An Empirical, Nonparametric Simulator for Multivariate Random Variables with Differing Marginal Densities and Nonlinear Dependence with Hydroclimatic Applications

Upmanu Lall,1,2 Naresh Devineni,3,4,* and Yasir Kaheil5

Multivariate simulations of a set of random variables are often needed for risk analysis. Given a historical data set, the goal is to develop simulations that reproduce the dependence structure in that data set so that the risk of potentially correlated factors can be evaluated. A nonparametric, copula-based simulation approach is developed and exemplified. It can be applied to multiple variables or to spatial fields with arbitrary dependence structures and marginal densities. The nonparametric simulator uses logspline density estimation in the univariate setting, together with a sampling strategy to reproduce dependence across variables or spatial instances, through a nonparametric numerical approximation of the underlying copula function. The multivariate data vectors are assumed to be independent and identically distributed. A synthetic example is provided to illustrate the method, followed by an application to the risk of livestock losses in Mongolia.

KEY WORDS: Copula; correlated risk; logspline; multivariate nonparametric simulator; spatial fields

1. INTRODUCTION

The recent focus on climate change and variability has increased the interest in methods for assessing multivariate risks. For instance, insurance underwriters may be interested in potentially correlated losses across a region related to rainfall, wind, or other climatic extremes. The agricultural sector may be interested in how correlated crop yields across a region could be modeled to better estimate the aggregate yield shortfall probabilities, or irrigation water requirements. Sectoral fund managers may be interested in portfolio risk assessment considering their potentially correlated assets. Typically, historical data on a set of these variables are used for such an assessment. Often, the joint likelihood of the occurrence of multiple variables (e.g., floods, tornadoes, heating degree days) at multiple locations may be of interest.

Hydrologists, agricultural specialists, and others may also be interested in the joint probability distribution of different climate and “outcome” (e.g., crop yields, prices, drought indices, heating indices, climate indices) variables. Sometimes, when seasonal climate forecasts or climate change scenarios are provided, the interest is in making sure that the climate variables provided as inputs for hydrologic or agricultural or economic simulations are mutually consistent, i.e., have a multivariate probability distribution that is consistent with that...
2. THEORY AND ALGORITHM

The literature on nonnormal multivariate dependence has recently seen significant growth in practical applications in hydrology, risk management, and finance, through the development and application of ideas related to copulas.\(^\text{(12–16)}\) For multiple random variables \(x\), the copula function\(^\text{(12)}\) \(C(.)\) was introduced as:

\[
F(x_1, x_2, \ldots, x_m) = P(X_1 < x_1, X_2 < x_2, \ldots X_m < x_m) \\
= P(F_1(x_1) < F_1(x_1), F_1(x_2) < F_2(x_2), \ldots, F_1(x_m) < F_m(x_m)) \\
= P(U_1 < F_1(x_1), U_2 < F_2(x_2), \ldots, U_m < F_m(x_m)) \\
= C(U_1, U_2, \ldots, U_m) \tag{1}
\]

where \(U_i = F_i(X_i)\), the distribution function of \(x_i\).

The copula, \(C(.)\), is a multivariate cumulative distribution function. When the copula \(C(.)\) is differentiable, the multivariate probability density \(f(x)\) can be expressed in terms of the marginal densities of its comprising variables and a copula density as:

\[
f(x) = f_1(x_1) f_2(x_2) \ldots f_m(x_m) c(u_1, u_2, \ldots, u_m) \tag{2}
\]

where \(f_i(x_i)\) is the univariate probability density for variable \(x_i\), \(u_i\) are uniformly distributed random variables, and \(c(.)\) is the copula density or dependence function of the \(u_i\).

A central idea that emerges from Equation (2) is that the multivariate density can be expressed in terms of individual marginal densities, which can be different for each variable and specified independently of each other, and a copula function. The interest in copulas has grown since it frees the practitioner from the traditional framework of multivariate normal distribution and linear correlation-based models. The linear dependence framework often does not always provide a satisfactory multivariate model even after transformation of marginals to normality. To develop a multivariate model, one needs to fit an appropriate probability density \(f_i(x_i)\) to the data for the \(i\)th variable, and an appropriate copula function across all variables. In both cases there are parametric and nonparametric choices.

Parametric choices may be hard to discriminate between, and may or may not provide a good fit to the data in all cases (e.g., in the presence of mixtures or regimes in state space). Nonparametric methods are limited by a modest ability to extrapolate beyond the range of the observed data, and hence are more
limited than parametric methods to the state space explored by the observed data set. As sample sizes increase, and each observation is drawn independently, the coverage of the state space can increase, and the ability to discriminate between parametric distributions, and to get a better representation by nonparametric methods, improves. We have a preference for an automatic procedure using nonparametric methods, but recognize its limitations as above.

2.1. Estimating the Marginal Densities

In many cases, investigators have their favorite choices of a restricted set of probability distributions to fit to univariate, i.i.d. data, e.g., log normal, log Pearson 3 or GEV for floods, beta for crop yields, or normal for seasonal temperature, and select the one that fits best from a limited set. There may or may not be an unequivocally “best” choice in a given setting based on the usual criteria, and hence there is uncertainty as to the “model choice” for any given data series. Since the early books\(^{17-19}\) and review paper of Ref. 20 nonparametric density estimation, using kernel, local polynomial, spline, or nearest neighbor density methods, has become increasingly popular for hydrometeorological, econometrics, agriculture, and other application areas. These methods do not assume an underlying functional form for the density, but approximate an arbitrary, differentiable density through local estimation. Typically, the resulting density estimates have higher variance but lower bias than a specific parametric estimator, but have the advantage that the model choice issue, which adds uncertainty or variance to the parametric estimation process, is avoided. In our experience, a particularly effective and automatic nonparametric density estimation method in the univariate case, especially for variables that may have a known lower or upper bound and for small sample sizes, is log spline density estimation\(^{21,22}\). Given the potentially diverse and large number of univariate density functions one may need to estimate, we propose the use of log spline density estimation for the functions \(f_i(x_i)\). It has lower bias and higher rates of finite sample convergence to the underlying density function than some traditional nonparametric density estimators, is adaptive to the data, has automatic parameter selection, and the integrated mean square error compares well with parametric density estimation for known density functions. The log spline density estimate is constructed as follows:

\[
f_i(x_i) = e^{\left(\sum_{k=1}^{K} \beta_k g_k(x_i) - \beta_0\right)}
\]

where \(g_k(x_i)\) represent \(K\) basis functions, which are taken to be cubic splines defined over a sequence of knots, \(t_1, \ldots, t_k\), and the \(\beta\) are their coefficients. The \(\beta\) are estimated using maximum likelihood with the constraint that the density integrates to 1, i.e., \(\int f_i(x_i) = 1\). We refer the reader to Ref. 21 for details of knot placement, selection of the number of knots, and the implementation of the algorithm. The method is fully automatic, in that the knot locations of the spline, the number of knots, and the coefficients of the basis functions are selected using penalized functions. The quantile function can be derived directly from the solution. The log spline package in \(R\) was used in the applications reported here. While this is the method used for univariate density estimation in this article, other methods, parametric or nonparametric, could also be used. The key point of the article is the development of the multivariate simulator and the choice of the univariate density function is not an issue.

2.2. The Copula Function and Dependence

A number of parametric copula functions and methods for estimating their parameters are available.\(^{23}\) Typically, the parametric copulas (e.g., the Gaussian, the Archimedean, and the Marshall-Olkin) and their parameters lead to particular ways by which the dependence of multiple variables in the tails of the distribution or overall are modeled. Upper and lower tail dependence can be modeled separately, which is an advantage for many of these functions. Copulas provide a rather general approach to modeling dependence. However, this can also be a drawback. Particularly as the dimension \(m\) of the multivariate space increases, selecting the best copula to fit and estimating it can be difficult\(^{24}\) unless there is a consistent parametric framework for the entire process, and the sample size is large. Nonparametric copulas can be fit using kernel functions.\(^{25}\) However, for successful and automatic implementation, these require boundary correction (recall that all the \(u_i\) are uniformly distributed between 0 and 1, and hence kernel boundary bias will dominate as the dimension \(m\) increases), and bandwidth selection. Here, we consider a strategy that entails sampling from the empirical copula.\(^{26}\)

The empirical copula \(C_{\text{emp}}\) of the data\(^{26}\) is a consistent estimator of the unknown copula \(C(U_1,\)}
$U_2 \ldots U_m$). Let the historical data be denoted as the set $A$: $(x_i, j = 1 \ldots n)$, where $n$ is the sample size, and $x \in \mathbb{R}^n$. Then define:

$$\hat{u}_j = \hat{F}(x_j) = \frac{r_j}{n+1}$$  \hspace{1cm} (4)

where $r_i$ is the vector of univariate ranks corresponding to each of the components of $x_i$.

The empirical copula can then be defined through the $C(\hat{u}_j)$. The empirical estimate of the univariate cumulative distribution function, $\hat{F}(x_j)$, is based on the plotting position formula.\(^{(27)}\) Other empirical estimates of the quantile function exist\(^{(28)}\) and could also be used. There is general uncertainty as to the best one to use since the “optimal” one varies by the underlying distribution. In our non-parametric framework, we choose to use the Weibull estimator,\(^{(27)}\) but also propose to account for the uncertainty by considering that each estimate may actually be a draw from a distribution centered on the estimate in Equation (4), rather than a point value. This leads to a smoothed bootstrap\(^{(29)}\) under simulation, and reduces the sensitivity to the actual plotting position formula used to estimate the empirical distribution function. Further, we consider an “out of sample” approach to the estimation of this distribution function. For each variable $x$, consider the following interval estimate of the cumulative distribution function corresponding to each historical observation, $x_{ij}, j = 1 \ldots n$. Let $v_{jk}$ be the sorted sample values obtained by dropping $x_{ij}$ from the sample, $k = 1 \ldots (n-1)$. Then:

$$\begin{align*}
x_{ij} < v_{i1} & \quad 0 < F(x_{ij}) < \frac{1}{n} \\
v_{(j-1)} < x_{ij} < v_{ij} & \quad \frac{j-1}{n} < F(x_{ij}) < \frac{j}{n} \\
x_{ij} > v_{(n-1)} & \quad \frac{n-1}{n} < F(x_{ij}) < 1
\end{align*}$$  \hspace{1cm} (5)

This can be represented as:

$$\hat{u}_{ij} \sim \begin{cases} 
U \left[ \frac{r_j - 0.5}{n+1}, \frac{r_j + 0.5}{n+1} \right] & \text{if } 2 < r_{ij} < n-1 \\
U \left[ 0, \frac{r_j + 0.5}{n+1} \right] & \text{if } r_{ij} = 1 \\
U \left[ \frac{r_j - 0.5}{n+1}, 1 \right] & \text{if } r_{ij} = n
\end{cases}$$  \hspace{1cm} (6)

where $U[a,b]$ refers to a uniform distribution with limits $a$ and $b$, and the $r_j$ is the rank of the $j$th observation of variable $x_i$ in the historical data set $A$. Now consider drawing a sample of the $\hat{u}_{ij}$ of length $n$. This sample will preserve the multivariate dependence across the $x$ as represented in the historical sample, as long as the $j$th position is maintained for all variables sampled consistent with the position in $A$. Essentially, such a sample will be a candidate representation of the empirical copula $C_{emp}$, sampled to account for the uncertainty in the empirical quantile function, and leads to a smoothing of the empirical quantile function used, via simulation, and hence to a smoothed bootstrap.

2.3. Simulation Algorithm

2.3.1. Case 1: Simulating Using Only a (Historical Data Vector)

In this case, we assume that we have a data set $\{A: x_j, j = 1 \ldots n\}$, such that each of the $n$ vectors is independent and identically distributed, and we wish to draw multiple random samples $B$, each of size $n$, where the dependence across the components of $x$ is preserved. The proposed algorithm is described below:

1. Generate univariate sample of length $n$ for each variable $i$
   a. Estimate the $m$ univariate probability density functions $f_i(x_i)$ using the data $A$ and logspline density estimation.
   b. From each fitted $f_i(x_i)$, draw a random sample $x_{ij}$ of length $n$ $\{x_{ij}, j = 1 \ldots n\}$.
   c. Transform each vector $\{x_{ij}, j = 1 \ldots n\}$ into its ranks $\{R_{ij}, j = 1 \ldots n\}$.

2. Generate samples from the random draw of the empirical copula
   a. Construct the empirical copula $C_{emp} \{z_j, j = 1 \ldots n\}$ from $A$, using Equation (5) and Equation (6), where $z_j$ is the represented as a rank matrix.
   b. Bootstrap vectors, i.e., draw $n$ vectors with replacement from $z_j$, and record them as $\{z_{ij}, j = 1 \ldots n\}$.

3. Assemble data set $B$ combining the univariate samples and the copula bootstrap
   a. Each element $w_i$ in the new sample is generated by setting $w_{ij} = x_{ij} \delta (z_{ij} - R_{ij})$ $j = 1 \ldots n$, $i = 1 \ldots m$

$$\begin{align*}
w_{ij} = x_{ij} \delta (z_{ij} - R_{ij}) & \quad j = 1 \ldots n, \quad i = 1 \ldots m
\end{align*}$$  \hspace{1cm} (7)

where $\delta(\cdot)$ is the delta function, equal to 1 if the argument is 0, and 0 else
b. The simulated data is now in $\{B: w_j, j = 1 \ldots n\}$

The key step in this process is in Equation (7). The $j$th vector drawn from the empirical copula has ranks (from 1: $n$) of each variable $i$, recorded as $z_{ij}$.
Next, from the new univariate sample drawn for the \( i \)th variable, we find the value \( x'_{ij} \) that has the same rank \( R_{ij}' \) in a sample of size \( n \) as the rank recorded in \( z_{ij} \). This allows us to reorder the univariate samples generated so that the structure of the empirical copula is reproduced. The following example provides a simple illustration of the algorithm.

Let us consider a multivariate vector \( \mathbf{x} \) of size 10 comprising four random variables with different marginal densities but a common underlying joint density. This is shown in Fig. 1 as a table. As in Step 1 in the algorithm, we generate a univariate sample of length 10 for each of the four variables. This is labeled \( x' \). We transform this matrix into its sorted matrix \( \mathbf{R}' \). Next, as in Step 2, we construct the rank matrix of \( \mathbf{x} \) (\( z \)) and generate bootstrap samples from this matrix to inform the underlying joint distribution (\( z' \)). The final step involves rearranging the matrix \( \mathbf{x}' \) into \( \mathbf{w} \) using \( \mathbf{z}' \) and \( \mathbf{R}' \). For instance, the first row (highlighted in orange (colors visible in on-line version)) in the matrix \( \mathbf{w} \) is constructed based on the ninth largest value in \( x'_{1j} \), 10th largest value in \( x'_{2j} \), and the sixth largest value in \( x'_{3j} \) obtained from the logspline generated univariate samples in \( x' \). Similarly, row six (highlighted in yellow) comprises of the fourth, seventh, sixth, and fifth largest values in \( x' \). The other rows are constructed using similar rearrangement.

2.3.2. Case 2: Simulating Using a (Historical Data Vector) and F (New Data Vector That Has Correct Marginal Distributions, But Lacks the Joint Dependence)

The primary difference in this situation is that we have a new sample \( \{\mathbf{F}: \bar{x}_j, j = 1 \ldots n_f\} \), where the data \( \bar{x}_j \) correspond to the statistics of the variables \( \mathbf{x} \), but have been drawn from changed conditions, e.g., they may be ensemble seasonal forecasts from a climate model, or ensembles across many climate models, or independent stochastic simulations of the variables that have the same univariate statistical properties as in \( \mathbf{x} \). We consider that the univariate distributions of these variables are properly calibrated as represented in the data available to us, but the spatial or across-variable dependence is not correct. This could happen, for instance, if a univariate bias correction were applied to each ensemble member based on the statistics of each climate model’s output over a period where instrumental data and model simulations were available.\(^3\) As illustrated in Ref. 30, such bias corrections typically will not properly address multivariate dependence. This could also occur for independent stochastic weather/climate models that preserve the statistics of the variables at a given location or grids but do not replicate the spatial variability. Step 1a of the algorithm in the previous section is changed to address this situation. The other steps do not change.

**Modified 1. Generate univariate sample of length \( n \) for each variable \( i \)

- a. Estimate the \( m \) univariate probability density functions \( f_j(x_i) \) using the data \( \{\mathbf{F}: \bar{x}_j, j = 1 \ldots n_f\} \) and logspline density estimation. Here, \( n_f \) is the total ensemble size available. It could be different for each variable, but generally will not be.
- b. From each fitted \( f_j(x_i) \), draw a random sample \( x'_{ij} \) of length \( n \) \( \{x'_{ij}, j = 1 \ldots n\} \).
- c. Transform each vector \( \{x'_{ij}, j = 1 \ldots n\} \) into its ranks \( \{R'_{ij}, j = 1 \ldots n\} \).
- d. Follow the procedure as in the previous section.

3. APPLICATIONS

In this section, we first demonstrate the applications of the proposed algorithm using a synthetically generated data set. This is followed by an application to livestock mortality rate across 327 soums (an administrative unit) in Mongolia.

3.1. Synthetic Experiment (Case 1)

Consider a four-dimensional vector of sample size \( n = 100 \) generated from a multivariate Gaussian copula distribution. Consider that these four variables have different marginal distributions: normal, lognormal, beta, and a mixture normal with the joint distribution informed by the Gaussian copula.

\[
\begin{align*}
    x_1 &\sim N(0, 1) \\
    \log(x_2) &\sim N(0, 1) \\
    x_3 &\sim \text{Beta}(2, 5) \\
    x_4 &\sim N(0, 0.5) + N(3, 0.5) \\
\end{align*}
\]

(8)

The pair-wise scatterplot and the univariate density plot of the sample of \( \mathbf{x} \) are shown in Fig. 2. The line in each scatterplot is a locally weighted scatterplot smoother estimated using a nonparametric local density estimator.\(^3\) It is evident that the pair-wise
dependence between the variables may be nonlinear. The algorithm described in Section 2.3.1 is now used to generate 100 samples of the matrix \( w \) of size 100\( \times \)4. For each of the 100 samples, we compare the marginal density and the pair-wise dependence of the simulated variables with the original data. In addition, several statistics of interest to compare univariate marginal distribution, pair-wise dependence, and the dependence in the extremes are also computed from \( w \) and compared to those for \( x \).

Fig. 1. Sample illustration of the nonparametric simulation algorithm.

**Table 1.**

<table>
<thead>
<tr>
<th>( x )</th>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( x_3 )</th>
<th>( x_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.47</td>
<td>1.09</td>
<td>0.26</td>
<td>0.59</td>
<td></td>
</tr>
<tr>
<td>-0.04</td>
<td>2.1</td>
<td>0.33</td>
<td>3.58</td>
<td></td>
</tr>
<tr>
<td>0.17</td>
<td>0.4</td>
<td>0.14</td>
<td>-0.35</td>
<td></td>
</tr>
<tr>
<td>0.82</td>
<td>0.86</td>
<td>0.47</td>
<td>3.67</td>
<td></td>
</tr>
<tr>
<td>-0.48</td>
<td>0.46</td>
<td>0.16</td>
<td>0.38</td>
<td></td>
</tr>
<tr>
<td>-0.65</td>
<td>0.35</td>
<td>0.05</td>
<td>-0.26</td>
<td></td>
</tr>
<tr>
<td>0.58</td>
<td>1.52</td>
<td>0.35</td>
<td>1.12</td>
<td></td>
</tr>
<tr>
<td>-1.61</td>
<td>0.42</td>
<td>0.16</td>
<td>-1.22</td>
<td></td>
</tr>
<tr>
<td>-0.17</td>
<td>0.56</td>
<td>0.23</td>
<td>1.51</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( z )</th>
<th>( z_1 )</th>
<th>( z_2 )</th>
<th>( z_3 )</th>
<th>( z_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>10</td>
<td>7</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>6</td>
<td>9</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>8</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>10</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>5</td>
<td>8</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( x' )</th>
<th>( x'_1 )</th>
<th>( x'_2 )</th>
<th>( x'_3 )</th>
<th>( x'_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.12</td>
<td>0.73</td>
<td>0.2</td>
<td>0.3</td>
<td></td>
</tr>
<tr>
<td>-0.58</td>
<td>0.43</td>
<td>0.13</td>
<td>0.69</td>
<td></td>
</tr>
<tr>
<td>-0.53</td>
<td>2.95</td>
<td>0.21</td>
<td>0.24</td>
<td></td>
</tr>
<tr>
<td>-1.81</td>
<td>0.68</td>
<td>0.43</td>
<td>-1.01</td>
<td></td>
</tr>
<tr>
<td>-0.01</td>
<td>0.58</td>
<td>0.15</td>
<td>-0.52</td>
<td></td>
</tr>
<tr>
<td>-0.34</td>
<td>0.1</td>
<td>0.21</td>
<td>3.1</td>
<td></td>
</tr>
<tr>
<td>0.83</td>
<td>1.12</td>
<td>0.29</td>
<td>3.03</td>
<td></td>
</tr>
<tr>
<td>0.1</td>
<td>1.01</td>
<td>0.15</td>
<td>1.25</td>
<td></td>
</tr>
<tr>
<td>-1.57</td>
<td>0.64</td>
<td>0.24</td>
<td>4.3</td>
<td></td>
</tr>
<tr>
<td>-1.55</td>
<td>1.83</td>
<td>0.22</td>
<td>0.52</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( z' )</th>
<th>( z'_1 )</th>
<th>( z'_2 )</th>
<th>( z'_3 )</th>
<th>( z'_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>9</td>
<td>10</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>10</td>
<td>7</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>3</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>6</td>
<td>9</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>10</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>8</td>
<td>7</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( w )</th>
<th>( w_1 )</th>
<th>( w_2 )</th>
<th>( w_3 )</th>
<th>( w_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.12</td>
<td>1.83</td>
<td>0.43</td>
<td>0.69</td>
<td></td>
</tr>
<tr>
<td>-0.58</td>
<td>1.01</td>
<td>0.21</td>
<td>0.52</td>
<td></td>
</tr>
<tr>
<td>-1.57</td>
<td>0.10</td>
<td>0.13</td>
<td>0.24</td>
<td></td>
</tr>
<tr>
<td>-0.58</td>
<td>1.01</td>
<td>0.21</td>
<td>0.52</td>
<td></td>
</tr>
<tr>
<td>-0.34</td>
<td>2.95</td>
<td>0.22</td>
<td>3.10</td>
<td></td>
</tr>
<tr>
<td>-0.58</td>
<td>1.01</td>
<td>0.21</td>
<td>0.52</td>
<td></td>
</tr>
<tr>
<td>-1.81</td>
<td>0.58</td>
<td>0.15</td>
<td>-1.01</td>
<td></td>
</tr>
<tr>
<td>0.83</td>
<td>0.73</td>
<td>0.29</td>
<td>4.30</td>
<td></td>
</tr>
<tr>
<td>0.12</td>
<td>1.83</td>
<td>0.43</td>
<td>0.69</td>
<td></td>
</tr>
<tr>
<td>0.10</td>
<td>1.12</td>
<td>0.24</td>
<td>1.25</td>
<td></td>
</tr>
</tbody>
</table>

**Fig. 3** shows the pair-wise scatterplot of the original matrix \( x \) in red circles and the simulated matrix \( w \) in gray circles. The diagonal plots show the comparison of the univariate density of the original variable in \( x \) (shown in red), and the simulated variable in \( w \) (shown in gray) for all the 100 scenarios. The three boxplots on the right are presented to compare the first three moments (mean, standard deviation, and skewness) of the marginal distributions of the original variables in \( x \) (shown in red dots) with the
simulated variables in \( W \) (boxplot from the simulations). The underlying marginal distribution as well as the joint distribution of the data is reproduced without any bias and with reasonable sampling variations using the algorithm. Dependence, as measured by the pair-wise correlation (Pearson’s rho, Spearman’s rho, and Kendall’s tau), and dependence in extremes, as measured by the tail dependence coefficient, are also compared with those for the original data. The lower triangle in Fig. 4 shows the pair-wise box plots of the Kendall tau, Spearman, and Pearson correlation coefficients between the variables in \( W \). Each box represents the interquartile range, and the line in the middle indicates the median computed from the 100 samples. The solid circle (in red) corresponds to the same statistic computed from the original data \( x \). These boxplots show the range of variations in the performance metrics from
Fig. 3. Pair-wise scatterplot of the data matrix $w$ simulated from $x$ using the algorithm under case 1. The pair-wise scatter of the data matrix $x$ is shown in red dots. The red lines in each section are locally weighted (LOWESS) scatterplot smooths for the original variables $(x)$. The univariate density plot of $w$ (100 simulations) along with the univariate density plot of $x$ (in red) using nonparametric local density estimator are shown in the diagonal.

The tail dependence coefficient estimates the conditional probability that one marginal distribution exceeds a threshold given that the second distribution has exceeded it. The tail dependence coefficient for two variables $X$ and $Y$ is given as:

$$
\lambda_u = \Pr(F(X) > t | F(Y) > t)
$$

where $F(X)$ and $F(Y)$ are the cumulative distribution functions of $X$ and $Y$, $t$ is a threshold (e.g., the 90th percentile), and $\Pr$ represents the conditional probability. $\lambda_u$ is bounded between 0 and 1 and $\lambda_u \sim 0$ indicates no (strong) tail dependence. Here, we estimate $\lambda_u$ using a nonparametric approach as illustrated in Ref. 34. A detailed
A Multivariate Nonparametric Simulator

Fig. 4. Boxplot of the pair-wise dependence measures for the simulated matrix \( w \) from the 100 scenarios (case 1). The dependence measures for the original data \( x \) are shown in red circles. The lower triangle shows the results from correlation coefficients and the upper triangle shows the results from upper tail dependence coefficient, \( \lambda_u \).

description of the theory and various parametric and nonparametric estimators of the tail dependence coefficients can be found in Refs. 32 and 35 through 37. The upper triangle in Fig. 4 shows the box plots of pair-wise \( \lambda_u \) estimated between the variables in \( w \), while \( \lambda_u \) for the original data \( x \) is shown in red circle. Fig. 4 shows that the tail dependence in the original data is also reproduced well using the simulation. The sampling variations we see in the boxplots in Fig. 4 are obtained as a result of sampling the empirical copula \( C_{\text{emp}} \) as a distribution around the point estimate of the plotting position to represent the uncertainty in estimating the nonparametric copula function of the underlying data.
3.2. Synthetic Experiment (Case 2)

In the second example, we consider the same $x$ as in the previous example, but choose a new sample $\tilde{x}$ for simulating the data. The main difference between the two examples is that in the first we used the simulation algorithm with $x$ to simulate $w$, while in the second example we use both $x$ and $\tilde{x}$ to simulate $w$. $\tilde{x}$ has the same marginal distributions as $x$, but does not have the dependence structure. Such a situation can arise, for instance, where a climate forecast ensemble that has been bias corrected variable by variable to match the parameters of a historical data set, but the cross-variable dependence is not reproduced. A similar approach to ours is presented in the literature.(6,7) The practical
difference is that we consider uncertainty in the estimation of the empirical distribution function, and consider samples of arbitrary size, while they basically just swap ranks using samples of equal size. Fig. 5 shows the pair-wise scatter plots and the univariate density plots of the data matrices \( x \) and \( \tilde{x} \). The red dots and the red line in each section represent the pair-wise scatter and the locally weighted scatterplot smooths for \( x \). The black dots in each section represent the scatter for \( \tilde{x} \), the new data matrix. The univariate density plots for \( x \) (shown in red) and \( \tilde{x} \) (shown dashed black) using nonparametric local density estimator are shown in the diagonal. As discussed above, we can see that \( \tilde{x} \) has similar marginal distributions as \( x \) for the individual processes, but the joint dependence is missing. The algorithm described in Section 2.3.2 is used with \( x \) and \( \tilde{x} \) to generate 100 samples of the matrix \( w \) of size 100×4. Random
draws from the marginal distributions of $\tilde{x}$ estimated using logspline density estimator are rearranged using the underlying copula approximation of $x$ to obtain samples of $w$. Fig. 6 shows the pair-wise scatterplot of the original matrix $x$ in red circles, new matrix $\tilde{x}$ in black circles, and the simulated matrix $w$ in gray circles. The diagonal plots show the comparison of the univariate density of the original variables in $x$ (shown in red), $\tilde{x}$ (shown in dashed black lines), and the simulated variable in $w$ (shown in gray) for all the 100 samples. The right-most column presents the comparison (original to simulated) of the distribution of the three moments of the marginal distribution. We can see that the underlying joint distribution (as informed by $x$) is now reproduced for $w$, with reasonable sampling variety.

As in the previous example, we now check the dependence measures (Pearson’ rho, Spearman’s rho, Kendall tau, and $\lambda(x)$) of the simulated data $w$ against the dependence in the original data $x$ and new data $\tilde{x}$. The lower triangle in Fig. 7 represents the dependence measured by correlation coefficients and the upper triangle in Fig. 7 represents the tail dependence measured by tail dependence coefficients. We can see that while there is no dependence in the variables in $\tilde{x}$ (correlation coefficients and the tail dependence is close to 0 as seen from black circles), the dependence in the simulated data in $w$ (as seen from the box plots) is in the range of the dependence in the original data $x$ (shown as solid red circles).

3.3. Simulating Livestock Mortality Rate Over Mongolia for Risk Management

In this section, we present the application of the algorithm for simulating spatially correlated livestock mortality rate data from 327 soums in Mongolia. Livestock herding is one of the important sectors in Mongolia. The agricultural and livestock sector in Mongolia accounts for 15% of the national gross domestic production and supports around 39% of population.\(^{(38)}\) Given its geographic location, livestock mortality due to harsh winters is not uncommon for Mongolia. Recent events include the early 2000’s sudden onset winter storms (termed as dzud) with severe temperatures that led to extreme livestock mortality rate.\(^{(39)}\) Consequently, several international organizations are now focusing on developing solutions for sustainable livelihood. Index-based livestock insurance products that are aimed to hedge against catastrophic herd loss are gaining popularity.\(^{(40,41)}\) An index-based insurance program based on livestock mortality rates by species and soum was recommended to the government of Mongolia by the World Bank. In 2005 they entered into a credit agreement to implement the Index-Based Livestock Insurance Program. The approach entails self-insurance up to a 6% mortality rate where the herders bear the cost of small losses and a combination of market-based insurance and a social safety net from 6% to 30% mortality rates. The final layer of catastrophic loss (>30% mortality) is borne by the government of Mongolia. Index insurance has lower transaction costs and quicker payouts in the event of the disaster.

For insurers to objectively develop policies and rates, an accurate assessment of the spatially correlated risks (as seen in mortality rates) and analysis of potential exposure zones or regions is warranted. We applied our algorithm to the soum-level time series of livestock mortality rates, under the assumption of stationarity. Thirty-nine years of data (from 1972 to 2010) for each soum were obtained from the National Statistics Office of Mongolia.\(^{(38)}\) Fig. 8 shows the livestock mortality rate for four different return periods estimated from the observed data using the logspline density estimate. Note that the southwest provinces of Bayanhongor, Govi-Altay, and Dzavhan have higher mortality rates (>30%) for a return period of 20 years. The algorithm as described in Section 2.3.1 is employed on the original data to generate 100 simulations of the $39 \times 327$ data matrix. The logspline density estimator was used with the variable domain bounded between 0 and 100. Fig. 9 shows the median return period level for the 2, 10, 20, and >20 years estimated from the 100 simulations. We can see that the stochastic simulator catalog performs reasonably well in reproducing the spatial risk across the country. The simulation strategy is not only able to replicate the local risk or exceedance interval at each soum, but also generates spatially consistent return period levels that can be used to create potential exposure zones and targeted policies.

In addition to simulating the mortality rate and estimating their return periods for identifying the potential exposure zones, we also conducted an indemnity analysis to estimate the annual aggregate payouts under the three layers of the index-based insurance program. In the first layer (mortality rate <6%), the losses are retained by the herders and managed through various risk mitigation measures and capital savings. Less frequent and more severe losses occurring at a mortality rate between 6% and 30%
Fig. 7. Boxplot of the pair-wise dependence measures for the simulated matrix $\mathbf{w}$ from the 100 scenarios (case 2). The dependence measures for the original data $\mathbf{x}$ and $\tilde{\mathbf{x}}$ are shown in solid red and black circles, respectively. The lower triangle shows the results from correlation coefficients and the upper triangle shows the results from upper tail dependence coefficients, $\lambda_{u}$.

are financed by private insurance companies, where the herders purchase commercial products at different premiums. Catastrophe events at a mortality rate greater than 30% are covered by a government-sponsored social safety net program. For each layer, we estimate the aggregate annual payout as follows:

$$\text{Total Self Payout}_t = \sum_{i=1}^{M} \text{Self Payout}_i^t;$$

$$M = 1 \ldots M \text{ soums};$$

$$t = 1 \ldots T \text{ years}$$

$$\text{Self Payout}_i^t = \begin{cases} 
0, & \text{Mortality Rate}_i^t = 0\% , \\
\text{Mortality Rate}_i^t > 6\% , & \\
1, & 0\% < \text{Mortality Rate}_i^t \leq 6\% 
\end{cases}$$

$$\text{Insurance Payout}_i^t = \begin{cases} 
0, & 0\% < \text{Mortality Rate}_i^t \leq 6\% , \\
\text{Mortality Rate}_i^t > 30\% , & \\
1, & 6\% < \text{Mortality Rate}_i^t \leq 30% 
\end{cases}$$
We estimated the total self payout, total insurance payout and the total government payout each year using the simulations from the spatial simulator catalog. In addition, we also estimated these under a null hypothesis that the mortality rate is not spatially correlated. For each scenario (correlated vs. uncorrelated mortality rate) we have 100 simulations to represent the uncertainty in the payout estimates. Fig. 10 shows the probability distribution of the total payouts for each layer under both assumptions. Fig. 10 also shows the probability distribution (as red line) of the total payouts for each layer computed using the historical mortality rate data. We can see from Fig. 10 that the total payouts estimated for each layer using the mortality rate simulations from the spatial simulator are consistent with the payouts estimated from the historical mortality rates, indicating that the algorithm simulates the spatial dependence reasonably well. Analysis of the indemnities under the no correlation case clearly illustrates the importance of considering the joint exposure of the soums while developing insurance policies and payout schemes. The median number of payouts from 100 simulations of 39 years each, for the 327 soums for layer 2 (private insurance payout) is 48, while the 95th percentile payouts under this scheme are 167. Similarly, the median number of government payouts is 0 and the 95th percentile payouts are 21. We have experimented with increasing the bootstrap samples up to 10,000 and find that the median payout distribution converges to a value...
similar to the one reported, with confidence limits that appear to converge as the number of bootstrap samples increase.

Forming a reliable covariance matrix for this dimension and sample size is also not easy, even if the underlying model were completely linear and Gaussian. Given the experience with the synthetic examples, we believe that the simulator does a credible job of reproducing the payout distribution, but this is a challenging problem, and further investigations as to dimension reduction by different methods and sensitivity analyses of the results are needed.

4. SUMMARY

Designing an automatic multivariate simulation system for high dimensional variables with varying marginal densities and non-Gaussian dependence is a challenging proposition. Our goal was to develop an automatic, nonparametric approach that does reasonably well in this setting, and can account for some of the uncertainties in estimating the multivariate dependence structure. Specifically, we do not address epistemic uncertainties, and consider primarily uncertainties related to the statistical generation of a sample from the state variables that are available. Further, the approach presented is limited by the state space that is effectively sampled in the historical data, and does not seek to extrapolate the dependence structure beyond what has been sampled. The limitations of such an approach are readily understood by considering the Lorenz\(^{42}\) attractor, which exhibits a “butterfly” shape in three-dimensional state space. One could sample values for the three variables \((x,y,z)\) at a high sampling rate, and have a large sample size, and yet have sampled only one wing of the butterfly, which may be in the \((x,z)\) or \((x,y)\) planes. Clearly, our resampling-based method will not be able to generate the simulations in the \((x,y)\) plane if the observed data were in the \((x,z)\) plane. However, a sample of the same size, but with coverage in both planes, and hence a lower sampling rate, would work quite well for our smoothed bootstrap approach to modeling the complex dependence in the three variables that is offered by the Lorenz data. Indeed, we note with synthetic examples that
Fig. 10. Indemnity analysis using the spatial simulator catalog (spatially correlated mortality rate) and the null model (uncorrelated mortality rate). The probability distribution of the total number of payouts under each layer of the insurance scheme is shown for the spatially correlated mortality rate case (top panel) and the uncorrelated case (bottom panel). The median and the 95th percentile number of self, insurance, and government payouts is also shown in the inset. The uncertainty in these estimates is represented in parentheses.

the tail dependence as well as the dependence in state space is reproduced with uncertainties that decrease with the size of the original sample, and also with the number of bootstrap simulations generated. We also note, that for the example where only one wing of the butterfly has been sampled, we do not know of a method that would be effective to reproduce the dependence across the three Lorenz variables—even a calibration of the Lorenz model to estimate parameters from the data may not be effective.

Our thinking was influenced by our past work in developing nonparametric smoothed bootstraps in low dimensional settings, and by the simple shuffling schemes in Refs. 6 and 7. The simple, automatic approach developed is implemented in R with the code available from the authors. It has a wide range of potential applications to assess portfolio risk when the variables of interest are a spatial process, or block maxima of climate variables or of stocks. The primary assumption is that the underlying process is stationary, and that the realizations of the variables are independent and come from the same distribution.

Extensions to consider include a spatio-temporal model, where each variable has a temporal evolution model, and spatial dependence is preserved across temporal simulations are in progress. Markovian and non-Markovian temporal processes can be considered in this framework. Parametric and other nonparametric approximations to the marginal distributions and the copula are also being considered.

ACKNOWLEDGMENTS

This work was partially supported by the American International Group (AIG) under the “Climate Informed Global Flood Risk the Assessment” project. U. Lall was also supported by an IPA from the U.S. Army Corps of Engineers.

REFERENCES


30. Lall U. Will hydrologists learn from the world around them? Empiricism, models, uncertainty and stationarity. AGU Fall Meeting Abstracts, 2010; 1:01.


