WATER RESOURCES RESEARCH, VOL. 35, NO. 10, PAGES 3089–3101, OCTOBER 1999

A *k*–nearest-neighbor simulator for daily precipitation and other weather variables

Balaji Rajagopalan
Lamont-Doherty Earth Observatory, Columbia University, Palisades, New York

Upmanu Lall
Utah Water Research Laboratory, Utah State University, Logan

Abstract. A multivariate, nonparametric time series simulation method is provided to generate random sequences of daily weather variables that "honor" the statistical properties of the historical data of the same weather variables at the site. A vector of weather variables (solar radiation, maximum temperature, minimum temperature, average dew point temperature, average wind speed, and precipitation) on a day of interest is resampled from the historical data by conditioning on the vector of the same variables (feature vector) on the preceding day. The resampling is done from the k nearest neighbors in state space of the feature vector using a weight function. This approach is equivalent to a nonparametric approximation of a multivariate, lag 1 Markov process. It does not require prior assumptions as to the form of the joint probability density function of the variables. An application of the resampling scheme with 30 years of daily weather data at Salt Lake City, Utah, is provided. Results are compared with those from the application of a multivariate autoregressive model similar to that of *Richardson* [1981].

1. Introduction

Crop yields and hydrological processes such as runoff and erosion are driven by weather variations. Recognizing the inherent variability in climate, it is often desirable to assess management scenarios for a number of likely weather sequences. Stochastic models are useful for simulating scenarios that are representative of the data. While there is a substantial literature for rainfall simulation and for other variables one at a time, only a few multivariate weather simulation models have been developed.

An objective of the work presented here was to generate daily weather sequences as inputs to the Weather Erosion Prediction Project (WEPP) of the U.S. Department of Agriculture (USDA). Six variables (solar radiation (SRAD), maximum temperature (TMX), minimum temperature (TMN), average wind speed (WSPD), average dew point temperature (DPT), and precipitation (P)) that are of interest to WEPP were considered to represent the daily weather state. Generally, a statistical method for generating daily weather sequences needs to consider the statistical dependence or correlation of the weather variables with each other on the same day, as well as their "persistence," i.e., dependence on the weather state on previous days. Solar radiation, dew point temperature, and maximum temperature are likely to be lower on rainy days than on dry days, while the wind speed and minimum temperature may be higher on rainy days than on dry days. Consequently, precipitation is chosen as the driving variable in a number of existing models. Typically [see *Jones et al*., 1972; *Nicks and Harp*, 1980; *Richardson*, 1981; *Rajagopalan et al*., 1997], daily precipitation is generated independently, and the other variables are generated by conditioning on precipitation events (i.e., whether a day is wet or dry). A precipitation occurrence and amount model (e.g., a two-state Markov model, with exponentially distributed rainfall amounts) is used to generate the sequence of dry and wet days and precipitation amount. The other variables are simulated using a lag 1 multivariate, autoregressive model with exogenous precipitation input (MAR-1). The work of *Rajagopalan et al*. [1997] differed from the earlier work. They used kernel density estimation to specify the univariate and multivariate probability densities needed for describing the stochastic processes of interest. Precipitation was generated independently from a nonparametric wet/dry spell model [*Lall et al*., 1996], and the other variables on a given day were generated by conditioning on the precipitation magnitude (rather than just the precipitation state) for the day and on the previous day's values for the weather variables.

The precipitation amount on a rainy day may also depend on the wind, the temperature, and the humidity as measured by the dew point temperature. Consequently, there is reason to consider dependence of the daily weather process on more than just precipitation as has traditionally been done. *Young* [1994], in a model similar in spirit to the one presented here, considers such dependence. In the approach adopted in this paper, precipitation is simulated along with the other variables, thereby capturing the mutual dependence of all six weather variables. The simulation strategy used is a direct resampling of the data using a conditional bootstrap based on nearest-neighbor probability density estimation. This approach does not require the specification of and estimation of the parameters of a parametric model (e.g., normal or lognormal) for the joint or conditional probability density of the variables.

A brief review of traditional methods for simulating weather variables is first provided. The general framework for the resampling strategy proposed here is presented next. The k–nearest-neighbor (k-NN) bootstrap algorithm is outlined. An application of the method to data from Salt Lake City is

Paper number 1999WR900028.
0043-1397/99/1999WR900028$09.00

then presented. Comparisons of the simulations from the k-NN bootstrap and from a more traditional autoregressive simulation model are provided.

2. Background

The general structure of some traditional methods [see *Jones et al*., 1972; *Bruhn et al*., 1980; *Nicks and Harp*, 1980; *Lane and Nearing*, 1989; *Richardson*, 1981] for simulating daily weather is discussed in this section. Precipitation is first generated independently, and the other variables are conditioned on the generated state of precipitation (i.e., rain or no rain on the day). The other variables are generated either from independent statistical distributions fitted separately to each of the variables for each of the two precipitation states (i.e., rain, no rain) or from independently or jointly fitted autoregressive models of order 1 (AR-1).

Usually, the year is divided into periods (seasons), and moments (mean, standard deviation, and skew) are calculated for each variable for each period for each precipitation state. The seasonal moments are used to fit probability distributions or models. Homogeneity of the process in each season is assumed. *Jones et al*. [1972], *Bruhn et al*. [1980], *Nicks and Harp* [1980], and *Lane and Nearing* [1989] divide the year into 14-day or 1-month periods. *Richardson* [1981] smoothed the means and standard deviations of each period and each precipitation state using Fourier series. The smoothed daily values of the means and standard deviations are subsequently used for deseasonalization.

Daily precipitation occurrence in these models is presumed to follow a first-order Markov chain with the daily precipitation amount generated from an assumed probability distribution (such as gamma, exponential, truncated normal, etc.) fitted to the historical daily amounts for each period. One approach to generate the other variables is to fit distributions independently for each variable for each period and for each precipitation state, under the assumption that each variable is conditionally independent and identically distributed (i.i.d.). This approach and its variants are used by *Jones et al*. [1972], *Bruhn et al*. [1980], and *Lane and Nearing* [1989]. In Lane and Nearing's model CLIGEN each variable is assumed to be an independent Gaussian variable for each month, with parameters dependent on the precipitation state transition (e.g., wet to wet, dry to wet, etc.). This approach does not consider the dependence between the variables and the serial dependence for each variable.

Nicks and Harp [1980] considered serial dependence of weather variables. They fit autoregressive models of order 1 (AR-1) independently to each variable for each period. *Richardson* [1981], who used a multivariate autoregressive model of order 1 (MAR-1), added the consideration of dependence across variables. These models suffer from the drawback of assuming the data to be normally distributed. As a result, only linear dependence between variables and precipitation states from one day to the next can be reproduced.

These approaches have four main drawbacks. First, since precipitation is exogenously provided, lag 0 and lag 1 correlations of the variables are often not properly reproduced. Second, the choice of a probability distribution function is often subjective and is rarely formally tested on a site-by-site basis. Third, there is reliance on an implicit Gaussian framework (e.g., AR or MAR) which preserves only linear dependence and poses problems for bounded variables. Fourth, the fitted models have limited portability in the sense that procedures/distributions used at one site may not be best at other sites. Transformations of variables can be used to justify the Gaussian AR or MAR framework. However, it is difficult to develop appropriate transformations in the setting considered here and preserve the proper statistical relationships in the untransformed space. All six of the variables considered here are in some sense bounded.

Katz [1996] observes that the Richardson model (1) does not preserve the lag 1 autocorrelation of the weather variables that are conditioned on precipitation amount, (2) underestimates the observed variance of monthly values of the weather variables, and (3) because of its conditional form (conditioning on precipitation state), leads to effects unanticipated by the user, as model parameters are varied. He notes that these problems are endemic to this class of models and provides ways by which the unconditional distributions of the weather variables in such a model can be derived and examined. The model of *Rajagopalan et al*. [1997] circumvents some of these problems; since the nonparametric density estimation does not require the transformation of the variables, wet and dry spell statistics are explicitly preserved, and nonlinear relations between the variables are approximated. However, it does not address the problems introduced by having an exogenous precipitation simulator. The kernel density estimation procedures also do not adapt the degree of density smoothing to the state space as well as the k-NN density estimates employed here.

A multivariate chain model for simulating daily minimum and maximum temperatures and precipitation was presented by *Young* [1994]. This model is similar to the model presented here in that a k-NN strategy is employed to select a day at random from the historical data set as a simulation for the three variables for the next day. Young uses multiple discriminant analysis to identify patterns in the three-dimensional data. The k nearest neighbors of the current day in terms of these patterns are identified, one of them is randomly selected, and its "next" day's values are adopted as the simulation for the current day's successor. Seasonal variations are not considered, and the number of nearest neighbors is selected by comparing the autocorrelograms of the simulated variables with those of the corresponding historical variables. The number of nearest neighbors selected (three to five) by this criterion is quite small. Young demonstrates the superiority of the approach over a first-order Markov chain model for the three variables in terms of a variety of statistics. His model preserves most notably the cross correlation between temperature and precipitation and the wet/dry spell statistics. He also notes some biases (e.g., reduced persistence and underestimation of the fraction of dry months) in the sequences simulated by his method. The work presented here is philosophically similar to the model of Young, but it differs in operational details. A connection to the Markov process, nonparametric density estimation, and nonlinear dynamical systems literature is also provided.

All the techniques discussed in this section focused on "short-range" statistical properties. It is known that such models will not likely reproduce the variance and related statistical attributes at longer aggregation periods (e.g., the interannual variance and dependence of seasonal precipitation). The model presented in this paper does not explicitly address this concern either.

Figures 1 and 2 show the pairwise scatterplot of the six variables for wet and dry days, respectively, for season 1 (Janu-

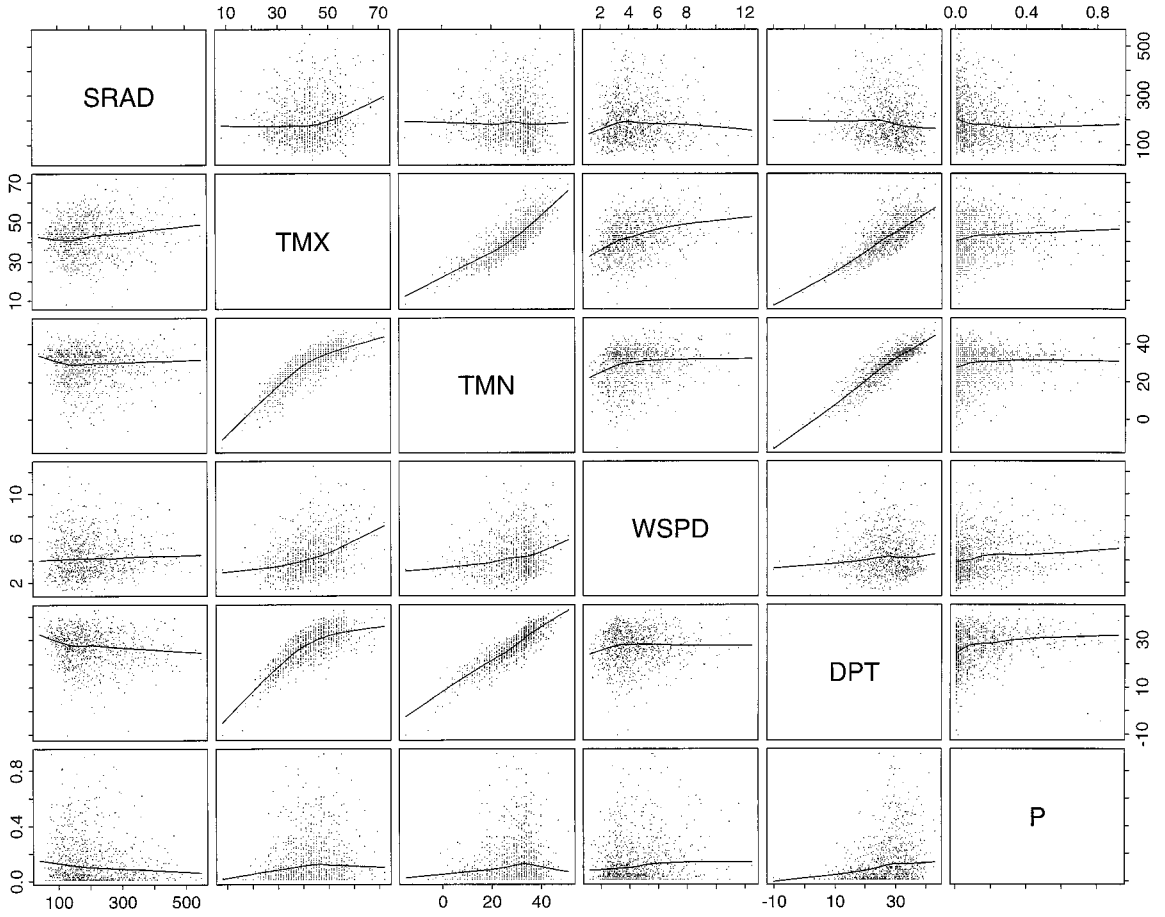

Figure 1. Pairwise scatterplot of SRAD, TMX, TMN, WSPD, DPT, and P for wet days, for season 1 at Salt Lake City. The lines in each section are the locally weighted scatterplot smoother (LOWESS) smooths.

ary–March) of the 1961–1991 data from Salt Lake City. The line in each scatterplot is a locally weighted scatterplot smooth (LOWESS: a moving-window–weighted local regression from *Cleveland* [1979]). We observe that the pairwise relationships between the variables can (1) be nonlinear and (2) differ for wet and dry days. There is also evidence (bottom row of Figure 1) for the dependence of the precipitation amount on some of the other variables (notably dew point temperature). This indicates that a strategy that directly includes precipitation in the set to be simulated may be better than one in which precipitation is generated exogenously to the other variables. Heteroskedasticity (nonconstant variance of errors from the smooth in each frame) is also observed. Transforms of individual variables are often used to develop cross-dependence relations that are approximately linear with relatively uniform scatter about the regression line. Given the varying "curvature" of the mean response and scatter in the pairwise relationships, it is not obvious that a useful set of univariate transformations that can address the multivariate dependence is feasible. The likely utility of a scheme that recognizes these factors and approximates the behavior locally in some sense is obvious.

3. Multivariate Markov Model and Bootstrap

Let us denote the time series of length n of the daily values of the six variables by $\mathbf{x}_t$, $t = 1, \cdots, n$. For now, assume that seasonality has been taken care of in some fashion and we are interested in resampling daily values $\mathbf{x}_t$, focusing only on dependence on m past values, i.e., $\mathbf{x}_{t-1}$, $\mathbf{x}_{t-2}$, $\cdots$, $\mathbf{x}_{t-m}$. The process $\mathbf{x}_t$ is thus considered to be a m-dependent multivariate Markov process. Synthetic sequences from such a model can be simulated if we specify the conditional distribution function $F(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_{t-2}, \cdots, \mathbf{x}_{t-m})$. The models discussed in section 2 belong to this general framework, with $m = 1$ and with the conditional distribution function $F(\mathbf{x}_t|\mathbf{x}_{t-1})$ described using parametric functions (Gaussian distributions for all variables except precipitation). The primary difference in this paper is that we implicitly use a nonparametric density estimate to resample from $F(\mathbf{x}_t|\mathbf{x}_{t-1})$.

The bootstrap [*Efron*, 1979] is a technique that prescribes a data-resampling strategy using the random mechanism that generated the data. Its applications for estimating confidence intervals and parameter uncertainty are well known [see *Härdle and Bowman*, 1988; *Tasker*, 1987; *Woo*, 1989; *Zucchini and Adamson*, 1989]. Usually, the bootstrap resamples with replacement from the empirical distribution function $F_n(\mathbf{x})$ of independent, identically distributed data, $\mathbf{x}_i$, $i = 1, \cdots, n$. This is equivalent to resampling the observations $\mathbf{x}_i$ with a probability of $1/n$. An algorithm for bootstrapping time series considering Markovian dependence was developed by *Lall and Sharma* [1996], who applied it to univariate, monthly streamflow data. This algorithm was motivated by nonparametric approaches to time series analysis using nearest-neighbor den-

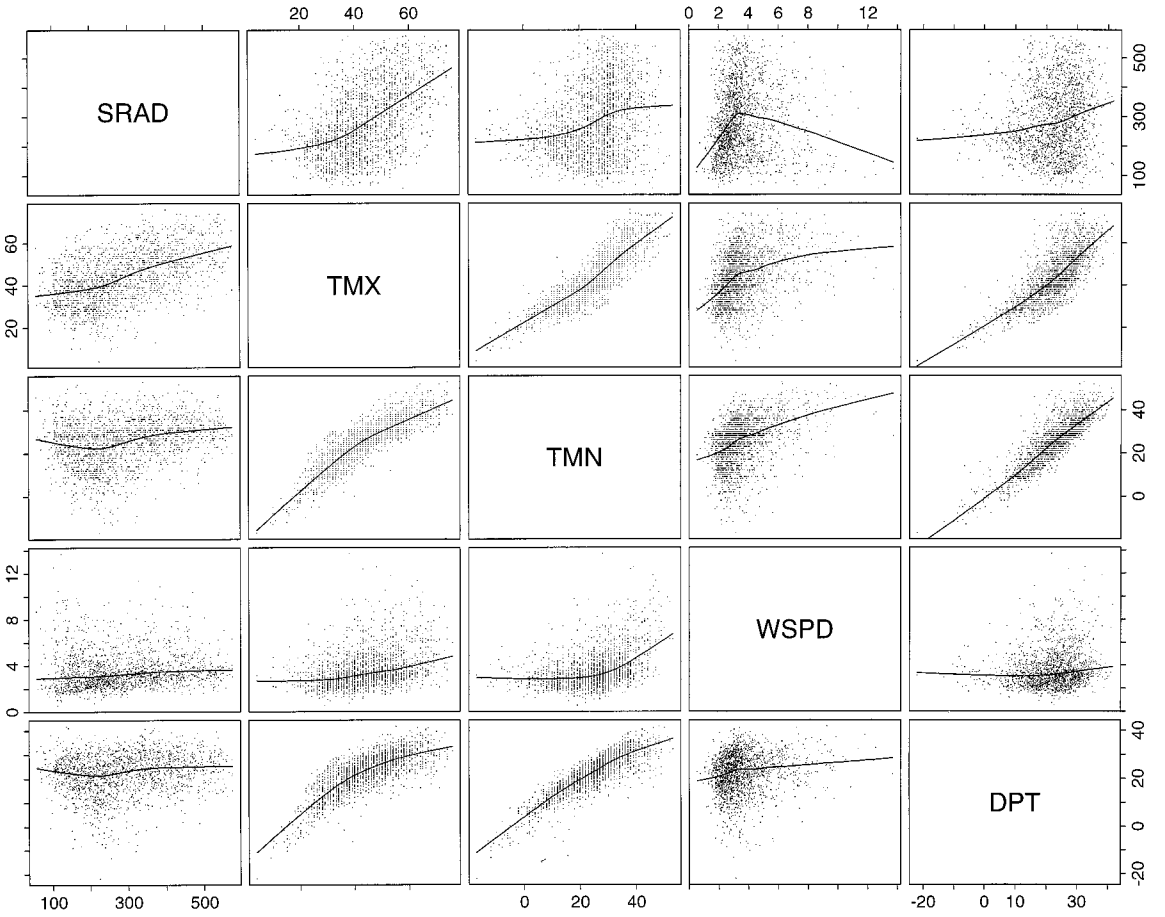

Figure 2. Pairwise scatterplot of SRAD, TMX, TMN, WSPD, and DPT for dry days for season 1 at Salt Lake City. The lines in each section are the LOWESS smooths.

sity and regression estimators of *Yakowitz* [1973, 1979, 1985, 1993]. We shall briefly motivate this algorithm in the context of the present work.

The Markov chain model for precipitation occurrence usually considers two states (wet and dry) and transition probabilities p_{ij} for transitions from state i to state j in the next time period. This is a nonparametric model, with an intuitively appealing structure. It has been noted [*Lall et al*., 1996; *Rajagopalan et al*., 1996] that it may be desirable to have more than two states in such models to recognize the role of precipitation magnitude. Increasing the number of states can provide a better stepwise approximation to the conditional distribution function $F(P_t|P_{t-1})$ of the associated Markov process for rainfall.

One can extend this thinking to the other five variables as well. Let us say that we partition each of these variables into p states and consider a Markov chain model for all the variables. For the multivariate problem in six variables, there are a total of p^6 states at each time step. Thus even for the rather coarse description of the process for $p = 2$ one needs to compute transition probabilities from 64 states to 64 states at the next time step. Clearly, the sample sizes needed to reliably estimate transition probabilities under this framework would be very large. As the number of states considered increases, the situation becomes rapidly intractable ($p = 5$ yields 15,625 states, and $p = 10$ gives 10^6 states). This is the well-known curse of dimensionality. Conceptually, we shall retain the nonparametric flavor of the Markov chain approach, but we shall strive to approximate the conditional distribution function $F(\mathbf{x}_t|\mathbf{x}_{t-1})$ in a more adaptive manner using nearest-neighbor density estimators.

We motivate this idea through Figure 3, where we show a plot between successive values for a synthetic, univariate time series. Note that while the correlation between $\mathbf{x}_t$ and $\mathbf{x}_{t-1}$ is zero, $\mathbf{x}_t$ depends directly on $\mathbf{x}_{t-1}$, with no random terms. Four states equally spaced between 0 and 1 for a Markov chain representation are considered. Consider resampling an $\mathbf{x}_t$, given that $\mathbf{x}_{t-1}$ corresponds to the whisker in the window marked as A. If we had observed this value of $\mathbf{x}_{t-1}$ several times, we could directly apply the bootstrap and resample directly from the successors (i.e., $\mathbf{x}_t$ values corresponding to each such occurrence) to these observations. Since we do not have such information, assuming that the conditional distribution function $F(\mathbf{x}_t|\mathbf{x}_{t-1})$ is smooth (i.e., differentiable with bounded derivatives) in a neighborhood of the point of interest, we can "borrow" the successors of neighboring values of $\mathbf{x}_{t-1}$ for the purpose. The windows A and B were based on 10 neighbors of the marked point. We can see that these moving windows are quite effective in capturing the local attributes of the transitions from $\mathbf{x}_{t-1}$ to $\mathbf{x}_t$. For the situation corresponding to window A, if we had used the four-state Markov chain model, all we would know is that $0.75 < \mathbf{x}_t < 1$ with probability 1 for all values in the range $0.25 < \mathbf{x}_{t-1} < 0.5$. Asymptotically, i.e., as the sample size tends to infinity, the size of the neigh-

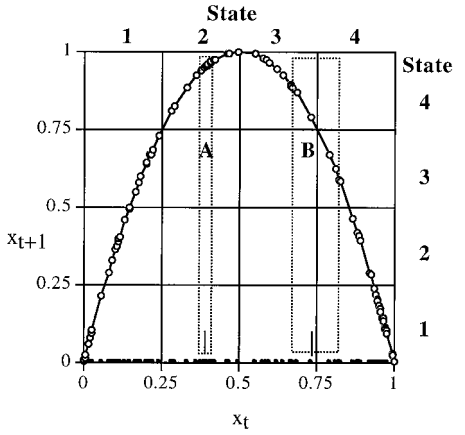

Figure 3. A plot of $\mathbf{x}_{t+1}$ versus $\mathbf{x}_t$ for the time series generated from the model $\mathbf{x}_{t+1} = [1 - 4(\mathbf{x}_t - 0.5)^2]$. The state space for $\mathbf{x}$ is discretized into four states as shown. Also shown are windows A and B with whiskers located over selected values of $\mathbf{x}_t$. These windows represent a k nearest neighborhood of the corresponding $\mathbf{x}_t$. In general, these windows will not be symmetric about the $\mathbf{x}_t$ of interest, and their width varies depending on the relative sampling density of $\mathbf{x}_t$. Note how one can think of state transition probabilities using these windows in much the same way as with the multistate Markov chain. However, the nearest-neighbor windows point directly to the region in which transitions are possible. A value of $\mathbf{x}_{t+1}$ conditional to point A or B can be bootstrapped by appropriately sampling and replacing one of the values of $\mathbf{x}_{t+1}$ that falls in the corresponding window. (From *Lall and Sharma* [1996].)

borhood dictated by a given number of neighbors k will shrink, and the approximation of the underlying conditional distribution function will improve.

In the multivariate setting, neighbors of the conditioning point correspond to data patterns that are similar to the pattern at the conditioning point. For a day with no rain, that is warm, with little wind, and no humidity, neighbors established by calculating the vector distance between the observations will be similar days. The values for the weather variables for the next day will be sampled as a vector from a historically similar day. Clearly, there is some utility to giving a higher probability to a day that is more similar to the conditioning day than the other "neighbors." Using a weight function that decays smoothly with distance can reduce the sensitivity to the number of nearest neighbors used for resampling. A weight function applied to the nearest neighbors that is natural in a certain sense and the choice of the number of nearest neighbors to use are discussed in some detail by *Lall and Sharma* [1996].

4. The k-NN Resampling Algorithm

The k-NN conditional resampling scheme is described in this section. All six daily weather variables (including precipitation) are considered simultaneously as members of a daily weather vector. Denote the vector time series of weather variables by $\mathbf{x}_t$, $t = 1, \cdots, n$, and assume for now that we have decided on a dependence structure, i.e., which and how many lags the future values will depend on and the number of nearest neighbors k to use. We shall call this conditioning set a "feature vector" and the simulated or forecasted vector the "successor." The strategy is to find the historical nearest neighbors of the current feature vector and to resample from their successors. Rather than resampling uniformly from the k successors, we use a discrete resampling kernel that is monotonically decreasing, is data adaptive, adapts automatically to the dimension of the feature vector and to boundaries of the sample space, and has an attractive probabilistic interpretation consistent with the nearest-neighbor method. Also presume for now that the data have been deseasonalized or that a treatment for seasonality is available that does not affect the algorithm presented in section 4.1. We deseasonalize the time series of each of the variables by removing the calendar day's mean and dividing by the calendar day's standard deviation computed over the historical record. The $\mathbf{x}_t$ referred to are deseasonalized variates. The final results presented are obtained by multiplying the daily values generated by the standard deviation for that date and by adding the mean for that date.

We now present an annotated algorithm for resampling weather variables adopted here that considers day-to-day dependence between the variables. This algorithm is applied for a given season (e.g., 3 months, 1 month) and is initialized by the $\mathbf{x}_t$ values for the last day of the previous season.

4.1. Flow Chart for Resampling

The key steps in the algorithm are (1) identifying a current conditioning vector of the six weather variables, (2) determining its k nearest neighbors in state space, (3) identifying, for each of these k nearest neighbors, a successor vector comprising the next day's values for the six variables, (4) resampling one of these vectors to represent the next day's weather using a kernel or weight function, and (5) repeating this process.

1. Define the composition of the feature vector $\mathbf{D}_t$ of dimension d.

$$\mathbf{D}_t : \mathbf{x}_{t-1}$$

Here we have chosen to use the vector of the (six) deseasonalized variables of interest on the previous day as the feature vector. One could add, if desired, other information, such as the value of an atmospheric flow index (e.g., the Southern Oscillation Index) on the same day or averaged over the past month and/or additional lags (e.g., $\mathbf{D}_t : [\mathbf{x}_{t-1}, \mathbf{x}_{t-1}, \cdots, \mathbf{x}_{t-L}]$ where L is the number of terms in the model). *Katz and Parlange* [1995] fit stochastic models for daily precipitation conditional on a monthly index of large-scale atmospheric circulation.

2. Denote the current feature vector as $\mathbf{D}_i$ and determine its k nearest neighbors among the historical state vectors $\mathbf{D}_m$ using the weighted Euclidean distance

$$r_{im} = \sqrt{\left[\sum_{j=1}^{d} w_j (v_{ij} - v_{mj})^2 \right]} \tag{1}$$

where $v_{(\ )j}$ is the jth component of $\mathbf{D}_{(\ )}$ and the w_j are weights. Here we chose the weights w_j as "scaling" weights (e.g., $1/s_j$), where s_j is some measure of scale such as the standard deviation or range of v_j. The weighted euclidean distance may also be computed as $(r_{im}^2 = (\mathbf{v}_i - \mathbf{v}_m)^{\mathrm{T}} \Sigma^{-1} (\mathbf{v}_i - \mathbf{v}_m))$, where Σ is the covariance matrix of $\mathbf{D}$ and $\mathbf{v}_i$ and $\mathbf{v}_m$ represent the values of $\mathbf{D}$ at points i and m. The weights w_j may thus be specified a priori, as is done here, or they may be chosen to provide the best forecast for a particular successor in a least squares sense [see *Yakowitz and Karlsson*, 1987]. The latter would be the desirable method, but it adds substantially to the

computational burden. Multiple discriminant analysis as used by *Young* [1994] would be another choice for neighbor identification.

3. Denote the ordered set of nearest-neighbor indices by $J_{i,k}$. An element $j(i)$ of this set records the time t associated with the jth closest $\mathbf{D}_m$ to $\mathbf{D}_i$. Denote $\mathbf{x}^s_{i,j}$ as the successor to $\mathbf{D}_{j(i)}$. If the data are highly quantized, it is possible that a number of observations may be the same distance from the conditioning point. The resampling kernel defined in step 4 is based on the order of elements in $J_{i,k}$. Where a number of observations are the same distance away, the original ordering of the data can impact the ordering in $J_{i,k}$. To avoid such artifacts, we copy the time indices t into a temporary array that is randomly permuted prior to distance calculations and creation of the list $J_{i,k}$.

4. Define a discrete kernel $K[j(i)]$ for resampling one of the $\mathbf{x}^s_{i,j}$ as follows:

$$K[j(i)] = \frac{1/j}{\sum_{j=1}^{k} 1/j} \tag{2}$$

where $K[j(i)]$ is the probability with which $\mathbf{x}^s_{i,j}$ is resampled. Note that this resampling kernel is the same for any i and can be computed and stored prior to the start of the simulation. *Lall and Sharma* [1996] develop this kernel through a local Poisson approximation of the probability density function of state space neighbors.

5. Using the discrete probability mass function $K[j(i)]$, resample a $\mathbf{x}^s_{i,j}$, update the current feature vector, and proceed to step 2 if additional simulated values are needed.

4.2. Choosing the Number of Neighbors k and Model Order L

The user-selectable parameters of the k-NN daily weather simulator are the number of nearest neighbors k used for conditioning and the number of terms L in the model. One could also use other conditioning variables (e.g., climate state). Traditional time series simulation models often use criteria that evaluate the mean square forecast error (corrected for the degrees of freedom of the model) to choose model parameters. *Lall and Sharma* [1996] suggest the use of the generalized cross validation (GCV) score function to choose k and the number of lags L for the Markov model (set to 1 in the applications reported here). This is similar to the use of the Akaike information criteria (AIC) in the traditional autoregressive moving average (ARMA) framework. In our context the GCV score function is given as

$$\mathrm{GCV} = \frac{\sum_{i=1}^{n_1} \mathbf{e}_i^{\mathrm{T}} \mathbf{W} \mathbf{e}_i}{n_1 \left\{ 1 - \left[1 \Big/ \sum_{j=1}^{k} (1/j) \right] \right\}^2} \tag{3}$$

where n_1 is the total number of forecasts possible ($n - L$ here) with sample size n, $\mathbf{W}$ is a weight matrix, and $\mathbf{e}_i$ is the error vector of the ith vector $\mathbf{x}_i$, defined as

$$\mathbf{e}_i = \mathbf{x}_i - \mathbf{x}_i^f \tag{4}$$

$$\mathbf{x}_i^f = \sum_{j=1}^{k} \frac{1/j}{\sum_{m=1}^{k} 1/m} \mathbf{x}^s_{i,j} \tag{5}$$

where $\mathbf{x}_i$ is a recorded value that is to be forecast from L prior lags $(\mathbf{x}_{i-1}, \mathbf{x}_{i-2}, \cdots, \mathbf{x}_{i-L})$, $\mathbf{x}^s_{i,j}$ is the successor vector corresponding to the jth nearest neighbor of the feature vector on which we are conditioning to resample a vector corresponding to $\mathbf{x}_i$, and $\mathbf{x}_i^f$ is the k-NN "forecast" vector corresponding to $\mathbf{x}_i$.

The GCV score above is a measure of the expected predictive mean square error for the k-NN forecasts of the six variables of interest on a given day. The forecasts are formed as the weighted average of the successors of the k nearest neighbors of the feature vector. The weight matrix $\mathbf{W}$ can be specified a priori to recognize the relative importance the user wishes to assign to each of the six variables. One choice of the weight matrix is a diagonal matrix with the scaling weights (i.e., $w_{jj} = 1/s_j$, other $w_{ij} = 0$). The number of nearest neighbors k to use and the order L of the model can be chosen as the values that minimize the GCV score over the data.

For parametric models a maximum likelihood or method of moments estimation of the parameters "ensures" that the relative frequency distribution of the data and the simulated sequences will match. The selection of the model parameterization using GCV- or AIC-like criteria is appropriate if the model errors are normally distributed and the conditional mean of the state variables is of primary interest. Departures between the frequency distributions of the historical data and the simulated sequences are presumed to be due to model misspecification. They may be addressed by reexamining the probability distribution models and/or the model order used. In the nonparametric framework employed here, different values of k will lead to a different bias or variance associated with the approximation of the underlying frequency distributions, and changing the model order will affect the persistence statistics. Thus one should be able to control how well the frequency distributions of state variables generated match those of the historical data through an appropriate choice of k. In this regard, a GCV-based choice of k and L may be suboptimal, since it only considers the performance of the model with respect to the conditional mean and variance. In practice, the user may wish to experiment with the choice of k around the value selected by GCV to "tune" the sequences generated so that a broad range of statistics is matched. *Young* [1994] based his choice on the match between the autocorrelation function of simulated and historical variables.

With regard to the model order L it is interesting to note that for dynamical systems, Takens embedding theorem suggests that a Markov model with a d-dimensional state space can be shown to be equivalent to a univariate Markov model of order $(d + 1)$ in terms of any one of the variables. The reader is referred to *Sangoyomi et al.* [1996] and *Abarbanel and Lall* [1996] for a discussion of this point. Thus, even with a lag 1 model, with the six state variables assumed to be interdependent, one may be able to reproduce persistence in precipitation and other variables without recourse to additional lags.

Here we considered only a lag 1 model and for computational ease used a prescriptive choice for k of $\sqrt{n}$. Asymptotic arguments [*Fukunaga*, 1990] suggest that k should be chosen so as to be proportional to $n^{4/(d+4)}$, where d is the dimension of the vector for which the nearest-neighbor density is to be

estimated, with the constant of proportionality dependent on the underlying density. For the sample sizes under consideration here the choice of $\sqrt{n}$ was found to give good results for the simulated statistics of **x**.

5. Model Application and Performance Measures

The k-NN simulator was applied to daily weather data from Salt Lake City, Utah. Thirty years of daily weather data were available from the period 1961–1991. Salt Lake City is at 40°46′N latitude, 111°58′W longitude and at an elevation of 1288 m. Most of the precipitation comes in the form of winter snow. Rainfall occurs mainly in spring, with some in fall.

The data for each of the six variables were first deseasonalized by subtracting the mean for the calendar day and dividing by the standard deviation of the variable for the calendar day. Recognizing that the dynamics of the weather system may have seasonal variation beyond that represented in the mean and variance, the simulator was applied on seasonally segregated data. Fixed or moving window seasons can be employed. Fixed seasons were used in the applications presented here to facilitate comparisons with the MAR models, which are usually used with fixed seasons. The year is divided into four periods or seasons (season 1 (January–March), season 2 (April–June), season 3 (July–September), and season 4 (October–December)). Simulations for days in any particular period are made using the historical data of that season. Comparisons of statistics are then made for each season. We have also used a moving window of some width (e.g., 90 days), centered on the calendar day of interest, rather than a fixed season demarcation. Comparable results are obtained by using such a moving window rather than fixed seasons. We shall outline first the experiment design and then some measures of performance used to judge the utility of the model.

5.1. Experiment Design

The algorithm described in section 4.1 is applied to the Salt Lake City data, and selected statistics of the simulated traces are compared with those from a MAR-1 model. The main steps are as follows: (1) The daily weather variables are generated following the simulation algorithm described in section 4. (2) Twenty-five synthetic records of 30 years each (i.e., the historical record length) are simulated using the k-NN model. (3) The statistics of interest described in section 5.2 are computed for each simulated record for each season and compared to statistics of the historical record using box plots.

5.2. Performance Measures

The following statistics were considered to be of interest in comparing the historical record and the simulated record of weather variables. None of these statistics is explicitly specified in fitting the k-NN model. Consequently, their successful reproduction can be considered a sign of success for the method. All computed statistics are for daily values of each variable and refer to each season. Moments are mean, standard deviation, skew, and coefficient of variation. Relative frequencies are 25th and 75th quantiles of the 30-year record and in some cases the largest or smallest values in a 30-year record. Dependence is pairwise lag 0 and lag 1 cross correlation across all variables.

6. Results

The statistics of interest calculated from the simulations are compared with those for the historical record using box plots. A box in the box plots (e.g., Figure 4) indicates the interquartile range of the statistic computed from 25 simulations, and the line in the middle of the box indicates the median simulated value. The solid lines correspond to the statistic of the historical record. The box plots show the range of variation in the statistics from the simulations and also show the capability of the simulations to reproduce historical statistics. Only selected results are shown here to save space. (The detailed results are available from the authors or on the Web from the Utah Water Research Laboratory, Utah State University at http://pub.uwrl.usu.edu/~ulall/knnweather.) In summary, the moments and relative frequency measures of SRAD, TMX, TMN, WSPD, DPT, and P are reproduced by the k-NN simulations with reasonable variety and without bias using the default choice of k.

Illustrative statistics of wet and dry spell lengths simulated are shown in Figures 4 and 5. Figure 4 provides the box plots of average wet spell length, standard deviation of wet spell length, fraction of wet days, and length of longest wet spell length for each season. Figure 5 shows the box plots of these statistics for the dry spell length. The box plots in Figures 4 and 5 show that the historical wet and dry spell statistics are well reproduced by the simulations even though they were not explicitly modeled. Note in Figure 5 that spells longer than those in the historical record can be generated by the k-NN resampling procedure applied at a daily time step.

Figures 6 and 7 show the box plots of the lag 0 cross correlation and lag 1 cross correlation between the variables, respectively. Figure 8 shows the lag 1 autocorrelation of each of the variables for all four seasons. As can be seen from Figures 6–8, the historical correlations are reproduced well by the simulations in all four seasons.

We compared the k-NN resampling approach with a slightly modified version of the MAR-1 model developed by *Richardson* [1981] that was discussed in section 2. Instead of adopting the Markov chain for generating the daily precipitation we use a nonparametric wet/dry spell model as developed by *Lall et al*. [1996] to generate the daily precipitation (both state and amount). Instead of the Fourier series smoothing of the 14-day period means and standard deviations of each variable, we calculate the wet and dry day means and standard deviations of each variable for each calendar day and use a discrete nonparametric smoother [see *Rajagopalan and Lall*, 1995] to smooth these variables. The time series of each of the variables is then reduced to residual elements by subtracting appropriate means and dividing by the appropriate standard deviations. By appropriate we mean the wet day or dry day means and standard deviations, depending upon the precipitation state. The residual elements are assumed to be serially uncorrelated and normally distributed. MAR-1 is then fit to these residuals as per procedures detailed by *Salas et al*. [1980]. A precipitation sequence is first generated from the nonparametric wet/dry spell model. A vector of residuals of the weather variables is simulated from the MAR-1 model fitted for the season. Depending on precipitation state of the day, the residuals are multiplied by the appropriate standard deviation and added to the appropriate mean to recover daily weather variable values.

The wet/dry spell precipitation model reproduced the statistics of precipitation amount and spell length as shown by *Lall et al*. [1996]. The mean values of SRAD, TMX, TMN, WSPD, and DPT were reproduced by the MAR model. However, the variance, skew, and quantiles were often biased (sometimes significantly) in the MAR simulations. This is due in part to the

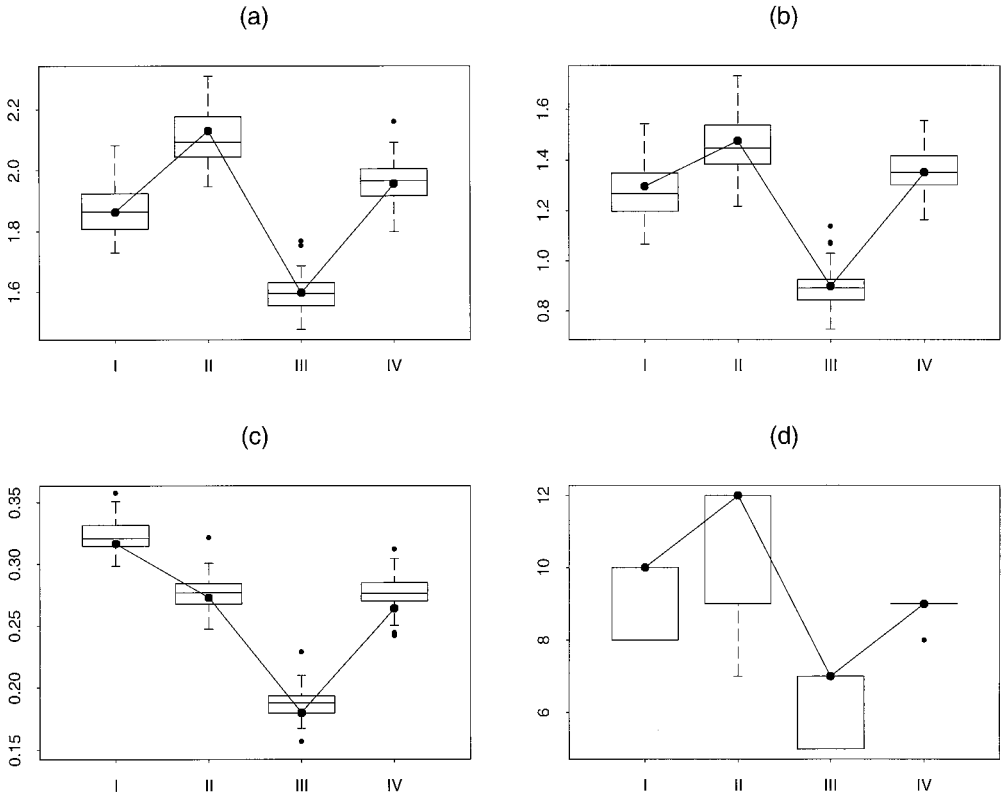

Figure 4. Box plots of statistics of wet spell length: (a) mean wet spell length, (b) standard deviation of wet spell length, (c) fraction of wet days, and (d) longest wet spell length from k-NN simulations along with the historical values for the four seasons. Roman numerals indicate the four seasons.

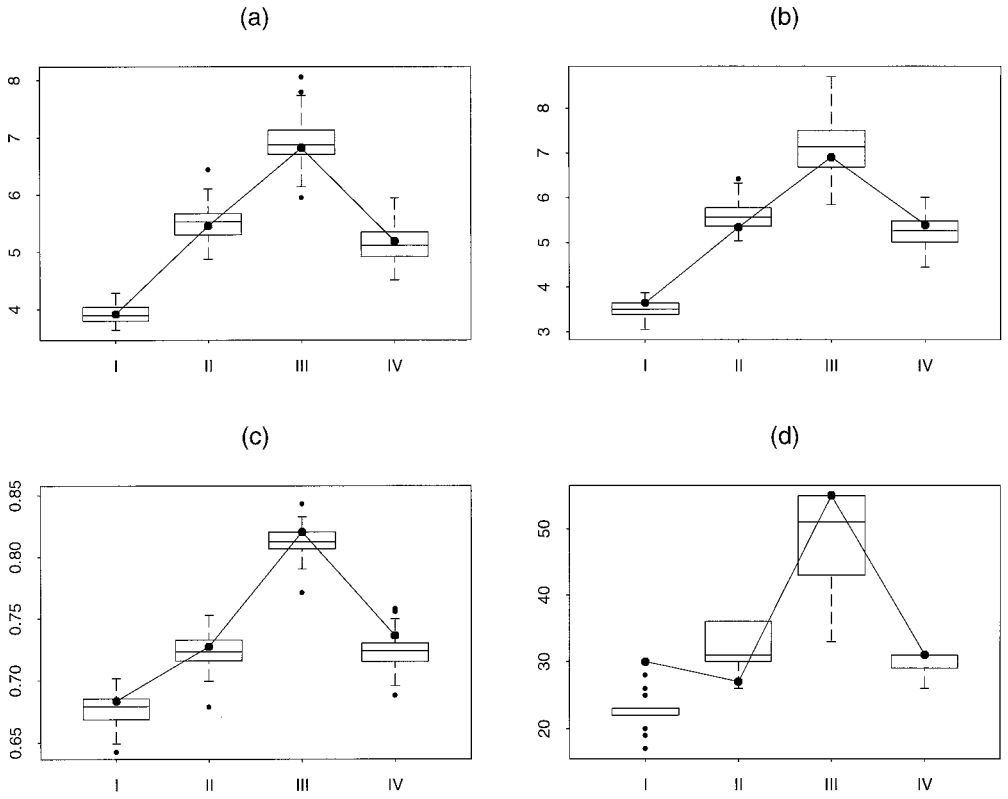

Figure 5. Box plots of statistics of dry spell length: (a) mean dry spell length, (b) standard deviation of dry spell length, (c) fraction of dry days, and (d) longest dry spell length from k-NN simulations along with the historical values for the four seasons.

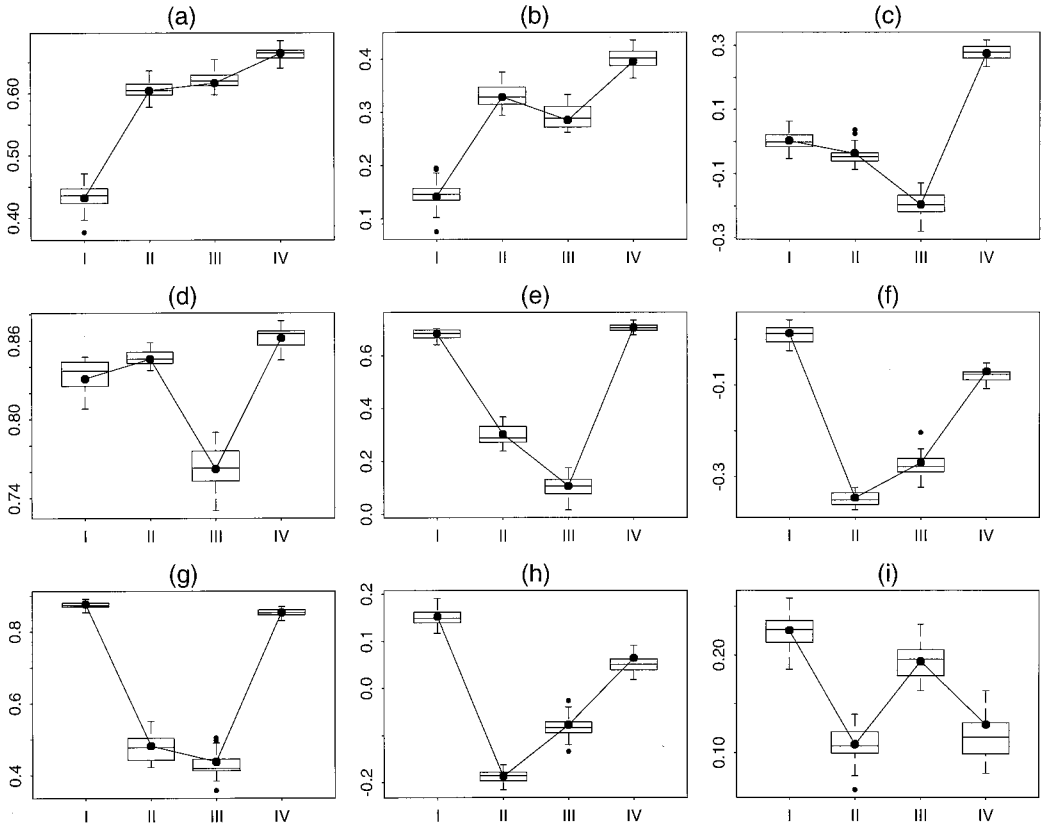

Figure 6. Box plots of lag 0 cross correlation between (a) SRAD and TMX, (b) SRAD and TMN, (c) SRAD and DPT, (d) TMX and TMN, (e) TMX and DPT, (f) TMX and P, (g) TMN and DPT, (h) TMN and P, and (i) DPT and P from k-NN simulations along with the historical values for the four seasons.

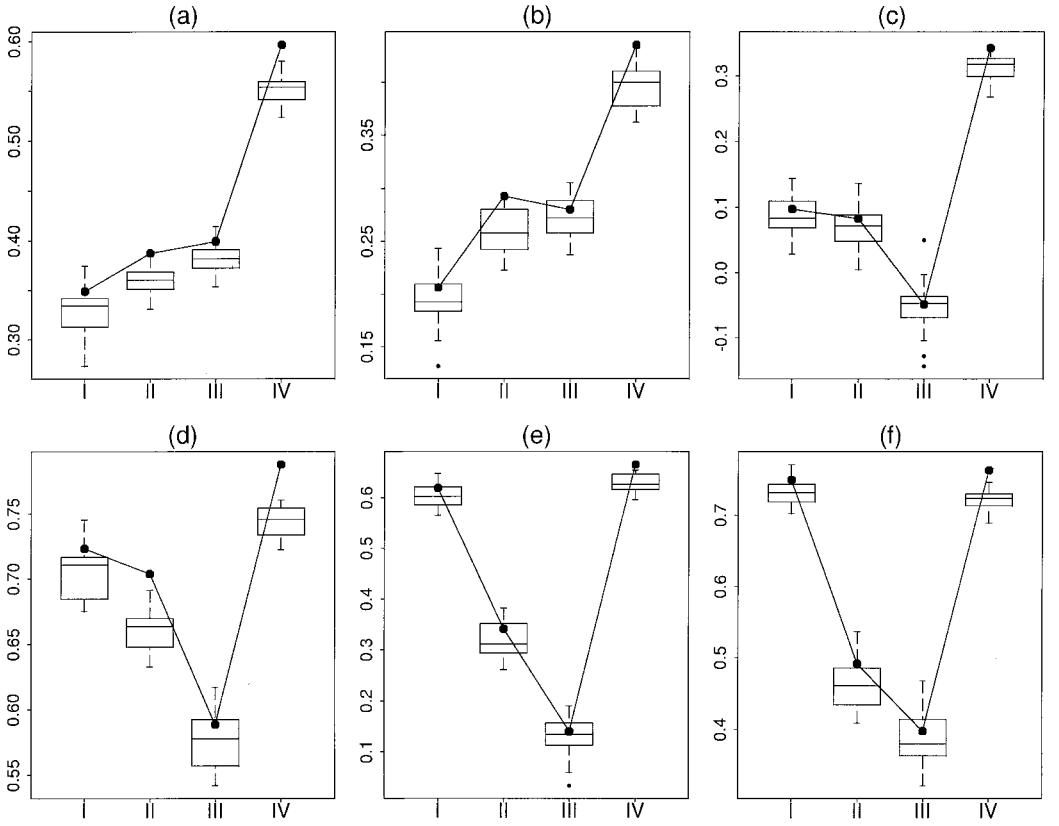

Figure 7. Box plots of lag 1 cross correlation between (a) SRAD and TMX, (b) SRAD and TMN, (c) SRAD and DPT, (d) TMX and TMN, (e) TMX and DPT, and (f) TMN and DPT from k-NN simulations along with the historical values for the four seasons.

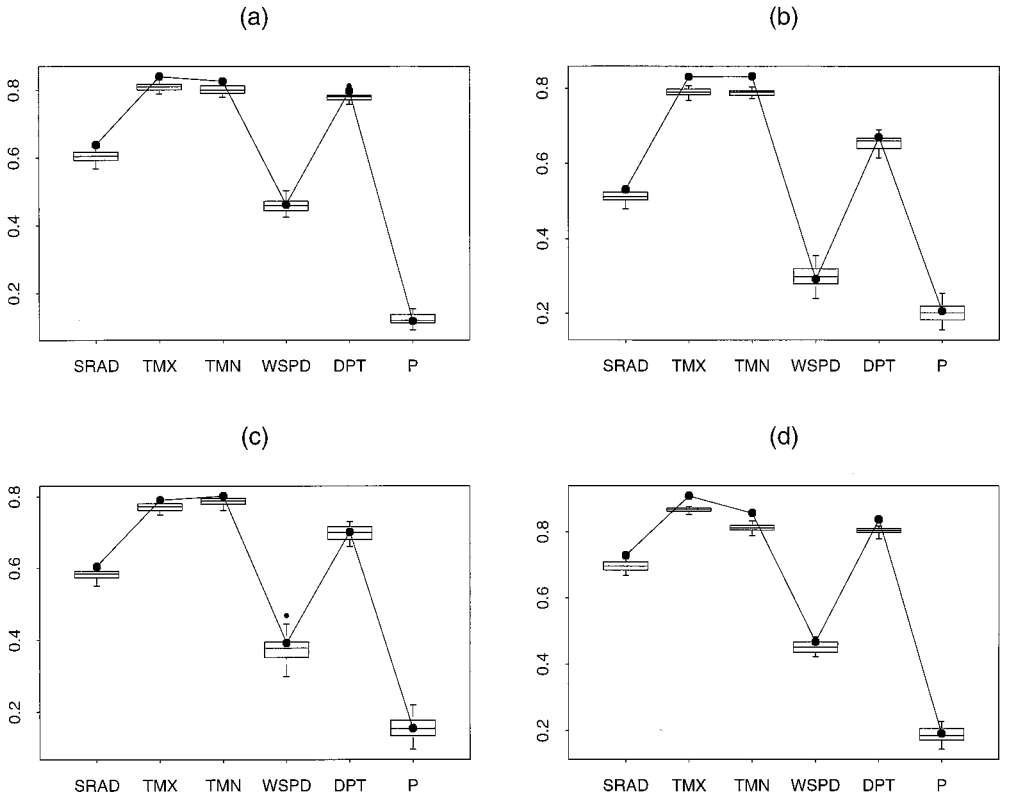

Figure 8. Box plots of lag 1 autocorrelation of SRAD, TMX, TMN, WSPD, and DPT for (a) season 1, (b) season 2, (c) season 3, and (d) season 4 from k-NN simulations along with the historical values.

behavior of the two stage models noted by *Katz* [1996] and in part due to the fact that the marginals were not transformed to Gaussian prior to simulation. To save space, the MAR results for these simulations are not presented since the main focus of our comparison across these models is the manner in which correlations are preserved.

Figures 9, 10, and 11 show the box plots of lag 0 cross correlation, lag 1 cross correlation, and lag 1 autocorrelation of the variables. A comparison of Figures 9–11 with the corresponding figures for the k-NN simulation (Figures 6–8) reveals the superiority of the k-NN simulator in preserving the cross-dependence and serial dependence terms. This observation, coupled with the better reproduction of the basic univariate statistics, and the relative simplicity of the algorithm suggest its utility for generating multivariate, daily weather sequences.

While the k-NN weather simulator was designed to reproduce only daily or short-term statistics, the statistics of seasonal totals appear to be reproduced effectively as well for the Salt Lake City data, as shown in Figure 12. The mean and variance of the annual total precipitation were also preserved (not shown here) with a slight downward bias in the variance. This indicates the potential capability of the k-NN simulator to capture interannual statistics and, consequently, low-frequency climate variability.

7. Summary and Conclusions

A multivariate k-NN resampling scheme with lag 1 dependence was illustrated for six daily weather variables. Its ability to successfully reproduce sample statistics was demonstrated. We see that the properties of the precipitation spell structure and amount are also preserved. The need for separate precipitation and weather models is thus obviated. We feel that this approach recognizes the mutual dependence between the six weather variables better than past approaches.

A Markovian interpretation of the k-NN model described here is apparent upon thinking about the manner in which the one-step transition process works. The value to be simulated at the next time step can be thought of as a transition to any of the states within a neighborhood of the state of the current time. The conditional probability density function (pdf) can be viewed as an approximation to the transition probabilities. Thus the k-NN model implemented here can be seen as a one-step Markov model with the transitions estimated nonparametrically. With a few exceptions [*Young*, 1994] the model presented here represents a philosophical as well as practical departure from the methods used for daily weather simulation.

The k-NN approach presented improves on the kernel-based nonparametric simulation approach developed by *Rajagopalan et al*. [1997]. In that model, precipitation was generated exogenously using a nonparametric wet/dry spell model, and the other variables were conditioned on precipitation on the day and on the other variables on the prior day. The 11-dimensional (six variables on the current day and five variables on the preceding day) joint probability density needed for simulation was estimated using a multivariate Gaussian kernel function and was used with a bandwidth chosen appropriate for estimating a multivariate Gaussian density. The simulations from that model failed to adequately reproduce the lag 0 and lag 1 correlations between the variables, especially with pre-

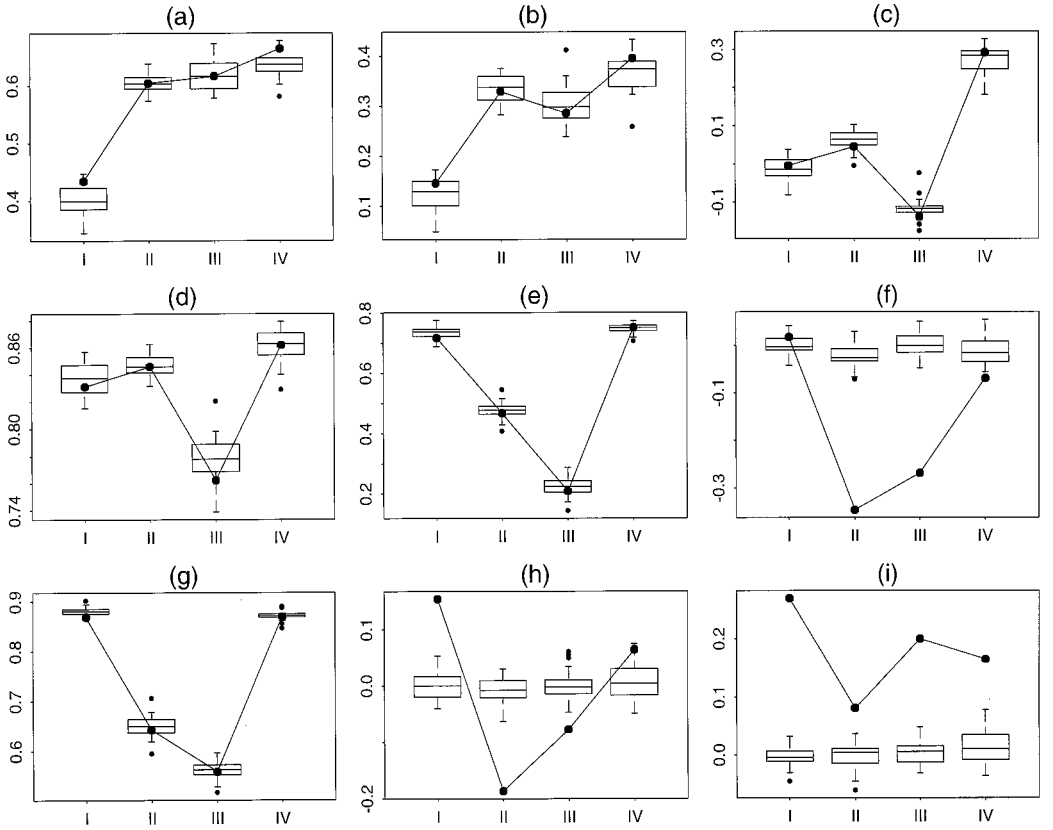

Figure 9. Box plots of lag 0 cross correlation between (a) SRAD and TMX, (b) SRAD and TMN, (c) SRAD and DPT, (d) TMX and TMN, (e) TMX and DPT, (f) TMX and *P*, (g) TMN and DPT, (h) TMN and *P*, and (i) DPT and *P*, from MAR-1 simulations along with the historical values for the four seasons.

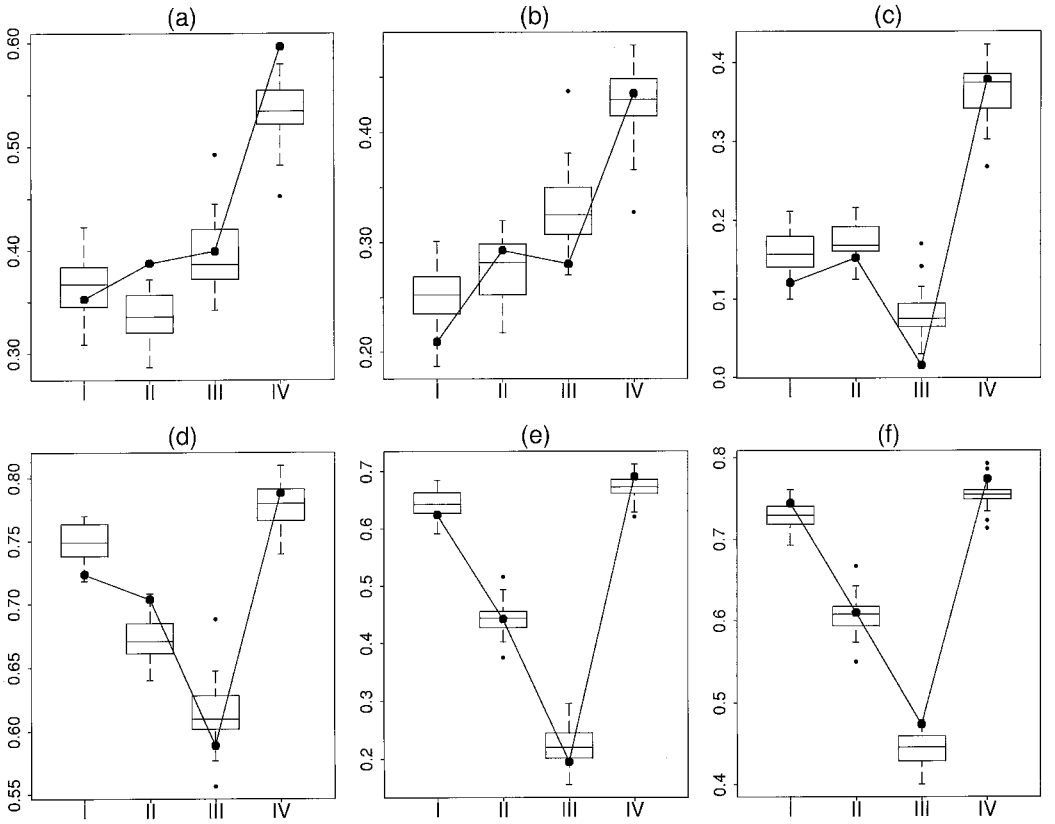

Figure 10. Box plots of lag 1 cross correlation between (a) SRAD and TMX, (b) SRAD and TMN, (c) SRAD and DPT, (d) TMX and TMN, (e) TMX and DPT, and (f) TMN and DPT, from MAR-1 simulations along with the historical values for the four seasons.

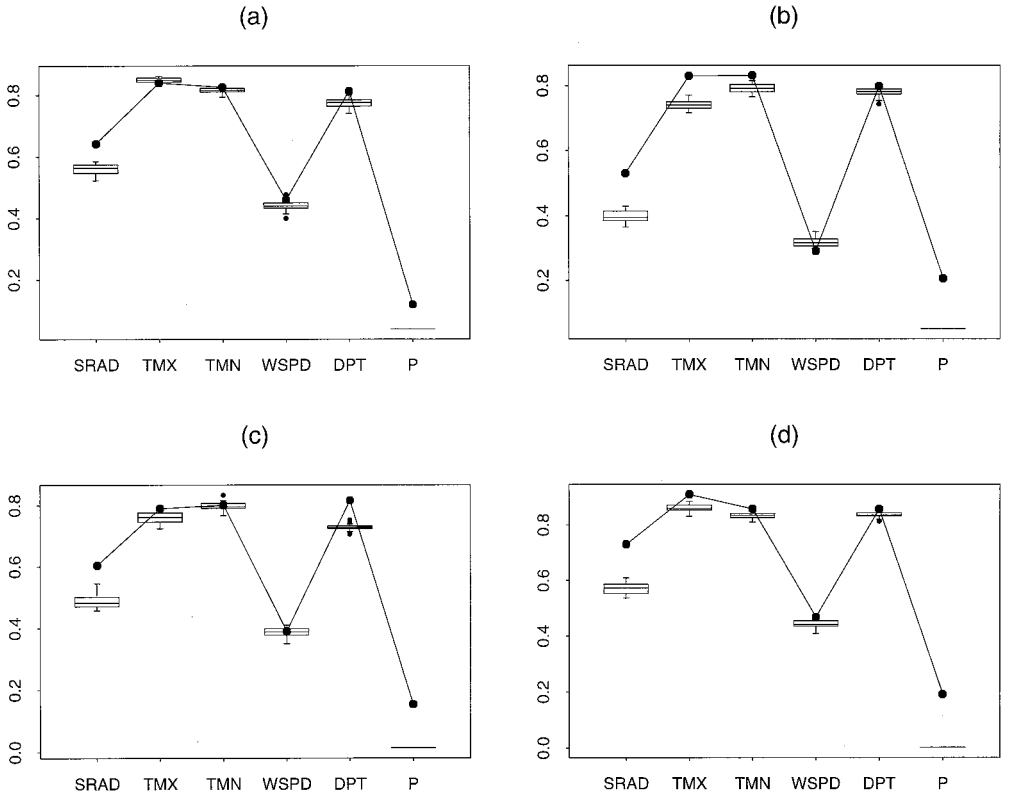

Figure 11. Box plots of lag 1 autocorrelation of SRAD, TMX, TMN, WSPD, and DPT for (a) season 1, (b) season 2, (c) season 3, and (d) season 4, from MAR-1 simulations along with the historical values.

cipitation. This is to be expected since precipitation is supplied exogenously to the model, unlike in the k-NN approach used here. We had the same experience with a MAR-1 model that was applied in the same manner. The kernel-density–based resampler was also biased near the boundaries of the domain and allowed the generation of values that were outside the variable bounds. While these problems can potentially be rectified at the expense of additional variance in the simulations, the k-NN resampler does not suffer from them. The "bandwidth" of the k-NN resampler automatically adapts to the local density of points, being larger where the data are sparse and smaller where the data are dense. This is an advantage over the kernel-density–based approach, where a fixed bandwidth is used and the number of points used for each local resample can vary drastically (from zero to a large fraction of the sample). The fixed bandwidth methods typically do poorly in the tails, where data are sparse, and also near the modes of the density, where the modes may be smoothed out. Finally, the k-NN approach presented here is computationally faster than the kernel-density–based approach given by *Rajagopalan et al.* [1997], where the parameters were also chosen in an ad hoc manner.

The comparisons with the MAR-1 model presented here are arguably unfair to the MAR model since the variables in MAR were not transformed to be approximately normal prior to simulation. However, we have made other comparisons with the CLIGEN model of *Nicks et al.* [1995], which does transform precipitation and wind speed to be normal using a trans-

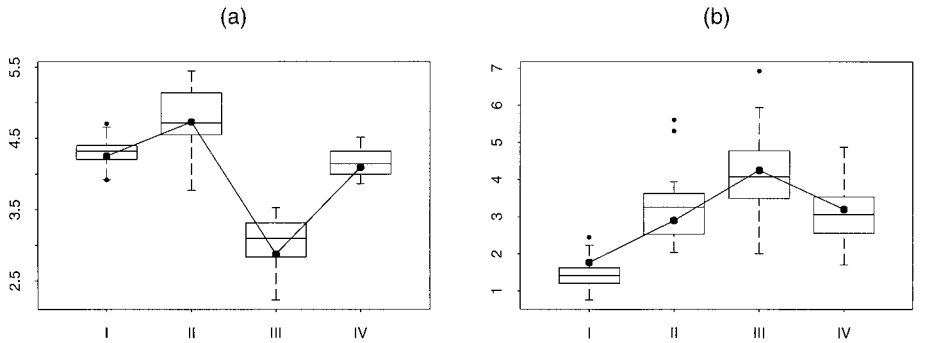

Figure 12. Box plots of (a) means of total seasonal precipitation and (b) variance of total seasonal precipitation for the four seasons along with the historical values.

formation from gamma to a normal distribution. Our conclusions are unchanged with respect to those comparisons. As was indicated in the section 2, working with transformed data in a multivariate setting is somewhat tenuous if unbiased real space statistics are of interest. The main point of our comparisons is that the k-NN approach is better at preserving the cross-dependence and frequency structure than earlier models that generate precipitation separately from other variables. This is useful, for instance, in getting the right attributes of snow versus rain and for erosion or crop modeling, where getting the right combination of meteorological variables makes a difference in the outcomes of interest.

Since it is a bootstrap, the simulations from the k-NN method do not produce values that have not been observed in the historical data. This is a major limitation if extreme values outside the available record are of interest. One can readily devise a strategy that allows nearest-neighbor resampling with perturbation of the historical data in the spirit of the traditional autoregressive models, i.e., conditional expectation with an added random innovation. First, one evaluates the conditional forecast $\mathbf{x}_{i,j}^s$ as shown in (5). Then one proceeds through the simulation by estimating the nearest-neighbor regression forecast relative to a conditioning vector $\mathbf{D}_i$ and then adding to this one of the $\mathbf{e}_j$ corresponding to a data point j that lies in the k–nearest-neighborhood $J_{i,k}$. An innovation $\mathbf{e}_j$ is then chosen using the resampling kernel $K[j(i)]$ in the same manner as the successor $\mathbf{x}_{j(i)}$ was chosen in section 4.1. This scheme will perturb the historical data points in the series, with innovations that are representative of the neighborhood, and will thus "fill in" between the historical data values, as well as extrapolating beyond the sample. The computational burden is increased, and there is a possibility that the bounds on the variables will be violated during simulation. However, there may be situations where the investigator may wish to adopt this strategy.

Acknowledgments. Partial support of this work by the U.S. Forest Service under contract notes INT-915550-RJVA and INT-92660-RJVA, Amend 1 is acknowledged. Partial support for the first author by NOAA grants UCSIOCU01556601D, NA36GP0074-02, and NA56GP0221 is also thankfully acknowledged. Comments and suggestions from Ashish Sharma, Charlie Luce, David Tarboton, Richard Katz, and two anonymous reviewers were helpful in preparing this manuscript. R. Katz is also thanked for bringing K. C. Young's work to our attention.

References

Abarbanel, H. D. I., and U. Lall, Nonlinear dynamics of the Great Salt Lake: System identification and prediction, *Clim. Dyn*., *12*, 237–297, 1996.

Bruhn, J. A., W. E. Fry, and G. W. Fick, Simulation of daily weather data using theoretical probability distributions, *J. Appl. Meteorol*., *19*, 1029–1036, 1980.

Cleveland, W. S., Robust locally weighted regression and smoothing scatter plots, *JASA J. Am. Stat. Assoc*., *74*, 829–836, 1979.

Efron, B., Bootstrap methods: Another look at the Jackknife, *Ann. Stat*., *7*, 1–26, 1979.

Fukunaga, K., *Introduction to Statistical Pattern Recognition*, Academic, San Diego, Calif., 1990.

Härdle, W., and A. W. Bowman, Bootstrapping in nonparametric regression: Local adaptive smoothing and confidence bands, *JASA J. Am. Stat. Assoc*., *83*, 102–110, 1988.

Jones, W., R. C. Rex, and D. E. Threadgill, A simulated environmental model of temperature, evaporation, rainfall, and soil moisture, *Trans. ASAE*, *15*, 366–372, 1972.

Katz, R., Use of conditional stochastic models to generate climate change scenarios, *Clim. Change*, *32*, 237–255, 1996.

Katz, R., and M. B. Parlange, Generalizations of chain-dependent processes: Application to hourly precipitation, *Water Resour. Res*., *31*(5), 1331–1341, 1995.

Lall, U., and A. Sharma, A nearest neighbor bootstrap for time series resampling, *Water Resour. Res*., *32*(3), 679–693, 1996.

Lall, U., B. Rajagopalan, and D. G. Tarboton, A nonparametric wet/dry spell model for resampling daily precipitation, *Water Resour. Res*., *32*(9), 2803–2823, 1996.

Lane, L. J., and M. A. Nearing, USDA Water Erosion Prediction Project: Hillslope profile model documentation, *NSERL Rep. 2*, Natl. Soil Erosion Res. Lab., U.S. Dep. of Agric., Agric. Res. Serv., West Lafayette, Indiana, 1989.

Nicks, A. D., and J. F. Harp, Stochastic generation of temperature and solar radiation data, *J. Hydrol*., *48*, 1–7, 1980.

Nicks, A. D., L. J. Lane, and G. A. Gander, Weather generator, USDA Water Erosion Prediction Project: Technical documentation, chap. 2, *NSERL Rep. 10*, pp. 2.1–2.21, Natl. Soil Erosion Res. Lab., U.S. Dep. of Agric., Agric. Res. Serv., West Lafayette, Indiana, 1995.

Rajagopalan, B., and U. Lall, A kernel estimator for discrete distributions, *J. Nonparametric Stat*., *4*, 409–426, 1995.

Rajagopalan, B., U. Lall, and D. G. Tarboton, A nonhomogeneous Markov model for daily precipitation simulation, *J. Hydrol. Eng*., *1*(1), 33–40, 1996.

Rajagopalan, B., U. Lall, D. G. Tarboton, and D. S. Bowles, Multivariate nonparametric resapling scheme for generation of daily weather variables, *Stochastic Hydrol. Hydraul*., *11*(1), 523–547, 1997.

Richardson, C. W., Stochastic simulation of daily precipitation, temperature, and solar radiation, *Water Resour. Res*., *17*(1), 182–190, 1981.

Salas, J. D., J. W. Delleur, V. Yevjevich, and W. L. Lane, *Applied Modeling of Hydrologic Time Series*, Water Resour. Publ., Fort Collins, Colo., 1980.

Sangoyomi, T., U. Lall, and H. D. I. Abarbanel, Nonlinear dynamics of the Great Salt Lake: Dimension estimation, *Water Resour. Res*., *32*(1), 149–159, 1996.

Tasker, G. D., Comparison of methods for estimating low flow characteristics of streams, *Water Resour. Bull*., *23*(6), 1077–1083, 1987.

Woo, M. K., Confidence intervals of optimal risk-based hydraulic design parameters, *Can. Water Resour. J*., *14*(1), 10–16, 1989.

Yakowitz, S., A stochastic model for daily river flows in an arid region, *Water Resour. Res*., *9*(5), 1271–1285, 1973.

Yakowitz, S., Nonparametric estimation of Markov transition functions, *Ann. Stat*., *7*, 671–679, 1979.

Yakowitz, S., Nonparametric density estimation, prediction, and regression for Markov sequences, *JASA J. Am. Stat. Assoc*., *80*, 215–221, 1985.

Yakowitz, S., Nearest neighbor regression estimation for null-recurrent Markov time series, *Stochastic Processes Their Appl*., *48*, 311–318, 1993.

Yakowitz, S., and M. Karlsson, Nearest neighbor methods with application to rainfall/runoff prediction, in *Stochastic Hydrology*, edited by J. B. Macneil and G. J. Humphries, pp. 149–160, D. Reidel, Norwell, Mass., 1987.

Young, K. C., A multivariate chain model for simulating climatic parameters from daily data, *J. Appl. Meteorol*., *33*, 661–671, 1994.

Zucchini, W., and P. T. Adamson, Bootstrap confidence intervals for design storms from exceedance series, *Hydrol. Sci. J*., *34*(1), 41–48, 1989.

U. Lall, Utah Water Research Laboratory, Utah State University, UMC 82, Logan, UT 84322-8200. (ulall@cc.usu.edu)

B. Rajagopalan, Lamont-Doherty Earth Observatory, Columbia University, P. O. Box 1000 Rt. 9W, Palisades, NY 10964-8000. (rbala@rosie.ldgo.columbia.edu)

(Received February 11, 1998; revised January 27, 1999; accepted January 28, 1999.)