Random Sampling from a Truncated Multivariate Normal Distribution

J. A. BRESLAW
Department of Economics, Concordia University
1455 de Maisonneuve West, Montreal, P.Q. H3G-1M8, Canada

(Received and accepted November 1993)

Abstract—This note critiques a procedure suggested by Ahmad and Abd-El-Hakim for drawing random samples from a truncated bivariate normal distribution. An alternative methodology using the Gibbs sampler is described. The techniques can be used for sampling from a truncated multivariate normal distribution. The GAUSS code is included as an appendix.

Keywords—Gibbs sampler, Multivariate normal distribution, Truncated distribution, Simulation methods.

1. INTRODUCTION

Recently, Ahmad and Abd-El-Hakim [1] described a simple algorithm for sampling random variables from a truncated standard normal distribution for both the univariate and bivariate case. The univariate case uses the Tadikamalla and Johnson [2] acceptance/rejection methodology, which is standard. For the bivariate case with correlation $\rho$, we require

$$K + N^2(0, \rho, \theta, \theta)$$

such that for observation $i$,

$$\alpha_i^- \leq Y_i < \alpha_i^+,$$

where $\alpha_i^-$ and $\alpha_i^+$ are $2 \times 1$ vectors of lower and upper truncation points, respectively. Let $\{X_1, X_2\}$ be two independent samplings from a univariate normal distribution. The method utilized by Ahmad and Abd-El-Hakim is to first sample $X_1$ unconditionally from the univariate truncated normal distribution with truncation points $\{\alpha_1^-, \alpha_1^+\}$. $Y_1$ is set equal to $X_1$. $Y_2$ is derived from the univariate conditional distribution, conditional on $Y_1$. Thus, $Y_2 \sim N_1(\rho Y_1, \sqrt{1 - \rho^2})$.

Hence,

$$\alpha_2^- \leq Y_2 < \alpha_2^+, \quad \frac{\alpha_2^- - \rho Y_1}{\sqrt{1 - \rho^2}} \leq X_2 < \frac{\alpha_2^+ - \rho Y_1}{\sqrt{1 - \rho^2}}.$$  (3)

Hence, $X_2$ is sampled from the univariate distribution with delineation points given in (3), $Y_2$ is set to $\rho X_1 + \sqrt{1 - \rho^2} X_2$.

$Y = \{Y_1, Y_2\}$ has the required covariance structure, and satisfies the given truncation points. Unfortunately, $Y$ does not exhibit the required distribution. To see this, we note that the first vector, $Y_1$ is identical to $X_1$, where $X_1$ is just a draw from a univariate normal between $\alpha_1^-$ and $\alpha_1^+$. Hence, this draw is independent of $Y_2$, which violates the true multivariate nature of $Y$. 
On the other hand, \( Y_2 \) will be drawn conditional on \( Y_1 \); the effect is to introduce a structure that is not present in the true truncated distribution. To illustrate this point, consider a bivariate distribution with correlation \( \rho = 0.9 \); this is shown in Figure 1. Now consider a drawing from the northwest quadrant, corresponding to the region \( \{ Y_1 \leq 0, Y_2 \geq 0 \} \). This is shown in Figure 2 for a sample of 2,000 based on a “brute force” rejection/acceptance basis—that is, we sample from \( Y \sim N_k(0, \Omega) \), and select the first 2,000 observations that fall in this quadrant. Figure 3 shows a 2000 sampling for the same quadrant using the Ahmad and Abd-El-Hakim (A & A) methodology. As can be seen, the value of \( Y_1 \) is, on average, far too large for a sample from \( N_2(0, \Omega) \), and as a consequence, \( Y_2 \) is in general too small. Figure 4 on the other hand, which uses the Gibbs sampler\(^1\) described below, has a distribution very similar to Figure 2.

\[ \text{Figure 1. Untruncated BNDF.} \]

\[ \text{Figure 2. TBNDF: brute force.} \]

2. A GIBBS SAMPLER

A method for sampling from a multivariate normal distribution has been proposed by Hajives-siliou and McFadden [3]. Consider the problem of drawing a sample from a truncated multivariate normal distribution of dimension \( k \). That is, let \( \eta \sim N_k(0, \Omega) \), and that for observation \( i \), \( \eta_i \) occurs in a cell with known upper and lower delineations \( (\alpha_i^+, \alpha_i^-) \). We require a sample, \( \tilde{\eta} \).

\(^1\)The number of rounds \( J \) was set to 5.
drawn from the same distribution, such that \( \hat{\eta}_i \) falls within the same cell. This is achieved using the Gibbs sampler, which was introduced by Geman and Geman [4] and Tanner and Wong [5]. This sampler relies on the fact that iterative recursive sampling from conditional distributions results in a sequence of random variables which converge in distribution to the joint distribution. Thus, a sample from a truncated multivariate normal distribution can be derived from a recursive procedure using draws from a univariate truncated normal distribution which itself can be drawn smoothly from a uniform distribution. This procedure has been used by Hajivassiliou and McFadden [3] to simulate logarithmic derivatives of rectangular probability integrals, by Hajivassiliou, McFadden and Rudd [5] to simulate multivariate normal orthant probabilities, and by McCulloch and Rossi [7] for a Bayesian analysis of the MNP model.

**Lemma 1.** Let \( \hat{\eta}^0 \in B \) be any initial feasible sample. A recursive process is used consisting of \( J \) rounds, \( j = 1, \ldots, J \). In each round, \( j \), the \( k \) draws from a truncated univariate distribution are made for each of \( \hat{\eta}_i^j, i = 1, \ldots, k \) in turn. At step \( i \) in round \( j \), \( \hat{\eta}_i^{j-1} \) and \( \hat{\eta}_i^{j-1} \) have been chosen. Define:

\[
\hat{\eta}_i^j = \mu_i + \sigma_i \Phi^{-1} \left( (\Phi(Z^+) - \Phi(Z^-))\tau + \Phi(Z^-) \right),
\]

where:
\( \Phi \) is the univariate normal distribution function
\( \Phi^{-1} \) is the inverse univariate normal distribution function
\( \tau \) is the uniform density function over the range \([0,1]\)
\( \mu_{ij} \) is the conditional mean of \( \eta_i^j \) at step \( i \) in round \( j \)
\( \sigma_i \) is the conditional standard deviation of \( \eta_i \)

\[
\begin{align*}
Z^+ &= (\hat{\alpha}_i^+ - \mu_{ij}) / \sigma_i \\
Z^- &= (\hat{\alpha}_i^- - \mu_{ij}) / \sigma_i \\
\end{align*}
\]

Then the limiting distribution of \( \eta \) is \( N_k(0, \Omega) \).

\textbf{PROOF.} See [3].

We note that if \( k = 1 \), we have the univariate case; in this situation, the recursive process is not required, since the conditional distribution is the joint distribution. Equation (4) provides a direct method for sampling from a univariate normal truncated distribution, without the need for acceptance/rejection, and is easily programmable, since both \( \Phi \) and \( \Phi^{-1} \) are standard functions, as is \( \tau \). \( \eta^0 \) can be any initial feasible sample, and is usually drawn from \( k \) independent draws from the truncated univariate distribution; consequently, the initial sample will not have the correct covariance structure. While asymptotically it makes no difference, a better starting point is the Ahmad and Abd-El-Hakim estimator, and since the extra computational cost is negligible, this would seem to be a worthwhile procedure.

\textbf{REFERENCES}

This appendix provides the GAUSS code for sampling from a multivariate truncated normal density function using the Gibbs sampler.

```gauss
//**************************************************************
// RNDTN.SRC - Creates a matrix of (pseudo) random variables distributed
// truncated multivariate normal distribution using the Gibbs sampler.
// Purpose: Creates a matrix of (pseudo) random variables distributed
// truncated multivariate normal distribution.
// Format: y = RNDTN(xh,xl,mu,omega);
// Input xh Kx1 or KxN matrix, the upper limits of the K-variate
// normal density function
// xl Kx1 or KxN matrix, the lower limits of the K-variate
// mu Kx1 or KxN matrix, means of the K-variate normal
// density function
// omega KxK symmetric, positive definite covariance matrix
// of the K-variate normal density function.
// _rtnrep scalar, the number of Gibbs replications (default = 20).
// _rtnpnt scalar, 1 - print iteration number (default = 0).
// Output: y 1xk vector or Kxk matrix of random numbers derived
// from the multivariate normal density function between
// the limits given by xh and xl.
// Remarks:
// The Gibbs Sampler is based on a Markov chain that utilizes
// univariate truncated normal densities to construct conditional
// variates, and has the truncated multivariate normal as its
// limiting distribution. See V. Hajivassiliou (1992), "Simulation
// Estimation Methods for Limited Dependent Variable Models" in
// Handbook of Statistics, Vol 11 (Econometrics), C.S. Maddala,
// Example:
// let xh = 2 1;
// let xl = 0 -1;
// omega[2,2] = 1 .0 .0 1;
// let mu[5,2] = 3 3 3 0 0
// 0 0 0 0 0;
// z = rndtn(xh,xl,mu,omega);
// This simulates the bivariate truncated normal density
// function over the specified delineation range for five
// observations with specified means.
//**************************************************************

external matrix _rtnrep;
external matrix _rtnpnt;
proc rndtn; endp;
proc cdnval; endp;
proc incdfn; endp;
```

---

Truncated Multivariate Normal Distribution

APPENDIX

GAUSS CODE
proc rndtn(xh,xl,mu,omega);
   /* generates a random variate from the truncated multivariate normal pdf */
   using a Gibbs sampler */
   local n, k, ysim, iter, j, cmu, csig, dl, dh;
   n = maxc(cols(mu)|cols(xh)|cols(xl));
   k = rows(omega);
   ysim = zeros(n,k);
   iter = 1; do until iter > _rtnrep;
      if (_rtnpt); screen on;
      locate 8,10; "iteration " iter;
      endif;
      j = 1; do until j > k;
         if (iter == 1);
            cmu = mu[j,1];
            csig = sqrt(omega[j,j]);
         else;
            {cmu,csig} = cdnval(ysim,mu,omega,j);
         endif;
         dh = cdfn((xh[j,1]-cmu)./csig);
         dl = cdfn((xl[j,1]-cmu)./csig);
         ysim[,j] = cmu + csig*incdfn((dh-dl).*rndu(n,1) + dl);
         j = j+1;
      endo;
      iter = iter + 1;
   endo;
   retp(ysim);
endp;
/**------------------------------------------------------------------*/
proc (2) = cdnval(ysim,mu,omega,j);
   /* returns the univariate conditional mean and variance */
   local n, k, ix, sig11, sig21, sigcdn, mucdn;
   n = cols(mu); k = rows(omega);
   ix = seqa(1,1,k);
   ix[1] = miss(0,0);
   ix = packr(ix);
   sig11 = omega[ix,ix];
   if (k == 1); retp(mu',sig11); endif;
   sig21 = omega[ix,1];
   isig22 = inv(omega[ix,ix]);
   sigcdn = sqrt(sig11 - sig21*isig22*sig21);
   mucdn = mu[1,1] + (ysim[,ix] - mu[ix,1])*isig22*sig21;
   retp(mucdn,sigcdn);
endp;
/**------------------------------------------------------------------*/
proc incdfn(p);
   /* Calculates the inverse of a standard normal CDF given a value between zero and one. */
   local v,t,n,d,t2,tlow,thgh;
   tlow = 1e-308; thgh = 1 - 1e-16;
   p = tlow.*(p .< tlow) + p.*(p .ge tlow);
   p = thgh.*(p .> thgh) + p.*(p .le thgh);
   t=sqrt(-2*ln(abs((p.>0.5)-p)));
   t2=t-2;
   n=2.515517+0.802853*t+0.010328*t2;
   d=1+1.432786*t+0.189269*t2+0.001308*t-3;
   x=t-(n./d);
   retp(x);
endp;