Title

mvnlattice( ) — Lattice rule approximation of the MVN probability function

Syntax

real rowvector `mvlattice(`*real vector a*`,` *real vector b*`,` *real matrix R* `[,` *real scalar maxn* `[,` *eps*`]])`

where inputs are

a: real vector of length n (lower bounds of integration)
b: real vector of length n (upper bounds of integration)
R: an $n \times n$ real matrix (variance-covariance or correlation)
maxn: real scalar (maximum number of quadrature points, default 1.0e4)
eps: real scalar (desired absolute error of integration, default 1.0e-4)

Here, n is the dimension of the MVN distribution. Missing values denote $-\infty$ in *a* and $+\infty$ in *b*.

Description

This routine numerically approximates the integral of the multivariate normal (MVN) density function using a rank-1 lattice rule, a form of constant weight quadrature. The lower and upper bounds of integration are contained in the vectors *a* and *b*, respectively. Missing values denote $-\infty$ in *a* and $+\infty$ in *b*. The matrix *R* contains either the MVN variance-covariance or its correlation matrix. The optional scalar *maxn* sets the maximum number of points to use in the lattice rule; the default is 1.0e4. Scalar *eps* sets the targeted absolute error of the quadrature; the default is 1.0e-4.

Remarks

Remarks are presented under the headings

Introduction
Error analysis
Choice of lattice points

Introduction

`mvnlattice` numerically approximates the integral of the multivariate normal density

$$F(\mathbf{a},\mathbf{b}|\mathbf{R}) = \frac{1}{(2\pi)^{\frac{n}{2}}\,|\mathbf{R}|^{\frac{1}{2}}} \int_{a_1}^{b_1} \cdots \int_{a_n}^{b_n} \exp\left(-\tfrac{1}{2}\mathbf{y}'\mathbf{R}^{-1}\mathbf{y}\right) d\mathbf{y}$$

using a rank-1 lattice rule, an equal weight quadrature rule for smooth one-periodic functions over the unit hypercube. The form of the rank-1 lattice rule used is

$$Q(f) = \frac{1}{N}\sum_{j=0}^{N-1} f\left(\left\{\frac{j}{N}\mathbf{z}\right\}\right),$$

where $\mathbf{z}$ is an integral vector of length n that has no common factor with N (a prime), and $\{x\}$ denotes the fractional part of x. The components of $\mathbf{z}$ come from the one parameter Korobov sequence

$$\mathbf{z}(\ell) = \left(1, \ell, \ell^2 \text{mod N}, \ldots, \ell^{n-1} \text{mod N}\right),$$

where the value of ℓ is chosen such that the points generated by $\left\{\frac{j}{N}\mathbf{z}\right\}$ are "good lattice points" in the sense that the bound on the error of integration converges to zero at an appropriately fast rate as N gets large. See Chapter 4 of of Sloan and Joe (1994) for a discussion of the method of good lattice points.

The computational algorithm for transforming the integral over a unit hypercube is taken from Genz (1992) and his discussion is repeated here. The algorithm is carried out by first taking the Cholesky factorization of the variance-covariance $\mathbf{R} = \mathbf{T}\mathbf{T}'$ and making the change of variables $\mathbf{y} = \mathbf{T}\mathbf{x}$ so that $d\mathbf{y} = |\mathbf{T}|d\mathbf{x} = |\mathbf{R}|^{\frac{1}{2}}d\mathbf{x}$. The bounds of integration can be rewritten as $\mathbf{a} \leq \mathbf{T}\mathbf{x} \leq \mathbf{b}$ so that $\frac{a_1}{t_{11}} \leq x_1 \leq \frac{b_1}{t_{11}}$ and

$$\frac{\left(a_i - \sum_{j=1}^{i-1} t_{ij}x_j\right)}{t_{ii}} \leq x_i \leq \frac{\left(b_i - \sum_{j=1}^{i-1} t_{ij}x_j\right)}{t_{ii}},$$

for $i = 2, \ldots, n$. The transformation $x_i = \Phi^{-1}(v_i)$, where $\Phi(\cdot)$ is the standard normal distribution function, gives new bounds $c_1 = \Phi\left(\frac{a_1}{t_{11}}\right)$, $d_1 = \Phi\left(\frac{b_1}{t_{11}}\right)$ and for $i = 2, \ldots, n$

$$c_i(v_1, \ldots, v_{i-1}) = \Phi\left(\frac{a_i - \sum_{j=1}^{i-1} t_{ij}\Phi^{-1}(v_j)}{t_{ii}}\right)$$
$$d_i(v_1, \ldots, v_{i-1}) = \Phi\left(\frac{b_i - \sum_{j=1}^{i-1} t_{ij}\Phi^{-1}(v_j)}{t_{ii}}\right)$$

Making the substitution and noting that $dx = \dfrac{dv}{\phi(\Phi^{-1}(v))}$, where $\phi(x)$ is the standard normal density function, the integral simplifies to, $F(\mathbf{a}, \mathbf{b}) = \displaystyle\int_{c_1}^{d_1}\int_{c_2(v_1)}^{d_2(v_1)} \cdots \int_{c_p(v_1,\ldots,v_{n-1})}^{d_p(v_1,\ldots,v_{n-1})} d\mathbf{v}$, at the cost of complicating the integration region. Genz (1992) solves this by substituting $v_i = c_i + u_i(d_i - c_i)$ yielding $F(\mathbf{a}, \mathbf{b}) = (d_1 - c_1)\displaystyle\int_0^1 (d_2 - c_2)\int_0^1 \cdots (d_n - c_n)\int_0^1 d\mathbf{u}$.

Iterating over $j = 0, \ldots, N-1$, the lattice rule is carried by first generating the n points of the j-th component of the lattice $\mathbf{u}_j = \left\{\frac{j}{N}\mathbf{z}\right\} = (u_{1j}, u_{2j}, \ldots, u_{nj})'$ and computing c_1, d_1, and $g_1 = d_1 - c_1$. Then for $i = 2, \ldots, n$ compute c_i, d_i using the formulas above, then $x_i = \Phi^{-1}\left(d_i u_{ij} + c_i(1 - u_{ij})\right)$ and $g_i = (d_i - c_i)g_{i-1}$. Denoting $f(\mathbf{u}_j) = g_n$ the lattice rule then can be expressed as $Q(f) = \frac{1}{N}\sum_{j=0}^{N-1} f(\mathbf{u}_j)$.

Commonly called the GHK simulator (Geweke, 1989; Hajivassiliou and McFadden, 1998; and Keane, 1994), most multinomial probit estimators use this technique with all $a_i = -\infty$ and an infinite sequence; such as pseudo-random uniform, Halton, or Hammersley sequence.

Error analysis

The algorithm used by `mvnlattice` for error analysis is taken from Genz and Brez (2002, 965-967), where the lattice is shifted using uniform pseudo-random deviates. For the lattice rule

$$Q(f) = \frac{1}{N}\sum_{j=0}^{N-1} f(\mathbf{u}_j),$$

with $\mathbf{u}_j = \left\{\frac{j}{N}\mathbf{z}\right\}$, shifting is performed by adding a psuedo-uniform random vector ϵ_i

$$Q_i(f) = \frac{1}{N}\sum_{j=0}^{N-1} f(\{\mathbf{u}_j + \epsilon_i\})$$

so that $Q_i(f)$ has expectation equal to F. The shifting procedure provides error analysis on the quadrature by generating p, say, uniform vectors and computing the mean and variance of $Q_i(f)$, $i = 1, \ldots, p$, $\bar{P} = (\sum_i^p Q_i(f))/p$ and $v = (\sum_i^p (Q_i(f) - \bar{P})^2)/(p(p-1))$. The computations are repeated with different primes (increasing in size), N_k, computing $\bar{P}_k$ and v_k, for $k = 1, \ldots$, and updating the overall integration estimate on the k-th repetition using,

$$P_k = \frac{P_{k-1} + \sigma_{k-1}(\bar{P}_k - P_{k-1})}{\sigma_{k-1} + v_k},$$

where $\sigma_k = \sigma_{k-1}v_k/(\sigma_{k-1} + v_k)$. A real scalar, w, is chosen so that we will be $100(1 - 1/w^2)\%$ confident that the true absolute error will be less than $w\sqrt{\sigma_k}$. The algorithm repeats for $k = 1, \ldots$, until $w\sqrt{\sigma_k}$ is less than the desired error or $\sum_k pN_k$ exceeds the maximum number of points. `mvnlattice` uses $w = 10$ and $p = 5$. This technique is also described in Sloan and Joe (1994).

The periodizing transform, $\widetilde{\mathbf{u}}_{ji} = |2\{\mathbf{u}_j + \epsilon_i\} - 1|$, is also adopted from Genz and Bretz (1992) since the lattice rule has better convergence properties for periodic integrands. An antithetic point is gained using $1 - \widetilde{\mathbf{u}}_{ji}$ effectively doubling the number of quadrature points and reducing the error. To further improve accuracy, the wider intervals for the bounds of integration, $(\mathbf{b} - \mathbf{a})$, are pivoted to the inside of the integration. The corresponding rows and columns of $\mathbf{R}$ matrix are pivoted as well.

Choice of lattice points

A criterion for "good lattice points" is derived for the class of functions that are one periodic and have a certain smoothness. The error for the lattice rule, $Q(f;\mathbf{z},N)$, using integral vector $\mathbf{z}$ and prime N is $|Q(f;\mathbf{z},N) - \int f|$, where $\int f$ is the integral of f. This error can be expressed as the sum of the Fourier coefficients for f whose absolute values are bounded. See Chapter 4 of Sloan and Joe (1994) and the references therein for details. For this class of functions a bound for the rank-1 lattice rule error can be achieved if $f = c\ g_\alpha$, for some constant $c > 0$, $g_\alpha(\mathbf{x}) = \prod_{k=1}^n G_\alpha(x_k)$. The function $G_\alpha(x)$, for α an even integer, can be expressed in terms of the Bernoulli polynomial of order α, B_α. $Q(g_\alpha;\mathbf{z},N)$ is then a worst case for the lattice rule using $\mathbf{z}(l)$. `mvnlattice` uses $\alpha = 2$, so $B_2(x) = x^2 - x + 1/6$ and $G_2(x) = 1 + 2\pi^2(x^2 - x - 1/6)$.

`mvnlattice` uses the rank-1 Korobov lattice generator $\mathbf{z}(\ell) = (1, \ell, \ell^2 \mathrm{mod}\ N, \ldots, \ell^{N-1}\mathrm{mod}\ N)$ such that the integer $\ell, 1 \le \ell \le (N-1)/2$, minimizes $Q(g_2;\mathbf{z}(\ell),N)$. Note that the search is only over $[1,(N-1)/2]$ since $g_2(\{j\mathbf{z}(\ell)/N\}) = g_2(\{j\mathbf{z}(N-\ell)/N\})$.

A Mata utility routine `rank1coef` carries out this search. The coefficients for dimensions $n = 2, \ldots, 20$ and prime $N = 307$ are

$$\ell = (119, 75, 48, 133, 97, 41, 153, 2, 108, 20, 20, 20, 20, 2, 100, 153, 41, 153, 39).$$

Recursively, we generate $z_k = \ell z_{k-1} \mathrm{mod}\ N, k = 2, \ldots, n$ and $z_1 = 1$, so for $n = 5$ we have $\mathbf{z} = (1, 133, 190, 96, 181)$. The following Mata code will generate the lattice points:

```
: N = 307
: n = 5
: l = 133
: z = 1
: for (i=2: i<=n; i++)
>   z = (z,mod(z[i-1]*l,N))
>
: z = z:/N
: X = J(N,n,0)
: x = J(1,n,0)
: for (i=2; i<=N; i++)
>   x = mod(x:+z,1)
>   X[i,.] = x
>
```

The matrix scatter plot displayed in Figure 1 shows the lattice points generated for the five dimensions. Note how evenly the points cover the unit interval.

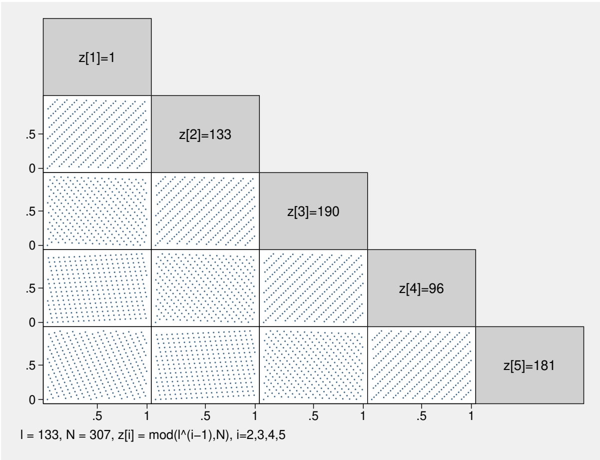

Figure 1. Rank-1 lattice points for $N = 309$, $n = 5$, and $\ell = 119$.

Conformability

`mvnlattice(`*a*, *b*, *R*, *maxn*, *eps*`)`

input:

a:	*1 x n*, or *n x 1*
b:	*1 x n*, or *n x 1*
R:	*n x n*
maxn:	*1 x 1*
eps:	*1 x 1*

output:

result:	*1 x 3*

Diagnostics

`mvnlattice(`*a*, *b*, *R*`)` returns a vector of length 3 in which the first element is the estimated integral, the second element its estimated absolute error of the integral, and the last element contains the number of points used.

Use `uniformseed` in order to reproduce the results.

▷ Example 1

In the examples below note that with duplicate calls to `mvnlattice` slightly different estimates are given due to the random shift of the lattice. To duplicate a result, use `uniformseed` to set the uniform random number generator seed.

```
: R = (1,.2,.3,-.4\.2,1,-.2, .5\.3,-.2,1,.3\-.4,.5,.3,1)
: mvnlattice((-.5,-.6,-1,-1.5),(2,0,1,.5),R)
                  1              2              3
    +----------------------------------------------+
  1 |    .0914858347    .0000361571           4630 |
    +----------------------------------------------+
: mvnlattice((-.5,-.6,-1,-1.5),(2,0,1,.5),R)
                  1              2              3
    +----------------------------------------------+
  1 |   .0914863778    .0000382281           4630  |
    +----------------------------------------------+
: mvnlattice((-.5,-.6,-1,-1.5),(2,0,1,.5),R,200000,1.0e-6)
                  1              2              3
    +----------------------------------------------+
  1 |   .0914875375    6.43827e-07         170260  |
    +----------------------------------------------+
: mvnlattice((-.5,-.6,-1,-1.5),(2,0,1,.5),R,200000,1.0e-6)
                  1              2              3
    +----------------------------------------------+
  1 |   .0914872398    3.69675e-07         170260  |
    +----------------------------------------------+
```

References

Fang, K.T. and Y. Wang 1994. *Number-Theoretic Methods in Statistics* London:Chapman & Hall.

Genz, A. 1992. Numerical computation of multivariate normal probabilities. *Journal of Computational and Graphical Statistics* 1: 141–149.

Genz, A. and F. Bretz 2002. Comparison of methods fot the computation of multivariate t probabilities. *Journal of Computational and Graphical Statistics* 11: 950–971.

Geweke, J. 1989. Bayesian inference in econometric models using Monte Carlo integration. *Econometrica* 57: 1317–1339.

Hajivassiliou, V. and D. McFadden 1998. The method of simulated scores for the estimation of LDV models. *Econometrica* 66: 863–896.

Keane, M. 1994. A computationally practical simulation estimator for panel data. *Review of Economics and Statistics* 76: 648–672.

Sloan, I.H. and S. Joe 1994. *Lattice Methods for multiple Integration.* Oxford University Press