高维正态概率积分中两个高斯型数值积分公式的比较 *1)

杨自强

（中国科学院计算数学与科学工程计算所）

COMPARISON OF TWO GAUSSIAN QUADRATURES IN MULTIVARIATE NORMAL INTEGRALS

Yang Ziqiang

(Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Sciences, Beijing)

Abstract

Both the classical Gauss-Hermite quadrature for $\int_{-\infty}^{\infty} f(x)e^{-x^2}dx$ and the little-known Gaussian quadrature for $\int_0^{\infty} f(x)e^{-x^2}dx$ given by Steen-Byrne-Gelbard (1969) can be used to evaluate the multivariate normal integrals. In the present paper, we compare the above quadratures for the multivariate normal integrals. The simulated results show that the efficiencies of two formulas have not the significant difference if the condition of integral is very good, however, when the dimension of integral is high or the condition of correlation matrix of the multivariate normal distribution is not good, Steen et.al. formula is more efficient. In appendix, an expanded table of Gaussian quadrature for Steen et.al. is given by the present author.

§1. 引　言

在高维正态积分估计中，有两个高斯型数值积分公式可用，一是 $\int_{-\infty}^{\infty} f(x)e^{-x^2}dx$ 的经典 Gauss-Hermite 公式，二是 $\int_0^{\infty} f(x)e^{-x^2}dx$ 的不太知名的 Steen-Byrne-Gelbard 公式 [1]. Thisted 的著作 [2] 对此有很好的介绍.

* 1996 年 9 月 6 日收到.

1) 国家自然科学基金资助项目.

1.1 经典公式 ——Gauss-Hermite 数值积分公式

Gauss-Hermite 数值积分公式是

$$\int_{-\infty}^{\infty} f(x)e^{-x^2}dx = \sum_{i=1}^{n} w_i^* f(x_i^*). \tag{1.1}$$

此式在概率积分的估计中扮演着重要的角色，式中 x_i^*，$i=1,2,\cdots,n$ 称为节点，它们是 n 阶正交多项式 $h_n(x)$ 的 n 个零点．由 $h_n(x)$ 的正交性，

$$\int_{-\infty}^{\infty} h_i(x)h_j(x)e^{-x^2}dx = 0, \quad i \neq j, \quad i,j = 0,1,2,\cdots, \tag{1.2}$$

可有如下的正交多项式序列：

$$h_0(x) = 1, \qquad h_1(x) = x,$$
$$h_{k+1}(x) = xh_k(x) - \frac{k}{2}h_{k-1}(x), \quad k = 1,2,\cdots. \tag{1.3}$$

在 (1.1) 中， 权w_i^* 由下式给出：

$$w_i^* = \frac{1}{2^{n-1}}\sqrt{\pi}\Gamma(n)/(h_n'(x_i^*)h_{n-1}(x_i^*)), \quad i = 1,2,\cdots,n. \tag{1.4}$$

熟知， Gauss-Hermite 公式有 $2n-1$ 阶的精度，即只要 $f(x)$ 是不超过 $2n-1$ 阶的多项式，则 (1.1) 式就精确成立．许多数值计算的教科书，例如 [2,3]，都能找到有关高斯型数值积分公式的数学背景，特别是 [3] 含有一个很大的 $w_i^*-x_i^*$ 表 ($n=2(1)64(4)96(8)136$).

原则上说， (1.1) 式可用于估计半无穷区间上的积分 $\int_0^{\infty} f(x)e^{-x^2}dx$. 事实上，令函数 g 是 f 关于原点的偶拓展，则有

$$\int_0^{\infty} f(x)e^{-x^2}dx = \frac{1}{2}\int_{-\infty}^{\infty} g(x)e^{-x^2}dx = \frac{1}{2}\sum_{i=1}^{n} w_i^* g(x_i^*). \tag{1.5}$$

但是 g 在原点往往只有低阶的可微性，从而上式会导致很不准确的结果，例如

$$\int_0^{\infty} xe^{-x^2}dx = \frac{1}{2}\int_{-\infty}^{\infty} |x|e^{-x^2}dx = \frac{1}{2}. \tag{1.6}$$

当使用 (1.5) 式估计时，在 $n=2,3,4$ 的条件下，相对误差竟出乎意料地分别高达 25.3%, 11.3%, 7.3%.

1.2. Steen-Byrne-Gelbard 的高斯型数值积分公式

为改进 Gauss-Hermite 公式， Steen-Byrne-Gelbard[1] 提出如下公式：

$$\int_0^{\infty} f(x)e^{-x^2}dx = \sum_{i=1}^{m} w_i f(x_i), \tag{1.7}$$

式中 x_i 是 m 阶正交多项式 $p_m(x)$ 的 m 个零点. 此处

$$\int_0^\infty p_i(x)p_j(x)e^{-x^2}dx = 0, \quad i \neq j, \quad i,j = 0,1,2,\cdots, \tag{1.8}$$

且有如下的正交多项式序列:

$$p_0(x) = 1, \quad p_1(x) = x - \frac{1}{\sqrt{\pi}},$$

$$p_{k+1}(x) = (x+\alpha_k)p_k(x) + \beta_k p_{k-1}(x), \quad k = 1,2,\cdots. \tag{1.9}$$

式中 α_k 与 β_k 有如下形式:

$$\alpha_k = -\frac{1}{2}p_k^2(0)/r_k, \quad \beta_k = -r_k/r_{k-1}, \tag{1.10}$$

$$r_k = \frac{1}{2}kr_{k-1} + \frac{1}{2}p_k(0)p_{k-1}(0), \quad k = 1,2,\cdots \tag{1.11}$$

$$r_0 = \frac{1}{2}\sqrt{\pi}.$$

(1.7) 式中的权是

$$w_i = r_{m-1}/(p'_m(x_i)p_{m-1}(x_i)), \quad i = 1,2,\cdots,m. \tag{1.12}$$

Steen 等人的公式 (1.7) 在半无穷区间 $(0,\infty)$ 上也有 $2m-1$ 阶的精度. 他们还在同一文章中导出了另一个高斯型公式 $\int_0^b f(x)e^{-x^2}dx$.

显然, 公式 (1.7) 也可以用于全区间 $(-\infty,\infty)$ 的积分. 因为

$$\int_{-\infty}^\infty f(x)e^{-x^2}dx = \int_0^\infty [f(-x)+f(x)]e^{-x^2}dx = \sum_{i=1}^m w_i[f(-x_i)+f(x_i)]. \tag{1.13}$$

即使把上式用于函数 $f(x) = |x|^{2k-1}$, $k = 1,2,\cdots,m$, 也能得到精确的结果, 但如上所述, 经典公式 (1.5) 却不能做到.

本文目的是比较 Steen 等人公式 (1.13) 与经典公式 (1.1) 在估计高维正态积分中的效率. 从表面上看, 在全区间 $(-\infty,\infty)$ 的积分中, 当 $f(x)$ 不是偶函数时, Steen 等人公式 (1.13) 需要计算 $2m$ 个被积函数值才有 $2m-1$ 阶精度, 而使用 $2m$ 个函数值的经典公式 (1.1) 却有 $4m-1$ 阶的精度. 所以, 如被积函数充分光滑宜用经典公式. 但是通过本文的大量模拟计算表明, 当积分条件十分好时, 两者没有明显的差异, 不过当积分维数较高或多维正态分布的相关矩阵条件不好时, Steen 等人的公式更为有效.

为方便应用 Steen 等人的公式, 在附录中给出由本文作者大大扩展了的 Steen 等人的 $w_i - x_i$ 表.

§2. 两个高斯型数值积分公式的比较

为比较经典的与 Steen 等人的数值积分公式 (1.1) 与 (1.13). 在计算中，前者我们可以使用 Stroud-Secrest[3] 给出的经典 $w_i^* - x_i^*$ 表或其 FORTRAN 程序计算，但后者只好使用我们自己的 $w_i - x_i$ 扩展表 $m \le 60$. 因为在原来的表中， $m \le 15$, 这对相关矩阵 R 条件不太好的高维正态积分来说显得小了些. 有关此表及其计算方法请见附录.

今引入如下记号： $N_k(\mu, \Sigma)$ 表示 k 维正态分布，其均值向量与协差阵依次是 $\mu = (\mu_1, \cdots, \mu_k)'$, $\Sigma = (\sigma_{ij})$. 为方便而不失一般性，设 $\mu_i = 0$, $\sigma_{ii} = 1$, 这时协差阵变为相关矩阵 $R = (r_{ij})$, 其密度函数和分布函数分别记为 $\phi_k(x_1, \cdots, x_k)$ 和 $\Phi_k(x_1, \cdots, x_k)$. 当 $k = 1$ 时，略去下标 k 简记为 $N(0,1)$, $\phi(x)$, $\Phi(x)$.

表 1 积分 (2.1) 的估计值

相关	维数	绝对误差	Steen 等人公式		经典的公式	
r_{ij}	k	eps	$n=2m$	估计值	n	估计值
0.10	12	$.5 \times 10^{-7}$	16	.77520152	12	.77520152
0.10	20	$.5 \times 10^{-7}$	14	.66798382	14	.66798382
0.20	12	$.5 \times 10^{-7}$	18	.79334386	16	.79334386
0.20	20	$.5 \times 10^{-7}$	22	.70401591	20	.70401591
0.40	12	$.5 \times 10^{-7}$	28	.83179585	30	.83179584
0.40	20	$.5 \times 10^{-7}$	34	.77168972	36	.77168973
0.50	12	$.5 \times 10^{-7}$	36	.85158725	40	.85158723
0.50	20	$.5 \times 10^{-7}$	42	.80362561	50	.80362561
0.60	12	$.5 \times 10^{-6}$	36	.8717307	42	.8717303
0.60	20	$.5 \times 10^{-6}$	42	.8346853	52	.8346850
0.80	12	$.5 \times 10^{-6}$	72	.9140427	104	.9140425
0.80	20	$.5 \times 10^{-6}$	84	.8959848	124	.8959843
0.90	12	$.5 \times 10^{-5}$	88	.937864	140	.937857
0.90	20	$.5 \times 10^{-5}$	100	.928266	156	.928250
0.95	12	$.5 \times 10^{-4}$	100	.95191	156	.95201
0.95	20	$.5 \times 10^{-4}$	112	.94646	164	.94666

下面是用前述两数值积分公式作对比计算的一些例子.

例 1. 相关系数为 $r_{ij} = a_i a_j$, $(i \ne j)$ 的多维正态积分. 此时 (见 [4] 或 [5,6]) 有

$$\Phi_k(t_1, \cdots, t_k) = \int_{-\infty}^{\infty} \Big[\prod_{i=1}^{k} \Phi\Big(\frac{t_i + a_i u}{\sqrt{1 - a_i^2}} \Big) \Big] \phi(u) du. \tag{2.1}$$

式中 $\Phi(t) = \dfrac{1}{\sqrt{2\pi}} \displaystyle\int_{-\infty}^{t} e^{-\frac{u^2}{2}} du$ 可由一个精度高而简单的表达式得到 (例如著名软件库 IMSL 的程序精度超过 14 位), 因此今后我们在计算上不再把它看作是个一维积分. 由 (1.1) 与 (1.13) 式得到的数值结果如表 1 所示. 为简单计，在对比计算中我们假设:

(i) 同一个积分内的相关系数都相等，即 $r_{ij} = r \ (i \neq j)$，于是 $a_i = \sqrt{r}$. (ii) 所有积分的上限都相同，此处取 $t_i = 2.0$. 表 1 中的节点数 n 及估计值由如下方法决定：令

$$I_n = \begin{cases} \sum_{i=1}^{n} w_i^* f(x_i^*), & n = 2(2)60(4)180, \quad \text{对经典的 } w_i^* - x_i^*, \\ \sum_{i=1}^{[n/2]} w_i[f(-x_i) + f(x_i)], & n = 2(2)60(4)120, \quad \text{对 Steen 等人 } w_i - x_i, \end{cases} \tag{2.2}$$

并假设 $n_j < n_{j+1} < n_{j+2}$ 是最早使下式成立的三个相邻节点数目

$$|I_{n_j} - I_{n_{j+1}}| < eps, \quad |I_{n_{j+1}} - I_{n_{j+2}}| < eps. \tag{2.3}$$

最后，我们取 n_{j+2} 与 $I_{n_{j+2}}$ 作为表 1 中的 n 与积分估计值. 对于一个确定的 eps, 若公式甲的 n 值小于公式乙的值时，我们说公式甲比乙好.

由表 1 可以看出：(i) 不论经典公式还是 Steen 等人公式，估计中所需的 n 和结果精度都依赖于相关系数与维数. (ii) 如果 r 十分小，经典公式稍优于 Steen 等人公式，但差异并不显著. (iii) 当 r 增大时，n 亦增大，但经典公式中的增幅大于 Steen 等人公式，从而使经典公式失去优势，当 $r = 0.40$ 时，Steen 等人公式已优于经典公式. (iv) 当 r 较大时，Steen 等人公式明显优于经典公式，n 的比率约为 2/3.

准分解假设下高维正态积分的降维算法

作者在相关矩阵 R准分解 的假设下建立了高维正态积分的降维算法 [5,6]. 粗略地说，准分解是指 R 的非对角元素中的大部分有 $r_{ij} = a_i a_j$ 形式，而其它部分有 $r_{ij} = a_i a_j + b_{ij}$ 形式，此处的 b_{ij} 为非零偏差. 这一算法使高维正态分布降为 2 或 3 维积分. 最简单的情形是只有如下一个非零偏差 b_{ij}.

$$r_{ij} = \begin{cases} a_i a_j, & i \neq j, \ (i,j) \neq (i_1, j_1), \\ a_{i_1} a_{j_1} + b_{i_1 j_1}, & i = i_1, \ j = j_1, \ i_1 \neq j_1. \end{cases} \tag{2.4}$$

此时我们有

$$\Phi_k(t_1, \cdots, t_k) = \int_{-\infty}^{\infty} g_{i_1 j_1}(u) \Big[\prod_{\substack{i=1 \\ i \neq i_1, j_1}}^{k} \Phi\Big(\frac{t_i + a_i u}{\sqrt{1 - a_i^2}} \Big) \Big] \phi(u) du, \tag{2.5}$$

式中

$$g_{i_1 j_1}(u) = \int_{-\infty}^{\infty} \Phi\Big(\frac{t_{i_1} + a_{i_1} u \pm \sqrt{|b_{i_1 j_1}| c_{i_1 j_1}} \ v}{\sqrt{1 - a_{i_1}^2 - |b_{i_1 j_1}| c_{i_1 j_1}}} \Big) \Phi\Big(\frac{t_{j_1} + a_{j_1} u + \sqrt{|b_{i_1 j_1}| / c_{i_1 j_1}} \ v}{\sqrt{1 - a_{j_1}^2 - |b_{i_1 j_1}| / c_{i_1 j_1}}} \Big) \phi(v) dv. \tag{2.6}$$

式中 “±” 号表示根式符号，即 sign $(b_{i_1 j_1})$. 上述算法把 Φ_k 简化为一个 2 维积分，其后用乘法规则便可得到估值. 说详细点便是先用 (1.1) 或 (1.13) 估计 (2.6), 接着再次使用 (1.1) 或 (1.13) 估计 (2.5).

例 2. 有一个著名的 3 维正态积分的例子，它最早由 Steck[7] 提出，其后为许多人计算过. 此例中的相关矩阵是 $R(r_{21}, r_{31}, r_{32}) = (0.7, 0.2, -0.4)$, 而上限是 $(t_1, t_2, t_3) = (1.2, 1.0, -0.5)$. 基于 Steck 近似式的改进, 作者 [5] 算出 $\Phi_3(1.2, 1.0, -0.5) = 0.22060958$. 此处我们使用上述降维算法 (2.5), (2.6) 及两个高斯型数值积分公式 (1.1), (1.13), 在不同的节点数情况下重算此积分如表 2 所示.

在这个 3 维例子中，相关矩阵的条件并不十分好，其特征值是 = (1.7329, 1.1694, 0.0977), 所以经典公式需用很多节点. 如果以获得较精确的估计值 0.2206095808 所需的最小 n 为准，则有如下的比值

$$\frac{\text{Steen 等人公式所需的 } n}{\text{经典公式所需的 } n} = \frac{60}{164} = 36.6\%.$$

值得注意的是这个比值仅仅是对一维积分而言的，而 (2.5) 是 2 维积分，所以两者的效率之比应是

$$\frac{\text{Steen 等人公式所需的计算时间}}{\text{经典公式所需的计算时间}} = \left(\frac{60}{164}\right)^2 = 13.4\%.$$

这是表明 Steen 等人公式明显优于经典公式的又一个例子.

表 2　一个著名的 3 维积分的估计值

n 或 $2m$	Steen 等人公式	经典的公式
20	.22067	.21523
30	.220611	.219322
40	.22060961	.22051490
50	.2206095804	.2206589288
60	.2206095808	.2206375011
70	.2206095808	.2206184542
80	.2206095808	.2206115779
90	不再改变	.2206098452
100		.2206095513
120		.2206095642
140		.2206095792
160		.2206095807
164		.2206095808
168		.2206095808
172		.2206095808
176		不再改变

例 3. 相关矩阵 R 满足 (2.4) 式的多维正态积分. 估计方法是降维法 (2.5),(2.6) 和高斯型数值积分公式 (1.1), (1.13). 本例使用乘同余法产生的随机数 (乘子： 16807, 模 Mod: $2^{31} - 1$, 初值： 370713) 来构造相关矩阵和积分上限 (在 (2.4) 式中， $a_i \sim U(-1, 1)$, $b_{ij} \sim U(-4.5, 4.5)$, $t_i \sim U(-0.5, 3.5)$). 收敛准则与例 1 同，但此处有 $n = 2(2)40(4)60(8)120$ 或 180, 而 $eps = .5 \times 10^{-5}$ 或 $.5 \times 10^{-6}$. 为简单计，我们只估计 8 维积

分，且 (2.4) 式中的非零偏差位置是 $(i_1, j_1) = (2, 1)$. 我们根据行列式 $|R|$ 值把 R 分成 9 组，每组计算所需的节点的平均数 $\bar{n}$ 列于表 3. 计算中共产生了 400 个相关矩阵，但有 37 个因 $|R| \to 0$ 而被剔除.

表 3 显示， Steen 等人公式所需的节点总数小于经典公式，其总平均数之比为 $24.2/33.8 = .72$ (对于 $eps = .5 \times 10^{-5}$) 或 $30.6/45.5 = .67$ (对于 $eps = .5 \times 10^{-6}$). 但运行时间之比为 $.72^2 = .52$ 或 $.67^2 = .45$. 对于高精度而言， Steen 等人公式更优. 此表再次显示了类似表 1 的结论. 进一步研究还可以发现，仅当正态分布的相关阵 R 条件极好时 (指 $1 - |R| < 0.5$), 经典公式才略快于 Steen 等人的公式. 但 $1 - |R| < 0.5$ 在 8 维场合是不多见的 (若设各相关系数相等 (记为 r), 则 $1 - |R| < 0.5$ 与 $|r| < 0.2$ 对应，实际模拟计算时它出现的频率仅为 7%), 并且此时两公式所需的节点数都很少，因而计算时间的差异不明显. 与此相反，在其它场合 (出现频率约 93%), 经典公式比 Steen 等人公式需要更多节点，且随着 R 矩阵条件变坏，计算时间差距迅速拉大. 总而言之，当积分条件十分好时，两公式没有明显的差异. 然而，对大多数情形而言，Steen 等人公式比经典公式更优.

表 3 估计 363 个 8 维积分的平均节点数 $\bar{n}$

$1 - \|R\|$	样本大小		Steen 等人公式的 $\bar{n}$		经典公式的 $\bar{n}$	
	频数	%	$eps = \frac{1}{2}10^{-5}$	$\frac{1}{2}10^{-6}$	$eps = \frac{1}{2}10^{-5}$	$\frac{1}{2}10^{-6}$
[.000, .500)	27	7.4%	12.7	15.3	11.8	14.6
[.500, .600)	29	8.0%	17.2	21.2	19.3	26.0
[.600, .700)	32	8.8%	16.8	20.3	18.0	22.6
[.700, .800)	43	11.9%	19.0	22.8	20.3	27.3
[.800, .900)	73	20.1%	21.0	25.8	26.2	34.7
[.900, .940)	51	14.1%	26.5	34.5	38.9	51.3
[.940, .970)	44	12.1%	29.8	38.5	47.1	63.8
[.970, .990)	35	9.6%	36.3	47.9	61.3	83.8
[.990, .999)	29	8.0%	28.6	50.2	62.8	88.8
合计	363	100.0%	24.2	30.6	33.8	45.5

附录：高斯型数值积分公式 $w_i - x_i$ 表 (扩展)

Steen-Byrne-Gelbard[1] 曾给出一个用于估计积分 $\int_0^\infty f(x)e^{-x^2}dx = \sum_{i=1}^m w_i f(x_i)$ 的 $w_i - x_i$ 表. 它也可以用于计算 $\int_{-\infty}^\infty f(x)e^{-x^2}dx = \sum_{i=1}^m w_i[f(-x_i) + f(x_i)]$. 在他们的表中， $m = 2, 3, \cdots, 15$, 这对相关矩阵 R 条件不太好的高维正态积分来说显得小了些. 为能得到较高精度的结果，本文作者把该表扩展为 $m = 2(1)60$ 且精确到 33 位. 在计算此表时，由于 (1.11) 式中的序列 $\{r_1, r_2, \cdots, r_m\}$ 在递推时误差迅速传播，例如其放大因子在 m =15,30,60 时近似为 10^7, 10^{15}, 10^{30}, 因此我们采用中国科学院计算数学与

科学工程计算研究所的任意精度软件，其中间结果字长取 120 位．但限于篇幅，本文只开列其中的一部分 ($m = 2(1)14(2)20, 24, 28, 35$, 且精确到 15 位)．当 $m \le 15$ 时，某些值的尾部数字与 Steen 等人的不完全相同，这是因为我们的结果更为精确．值得指出的是，如果 $f(x)$ 是有界函数，在本表中的每个 m 内，其最后的一些 w_i 与 x_i 值可因 w_i 的值太小而被略去．例如把 $w_i < 0.1 \times 10^{-9}$ 及其对应的 x_i 都删去，则计算多维正态积分的结果与使用原来的表几乎没有差别 (因为大多数的差小于 0.5×10^{-10}, 且没有一个超过 0.1×10^{-9})，但我们却节省了可观的运行时间．例如当 $m = 28$ 时，仅需使用其前 21 项．

$m = 2$, Wi	Xi →	Wi	Xi
.640529179684379D+00	.300193931060839D+00	.245697745768379D+00	.125242104533372D+01
$m = 3$, Wi	Xi →	Wi	Xi
.446029770466658D+00	.190554149798192D+00	.396468266998335D+00	.848251867544577D+00
.437288879877644D−01	.179977657841573D+01		
$m = 4$, Wi	Xi →	Wi	Xi
.325302999756919D+00	.133776446996068D+00	.421107101852062D+00	.624324690187190D+00
.133442500357520D+00	.134253782564499D+01	.637432348625728D−02	.226266447701036D+01
$m = 5$, Wi	Xi →	Wi	Xi
.248406152028443D+00	.100242151968216D+00	.392331066652399D+00	.482813966046201D+00
.211418193076057D+00	.106094982152572D+01	.332466603513439D−01	.177972941852026D+01
.824853344515629D−03	.266976035608766D+01		
$m = 6$, Wi	Xi →	Wi	Xi
.196849675488598D+00	.786006594130979D−01	.349154201525395D+00	.386739410270631D+00
.257259520584421D+00	.866429471682044D+00	.760131375840057D−01	.146569804966352D+01
.685191862513597D−02	.217270779693900D+01	.984716452019267D−04	.303682016932287D+01
$m = 7$, Wi	Xi →	Wi	Xi
.160609965149261D+00	.637164846067008D−01	.306319808158099D+00	.318192018888619D+00
.275527141784906D+00	.724198989258373D+00	.120630193130784D+00	.123803559921509D+01
.218922863438067D−01	.183852822027095D+01	.123644672831057D−02	.253148815132768D+01
.110841575911059D−04	.337345643012458D+01		
$m = 8$, Wi	Xi →	Wi	Xi
.134109188453360D+00	.529786439318511D−01	.268330754472639D+00	.267398372167765D+00
.275953397988422D+00	.616302884182400D+00	.157448282618790D+00	.106424631211622D+01
.448141099174629D−01	.158885586227006D+01	.536793575602533D−02	.218392115309586D+01
.202063649132411D−03	.286313388370808D+01	.119259692659534D−05	.368600716272440D+01
$m = 9$, Wi	Xi →	Wi	Xi
.114088970242111D+00	.449390308011905D−01	.235940791223676D+00	.228605305560523D+00
.266425473630252D+00	.532195844331623D+00	.183251679101671D+00	.927280745338049D+00
.713440493066984D−01	.139292385519585D+01	.139814184155625D−01	.191884309919739D+01
.116385272078542D−02	.250624783400570D+01	.305670214897907D−04	.317269213348120D+01
.123790511337534D−06	.397889886978974D+01		

$m = 10$, Wi	Xi →	Wi	Xi
.985520975190362D−01	.387385243256994D−01	.208678066608076D+00	.198233304012949D+00
.252051688403725D+00	.465201111814507D+00	.198684340038460D+00	.816861885591907D+00
.971984227601550D−01	.123454132402774D+01	.270244164355872D−01	.170679814968865D+01
.380464962250372D−02	.222994008892444D+01	.228886243045298D−03	.280910374689825D+01
.434534479845945D−05	.346387241949537D+01	.124773714818325D−07	.425536180636561D+01
$m = 11$, Wi	Xi →	Wi	Xi
.862207055348204D−01	.338393212317745D−01	.185767318954432D+00	.173955727710236D+00
.235826124129156D+00	.410873840972387D+00	.205850326842101D+00	.726271784259897D+00
.119581170616438D+00	.110386324646491D+01	.431443275887789D−01	.153229503457537D+01
.886764989495983D−02	.200578290246814D+01	.927141875111555D−03	.252435214151921D+01
.415719321683689D−04	.309535170986922D+01	.586857646864747D−06	.373947860994358D+01
.122714514000882D−08	.451783596718736D+01		
$m = 12$, Wi	Xi →	Wi	Xi
.762461467930431D−01	.298897007696644D−01	.166446068879474D+00	.154204878265825D+00
.219394898128707D+00	.366143962974312D+00	.207016508679094D+00	.650881015845205D+00
.137264362796474D+00	.994366869880792D+00	.605056743489164D−01	.138589120364956D+01
.165538019564075D−01	.181884860842823D+01	.258608378835667D−02	.229084273867285D+01
.206237541067489D−03	.280409679339362D+01	.706650986752706D−05	.336727070416293D+01
.759131547256598D−07	.400168347567348D+01	.118195417166772D−09	.476821628798986D+01
$m = 13$, Wi	Xi →	Wi	Xi
.680463904418562D−01	.266511266223847D−01	.150057211706640D+00	.137891855469939D+00
.203606639691744D+00	.328828675158344D+00	.204104355198729D+00	.587378531473707D+00
.150119228251119D+00	.901480884535393D+00	.774536315632889D−01	.126129650258238D+01
.264891667292508D−01	.166003713190544D+01	.562343031211025D−02	.209410900418749D+01
.683241179366847D−03	.256320702525468D+01	.424853319211805D−04	.307091234102964D+01
.113557101360564D−05	.362669201180255D+01	.946453645906371D−08	.425220740047932D+01
.111810461699377D−10	.500800834323777D+01		
$m = 14$, Wi	Xi →	Wi	Xi
.612109812196030D−01	.239567892441074D−01	.136062058638519D+00	.124240344109914D+00
.188856801742046D+00	.297338568916482D+00	.198577828793095D+00	.533329214443288D+00
.158617339225760D+00	.821873189116935D+00	.928167848032935D−01	.115406707387441D+01
.379316402929447D−01	.152327479144572D+01	.102563915568667D−01	.192533821047710D+01
.172277191301072D−02	.235860076664477D+01	.165956353055286D−03	.282409375484497D+01
.819589395640479D−05	.332626935870514D+01	.173876626376522D−06	.387510499098758D+01
.114293991639056D−08	.449243807160692D+01	.104120023691655D−11	.523843136267529D+01
$m = 16$, Wi	Xi →	Wi	Xi
.505246320213779D−01	.197536584600773D−01	.113608556894151D+00	.102802245237917D+00
.162921292314545D+00	.247397669452455D+00	.183562801116246D+00	.446696225961683D+00
.165438637755610D+00	.693073720302000D+00	.116572490553503D+00	.979404170330730D+00
.619996960991566D−01	.129978932127704D+01	.239197096186835D−01	.164985424039743D+01
.640991442405013D−02	.202680815216887D+01	.113569531068878D−02	.242945049160214D+01
.125286221329562D−03	.285826652854327D+01	.795049571962246D−05	.331576927503870D+01
.259000761941506D−06	.380737711675590D+01	.361154913974278D−08	.434360634547017D+01
.153767791618984D−10	.494637720404839D+01	.867420445249463D−14	.567501793404192D+01

$m = 18$, Wi	Xi →	Wi	Xi
.426142619814402D−01	.166490322202372D−01	.965666206000141D−01	.868590621084087D−01
.141519805800333D+00	.209868348130364D+00	.166987766912260D+00	.380820011068889D+00
.163315996982125D+00	.594030720753567D+00	.130699466898224D+00	.843863021635420D+00
.833782286268739D−01	.112530737639273D+01	.411112305586427D−01	.143428771048619D+01
.151576539874531D−01	.176777393564494D+01	.403326905747473D−02	.212379869512189D+01
.744434183403738D−03	.250145945877649D+01	.909462923551198D−04	.290097223563891D+01
.693223179081282D−05	.332384856894201D+01	.304417379930170D−06	.377331513435006D+01
.685706226732795D−08	.425524711266144D+01	.657362956131058D−10	.478037806093043D+01
.189340654173429D−12	.537053657981972D+01	.691219900671972D−16	.608421686390343D+01
$m = 20$, Wi	Xi →	Wi	Xi
.365679216320084D−01	.142795096999167D−01	.833175344016776D−01	.746313003921914D−01
.123954178541192D+00	.180861563058038D+00	.151028580007023D+00	.329433356064288D+00
.156414467004551D+00	.516050543061530D+00	.136992234955135D+00	.736255457580891D+00
.995292472472201D−01	.985873575037527D+00	.585337706979063D−01	.126128901610277D+01
.271357803755605D−01	.155957964520966D+01	.964473365921541D−02	.187856191930298D+01
.255171509156881D−02	.221679416538765D+01	.486448997660902D−03	.257357782082628D+01
.643514517998962D−04	.294898974678723D+01	.564239148952609D−05	.334398137986164D+01
.309083652232771D−06	.376059993291077D+01	.975641105342091D−08	.420244260042114D+01
.157608671739272D−09	.467560884779448D+01	.107594574673723D−11	.519090168697500D+01
.216986354627585D−14	.576998516556776D+01	.531122306167734D−18	.647055838706458D+01
$m = 24$, Wi	Xi →	Wi	Xi
.280238959590632D−01	.109359185163595D−01	.642946345525286D−01	.572930968311085D−01
.974668000780112D−01	.139405790626568D+00	.123594349192342D+00	.255269852080956D+00
.137542221830958D+00	.402320285445140D+00	.135093024963885D+00	.577723113303123D+00
.116022414306451D+00	.778623923478134D+00	.857888724129206D−01	.100233650765349D+01
.536346528887674D−01	.124646689598206D+01	.278146356079732D−01	.150898244087682D+01
.117319281117153D−01	.178824185314603D+01	.394392976487992D−02	.208300245051669D+01
.103442916350338D−02	.239241857850232D+01	.206855397653932D−03	.271604260782054D+01
.307321487725170D−04	.305383853931166D+01	.329153125885586D−05	.340621904278762D+01
.245049048841933D−06	.377412101815416D+01	.121121482290497D−07	.415914532005569D+01
.374171410206439D−09	.456381026268971D+01	.664465990717029D−11	.499202557000314D+01
.599396513601045D−13	.545004471298922D+01	.224762744818840D−15	.594862649436402D+01
.240524057745805D−18	.650903854987767D+01	.287509416649120D−22	.718754093890710D+01
$m = 28$, Wi	Xi →	Wi	Xi
.223532497222130D−01	.871958802900555D−02	.514934995302115D−01	.457490961416688D−01
.789023891951536D−01	.111598719323223D+00	.102340382250321D+00	.205047814001677D+00
.118642537346178D+00	.324479237601588D+00	.124476185823818D+00	.468031827732541D+00
.117870614214242D+00	.633739546989720D+00	.997975541541843D−01	.819650499831922D+00
.746101499956902D−01	.102391838604951D+01	.485727346210531D−01	.124486554918741D+01
.271391810702547D−01	.148102140076746D+01	.128229720032950D−01	.173114216238478D+01
.504718355236458D−02	.199421822350113D+01	.162964430501715D−02	.226947480089551D+01
.424753328116719D−03	.255637067983224D+01	.878438015598585D−04	.285459905699284D+01
.141451873195417D−04	.316409416157117D+01	.173602337532960D−05	.348504761717083D+01
.158404544180041D−06	.381793971772025D+01	.104311640611843D−07	.416359350290170D+01

.477870719294284D−09	.452326498944633D+01	.145328461477563D−10	.489879403190714D+01
.275705707927272D−12	.529286416352846D+01	.299219834340666D−14	.570947571444227D+01
.163377562010590D−16	.615488431127823D+01	.364240980541915D−19	.663971857583391D+01
.223927348516155D−22	.718484721177042D+01	.142213904410655D−26	.784528269454064D+01
$m = 35$, Wi	Xi →	Wi	Xi
.160914537442079D−01	.627452437874713D−02	.372149121733108D−01	.329697650070104D−01
.575998099389036D−01	.806346261399548D−01	.762855961940561D−01	.148685279325790D+00
.918370323497648D−01	.236317556708436D+00	.102446597411635D+00	.342564721917868D+00
.106345754668529D+00	.466351668183680D+00	.102456150285205D+00	.606547080145391D+00
.910477498468085D−01	.762010023860146D+00	.740306027292296D−01	.931628643027805D+00
.545796041453307D−01	.111435012785374D+01	.361394617045517D−01	.130920230157720D+01
.212828982894386D−01	.151530792303100D+01	.110385377208141D−01	.173189317023067D+01
.499276234766670D−02	.195829185140344D+01	.194985582474527D−02	.219394679771963D+01
.650889501294441D−03	.243840972076799D+01	.183795324846697D−03	.269134063626884D+01
.434265214582000D−04	.295250780925340D+01	.848664921759722D−05	.322178909513455D+01
.135464035489423D−05	.349917555993580D+01	.174179407392497D−06	.378477838999670D+01
.177615459097800D−07	.407884039249135D+01	.141094378983242D−08	.438175392550428D+01
.855076889230746D−10	.469408803453180D+01	.385598831371819D−11	.501662920852629D+01
.125513075329417D−12	.535044309852693D+01	.283883777466493D−14	.569697002087971D+01
.424823662936399D−16	.605817791980887D+01	.393986537552956D−18	.643681958749527D+01
.206651016632570D−20	.683689499902951D+01	.535016954720285D−23	.726456235470514D+01
.547723937028798D−26	.773018719612156D+01	.145950421348989D−29	.825401095248546D+01
.354156691581967D−34	.888922893671777D+01		

参 考 文 献

[1] N.M. Steen, G.D. Byrne, E.M. Gelbard, Gaussian quadratures for the integrals, *Mathematics of Computation*, **23**:2 (1969), 661–671.

[2] R.A. Thisted, Elements of Statistical Computing–Numerical Computation, Chapman & Hall, New York, 1988.

[3] A.H. Stroud, D.H. Secrest, Gaussian Quadrature Formulas, Prentice-Hall, Englewood Cliffs, New Jersey, 1966.

[4] Y.L. Tong, The Multivariate Normal Distribution, Springer-Verlag, New York, 1990.

[5] C.M. Zhang, Z.Q. Yang, Evaluation of multivariate normal probabilities, Proceedings of the Asian Conference on Statistical Computing, Beijing, 1993.

[6] 杨自强，张春明，高维正态概率积分的降维算法与 L_1 逼近，计算数学，**19**:1 (1997), 91–102.

[7] G.P. Steck, A table for computing trivariate normal probability, *Ann. Math. Statist.*, **29** (1958), 780–800.