R}_t$ as a vector of daily rainfall amounts or occurrences at multiple locations in a region that are to be simulated conditional to the predictors $\mathbf{X}_t$ that represent relevant atmospheric indicators. This general formulation is specific to a stochastic downscaling model [19] which forms one of the four examples presented later in the paper. Note also that in the notations used above, vectors or matrices have been represented as "bold", a notation that will be followed elsewhere.

Nearest neighbour based resampling methods use the classic bootstrap described by Yakowitz [35] and Efron and Tibshirani [7]. An important issue in nearest-neighbour resampling is the choice of a function $\psi(\cdot)$ which defines the proximity of K-nearest neighbours of a particular state. This study uses the K-nearest neighbour bootstrap formulation proposed by Lall and Sharma [14] which specifies the proximity $\psi(\cdot)$ in (1b) as:

$$\begin{aligned} \psi(\mathbf{X}_t - \mathbf{X}_i) &= \frac{1}{k} \quad \text{if } k \leqslant K \\ &= 0 \quad \text{if } k > K \end{aligned} \tag{2}$$

where k denotes the number of observations whose distance to $\mathbf{X}_t$ is less than or equal to the distance between $\mathbf{X}_t$ and $\mathbf{X}_i$ in the historical sample, and K is a specified maximum value of k. A commonly used distance formulation, the Euclidean distance $\xi_{t,i}$ between $\mathbf{X}_t$ and $\mathbf{X}_i$ is written as:

$$\xi_{t,i} = \sqrt{\sum_{j=1}^{m} \{s_j(X_{j,i} - X_{j,t})\}^2} \tag{3}$$

where the vector $\mathbf{X}_i$ consists of m predictor variables $X_{j,i}$, $j = 1, \ldots, m$, and s_j is the scaling weight for the jth predictor. It is also possible to use other distance measures (for example see [36,31,34]). For other non-parametric methods the exact formulation of (1a, b) varies (see for example, Sharma [26] for a formulation using kernel density estimation methods).

Predictor variables used in the calculation of Euclidean distances are made dimensionless through standardization using scaling weights. Scaling of variables imparts an equal weight to each predictor in selection of the nearest neighbours. Scaling is also necessary when the feature vector consists of combination of discrete and continuous variables [27,6]. The reciprocal of the sample standard deviation of each predictor variable is a common choice for the scaling weights [14,21,27, 10,3]. However, this distance formulation has been noted to result in an improper representation of the conditional distribution because of which application specific choices have been used in [4,6,3]. For similar reasons, Yates et al. [36] and Wójcik and Buishand [34] used modified representation of the distance measure in (3) by adopting the Mahalanobis distance, as an alternative to the Euclidean distance based on the inverse of the predictor covariance matrix. Souza Filho and Lall [31] used a modified Euclidean distance formulation based on coefficients from a fitted multiple linear regression model among the predictors and predictands. These formulations are discussed in detail later in the paper. The basic concept of expressing the contribution of each predictor variable in defining the response has also been recognized in the applications dealing with the non-linear dynamics of hydrologic processes (see for example [13,11]).

The Euclidean distance formulation in (3) considers all predictors to be equally important in the estimation of the conditional probability. In reality, however, each predictor may have unequal importance and some of them may be inter-correlated and may not be significant at all. These less important, correlated or otherwise redundant predictor variables tend to influence the estimated distance metric and hence the conditional probability, leading to the resampled series offering a biased representation of the true population it aims to simulate.

The effect of inclusion of surplus predictor variables is shown using a synthetic example in Fig. 1 wherein a sinusoidal series of the form $z = \sin(u) + \varepsilon$ is plotted. Thus, the z series is a function of only one predictor variable u, a uniformly distributed random number varying between 0 and 5π, ε being a Gaussian error term

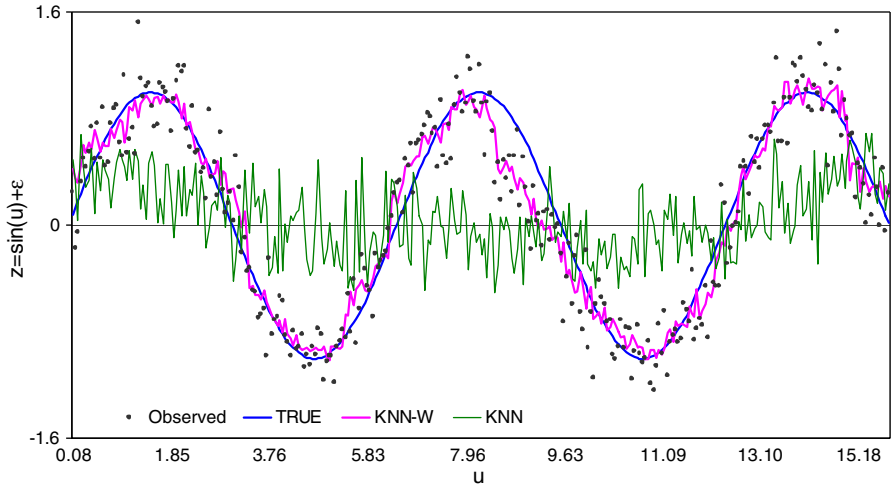

Fig. 1. True, observed and predicted data points using KNN and KNN-W formulations—non-linear synthetic dataset. The points on the graph indicate the series formed, thick continuous line indicates true sine curve, thin dashed line indicate the series generated by KNN and thin continuous line indicates the series generated by KNN-W.

($N(0, 0.2)$). To demonstrate the effect of inclusion of redundant predictor variables, we consider that the feature vector consists of three additional predictor variables, all uniformly distributed random variables having the same range as u, independent of u, z and each other. Giving all four predictors equal importance leads to a biased estimate of the conditional mean from the KNN conditional probability distribution. This bias is much stronger at the two boundaries of the data, and at the points that represent large changes in the slope (peaks and troughs). An alternative formulation (KNN-W) that considers an optimal scaling weight for each predictor does not suffer from the same deficiency, and is the focus of the present study.

The paper is organized as follows. A brief description of different KNN alternatives, our proposed model (KNN-W) and the procedure used to estimate the optimal scaling weights are presented first. In the subsequent section, we demonstrate the importance of inclusion of optimal scaling weights by applying the different KNN formulations to synthetic samples from known linear and non-linear models, and extend the work to downscale synoptic atmospheric patterns in predicting the rainfall at multiple point locations in a region near Sydney, Australia. The last section presents the summary and conclusions drawn from this research.

2. *K*-nearest neighbour resampling

2.1. Background

As described in the previous section, the *K*-nearest neighbour approach estimates the conditional probability on the basis of *K*-nearest neighbours of the conditioning vector $\mathbf{X}_t$. In essence, the K patterns in the historical record that are most similar to the conditioning vector are identified, and the K sets of corresponding predictands are specified as the most likely values the system may assume at time step t. A common way of defining similarity uses the Euclidean distance formulation in (3). Recognizing the fact that the correlated predictors may influence the selection of nearest neighbours, Yates et al. [36] and Wójcik and Buishand [34] used a modified representation of the distance measure in (3) by adopting the Mahalanobis distance, as an alternative to the Euclidean distance and adopting the standardization based on the inverse of the predictor covariance matrix. The Mahalanobis distance measure is defined by:

$$\xi_{t,i} = \sqrt{(\mathbf{X}_t - \mathbf{X}_i)^{\mathrm{T}} \mathbf{C}^{-1} (\mathbf{X}_t - \mathbf{X}_i)} \tag{4}$$

where T represents the transpose operation, $\mathbf{C}^{-1}$ is the inverse of the predictor ($\mathbf{X}$) covariance matrix and $\mathbf{X}_t$ and $\mathbf{X}_i$ have their usual meaning. Note that when non-diagonal terms in $\mathbf{C}$ are replaced with zero, the Mahalanobis distance in (4) reduces to the Euclidean distance formulation in (3) for the case where the scaling weights s_j represent the reciprocal of the sample standard deviation. While the Mahalanobis distance measure considers the existing dependence amongst the predictor variables, it ignores the dependence among the predictors and predictands, an important omission if some of the predictors are redundant in the relationship being explored.

Recognising the relative importance of each predictor in selecting the nearest neighbours, Souza Filho and Lall [31] modified the Euclidean distance estimation based on a weight on each predictor variable. This weight was estimated as the slope coefficient of a linear regression between standardized predictand and predictor sets, the resulting distance being expressed as:

$$\xi_{t,i} = \sqrt{\sum_{j=1}^{m} \{r_j s_j (X_{j,t} - X_{j,i})\}^2} \quad (5)$$

where r_j is the regression coefficient between the jth predictor and predictand, and $\mathbf{X}_t$, $\mathbf{X}_i$, s_j and m have their usual meaning. As the weight is estimated based on linear regression, the distance measure is optimal when dependence is linear. However, the estimated weights are not appropriate if the assumption of linearity is strongly violated, a common case with real datasets in hydrologic studies.

2.2. Modified KNN (KNN-W) model

In the modification proposed here, the Euclidean distance in (3) is modified to include an 'influence' weight for each predictor as follows:

$$\xi_{t,i} = \sqrt{\sum_{j=1}^{m} \beta_j \{s_j (X_{j,i} - X_{j,t})\}^2} \quad (6)$$

where, s_j is the scaling weight as defined earlier, and β_j is the influence weight associated with the jth predictor. The sum of all influence weight equals unity. Introduction of influence weights, $\boldsymbol{\beta}$ ($\boldsymbol{\beta} = [\beta_j]$, $j = 1, \ldots, m$) aims to define the information content of each predictor in the estimation of the conditional cumulative probability density $P(\mathbf{R}_t|\mathbf{X}_t)$ irrespective of whether the underlying relationship is linear or non-linear. In the simplest form, all influence weights, $\boldsymbol{\beta}$ carry equal values and (6) reduces to (3), the traditional KNN model. If the relationship between predictor and predictand is linear, the influence weight β_j closely follows the scaled value $r_j^2/\sum_{i=1}^{m} r_i^2$ of the regression coefficient r_j of the Souza Filho and Lall [31] KNN formulation in (5).

2.3. Estimation of the KNN-W influence weights

The scaling weight vector $\mathbf{s}$ ($\mathbf{s} = [s_j]$, $j = 1, \ldots, m$), number of nearest neighbours K and influence weight vector $\boldsymbol{\beta}$ are the key to estimating accurately, the conditional probability density in the KNN-W approach. The scaling weight vector $\mathbf{s}$ is estimated as the reciprocal of the sample standard deviation associated with each predictor variable. The influence weight vector $\boldsymbol{\beta}$ is ascertained based on an optimization procedure using leave-one-out cross-validation, while a trial-and error criteria is used to estimate K, as described next.

Cross-validation (CV) is a commonly used basis for measuring the predictive error associated with a model. Cross-validation involves developing the model using one part of the sample and assessing its accuracy by applying it on the other, the process being repeated to ensure that all observations are used in estimating the prediction error. Leave-one-out cross-validation (L1CV) is a special case of CV where the model is formulated at each observed data point using all observations except the observation under consideration, and applying the model to predict the observation that has been left out. The procedure is repeated at all observed data points and the differences of observed and predicted values at these data points provide a measure of the predictive error. For the current application, an appropriate set of values for the influence vector $\boldsymbol{\beta}$ is ascertained by minimizing the predictive error associated with the model as assessed using L1CV. In the formulation described next, this measure of the predictive error is used to specify the log-likelihood associated with the $\boldsymbol{\beta}$, the optimal $\boldsymbol{\beta}$ being selected as the vector that results in the maximum log-likelihood score.

It may be mentioned here that as the estimation of the best set of influence weights is exclusively based on the minimization of the predictive error computed using L1CV, the resulting conditional distribution may not be optimal when assessed using statistics at a different scale of aggregation (see for example the assessment of a daily rainfall generation model using the variability in rainfall at an annual time scale in Harrold et al. [10]). If the aim is to formulate influence weights that are optimal with respect to statistics estimated at multiple scales, the likelihood function would need to be suitably modified.

In addition to the influence weight vector $\boldsymbol{\beta}$, the number of K-nearest neighbours is also an unknown that is often difficult to specify in practice. The optimal value of K depends on the number of observations from which nearest neighbours are selected, the number of dependent variables and the nature of the response variable's conditional probability distribution we aim to characterize. If the conditional distribution exhibits low variability (meaning that the system has a high degree of determinism built into it) a smaller K should be used. In addition to the above factors, the optimal K has been noted to differ when the conditional distribution is used for time series simulation as compared to when used for prediction (see for example [6]). In general, the optimal K will be smaller when the purpose is time series simulation than when the purpose is prediction. However, a small value of K may lead to significant duplication of the historic record in the simulated series. For a sufficient sample size ($\approx$100 observations) and small number of predictors ($<$6), Lall and Sharma [14] suggested that, as a basic guideline, K can be chosen as the square root of the length of the observations. For time series simulation, Buishand and Brandsma [6] found better results with small K ($\leqslant$5). Lall and Sharma [14] have also advocated use of generalized cross-validation for obtaining the optimal K value for a particular application. Jayawardena et al. [12] investigated selection of number of nearest neighbours based on generalized degree of freedom using the dynamical system approach and found

better results in comparison to the case when nearest neighbours are arbitrarily selected.

With modifications in the likelihood formulation, the approach we follow for estimating the influence weights $\boldsymbol{\beta}$ can also be used for estimating the optimal value of K for a particular application, number of predictor variables and length of observations. As the aim of this study is to illustrate the importance of inclusion of the influence weights in a multi-predictor formulation of the K-nearest neighbour resampling, we opt not to include the number of nearest neighbours in the optimization procedure described in the next section.

2.4. Adaptive metropolis algorithm

In the present study we have used the recently proposed adaptive metropolis (AM) sampling approach [8,17] to estimate the optimal values of the influence weights $\boldsymbol{\beta}$. The optimization procedure is based on the maximization of log likelihood for continuous data:

$$l(\mathbf{R}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi\sigma^2) - \frac{1}{2\sigma^2}\sum_{t=1}^{N}[R_t - g(\mathbf{X}_t;\boldsymbol{\theta})]^2 \tag{7}$$

where $l(\mathbf{R}|\boldsymbol{\theta})$ is the log likelihood, R_t is the observed and $g(\mathbf{X}_t;\boldsymbol{\theta})$ the predicted value at time step t, $\mathbf{X}_t$ is the predictor vector at time t, σ^2 is the predictive error variance, N is total number of observations, and $\boldsymbol{\theta}$ is the set of model parameters. For the present application, we include σ^2, the predictive error variance and $\boldsymbol{\beta}$, the influence weights as the parameters $\boldsymbol{\theta}$ to be optimised. Note that $g(\mathbf{X}_t;\boldsymbol{\theta})$ is estimated using leave-one-out cross-validation as described in the previous section, as the expected value of the conditional probability distribution specified in (2) and (6).

For the downscaling application presented later in this section, the rainfall values at all stations are pooled together to calculate the likelihood in (7). Thus, the number of observations N is $n \times n_s$, where n_s is the number of stations and n is the total number of observations at each station.

The above likelihood formulation was used to ascertain the optimal parameter vector $\boldsymbol{\theta}$ using the adaptive metropolis (AM) algorithm [8,17]. This algorithm is characterised by a proposal distribution based on the estimated posterior covariance matrix of the parameters. At step t, Haario et al. [8] consider a multi-variate normal proposal $N(\boldsymbol{\theta}_t, \mathbf{C}_t)$ with mean given by the current value, where $\mathbf{C}_t$ is the proposal covariance. The covariance $\mathbf{C}_t$ has a fixed value ($\mathbf{C}_0$) for the first few iterations and is updated after a t_0 iterations as:

$$\mathbf{C}_t = \begin{cases} \mathbf{C}_0 & t \leqslant t_0 \\ \eta\,\mathrm{Cov}(\boldsymbol{\theta}_0, \ldots, \boldsymbol{\theta}_{t-1}) + \eta\varepsilon\mathbf{I}_d & t > t_0 \end{cases} \tag{8}$$

where d is total number of parameters included in the optimization algorithm, $\mathbf{I}$ is a $d \times d$ identity matrix, ε is a parameter chosen to ensure $\mathbf{C}_t$ does not become singular and η is a scaling parameter to ensure reasonable acceptance rates of the proposed states.

The steps involved in implementation of the AM algorithm are:

(a) Initialize $t = 0$ and set $\mathbf{C}_0$ as a diagonal matrix with each diagonal term representing the variance associated with the prior distribution for each unknown.
(b) Update $\mathbf{C}_t$ for the current iteration number t using (8).
(c) Generate a proposed value $\boldsymbol{\theta}^*$ for $\boldsymbol{\theta}$ where $\boldsymbol{\theta}^* \sim N(\boldsymbol{\theta}_t, \mathbf{C}_t)$. If parameter values are outside the parameter constraints, repeat the step again. The constraints for the $\boldsymbol{\beta}$ parameters are defined as:

$$\sum_{j=1}^{m}\beta_j = 1.0 \quad \text{and} \quad 0 < \beta_j < 1.0 \tag{9}$$

These conditions ensure that the relative magnitude of each parameter is retained and it remains bounded. New values of each $\boldsymbol{\beta}$ are generated between 0 and 1 and are summed up. If the sum of $\boldsymbol{\beta}$ is nearly equal to 1 (± 0.01 in the present application), these are retained otherwise procedure is repeated again with the new set of $\boldsymbol{\beta}$ values. Retained set of $\boldsymbol{\beta}$ is rescaled again so that it sums to one.
(d) Compute the log likelihood $l(\mathbf{R}|\boldsymbol{\theta}^*)$ using (7) following the L1CV procedure.
(e) Calculate the acceptance probability, α, of the proposed parameter values using:

$$\alpha = \min\{1, \exp[l(\mathbf{R}|\boldsymbol{\theta}^*) + p(\boldsymbol{\theta}^*) - l(\mathbf{R}|\boldsymbol{\theta}_t) - p(\boldsymbol{\theta}_t)]\} \tag{10}$$

where $p(\boldsymbol{\theta})$ is the log prior distribution of $\boldsymbol{\theta}$, the priors being specified as a uniform PDF over $[0, 1]$ for each element of the influence weight vector $\boldsymbol{\beta}$, and a scaled inverse-chi-square distribution $\chi^{-2}(v, \lambda)$ for the error variance σ^2, the parameters v and λ being specified depending on the nature of the error associated with the true model. In the synthetic examples presented in the next section, v and λ are specified as 0.5 and 10 whereas for the real example these values are specified as 1.0 and 5, respectively.
(f) Generate $u \sim U[0, 1]$.
If $u < \alpha$, accept $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}^*$, otherwise set $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t$.
(g) Repeat (b)–(f) a sufficient number of times to ensure that the posterior distribution of the parameter vector $\boldsymbol{\theta}$ has been sampled exhaustively.

The scaling parameter η and the stopping criterion for our algorithm are based on the recommendations of Marshall et al. [17] and Haario et al. [8]. A reasonable

approach is to start the optimization with initial set of $\boldsymbol{\beta}$ parameters being assigned equal values satisfying (9), and consider σ^2 as the variance of the predictive error estimated using these parameter values. In the results reported next, the optimal $\boldsymbol{\theta}$ has been chosen as the one that results in the maximum log-likelihood in (7) from the assigned number of iterations. Standard convergence tests [17] were used to assess the adequacy of the number of iterations (15,000) assigned for each application, details on which are presented next.

3. Applications

This section illustrates the impact of using optimal influence weights for K-nearest neighbour resampling through four carefully designed experiments and four KNN resampling alternatives. The first three experiments use data from known synthetic models and are designed to test the following hypotheses: (a) the influence weight should assume a value of zero when a predictor variable is redundant; (b) the influence weight should be dictated by the information content each predictor has about the predictand(s); (c) the influence weight should be able to account for the influence of multi-collinearity among the predictor variables; and, (d) inclusion of influence weights should result in an improved performance of the model. The last experiment presents an application to downscale rainfall using a system of atmospheric circulation predictors, the aim being to illustrate the improvements over the other KNN formulations.

The resampling alternatives considered in the study differ in the way the scaling of variables is accomplished. The different scaling alternatives are: (i) reciprocal of the standard deviation (KNN), defined using (3); (ii) inverse of the predictor covariance (KNN-M1), defined using (4); (iii) regression coefficient vector of predictor and predictand and reciprocal of the standard deviation (KNN-M2), defined using (5) and (iv) influence weights and reciprocal of the standard deviation (KNN-W), defined using (6). While the second alternative (KNN-M1) is based on the Mahalanobis distance measure, other alternatives consider the Euclidean distance measure as the choice of the function in identifying the proximity of K-nearest neighbours. It should be noted that the KNN-M1 is based on the nearest neighbour probability metric defined in (1) and (2) and is different to that outlined in Yates et al. [36] which, if used, selects the nearest neighbour with a significantly higher probability. For KNN-M2, regression coefficients relating the predictors and predictand are estimated by carrying out a multiple linear regression analysis. For the real example, a pooled multiple linear regression is carried out instead, for reasons that are discussed later in the section.

3.1. Case studies using synthetic datasets

For all synthetic examples, the number of nearest neighbours K was specified equal to square root of number of observations. This choice was compared with a trial-and-error based estimate and found to be similar. The performances of all the approaches are evaluated by predicting the values in a leave-one-out cross-validation setting and comparing the mean square error (MSE). The mean square error (MSE) is calculated using the following:

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(z_i - \hat{z}_i)^2 \tag{11}$$

where, z_i and $\hat{z}_i$, respectively, are the ith observed and predicted values (observation or statistic) and N is the total number of such values. Here, $\hat{z}_i$ is estimated using leave-one-out cross-validation, and hence MSE represents the predictive error that can be expected when the model is applied to new data.

3.1.1. Linear dataset

A sample of 200 observations was generated using the following linear model:

$$z_i = x_{1,i} + x_{2,i} + \varepsilon_i \tag{12}$$

where, i varies from 1 to 200, $\mathbf{x}_1$ and $\mathbf{x}_2$ are generated from a standard normal PDF $N(0,1)$, and $\boldsymbol{\varepsilon}$ is a noise term also from a normal distribution $N(0,0.7)$. Thus, predictand $\mathbf{z}$ is a known function of $\mathbf{x}_1$ and $\mathbf{x}_2$ with both contributing equally. To evaluate the capability of the method at identifying redundant predictors, two additional variables, $\mathbf{x}_3$ and $\mathbf{x}_4$, both independent and $N(0,1)$ are also included as predictors in the KNN resampling algorithm. Given this configuration, an optimal outcome from the KNN-W should assign nearly equal influence weights (β_1 and β_2) to both $\mathbf{x}_1$ and $\mathbf{x}_2$ and negligible weights (β_3 and β_4) to $\mathbf{x}_3$ and $\mathbf{x}_4$ variables. On the other hand, as KNN considers all variables to be equally important and assigns equal weights to all of them, one would expect the predictive error associated with the KNN to be larger than that for the KNN-W. As the relationship among predictors and predictand is linear, KNN-M2 is expected to perform as well as KNN-W. Also, as all predictors are independent of each other, performance of KNN-M1 should be similar to KNN.

Fig. 2 presents the box-plots of the posterior distribution of influence weights for KNN-W. The optimized values of these parameters are also presented in Table 1. This table also includes the influence weights for KNN and KNN-M2 formulations. For KNN-M2 these weights are equivalent to $r_j^2/\sum_{i=1}^{m} r_i^2$, where r_j is regression coefficient for ith predictor as specified in (5). The specification of influence weights for KNN-M1 model

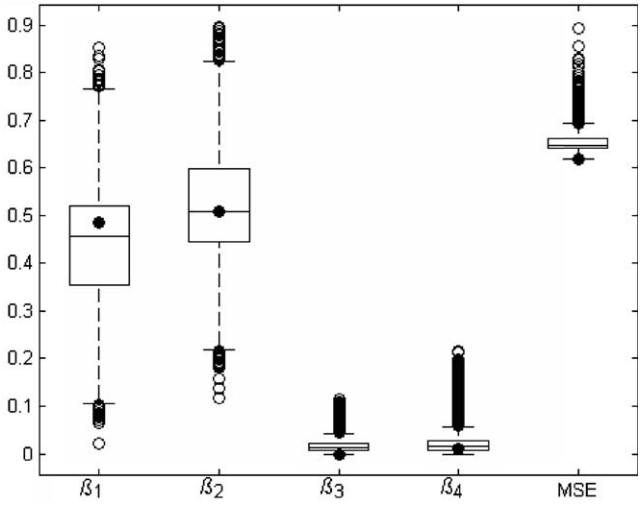

Fig. 2. Box plot of influence weights $\boldsymbol{\beta}$ for each predictor of KNN-W—linear synthetic dataset (Eq. (12)). Filled circles (●) indicate optimized values of influence weights $\boldsymbol{\beta}$ and associated mean square error (MSE). The whiskers extend to 1.5 times the inter-quartile range of the data. Observations outside the whiskers are represented as "○".

Table 1
Optimized values of influence weights ($\boldsymbol{\beta}$) for each predictor and related mean square error (MSE) (using Eq. (11))—linear synthetic dataset (Eq. (12))

Formulation	Influence weights				MSE × 10^2
	β_1	β_2	β_3	β_4	
KNN-W	0.500	0.470	0.020	0.010	64
KNN	0.250	0.250	0.250	0.250	84
KNN-M1	–	–	–	–	85
KNN-M2	0.506	0.490	0.004	0.000	68

The influence weights for KNN-M2 are specified as $r_j^2/\sum_{i=1}^{m} r_i^2$, where r_j is the regression coefficient for the jth predictor and m is number of predictors.

is not straightforward. As can be seen from these results, the KNN-W is able to recognize the inherent structure of the model by identifying variables x_3 and x_4 as redundant predictors with negligible influence weights, and variables x_1 and x_2 as significant predictors with almost equal influence weights, thus establishing the first two of the four hypotheses we seek to test. As the relationship among the predictors and the predictand is linear, KNN-M2 is also able to identify the true nature of the relationship and the scaled values of regression coefficients are found to be similar to the KNN-W influence weights. Table 1 also presents the mean square error (MSE) for all KNN formulations ascertained based on the ability of these models to correctly predict the values using leave-one-out cross-validation. The improved performance of the KNN-W as compared to standard KNN formulation establishes the last of the four hypotheses that are formulated. As expected, KNN-M2 provides MSE similar to KNN-W while KNN-M1 results are similar to KNN.

3.1.2. Non-linear dataset

The second case study is based on a non-linear dataset. A number of theoretical studies relating non-linear dynamics present proof of the possibility of very simple deterministic relationships, like the one presented next, resulting in highly complex outcomes, with sensitive dependence on initial conditions [9,18]. For this study, we generate a sample of 300 observations from an order-1 self-exciting threshold autoregressive (SETAR) non-linear model as described in Tong [33, pp. 99–101] and Lall and Sharma [14]. The model structure is similar to that of an order one autoregressive model except that the parameters of the model adopt different values when crossing a specified threshold. As discussed in Lall and Sharma [14], the SETAR model structure resembles a stream flow record from a catchment having multiple causative mechanisms (such as an overland flow and a coupled snowmelt and overland flow driven response). The order-1 SETAR model is written as:

$$\begin{aligned} z_i &= 0.4 + 0.8 z_{i-1} + \varepsilon_i \quad \text{if } z_{i-1} \leqslant 0.0 \\ z_i &= -1.5 - 0.5 z_{i-1} + \varepsilon_i \quad \text{otherwise} \end{aligned} \tag{13}$$

where i varies from 1 to 300, $\boldsymbol{\varepsilon}$ represents noise from a normal distribution $N(0, 0.60)$, and z denotes the response from the model. The auto correlation function (ACF) and the partial auto correlation function (PACF) provide a useful basis for ascertaining the structure of an autoregressive moving average (ARMA) model [24]. Fig. 3 presents the ACF and PACF for the series generated using (13). A Markov order 6 dependence structure can be concluded from these plots, under the assumption that the dependence is linear. We use these results to construct a feature vector consisting of six variables $[z_{i-1}, z_{i-2}, \ldots, z_{i-6}]$ in the results presented next. As with the previous example, we expect KNN-W to assign near unit influence weight to the first predictor ($\beta_1 \approx 1$) and negligible influence weights to the remaining five predictors ($\beta_2, \beta_3, \beta_4, \beta_5$ and $\beta_6 \approx 0$). As now the relationship among predictors and predictand is non-linear, we expect the performance of KNN, KNN-M1 and KNN-M2 to be inferior as compared to the KNN-W.

The optimal values of influence weights are presented in Table 2. This table also includes the influence weights for KNN and KNN-M2 and associated MSE values for all the formulations. As can be seen from these results, the KNN-W is able to identify the five additional variables as surplus predictors with negligible influence weights, and the lag one variable as the sole significant predictor with a weight nearing unity. As can be inferred from this table, the KNN-W results offer improvement over other KNN formulations in terms of MSE.

An additional test was conducted to evaluate the performance of all approaches to simulate the non-linearity between the response and predictors. The predicted observations from each model (z_i) are divided into two

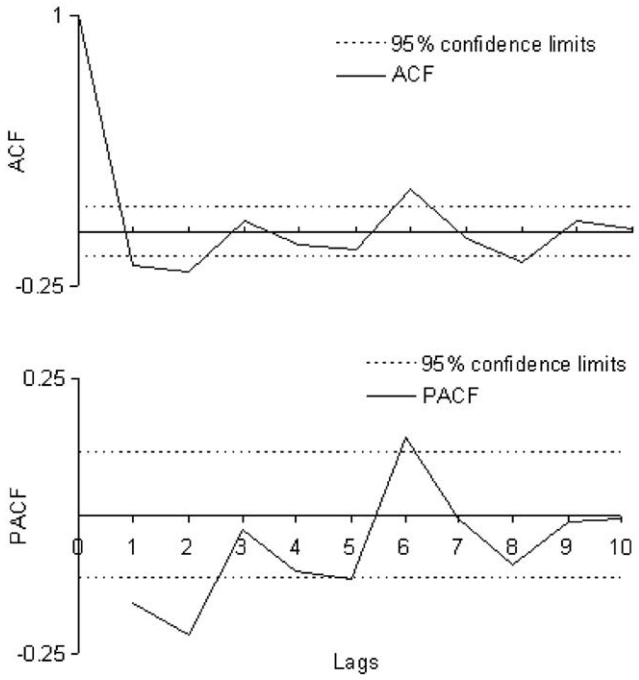

Fig. 3. Sample auto correlation function (ACF) and partial auto correlation function (PACF) plots—non-linear SETAR model.

sub-sets depending on the values at the previous time step (z_{i-1}) being $\leqslant 0$ or >0. A linear regression model is fitted to each sub-set and regression coefficients estimated. Table 3 presents the values of these coefficients for all models and the synthetic sample used. It is apparent that the non-linear dependence relationship between z_i and z_{i-1} is best simulated by the KNN-W formulation.

3.1.3. Multi-collinear dataset

The third case study evaluates the performances of KNN-W and other KNN formulations when the predictors are highly correlated. The study is based on a sample of 200 observations generated using the following linear model:

$$z_i = x_{1,i} + x_{2,i} + x_{3,i} + \varepsilon_i \tag{14}$$

where, i varies from 1 to 200, $\mathbf{x}_1$ and $\mathbf{x}_3$ are generated from a standard normal PDF $N(0,1)$ and $\boldsymbol{\varepsilon}$ is a noise term also from a normal distribution $N(0,0.5)$. In order to evaluate the affect of multi-collinearity among the predictors on the performance of KNN formulations, $\mathbf{x}_2$ is set equal to $\mathbf{x}_1$ times $U(0,2)$, with $U(0,2)$ representing a uniformly distributed random variate between 0 and 2. Thus, predictand $\mathbf{z}$ is a known function of $\mathbf{x}_1$, $\mathbf{x}_2$ and $\mathbf{x}_3$ with $\mathbf{x}_1$ and $\mathbf{x}_2$ being highly correlated (coefficient of correlation 0.87). An additional predictor variable, $\mathbf{x}_4$, also $N(0,1)$, is included as a redundant predictor.

Table 3
Regression coefficients (slope and intercept) obtained from generated $[z_i, z_{i-1}]$ pairs for $z_{i-1} \leqslant 0$ and $z_{i-1} > 0$—SETAR model dataset (Eq. (13))

Model formulations	Regression coefficients			
	$z_{i-1} \leqslant 0.0$		$z_{i-1} > 0.0$	
	Slope	Intercept	Slope	Intercept
True values	0.800	0.400	−0.500	−1.500
Generated data	0.700	0.317	−0.543	−1.430
KNN-W	0.626	0.217	−0.830	−1.154
KNN	0.322	−0.200	−0.856	−0.782
KNN-M1	0.303	−0.228	−0.887	−0.741
KNN-M2	0.456	−0.010	−1.012	−0.843

Note the performance of KNN-W is simulating coefficients that are closest to those of the synthetic sample used.

For the multi-collinear data set, as KNN assigns equal weights to all predictors, one would expect the predictive error associated with the KNN to be larger than that for the KNN-W. As the relationship among predictors and predictand is still linear, KNN-M2 is expected to perform as well as KNN-W. Also, as two predictors are highly correlated, performance of KNN-M1 is intuitively expected to be better than KNN, an expectation that is contradicted by the results (Table 4).

The influence weights and MSE's for the various KNN formulations are presented in Table 4. As can be seen from these results, the KNN-W and KNN-M2 assign weights that are comparable to each other (noting that the weights for x_1 and x_2 exhibit greater variability because of the high correlation). As was expected, KNN-W and KNN-M2 perform better than KNN. Interestingly, KNN-M1 performs worse than the regular KNN formulation. We feel that this is because the influence of each predictor in KNN-M1 is based on the

Table 2
Optimized values of influence weights ($\boldsymbol{\beta}$) for each predictor and related mean square error (MSE) (using Eq. (11))—non-linear SETAR model (Eq. (13))

Formulation	Influence weights						$MSE \times 10^2$
	β_1	β_2	β_3	β_4	β_5	β_6	
KNN-W	0.909	0.008	0.006	0.020	0.026	0.031	41
KNN	0.167	0.167	0.167	0.167	0.167	0.167	56
KNN-M1	–	–	–	–	–	–	59
KNN-M2	0.288	0.422	0.030	0.049	0.060	0.151	47

The influence weights for KNN-M2 are specified as $r_j^2 / \sum_{i=1}^{m} r_i^2$, where r_j is the regression coefficient for the jth predictor and m is number of predictors.

Table 4
Optimized values of influence weights (**β**) for each predictor and related mean square error (MSE) (using Eq. (11))—multi-collinear dataset (Eq. (14))

Formulation	Influence weights				MSE × 10^2
	β_1	β_2	β_3	β_4	
KNN-W	0.505	0.354	0.139	0.001	52
KNN	0.250	0.250	0.250	0.250	65
KNN-M1	–	–	–	–	96
KNN-M2	0.351	0.407	0.240	0.002	55

The influence weights for KNN-M2 are specified as $r_j^2/\sum_{i=1}^{m} r_i^2$, where r_j is the regression coefficient for the jth predictor and m is number of predictors.

dependence characteristics between the predictors, without taking account of the dependence that exists between the predictors and the predictand. Were the influence of all predictors on the response similar, and they were correlated with each other, one could have expected the KNN-M1 to outperform all other alternatives. As that is not the case here, the redundant predictor ends up having a stronger influence, leading to poor predictive results.

We have demonstrated that inclusion of influence weights in defining Euclidean distances and thereby estimating the conditional PDF results in an improvement in the resampling procedure in both linear and non-linear settings. Similar improvements can be expected in an application where the true form of the underlying model is not known. This is demonstrated next by downscaling rainfall at a network of raingauges near Sydney, Australia.

3.2. Downscaling of rainfall near Sydney, Australia

The downscaling application uses 43 years (1960–2002) of daily rainfall at 15 stations located around Sydney, New South Wales, Australia for June (winter in the southern hemisphere). In an earlier study Mehrotra et al. [19] identified the mean sea level pressure (MSLP), geopotential heights (GPH) at 700 h Pa and their gradients as plausible atmospheric circulation indicators of the rainfall patterns in this region. Consequently, based on the results of their study and the recommendations of the meteorologists about the rainfall driving mechanism about the region, a total of six atmospheric circulation variables are identified as potential predictors for the downscaling example. These consist of averaged MSLP and the east–west and north–south gradients of MSLP of the current as well as previous days over the study region. The previous day average rainfall of the region is also identified as an additional predictor variable for this example. The predictands in this application are daily rainfall amounts at the 15 stations. A trial and error procedure is performed to find out the optimal value of K, the number of nearest neighbours, and consequently a value of $K = 15$, is adopted for use in all the approaches considered for this example.

The influence weights for this example are estimated using the optimization procedure and the likelihood measure in (7) as described in the Section 2. Once the influence weights are identified, we evaluate the performances of all KNN formulations by predicting daily rainfall amounts in a leave-one-out cross-validation setting. The leave-one-out cross-validation is performed by downscaling for day t based on the conditional CDF in (2) and (3) with neighbours that do not include the data for day t. Results from all the models are ascertained by resampling 100 realisations of rainfall amounts for each day of the observed record, based on which various performance measures are computed.

The numerical comparison of these models is based on the estimation of the mean square error (MSE) for selected attributes of generated rainfall amounts and occurrence. A graphical comparison is performed on the basis of a performance measure indicating the success at simulating wet or dry days similar to what is observed. The mean square error in (11) for rainfall amount/occurrence/associated attribute can be written as:

$$\mathrm{MSE} = \frac{1}{N_S TN} \sum_{l=1}^{N_S} \sum_{j=1}^{T} \sum_{i=1}^{N} (x_{l,i} - \hat{x}_{l,j,i})^2 \tag{15}$$

where, $x_{l,i}$ is the ith observed value at lth station, $\hat{x}_{l,j,i}$ is the ith predicted rainfall amounts/occurrences/attribute at the lth station and for the jth realization, N indicates number of observations, N_S is number of stations and T is number of realizations. Note that the number of observations is more (30×43) when estimating the MSE for daily rainfall amount/occurrence, and less when the MSE refers to amounts/occurrences under a specific category (such as days on which fewer than 33% stations had rain) or a rainfall attribute (such as monthly maximum rainfall). Note also that the series of rainfall amount includes days with both zero and non-zero amounts and results for the monthly maximum rainfall compare the historical monthly maximum value to what is downscaled on the same day for each year.

Table 5 presents the optimized values of influence weights for all predictors and for KNN, KNN-W and KNN-M2 approaches. As can be seen from the table, KNN-W and KNN-M2 identify MSLP and north–south gradient of MSLP as the most significant predictors. All KNN formulations are also found adequate in reproducing successfully the spatial rainfall distribution (results not included in the paper) due to the resampling of an entire observed rainfall vector for a given day at all stations in a single go. Table 6 provides the MSE for various rainfall attributes of interest. Also included are the percentage of stations at which a given model

Table 5
Optimized values of influence weights (**β**) for each predictor—multi-site rainfall dataset

Formulation	Influence weights[a]						
	β_1	β_2	β_3	β_4	β_5	β_6	β_7
KNN-W	0.107	0.314	0.249	0.065	0.027	0.186	0.051
KNN	0.143	0.143	0.143	0.143	0.143	0.143	0.143
KNN-M1	–	–	–	–	–	–	–
KNN-M2	0.145	0.340	0.340	0.007	0.049	0.116	0.002

The influence weights for KNN-M2 are specified as $r_j^2/\sum_{i=1}^{m} r_i^2$, where r_j is the regression coefficient for the jth predictor and m is number of predictors.

[a] Predictor 1 represents average wetness fraction of the previous day, predictors 2, 3, 4 and 5, 6 and 7, respectively, represent the average MSLP, north–south gradient of MSLP and east–west gradient of MSLP of the current and the previous day.

is found to have the highest success in reproducing each of the rainfall attributes tabled. Note that the rainfall occurrence results are based on the assumption that a day is wet if the rainfall amount on that day is greater than or equal to 0.3 mm (after [10,5]). As can be inferred from this table, inclusion of influence weights offers improvements in the predicted results at a majority of stations. It is important to note that these improvements hold even when the stations are segregated based on the fraction on which rain occurs. Hence one can conclude that the KNN-W offers a better downscaling performance than the other alternatives across a range of rainfall events (spotty convective events where few stations receive rainfall to sustained frontal events where most of the stations record rain).

To evaluate further the performance of a model, attention is paid to the representation of the daily rainfall occurrences on each day classified as wet or dry. This offers an additional means of evaluating the performance of KNN-W as the influence weight estimation procedure is based on the use of rainfall amounts and hence does automatically translate into an improved performance of the model when evaluating rainfall occurrence results. Fig. 4 presents the success rates of model predicted conditional and unconditional wet and dry days at each station. The unconditional success rate is defined as the ratio of number of days when both observed and predicted values are either wet or dry to the total number of days in the observed record. In case of perfect match the success rate approaches unity. Similarly, the conditional success rate of previous day being wet (dry) is defined as the ratio of number of days when both observed and predicted values of the current day are either wet or dry with the previous day value in the observed and predicted record being wet (dry). The accurate reproduction of conditional wet and dry days defines the day to day persistence of the daily rainfall and also helps in representing the wet and dry spells of longer durations. As such these are of prime concern in catchment management studies. All the models under estimate the success rates of wet and dry days. However, KNN-W offers better results in reproducing these statistics at majority of stations.

The overall divergence of predicted results from the observed ones could be owing to the number of stations and the choice of predictor variables considered in the present application. This may be especially important when evaluating the performance of KNN-M2 in the above example. The basic requirement of KNN-M2 lies in estimating accurately the regression coefficients of predictors and predictand. As our real data set contains multiple predictands (raingauge stations), pooled regression was used to estimate the regression coefficients of KNN-M2 as per Souza Filho and Lall [31]. However, it was found that with a higher number of stations (similar experimental settings as of Mehrotra et al. [19] and Mehrotra and Sharma [20]) than was used in the above example, the basic assumption of pooled-regression became invalid. This was due to the estimated regression coefficients being significantly different when only coast-

Table 6
Mean square error (MSE) for selected attributes of generated rainfall amounts and occurrences (using Eq. (15)), and percentage of stations with minimum MSE—multi-site rainfall dataset

Attributes	Mean square error (MSE)				Percentage of stations with minimum MSE			
	KNN-W	KNN	KNN-M1	KNN-M2	KNN-W	KNN	KNN-M1	KNN-M2
Daily rainfall amount (mm^2)	89.1	97.3	96.8	97.7	87	0	13	0
Daily rainfall occurrence	0.258	0.268	0.275	0.264	93	0	7	0
Daily maximum rainfall (mm^2)	1079	1184	1203	1143	80	0	0	20
Less than 33% of stations are wet								
Amounts (mm^2)	11.9	14.7	14.9	13.7	100	0	0	0
Occurrence	0.178	0.184	0.188	0.187	80	7	13	0
33–66% of stations are wet								
Amounts (mm^2)	23.1	24.3	24.2	25.6	53	7	33	7
Occurrence	0.124	0.126	0.129	0.125	60	7	7	26
Greater than 66% of stations are wet								
Amounts (mm^2)	111.2	120.5	119.5	120.8	73	0	20	7
Occurrence	0.122	0.129	0.134	0.122	50	0	0	50

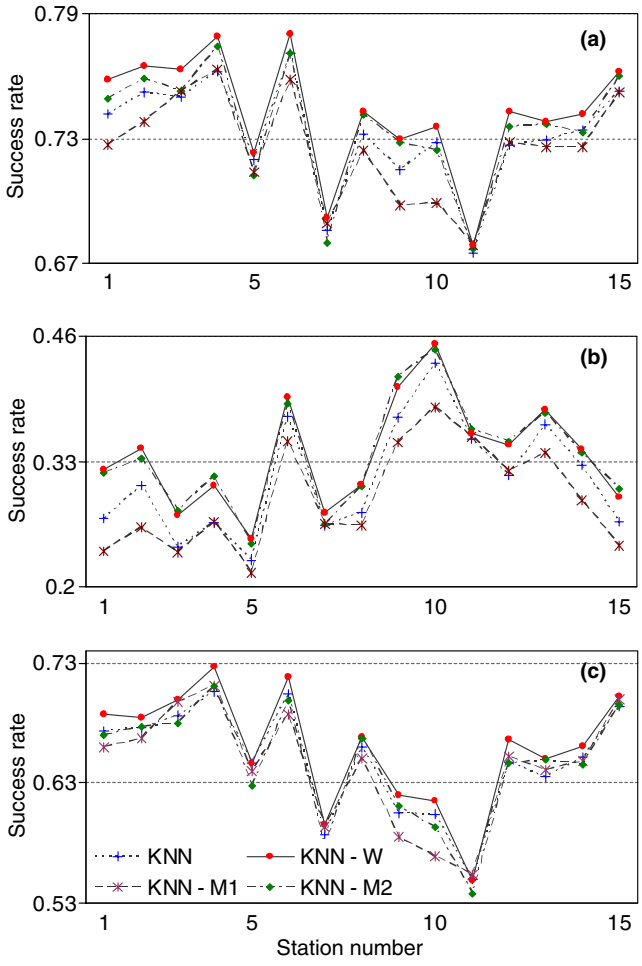

Fig. 4. Success rates of number of days being wet and dry (a) unconditional, (b) conditional on previous day being wet and (c) conditional on previous day being dry in the predicted record for KNN, KNN-W, KNN-M1 and KNN-M2 at each station—multi-site rainfall dataset.

al rainfall stations were used, as compared to the case where inland stations were used (results not presented here for lack of space but available from authors on request). It was only after this testing that the above experimental design (15 raingauge stations located near the coastal region) was finalized under the assumption that they represent a meteorologically homogeneous region. This represents a limitation of the pooled regression approach and indirectly of KNN-M2 when multiple responses are being modelled.

It was also felt that the atmospheric circulation variables selected as potential predictors are not adequate in defining the temporal correlation of rainfall at each station successfully. We feel that inclusion of a better mechanism explaining the lagged spatial distribution of rainfall patterns can offer improved results. Readers are refereed to one such downscaling formulation in Mehrotra and Sharma [20], which uses the proposed influence weight logic coupled with a framework for defining persistence in the observed rainfall pattern at the previous time-step as the basis for downscaling rainfall occurrence in a network of raingauges. Defining the relative contribution of each predictor variable by influence weights, however, to some extent, helps overcome this deficiency, which is the main focus of the present study.

4. Summary and conclusions

We presented a variation of the traditional KNN time series resampling approach to deal with the problem of using multiple predictors in the specification of the conditional probability density. The utility of the proposed approach is assessed by applying it to linear, non-linear and multi-collinear synthetic datasets and comparing the results with the other KNN formulations. The results of the study indicate that the proposed modification correctly identifies and simulates both linear and non-linear dependence between the predictor and predictands, even when predictors are strongly inter-correlated. The modified formulation is also applied to downscale daily rainfall using multiple predictors and improvements in the results as compared to other KNN formulations are noted.

The influence of magnitude of unexplained noise term on the performances of various models was also investigated thoroughly in the study using the synthetic datasets. This was achieved by varying the value of ε in Eqs. (12)–(14). All the models were run on the datasets obtained by varying the ε in increments of 0.2 starting from 0.2 and ending at 1. As expected, the capability of KNN-W in identifying the true form of relationship decreases with the increase in the noise in the data. However, still the MSE is found the lowest in comparison to other KNN formulations. The effect of noise, number of predictors and non-linearity of predictor and predictands on the selection of the number of nearest neighbours was also investigated. In general, with the increase in the noise, better results are obtained with the large K. However, with the increase in the number of predictors and non-linearity of the relationship of predictor and predictands, the optimal K tends to decrease (detailed results of these investigations are available on request from the authors).

The KNN-M1 considers existing dependence amongst the predictor variables, however, ignores the dependence among the predictors and predictands. KNN-M2 works best when predictors and response are linear or near-linear, and is a considerably simpler alternative equivalent to KNN-W for modelling such systems. However, with multiple response variables, estimation of the regression coefficients using pooled-regression poses difficulties, as one is required to assume that the same relationship is valid between each response and the

predictor set. This assumption is found especially wanting when variability amongst the responses is due to multiple causative mechanisms. The KNN-W approach is far more general than any of the existing KNN alternatives, and is worthy of consideration whenever the system behaves in a non-linear manner with different influences of the associated predictors.

One of the significant advantages of the KNN framework is the simplicity with which multiple predictor variables can be accommodated in the feature vector. The proposed modification can be used to find the significance of the added predictor(s) as compared to the existing predictors in defining the predictand(s). It is also possible to include the number of nearest neighbour K, for the given dataset, in the optimization procedure using a modified version of the model likelihood. The "curse of dimensionality" [25] is less felt in the KNN-W formulation due to the reduced weight associated with redundant or irrelevant predictor variables. Consequently, formulation of higher dimensional nonparametric models, such as those used for disaggregation [32], multi-variate stochastic simulation [24] or downscaling [19] is simplified through the use of the influence weight logic proposed here.

Another possible field of application of the influence weight logic presented here would be studies dealing with the issue of non-linearity in the hydrological processes using the nearest neighbour approach. The influence weight logic can help understand better the behaviour of many hydrologic and meteorologic phenomena which have been shown to exhibit non-linear deterministic behavior (e.g., [16,22,1,2,29,30]).

Acknowledgments

This work was partially funded by the Australian Research Council. We wish to acknowledge the constructive comments of Bellie Sivakumar, one of the AWR reviewers, whose inputs greatly benefited the quality of our presentation.

References

[1] Abarbanel HDI, Lall U. Nonlinear dynamics of the Great Salt Lake: system identification and prediction. Clim Dyn 1996;12:287–97.
[2] Abarbanel HDI, Lall U, Moon YI, Mann M, Sangoyomi T. Nonlinear dynamics and the Great Salk Lake: a predictable indicator of regional climate. Energy 1996;21(7/8):655–66.
[3] Beersma JJ, Buishand TA. Multi-site simulation of daily precipitation and temperature conditional on the atmospheric circulation. Clim Res 2003;25:121–33.
[4] Brandsma T, Buishand TA. Simulation of extreme precipitation in the Rhine basin by nearest neighbour resampling. Hydrol Earth Syst Sci 1998;2:195–209.
[5] Buishand TA. Some remarks on the use of daily rainfall models. J Hydrol 1978;36:295–308.
[6] Buishand TA, Brandsma T. Multisite simulation of daily precipitation and temperature in the Rhine basin by nearest-neighbor resampling. Water Resour Res 2001;37:2761–76.
[7] Efron B, Tibshirani RJ. An introduction to the bootstrap. New York: Chapman and Hall; 1993.
[8] Haario H, Saksman E, Tamminem J. An adaptive Metropolis algorithm. Bernoulli 2001;7(2):223–42.
[9] Henon M. A two-dimensional mapping with a strange attractor. Commun Math Phys 1976;50:69–77.
[10] Harrold TI, Sharma A, Sheather SJ. A nonparametric model for stochastic generation of daily rainfall occurrence. Water Resour Res 2003;39(10):1–11.
[11] Jayawardena AW, Lai F. Analysis and prediction of chaos in rainfall and stream flow time series. J Hydrol 1994;153:23–52.
[12] Jayawardena AW, Li WK, Xu P. Neighborhood selection for local modeling and prediction of hydrological time series. J Hydrol 2002;258:40–57.
[13] Laio F, Porporato A, Revelli R, Ridolfi L. A comparison of nonlinear flood forecasting methods. Water Resour Res 2003;39(5):1129. doi:10.1029/2002/WR001551.
[14] Lall U, Sharma A. A nearest neighbor bootstrap for time series resampling. Water Resour Res 1996;32:679–93.
[15] Lall U, Rajagopalan B, Tarboton DG. A nonparametric wet/dry spell model for resampling daily precipitation. Water Resour Res 1996;32(9):2803–23.
[16] Lorenz EN. Deterministic nonperiodic flow. J Atmos Sci 1963;20:130–41.
[17] Marshall L, Nott D, Sharma A. A comparative study of Markov chain Monte Carlo methods for conceptual rainfall-runoff modelling. Water Resour Res 2004;40. doi:10.1029/2003WR002378.
[18] May RM. Simple mathematical models with very complicated dynamics. Nature 1976;261:459–67.
[19] Mehrotra R, Sharma A, Cordery I. Comparison of two approaches for downscaling synoptic atmospheric patterns to multisite precipitation occurrence. J Geophys Res 2004;109:D14107. doi:10.1029/2004JD004823.
[20] Mehrotra R, Sharma A. A nonparametric nonhomogeneous hidden Markov model for downscaling of multi-site daily rainfall occurrences. J Geophys Res 2005;110:D16108. doi:10.1029/2004JD005677.
[21] Rajagopalan B, Lall U. A k-nearest neighbour simulator for daily precipitation and other weather variables. Water Resour Res 1999;35(10):3089–101.
[22] Rodriguez-Iturbe I, De Power FB, Sharifi MB, Georgakakos KP. Chaos in rainfall. Water Resour Res 1989;25(7):1667–75.
[23] Salas JD. Analysis and modelling of hydrologic time series. In: Maidment DR, editor. Handbook of hydrology. New York: McGraw-Hill; 1993. p. 19.1–19.72.
[24] Salas JD, Delleur JW, Yevjevich V, Lane WL. Applied modeling of hydrologic time series. CO, USA: Water Resource Publications; 1980. 482p.
[25] Scott DW. Multivariate density estimation: theory, practice and visualization. New York: John Wiley; 1992.
[26] Sharma A. Seasonal to interannual rainfall probabilistic forecasts for improved water supply management: Part 3—A nonparametric probabilistic forecast model. J Hydrol 2000;239:249–58.
[27] Sharma A, Lall U. A nonparametric approach for daily rainfall simulation. Math Comput Simulat 1999;48:361–71.
[28] Sharma A, O'Neill R. A nonparametric approach for representing interannual dependence in monthly streamflow sequences. Water Resour Res 2002;38(7):5.1–5.10.
[29] Sivakumar B. Chaos theory in hydrology: important issues and interpretations. J Hydrol 2000;227(1–4):1–20.

[30] Sivakumar B. Chaos theory in geophysics: past, present and future. Chaos, Solitons and Fractals 2004;19(2):441–62.
[31] Souza Filho F, Lall U. Seasonal to internannual ensemble streamflow forecasts for Ceara, Brazil: applications of a multivariate, semiparametric algorithm. Water Resour Res 2003; 39(11):1307. doi:10.1029/2002WR001373.
[32] Tarboton DG, Sharma A, Lall U. Disaggregation procedures for stochastic hydrology based on nonparametric density estimation. Water Resour Res 1998;34(1):107–19. 22.
[33] Tong H. Nonlinear time series analysis: a dynamical systems perspective. San Diego, Calif: Academic; 1990.
[34] Wójcik R, Buishand TA. Simulation of 6-hourly rainfall and temperature by two resampling schemes. J Hydrol 2003;273: 69–80.
[35] Yakowitz S. Nonparametric density estimation, prediction, and regression for Markov sequences. J Am Stat Assoc 1985;80(389): 215–21.
[36] Yates D, Gangopadhyay S, Rajagopalan B, Strzepek K. A technique for generating regional climate scenarios using a nearest-neighbor algorithm. Water Resour Res 2003;39(7):1–15.
[37] Young KC. A multivariate chain model for simulating climatic parameters from daily data. J Appl Meteorol 1994;33:661–71.