Journal of Computational and Applied Mathematics 16 (1986) 255–258
North-Holland

The numerical evaluation of certain multivariate normal integrals

Alan GENZ
Computer Science Department, Washington State University, Pullman, WA 99164, U.S.A.

David K. KAHANER
Scientific Computing Division, National Bureau of Standards, Gaithers, MD 20899, U.S.A.

Received 6 February 1986

Abstract: We show that a multivariate normal integral with tridiagonal covariance matrix can be computed efficiently using iterated integration.

Keywords: Numerical quadrature, normal integrals, multivariate normal integrals.

1. Introduction

An integral that occurs frequently in many applications is the multivariate normal integral

$$I(\boldsymbol{t}) = C\int_{-\infty}^{t_n}\int_{-\infty}^{t_{n-1}}\cdots\int_{-\infty}^{t_1}\exp\left(-\tfrac{1}{2}\boldsymbol{x}^{\mathrm{T}}M^{-1}\boldsymbol{x}\right)\,\mathrm{d}\boldsymbol{x}.$$

Here, $\boldsymbol{x}$ and $\boldsymbol{t}$ are n-tuples,

$$\boldsymbol{x}^{\mathrm{T}} = (x_1, x_2, \ldots, x_n),$$

M^{-1} is an n by n symmetric positive definite matrix (called the *covariance matrix*) and C is a constant defined so that $I(+\infty) = 1$,

$$C = \det(M)^{-1/2}(2\pi)^{-n/2}.$$

When $n = 1$ the evaluation of I is equivalent to computation of the error function, $\operatorname{erf}(t)$, for which rapid and reliable algorithms and software exist [1]. For $n > 1$ there are no general procedures which work as well, and for n greater than 3 or 4, evaluation of I can require large amounts of computer time. In this paper we describe a method which is applicable to the special case when the matrix M^{-1} is tridiagonal. There are no restrictions on any of the limits of integration but the most commonly occuring cases involve integration over a semi-infinite hyper-rectangle and we prefer to present the problem in this manner. The method was motivated by a problem from a statistician involving the distribution of certain order statistics, and is illustrated in Section 3.

2. Tridiagonal covariance matrix

If the covariance matrix was diagonal, then $I(t)$ could be written as an iterated integral, and each one dimensional integral could be done independently by using an *erf* algorithm. In that case the total amount of work would be n times the effort for a single one dimensional computation. On the other hand, if we treat $I(t)$ as an n-dimensional integral and apply, say, a product quadrature rule, then the effort will be exponential in n, i.e., of the form p^n, where p is a measure of the one dimensional work. For the case considered here we show that $I(t)$ can still be written as an iterated integral. The individual integrals can no longer be computed using *erf*, nor can they be done independently, but the total work will be shown to remain multiplicative in the dimensionality, of the form np^2.

We assume that the covariance matrix is tridiagonal (but not necessarily symmetric) and of the form

$$M^{-1} = \begin{pmatrix} a_1 & c_1 & 0 & & \cdots & 0 \\ b_1 & a_2 & c_2 & 0 & \cdots & 0 \\ 0 & b_2 & a_3 & c_3 & \cdots & 0 \\ \vdots & & & & & \vdots \\ 0 & \cdots & & & & c_{n-1} \\ 0 & \cdots & & & b_{n-1} & a_n \end{pmatrix}.$$

In this case we have

$$\boldsymbol{x}^{\mathrm{T}} M^{-1} \boldsymbol{x} = a_1 x_1^2 + (c_1 + b_1) x_1 x_2 + a_2 x_2^2 + \cdots + (c_{n-1} + b_{n-1}) x_{n-1} x_n + a_n x_n^2.$$

If we let

$$s_i = a_i + b_i,$$

then $I(t)$ becomes

$$I(\boldsymbol{t}) = C \int_{-\infty}^{t_n} \exp\left(-\tfrac{1}{2} a_n x_n^2\right) \int_{-\infty}^{t_{n-1}} \exp\left(-\tfrac{1}{2}\left(a_{n-1} x_{n-1}^2 + s_{n-1} x_{n-1} x_n\right)\right)$$
$$\cdots \int_{-\infty}^{t_2} \exp\left(-\tfrac{1}{2}\left(a_2 x_2^2 + s_2 x_2 x_3\right)\right) \int_{-\infty}^{t_1} \exp\left(-\tfrac{1}{2}\left(a_2 x_1^2 + s_1 x_1 x_2\right)\right) \mathrm{d}\boldsymbol{x}.$$

We next define the sequence of functions

$$f_1(t_1, x_2),\ f_2(\boldsymbol{t}_2, x_3), \ldots, f_{n-1}(\boldsymbol{t}_{n-1}, x_n) \quad \text{with } \boldsymbol{t}_i = (t_1, t_2, \ldots, t_i),$$

by

$$f_1(t_1, x_2) = \int_{-\infty}^{t_1} \exp\left(-\tfrac{1}{2}\left(a_2 x_1^2 + s_1 x_1 x_2\right)\right) \mathrm{d}x_1,$$

and

$$f_i(\boldsymbol{t}_i, x_{i+1}) = \int_{-\infty}^{t_i} \exp\left(-\tfrac{1}{2}\left(a_i x_i^2 + s_i x_i x_{i+1}\right)\right) f_{i-1}(\boldsymbol{t}_{i-1}, x_i)\, \mathrm{d}x_i, \quad i = 2, 3, \ldots, n-1.$$

The integral $I(t)$ then becomes

$$I(\boldsymbol{t}) = C \int_{-\infty}^{t_n} \exp\left(-\tfrac{1}{2} a_n x_n^2\right) f_{n-1}(\boldsymbol{t}_{n-1}, x_n)\, \mathrm{d}x_n.$$

We have thus established that $I(\boldsymbol{t})$ can be written as an iterated integral, and can therefore be evaluated numerically as an iterated sequence of one dimensional integrals.

There are a number of different methods and rules that could be used for the one dimensional integrals, including automatic and adaptive algorithms; a good discussion of these is provided in the book [2]. To illustrate the general result we assume that we have an appropriate integration rule for each of the integrals that define the functions f_i, and that these rules are given in the form

$$f_i(\boldsymbol{t}_i, x_{i+1}) \approx \sum_{j=1}^{p_i} w_{ij} \exp\left(-\tfrac{1}{2}\left(a_i x_{ij}^2 + s_i x_{ij} x_{i+1}\right)\right) f_{i-1}(\boldsymbol{t}_{i-1}, x_{ij}),$$

where

$$w_{i1}, w_{i2}, \ldots, w_{ip_i} \quad \text{and} \quad x_{i1}, x_{i2}, \ldots, x_{ip_i}$$

are the quadrature weights and nodes, respectively, for the integral defining f_i. We also assume that we have an integration rule for the final integral in the form

$$I(\boldsymbol{t}) \approx C \sum_{j=1}^{p_n} w_{nj} \exp\left(-\tfrac{1}{2} a_n x_{nj}^2\right) f_{n-1}(\boldsymbol{t}_{n-1}, x_{nj})\, \mathrm{d}x_n,$$

with weights and nodes w_{nj} and x_{nj}. The approximate computation of $I(\boldsymbol{t})$ requires the approximate evaluation of the functions $f_{i-1}(\boldsymbol{t}_{i-1}, x_i)$ at the points x_{ij} for $j = 1, 2, \ldots, p_i$ and $i = 2, 3, \ldots, n$, and the final rule computation for $I(\boldsymbol{t})$. The function f_1 can be approximated with p_1 evaluations for each fixed x_2, independent of any other variables. Thus if f_2 is approximated by a p_2 point rule, f_1 requires $p_1 p_2$ evaluations for each fixed x_3, thus a total of $p_2 p_3$ evaluations, independent of the other variables. The total time complexity for the approximate evaluation of $I(\boldsymbol{t})$ is therefore

$$\mathrm{O}(p_1 p_2 + p_2 p_3 + \cdots + p_{n-1} p_n + p_n).$$

This should be compared with

$$\mathrm{O}(p_1 p_2 \cdots p_n),$$

which is the time complexity for the approximate evaluation of $I(\boldsymbol{t})$ if an n-dimensional product rule is used directly.

3. An example

For the problem mentioned in Section 1, the matrix M was defined by

$$m_{ij} = \begin{cases} 4i(1 - j/(n+1)) & \text{if } i > j, \\ 4j(1 - i/(n+1)) & \text{otherwise.} \end{cases}$$

The dimensionality of the problem n, was 9, and region of integration had all values of t_i equal, i.e., $\boldsymbol{t} = (t, t, \ldots, t)$.

In this case the matrix M has a tridiagonal inverse which is given exactly by,

$$\boldsymbol{a} = (a_1, \ldots, a_9) = (6.48, 20.48, 35.28, 46.08, 50, 46.08, 35.28, 20.48, 6.48),$$

and

$$\boldsymbol{b} = \boldsymbol{c} = -(5.76, 13.44, 20.16, 24, 20.16, 13.44, 5.76).$$

After some experimentation, a cutoff value of -2.5 was chosen to replace the infinite lower limits on the integrals and a 2 point Gauss rule compounded 40 times was used for all of the integration rules ($p_i = 80$). This gave final results accurate to approximately 5 decimal digits for $t = 0, 0.1, \ldots, 1.5$, using a few seconds of time on a VAX 11/750. We were able to verify the accuracy by repeating the computations with different numbers of points. Additionally, for this problem, it can be shown that

$$I(\mathbf{0}) = 1/(n+1),$$

and we also compared our results against this value. Our earlier attempts to evaluate the integral (using a well known multidimensional adaptive quadrature routine) without taking advantage of the special structure of the covariance matrix required a couple of hours of VAX time to achieve only 1 or 2 digits accuracy for $t < 0.75$. the results for larger t were not even accurate in the first digit. Of course, the statistician was only interested in values of t which included 95% of the area, and these were all larger than $t = 1$.

References

[1] International mathematical and Statistical Library (IMSL), Sixth Floor, NBC Building, 7500 Bellaire Boulevard, Houston Texas, 77036-5085.

[2] P.J. Davis and P. Rabinowitz, *Methods of Numerical Integration* (Academic Press, New York, 2nd ed., 1984).

[3] R.C. Milton, Computer evaluation of the multivariate normal integral, *Technometrics* **14** (4) 881–889.