Contents

7

Other integration methods

Most of our Monte Carlo problems involve estimating expectations and these can often be written as integrals. Sometimes it pays to treat the problems as integrals and apply non-random numerical integration methods. In other settings we may be able to combine Monte Carlo and other methods into hybrid estimators. For instance, a nearly exact numerical integral of a problem related to our own, may be used as a control variate, as described in §8.9. This chapter is specialized and may be skipped on first reading.

There is a large literature on numerical integration. Here, we look at a few of the main ideas. Of special importance are the midpoint rule and Simpson's rule, for integrating over a finite interval $[a, b]$. They are simple to use and bring enormous improvements for smooth functions, and extend well to small dimensions d. We will also see how the advantage of classical quadrature methods decays rapidly with increasing dimension. This phenomenon is a manifestation of Bellman's 'curse of dimensionality', with Monte Carlo versions in two classic theorems of Bakhvalov. This curse is about worst case results and sometimes we are presented with problems that are more favorable than the worst case. We consider some results on 'weighted spaces' from which the worst functions are excluded. We also study 'sparse grids' in which the number of evaluation points required grows very slowly with dimension.

The connection between Monte Carlo and integration problems is as follows. Suppose that our goal is to estimate $\mathbb{E}(g(\boldsymbol{y}))$ where $\boldsymbol{y} \in \mathbb{R}^s$ has probability density function p. We find a transformation function $\psi(\cdot)$, using methods like those in Chapter 5, such that $\boldsymbol{y} = \psi(\boldsymbol{x}) \sim p$ when $\boldsymbol{x} \sim \mathbf{U}(0,1)^d$. Then

$$\mathbb{E}(g(\boldsymbol{y})) = \int_{\mathbb{R}^s} g(\boldsymbol{y}) p(\boldsymbol{y}) \, \mathrm{d}\boldsymbol{y} = \int_{(0,1)^d} g(\psi(\boldsymbol{x})) \, \mathrm{d}\boldsymbol{x} = \int_{(0,1)^d} f(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x},$$

where $f(\cdot) = g(\psi(\cdot))$. As a result our Monte Carlo problem can be transformed into a d-dimensional quadrature. We don't always have $d = s$. This method does not work when acceptance-rejection sampling is included in the way we generate $\boldsymbol{y}$, because there is no a priori bound on the number of uniform random variables that we would need.

Since we're computing integrals and not necessarily expectations we use the symbol I for the quantity of interest. For instance, with $I = \int_a^b f(x)\,\mathrm{d}x$ we have $I = (b-a)\mu$ where $\mu = \mathbb{E}(f(x))$ for $x \sim \mathbf{U}[a,b]$.

When f has a simple closed form, there is always the possibility that I can be found symbolically. Tools such as Mathematica™, Maple™, and sage can solve many integration problems. When symbolic computation cannot solve the problem then we might turn to numerical methods instead.

Numerical integration is variously called quadrature or cubature. Some authors reserve quadrature for the case where $y \in \mathbb{R}$ because the integral is the limit of a sum of quadrilateral areas (rectangles or trapezoids). They then use cubature for more general input dimensions. Hypercubature might be even more appropriate, especially for $d \geqslant 3$, but that term is seldom used.

7.1 The midpoint rule

We start with a one-dimensional problem. Suppose that we want to estimate the integral

$$I = \int_a^b f(x)\,\mathrm{d}x$$

for $-\infty < a < b < \infty$.

The value of I is the area under the curve f over the interval $[a,b]$. It is easy to compute the area under a piecewise constant curve, and so it is natural to approximate f by a piecewise constant function $\hat{f}$ and then estimate I by $\hat{I} = \int_a^b \hat{f}(x)\,\mathrm{d}x$. We let $a = x_0 < x_1 < \cdots < x_n = b$ and then take t_i with $x_{i-1} \leqslant t_i \leqslant x_i$ for $i = 1, \ldots, n$, and put $\hat{f}(x) = f(t_i)$ whenever $x_{i-1} \leqslant x < x_i$. To complete the definition, take $\hat{f}(b) = f(b)$. Then

$$\hat{I} = \int_a^b \hat{f}(x)\,\mathrm{d}x = \sum_{i=1}^n (x_i - x_{i-1}) f(t_i).$$

If f is Riemann integrable on $[a,b]$ then $\hat{I} - I \to 0$ as $n \to \infty$ as long as $\max_{1\leqslant i\leqslant n}(x_i - x_{i-1}) \to 0$.

There is a lot of flexibility in choosing $\hat{f}$ but unless we have special knowledge about f we might as well use n equal intervals of length $(b-a)/n$ and take t_i in the middle of the i'th interval. This choice yields the ***midpoint rule***

$$\hat{I} = \frac{b-a}{n} \sum_{i=1}^n f\Big(a + (b-a)\frac{i-1/2}{n}\Big). \tag{7.1}$$

If we have constructed f so that $a = 0$ and $b = 1$ then the midpoint rule simplifies to

$$\hat{I} = \frac{1}{n}\sum_{i=1}^{n} f\Big(\frac{i-1/2}{n}\Big).$$

For example, if $z \sim \mathcal{N}(0,1)$ and we want to estimate $\mathbb{E}(g(z)) = \int_{-\infty}^{\infty} g(z)\varphi(z)\,\mathrm{d}z$ then we can use

$$\frac{1}{n}\sum_{i=1}^{n} g\Big(\Phi^{-1}\Big(\frac{i-1/2}{n}\Big)\Big).$$

In so doing, we are using the midpoint rule to estimate $\int_0^1 f(x)\,\mathrm{d}x$ where $f(x) = g(\Phi^{-1}(x))$. For now we will suppose that g is a bounded function so that f is also bounded. We revisit the problem of unbounded integrands on page 7.

For a smooth function and large n, the midpoint rule attains a much better rate than Monte Carlo sampling.

Theorem 7.1. *Let $f(x)$ be a real-valued function on $[a,b]$ for $-\infty < a < b < \infty$. Assume that the second derivative $f''(x)$ is continuous on $[a,b]$. Let $t_i = a + (b-a)(i-1/2)/n$ for $i = 1,\dots,n$. Then*

$$\left|\int_a^b f(x)\,\mathrm{d}x - \frac{b-a}{n}\sum_{i=1}^{n} f(t_i)\right| \leqslant \frac{(b-a)^3}{24n^2}\max_{a\leqslant z\leqslant b}|f''(z)|.$$

Proof. For any x between $x_{i-1} \equiv t_i - (b-a)/(2n)$ and $x_i \equiv t_i + (b-a)/(2n)$, we can write $f(x) = f(t_i) + f'(t_i)(x-t_i) + (1/2)f''(z(x))(x-t_i)^2$ where $z(x)$ is a point between x and t_i. Let $\hat{I} = ((b-a)/n)\sum_{i=1}^n f(t_i)$. Then

$$\begin{aligned}
|I-\hat{I}| &= \left|\int_a^b f(x)\,\mathrm{d}x - \frac{b-a}{n}\sum_{i=1}^{n} f(t_i)\right| \\
&= \left|\sum_{i=1}^{n}\int_{x_{i-1}}^{x_i} f(x) - f(t_i)\,\mathrm{d}x\right| \\
&= \left|\sum_{i=1}^{n}\int_{x_{i-1}}^{x_i} f'(t_i)(x-t_i) + \frac{1}{2}f''(z(x))(x-t_i)^2\,\mathrm{d}x\right| \\
&= \frac{1}{2}\left|\sum_{i=1}^{n}\int_{x_{i-1}}^{x_i} f''(z(x))(x-t_i)^2\,\mathrm{d}x\right|.
\end{aligned}$$

Because f'' is continuous on $[a,b]$ and that interval is compact, f'' is absolutely continuous there and hence $M = \max_{a\leqslant x\leqslant b}|f''(x)| < \infty$. To complete the proof we write

$$|I-\hat{I}| \leqslant \frac{M}{2}\sum_{i=1}^{n}\int_{x_{i-1}}^{x_i}(x-t_i)^2\,\mathrm{d}x = \frac{Mn}{2}\int_0^{x_1}(x-x_1/2)^2\,\mathrm{d}x \tag{7.2}$$

by symmetry. Then with $x_1 = (b-a)/n$,

$$\int_0^{x_1} (x - x_1/2)^2 \,\mathrm{d}x = 2\int_0^{x_1/2} x^2 \,\mathrm{d}x = \frac{2}{3}\left(\frac{x_1}{2}\right)^3 = \frac{(b-a)^3}{12n^3}. \tag{7.3}$$

The result follows by substituting (7.3) into (7.2). □

The midpoint rule is very simple to use and it works well on one-dimensional smooth functions. The rate $O(n^{-2})$ is much better than the $O(n^{-1/2})$ root mean square error (RMSE) from Monte Carlo. The proof in Theorem 7.1 is fairly simple. A sharper analysis, in Davis and Rabinowitz (1984, Chapter 4.3) shows that

$$\hat{I} - I = \frac{(b-a)^3}{24n^2} f''(\hat{z})$$

holds for some $\hat{z} \in (a, b)$, under the conditions of Theorem 7.1.

Error estimation is awkward for classical numerical integration rules. When $f''(x)$ is continuous on $[a, b]$ then the midpoint rule guarantees that $|\hat{I} - I| \leqslant (b-a)^3 M/(24n^2)$, where $M = \max_{a \leqslant z \leqslant b} |f''(z)|$. This looks like a 100% confidence interval. It would be, if we knew M, but unfortunately, we usually don't know M.

The midpoint rule is the integral of a very simple piecewise constant approximation to f. We could instead approximate f by a piecewise linear function over each interval $[x_{i-1}, x_i]$. If once again, we take equispaced values $x_i = a + i(b-a)/n$ we get the approximate function $\widetilde{f}$ that on the interval $[x_{i-1}, x_i]$ satisfies

$$\widetilde{f}(x) = f(x_{i-1}) + \frac{x - x_{i-1}}{x_i - x_{i-1}}\big(f(x_i) - f(x_{i-1})\big).$$

The integral of $\widetilde{f}(x)$ over $[a, b]$ yields the ***trapezoid rule***

$$\widetilde{I} = \frac{b-a}{n}\Big[\frac{1}{2}f(a) + \sum_{i=1}^{n-1} f(x_i) + \frac{1}{2}f(b)\Big].$$

The trapezoid rule is based on a piecewise linear approximation $\widetilde{f}$ to f instead of a piecewise constant one $\hat{f}$. For large n, the function $\widetilde{f}$ is usually much closer to f than $\hat{f}$ is, and so we might expect the trapezoid rule to attain a higher rate of convergence than the midpoint rule. But we would be wrong. Under the conditions of Theorem 7.1, $\widetilde{I} - I = -(b-a)^3 f''(\widetilde{z})/(12n^2)$ for some $\widetilde{z} \in (a, b)$ and so

$$|\widetilde{I} - I| \leqslant \frac{(b-a)^3}{12n^2} \max_{a \leqslant z \leqslant b} |f''(z)|.$$

The trapezoid rule integrates correctly any function f that is piecewise linear on each segment $[x_{i-1}, x_i]$, by using two evaluation points at the ends of the segments. The midpoint rule also integrates such a function correctly using just one point in the middle of each segment. The midpoint rule has benefitted

from an error cancellation. This kind of cancellation plays a big role in the development of classical quadrature methods.

Another fact about these two methods is worth mentioning: the ***bracketing inequality***. Suppose that we know $f''(x) \geqslant 0$ for all $x \in [a,b]$. Then $\hat{I} - I = f''(\hat{z})(b-a)^3/(24n^2) \geqslant 0$ and $\widetilde{I} - I = -f''(\widetilde{z})(b-a)^3/(14n^2) \leqslant 0$ and therefore

$$\hat{I} \leqslant I \leqslant \widetilde{I}. \tag{7.4}$$

Equation (7.4) supplies us with a computable interval certain to contain the answer (when $f'' \geqslant 0$ is continuous on $[a,b]$), that is, a 100% confidence interval. It is unusual to get such a good error estimate in a deterministic problem. Of course it requires knowledge that $f'' \geqslant 0$ everywhere. If $f''(x) \leqslant 0$ is continuous, then the inequalities in (7.4) are reversed and we still get a bracketing.

Singularities at the endpoints

The midpoint rule has a big practical advantage over the trapezoid rule. It does not evaluate f at either endpoint a or b. Many of the integrals that we apply Monte Carlo methods to diverge to infinity at one or both endpoints. In such cases, the midpoint rule ***avoids the singularity***.

There are numerous mathematical techniques for removing singularities. These include change of variable transformations as well as methods that write $\int f(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x} = \int f_0(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x} + \int f_1(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$ where f_0 has a singularity but $\int f_0(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$ is known, and f_1 has no singularity.

When we have no such analysis of our integrand, perhaps because it has a complicated problem-dependent formulation, or because we have hundreds of integrands to consider simultaneously, then avoiding the singularity is attractive. By contrast, the trapezoid rule does not avoid the endpoints $x = a$ and $x = b$. For such methods a second, less attractive principle is to ***ignore the singularity***, perhaps by using $f(x_i) = 0$ at any sample point x_i where f is singular. Davis and Rabinowitz (1984, p. 180) remark that ignoring the singularity is a "tricky business and should be avoided where possible".

If $|f(x)| \to \infty$ as $x \to a$ or b then of course we won't have f'' bounded on $[a,b]$, and so we can no longer expect an error of $O(n^{-2})$. The midpoint rule handles this singularity problem much more gracefully than the trapezoid rule does. See Lubinsky and Rabinowitz (1984) and references therein for information on convergence rates that are attained on integration problems containing singularities.

7.2 Simpson's rule

The midpoint and trapezoid rules are based on correctly integrating piecewise constant and linear approximations to the integrand. That idea extends naturally to methods that locally integrate higher order polynomials. The result is much more accurate integration, at least when the integrand is smooth.

As a next step, we find a three-point rule that correctly integrates any quadratic polynomial over $[0,1]$. It is enough to correctly integrate 1, x and x^2. If we evaluate the function at points 0, 1/2 and 1 and use a rule of the form $w_1 f(0) + w_2 f(1/2) + w_3 f(1)$, the correct weights w_j can be found by solving

$$\begin{pmatrix} 1 & 1 & 1 \\ 0 & 1/2 & 1 \\ 0 & 1/4 & 1 \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix} = \begin{pmatrix} 1 \\ 1/2 \\ 1/3 \end{pmatrix}.$$

That is, we take

$$\begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1/2 & 1 \\ 0 & 1/4 & 1 \end{pmatrix}^{-1} \begin{pmatrix} 1 \\ 1/2 \\ 1/3 \end{pmatrix} = \begin{pmatrix} 1/6 \\ 2/3 \\ 1/6 \end{pmatrix}.$$

By a change of variable, this three-point rule can be adapted to the interval $[0, 2h]$ through

$$\int_0^{2h} f(x)\,\mathrm{d}x \doteq 2h\Big(\frac{1}{6}f(0) + \frac{2}{3}f(h) + \frac{1}{6}f(2h)\Big), \tag{7.5}$$

which is exact for quadratic functions f. Now we split the interval $[a,b]$ into $n/2$ panels of width $2h$ (so n has to be even) and apply equation (7.5) in each of them. Letting f_i be a shorthand notation for $f(a+ih)$ where $h = (b-a)/n$, the result is

$$\begin{aligned} \int_a^b f(x)\,\mathrm{d}x &\doteq \frac{h}{3}\Big(f_0 + 4f_1 + 2f_2 + 4f_3 + \cdots + 2f_{n-2} + 4f_{n-1} + f_n\Big) \\ &= \frac{h}{3}\Big(f_0 + 4\sum_{j=1}^{n/2-1} f_{2j-1} + 2\sum_{j=1}^{n/2} f_{2j} + f_n\Big). \end{aligned} \tag{7.6}$$

Equation (7.5) is known as Simpson's rule. Equation (7.6) is a compound Simpson rule that is also called Simpson's rule. Equation (7.5) is exact for cubic functions, not just quadratics. As a result (7.6) is exact for a piecewise cubic approximation to f. If f has a continuous fourth derivative $f^{(4)}$ on $[a,b]$ then the error in Simpson's rule is

$$-\frac{(b-a)^4}{180\,n^4} f^{(4)}(z), \quad \text{for some } z \in (a,b).$$

7.3 Higher order rules

The idea behind Simpson's rule generalizes easily to higher orders. We split the interval $[a,b]$ into panels, find a rule that integrates a polynomial correctly within a panel, and then apply it within every panel to get a compound rule.

There are two main varieties of compound quadrature rule. For ***open rules*** we do not evaluate f at the end-points of the panel. The midpoint rule is open.

For ***closed rules*** we do evaluate f at the end-points of the panel. The trapezoid rule and Simpson's rule are both closed. Closed rules have the advantage that some function evaluations get reused when we increase n. Open rules have a perhaps greater advantage that they avoid the ends of the interval where singularities often appear.

Newton-Cotes

The trapezoid rule and Simpson's rule use $m = 2$ and $m = 3$ points respectively within each panel. In general, one can use m points to integrate polynomials of degree $m-1$, to yield the Newton-Cotes formulas, of which the trapezoid rule and Simpson's rule are special cases. The Newton-Cotes rule for $m = 4$ is another of Simpson's rules, called Simpson's 3/8 rule. Newton-Cotes rules of odd order have the advantage that, by symmetry, they also correctly integrate polynomials of degree m, as we saw already in the case of Simpson's rule.

The next two odd order rules are

$$\int_0^{4h} f(x)\,\mathrm{d}x \doteq \frac{h}{45}\Big(14f_0 + 64f_1 + 24f_2 + 64f_3 + 14f_4\Big), \quad \text{and} \tag{7.7}$$
$$\int_0^{6h} f(x)\,\mathrm{d}x \doteq \frac{h}{140}\Big(41f_0 + 216f_1 + 27f_2 + 272f_3 + 27f_4 + 216f_5 + 41f_6\Big).$$

These are the 5 and 7 point closed Newton-Cotes formulas. Equation (7.7) is known as Boole's rule.

High order rules should be used with caution. They exploit high order smoothness in the integrand, but can give poor outcomes when the integrand is not as smooth as they require. In particular, if a genuinely smooth quantity has some mild nonsmoothness in its numerical implementation f, then high order integration rules can behave very badly, magnifying this numerical noise.

As a further caution, Davis and Rabinowitz (1984) note that taking f fixed and letting the order m in a Newton-Cotes formula increase does not always converge to the right answer even for f with infinitely many derivatives. Lower order rules applied in panels are more robust.

The Newton-Cotes rules can be made into compound rules similarly to the way Simpson's rule was compounded. When the basic method integrates polynomials of degree r exactly within panels, then the compound method has error $O(n^{-r})$, assuming that $f^{(r)}$ is continuous on $[a, b]$.

As noted above, open rules are valuable because they avoid the endpoints where the function may be singular. Here are a few open rules:

$$\int_0^{2h} f(x)\,\mathrm{d}x = 2hf(h) + \frac{h^3}{3}f''(z),$$
$$\int_0^{4h} f(x)\,\mathrm{d}x = \frac{4h}{3}\Big(2f(h) - f(2h) + 2f(3h)\Big) + \frac{14h^5}{45}f^{(4)}(z), \quad \text{and}$$
$$\int_0^{5h} f(x)\,\mathrm{d}x = \frac{5h}{24}\Big(11f(h) + f(2h) + f(3h) + 11f(4h)\Big) + \frac{95h^5}{144}f^{(4)}(z).$$

$\mathbf{w}(x)$	**Rule**
$\mathbb{1}_{\lvert x\rvert\leqslant 1}$	Gauss-Legendre
$\exp(-x^2)$	Gauss-Hermite
$\mathbb{1}_{0\leqslant x<\infty}\exp(-x)$	Gauss-Laguerre
$\mathbb{1}_{\lvert x\rvert<1}(1-x^2)^{-1/2}$	Gauss-Chebyshev (1st kind)
$\mathbb{1}_{\lvert x\rvert\leqslant 1}(1-x^2)^{1/2}$	Gauss-Chebyshev (2nd kind)

Table 7.1: This table lists some weight functions and the corresponding families of Gauss quadrature rules.

In each case the point z is inside the interval of integration, and the error term assumes that the indicated derivative is continuous. The first one is simply the midpoint rule after a change of variable to integrate over $[0, 2h]$. The next two are from Davis and Rabinowitz (1984, Chapter 2.6). They both have the same order. The last one avoids negative weights but requires an extra point.

Gauss rules

The rules considered above evaluate f at equispaced points. A Gauss rule takes the more general form

$$\hat{I}_G = \sum_{i=1}^{m} w_i f(x_i)$$

where both x_i and w_i can be chosen to attain high accuracy.

The basic panel for a Gauss rule is conventionally $[-1, 1]$ or sometimes $\mathbb{R}$, and not $[0, h]$ as we used for Newton-Cotes rules. Also the target integration problem is generally weighted. That is, we seek to approximate

$$\int_{-\infty}^{\infty} f(x)w(x)\,\mathrm{d}x$$

for a weight function $w(x) \geqslant 0$. The widely used weight functions are multiples of standard probability density functions, such as the uniform, gamma, Gaussian and beta distributions; see Table 7.1. The idea is that having f be nearly a polynomial can be much more appropriate than requiring the whole integrand $f(x)w(x)$ to be nearly a polynomial.

Choosing w_i and x_i together yields $2m$ parameters and it is then possible to integrate polynomials of degree up to $2m-1$ without error. It follows that if $f(x) = p(x)$ where p is a polynomial with $p(x_i) = f(x_i)$ for $i = 1, \ldots, m$, then $\hat{I}_G(f) = I(f)$. Because of this, the Gauss rule is called an ***interpolatory quadrature*** formula.

The error in a Gauss rule is

$$\frac{(b-a)^{2m+1}(m!)^4}{(2m+1)[(2m)!]^3} f^{(2m)}(z)$$

$\mathbf{m}$	$\mathbf{x}_i$	$\mathbf{w}_i$
2	$\pm\frac{1}{\sqrt{3}}$	1
3	0	$\frac{8}{9}$
	$\pm\frac{1}{5}\sqrt{15}$	$\frac{5}{9}$
4	$\pm\frac{1}{35}\sqrt{525-70\sqrt{30}}$	$\frac{1}{36}\left(18+\sqrt{30}\right)$
	$\pm\frac{1}{35}\sqrt{525+70\sqrt{30}}$	$\frac{1}{36}\left(18-\sqrt{30}\right)$
5	0	$\frac{128}{225}$
	$\pm\frac{1}{21}\sqrt{245-14\sqrt{70}}$	$\frac{1}{900}\left(322+13\sqrt{70}\right)$
	$\pm\frac{1}{21}\sqrt{245+14\sqrt{70}}$	$\frac{1}{900}\left(322-13\sqrt{70}\right)$

Table 7.2: Gauss quadrature rules $\sum_{i=1}^m w_i f(x_i)$ to approximate $\int_{-1}^1 f(x)\,\mathrm{d}x$, for $m = 1,\dots,5$. From Weisstein, Eric W, "Legendre-Gauss Quadrature." MathWorld web site.—A Wolfram Web Resource. http://mathworld.wolfram.com/Legendre-GaussQuadrature.html

where $a < z < b$, provided that f has $2m$ continuous derivatives. Unlike Newton-Cotes rules, Gauss rules of high order have non-negative weights. We could in principle use a very large m, and the result will still converge to $I(f)$, for continuous f on $[-1,1]$. Trefethen (2008) includes a short program to compute a Gauss rule and considers m as large as $2^{10}+1$.

For the uniform weighting $w(x) = 1$ we can easily break the region $[-1,1]$ into panels. Then for n function evaluations the error will be $O(n^{-2m})$ assuming as usual that $f^{(2m)}$ is continuous on $[a,b]$. Gauss rules for uniform weights on $[-1,1]$ have the advantage that they can be used within panels. Several are listed in Table 7.2.

Clenshaw-Curtis rules

Gauss rules for large n are expensive to construct. Also, the Gauss rules are not generally nested, meaning that the points x_i used at one value of m are not necessarily used again for a larger value of m. Clenshaw-Curtis rules address both of these difficulties. This second point makes them very well suited to sparse grid methods for multidimensional integration, that we consider in §7.8.

As for Gauss rules, we consider integration over $[-1,1]$ and approximate the

integral I by

$$\hat{I}_{CC} = \sum_{i=1}^{m} w_i f(x_i).$$

The Clenshaw-Curtis rule evaluates f at

$$x_i = \cos\Big(\frac{(i-1)\pi}{m-1}\Big), \quad i = 1, \ldots, m.$$

For large m, these points become more dense near the endpoints of the interval $[-1, 1]$. The weights w_i are chosen so that

$$\sum_{i=1}^{m} w_i x_i^r = \int_{-1}^{1} x^r \, \mathrm{d}x, \quad r = 0, 1, \ldots, m-1.$$

Thus, like the Gauss rule, Clenshaw-Curtis is interpolatory, though it is only exact for polynomials of degree up to $m-1$ as compared to degree $2m-1$ for the Gauss rule.

If $m_k = 2^k + 1$ then

$$x_i = \cos\Big(\frac{(i-1)\pi}{2^k}\Big) = \cos\Big(\frac{(i'-1)\pi}{2^{k+1}}\Big), \quad i' = 2i - 1$$

and so x_i appears again as a point in the Clenshaw-Curtis rule for m_{k+1}.

A short program to compute Clenshaw-Curtis integral estimate appears in Trefethen (2008). It uses a fast Fourier transform to compute the weights, and that paper includes examples with m as large as $2^{20} + 1$. Like Gauss rules, but unlike Newton-Cotes, the Clenshaw-Curtis formulas have nonnegative weights w_i.

Trefethen (2008), citing numerous prior papers, considers Clenshaw-Curtis rules to be almost as good as Gauss rules, despite having only about half of the latter's degree of polynomial correctness. In his numerical examples, the Gauss rules were much better for high degree polynomials, but not for more complicated functions such as $|x|^3$ or $|x + 1/2|^{1/2}$ on $[-1, 1]$. Where there was a great difference, it was because the Gauss rules were exceptionally good, and not because the Clenshaw-Curtis rules were very bad.

7.4 Fubini, Bahkvalov and curse of dimension

Classical quadrature methods are very well tuned to one-dimensional problems with smooth integrands. A natural way to extend them to multi-dimensional problems is to write them as iterated one-dimensional integrals, via Fubini's theorem. When we estimate each of those one-dimensional integrals by a quadrature rule, we end up with a set of sample points on a multi-dimensional grid. Unfortunately, there is a curse of dimensionality that severely limits the accuracy of this approach.

To see informally what goes wrong, let $f(x, y)$ be a function on $[0, 1]^2$. Now write

$$I = \int_0^1 \int_0^1 f(x, y)\,\mathrm{d}x\,\mathrm{d}y = \int_0^1 I(y)\,\mathrm{d}y$$

where $I(y) = \int_0^1 f(x, y)\,\mathrm{d}x$. Suppose that we use the same m point integration rule with weight u_i on point x_i, for any value of y, getting

$$\hat{I}(y) = \sum_{i=1}^{m} u_i f(x_i, y) = I(y) + E_1(y),$$

where $|E_1(y)| \leqslant Am^{-r}$ holds for some $A < \infty$ and all $0 \leqslant y \leqslant 1$. The value of r depends on the method we use and how smooth f is. Next we use an n point rule to average over y getting

$$\hat{I} = \sum_{j=1}^{n} v_j \hat{I}(y_j) = \sum_{j=1}^{n} v_j \big(I(y_j) + E_1(y_j)\big) = \int_0^1 I(y)\,\mathrm{d}y + E_2 + \sum_{j=1}^{n} v_j E_1(y_j)$$

where $|E_2| \leqslant Bn^{-s}$ for some $B < \infty$ and s depending on the outer integration rule and on how smooth $I(y)$ is.

The total error is a weighted sum of errors E_1 at different points y_j plus the error E_2. We suppose that the weighted sum of $E_1(y_j)$ is $O(m^{-r})$. This happens if $\sum_{j=1}^{n} |v_j| < C$ holds for all n because we assumed that $|E_1(y)| \leqslant Am^{-r}$ holds with the same $A < \infty$ for all $y \in [0, 1]$.

The result is that $|\hat{I} - I| = O(m^{-r} + n^{-s})$. In other words, the convergence rate that we get is like the worst of the two one-dimensional rules that we combine.

More generally, if we are using a d-dimensional grid of points and a product rule

$$\hat{I} = \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \cdots \sum_{i_d=1}^{n_d} \Big(\prod_{j=1}^{d} w_{ji_j} \Big) f(x_{i_1}, x_{i_2}, \ldots, x_{i_d})$$

we cannot expect the result to be better than what we would get from the worst of the rules we have used. Suppose that rule j is the least accurate. Then $\hat{I}$ could hardly be better than

$$\sum_{i=1}^{n_j} w_{ji} I_j(x_{ji})$$

where $I_j(x_j)$ is the (exact) integral of f over all variables other than x_j.

Getting the worst one-dimensional rate leads to a curse of dimensionality. Suppose that we use the same n point one-dimensional quadrature rule on each of d dimensions. As a result we use $N = n^d$ function evaluations. If the one-dimensional rule has error $O(n^{-r})$, then the combined rule has error

$$|\hat{I} - I| = O(n^{-r}) = O(N^{-r/d}).$$

Even modestly large d will give a bad result. The value r is the smaller of the number of continuous derivatives f has and the number that the quadrature

rule is able to exploit. Taking $r \gg d$ won't help in practice, because high order rules get very cumbersome and many of them are prone to roundoff errors.

This curse of dimensionality is not confined to sampling on grids formed as products of one-dimensional rules. Any quadrature rule in high dimensions will suffer from the same problem. Two important theorems of Bakhvalov, below, make the point.

Theorem 7.2 (Bakhvalov I)**.** *For $0 < M < \infty$ and integer $r \geqslant 1$, let*

$$C_M^r = \left\{ f : [0,1]^d \to \mathbb{R} \mid \left| \frac{\partial^r f(\boldsymbol{x})}{\partial x_1^{\alpha_1} \cdots \partial x_d^{\alpha_d}} \right| \leqslant M, \quad \forall\, \alpha_j \geqslant 0 \text{ with } \sum_{j=1}^d \alpha_j = r \right\}.$$

Then there exists $k > 0$ such that for any $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in [0,1]^d$ and any $w_1, \ldots, w_n \in \mathbb{R}$, there is an $f \in C_M^r$ with

$$\left| \sum_{i=1}^n w_i f(\boldsymbol{x}_i) - \int_{[0,1]^d} f(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x} \right| \geqslant k n^{-r/d}$$

Proof. This is given as Theorem 3.1 in Dimov (2008). □

Bakhvalov's theorem makes high-dimensional quadrature seem intractable. There is no way to beat the rate $O(n^{-r/d})$, no matter where you put your sampling points $\boldsymbol{x}_i$ or how cleverly you weight them. At first, this result looks surprising, because we have been using Monte Carlo methods which get an RMSE of $O(n^{-1/2})$ in any dimension. The explanation is that in Monte Carlo sampling we pick one single function $f(\cdot)$ with finite variance σ^2 and then in sampling n uniform random points, get an RMSE of $\sigma n^{-1/2}$ for the estimate of that function's integral. Bahkvalov's theorem works in the opposite order. We pick our points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$, and their weights w_i. Then Bakhvalov finds a function f with r derivatives on which our rule makes a large error.

When we take a Monte Carlo sample, there is always some smooth function for which we would have got a very bad answer. Such worst case analysis is very pessimistic because the worst case functions could behave very oddly right near our sampled $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$, and the worst case functions might look nothing like the ones we are trying to integrate. Lower bounds like Bahkvalov's are often constructed by choosing f equal to 0 at every $\boldsymbol{x}_i$ we used and then maximizing the integral of f subject to the constraints we have imposed on the partial derivatives of f. The integrand or integrands we are really interested in might not be so nasty.

Bakhvalov has a counterpart to Theorem 7.2 which describes random sampling. Whatever way we choose to sample our input points, there exists a smooth function with a large RMSE:

Theorem 7.3 (Bakhvalov II)**.** *For $0 < M < \infty$ and integer $r \geqslant 1$, let C_M^r be as given in Theorem 7.2. Let $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ be random elements of $[0,1]^d$ and*

let $w_1, \dots, w_n \in \mathbb{R}$. Then there exists $k > 0$ such that some function $f \in C_M^r$ satisfies

$$\mathbb{E}\biggl(\Bigl(\sum_{i=1}^n w_i f(\boldsymbol{x}_i) - \int_{[0,1]^d} f(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}\Bigr)^2\biggr)^{1/2} \geqslant k n^{-1/2-r/d}.$$

Proof. This is given as Theorem 3.2 of Dimov (2008). □

In Theorem 7.3, the worst case function f is chosen knowing how we will sample $\boldsymbol{x}_i$, but not knowing the resulting values $\boldsymbol{x}_i$ that we will actually use. Here we see an RMSE of at least $O(n^{-1/2-r/d})$ which does not contradict the MC rate. There is a curse of dimension only in the extent to which we can improve on MC, namely a factor proportional to $n^{-r/d}$.

7.5 Hybrids with Monte Carlo

We can hybridize quadrature and Monte Carlo methods by using each of them on some of the variables. For example, to approximate

$$I = \int_{[0,1]^d} \int_{[0,1]^s} f(\boldsymbol{x}, \boldsymbol{y})\,\mathrm{d}\boldsymbol{x}\,\mathrm{d}\boldsymbol{y}$$

we might take $\boldsymbol{x}_1, \dots, \boldsymbol{x}_m \in [0,1]^s$, $w_1, \dots, w_m \in \mathbb{R}$ and draw $\boldsymbol{y}_1, \dots, \boldsymbol{y}_n \sim \mathbf{U}(0,1)^d$ independently. Then the hybrid estimate is

$$\hat{I} = \frac{1}{n}\sum_{i=1}^n \sum_{j=1}^m w_j f(\boldsymbol{x}_j, \boldsymbol{y}_i). \tag{7.8}$$

Our hybrid has a curse of dimension driven by the size of s and it has variance σ^2/n where

$$\sigma^2 = \mathrm{Var}\biggl(\sum_{j=1}^m w_j f(\boldsymbol{x}_j, \boldsymbol{y})\biggr) = \sum_{j=1}^m \sum_{j'=1}^m w_j w_{j'} \mathrm{Cov}(f(\boldsymbol{x}_j, \boldsymbol{y}), f(\boldsymbol{x}_{j'}, \boldsymbol{y})).$$

We might expect this hybrid to be useful when s is not too large and $f(\boldsymbol{x}, \boldsymbol{y})$ is very smooth in $\boldsymbol{x}$. Then the inner sum in (7.8) is well handled by quadrature. If additionally, f has large variance in $\boldsymbol{x}$ at any fixed value for $\boldsymbol{y}$, then our quadrature may be much better than using Monte Carlo on both $\boldsymbol{x}$ and $\boldsymbol{y}$.

If we could integrate out $\boldsymbol{x}$ in closed form then we could use the estimate $\hat{I} = (1/n)\sum_{i=1}^n h(\boldsymbol{y}_i)$ where $h(\boldsymbol{y}) = \mathbb{E}(f(\boldsymbol{x}, \boldsymbol{y}) \,|\, \boldsymbol{y})$ for $\boldsymbol{x} \sim \mathbf{U}(0,1)^d$. This is the method called conditioning in §8.7. The hybrid (7.8) is conditioning with a numerical approximation to h. The term ***preintegration*** is also used.

Hybrids of Monte Carlo and quasi-Monte Carlo methods are often used. See Chapter 17. They take the form $\hat{I} = (1/n)\sum_{i=1}^n f(\boldsymbol{x}_i, \boldsymbol{y}_i)$ for a quadrature rule with $\boldsymbol{x}_i \in [0,1]^s$ and Monte Carlo samples $\boldsymbol{y}_i \sim \mathbf{U}(0,1)^d$.

7.6 Laplace approximations

The Laplace approximation is a classical device for approximate integration. This section uses the statistical concept of Fisher information.

Suppose that we seek to estimate $I = \int_{\mathbb{R}^d} f(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$ and $\log(f(\boldsymbol{x}))$ has Taylor expansion $b + \frac{1}{2}(\boldsymbol{x}-\boldsymbol{x}_*)^{\mathsf{T}} H(\boldsymbol{x}-\boldsymbol{x}_*)$ around its maximizer $\boldsymbol{x}_*$, where H is the Hessian matrix for $\log(f(\cdot))$ at $\boldsymbol{x}=\boldsymbol{x}_*$. If H is negative definite, then $\Sigma = -H^{-1}$ is a covariance matrix and we may write

$$I \approx e^b \int e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{x}_*)^{\mathsf{T}}\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{x}_*)}\,\mathrm{d}\boldsymbol{x} = e^b(2\pi)^{d/2}|\Sigma|^{-1/2} \tag{7.9}$$

where $|\Sigma|$ is the determinant of Σ, using the known normalization of the $\mathcal{N}(\boldsymbol{x}_*, \Sigma)$ distribution. Note that $|\Sigma| = |-H^{-1}|$ which is the absolute value of $1/|H|$.

The Laplace approximation is very accurate when $\log(f(\boldsymbol{x}))$ is smooth and the quadratic approximation is good where f is not negligible. Such a phenomenon often happens when $\boldsymbol{x}$ is a statistical parameter subject to the central limit theorem, $f(\boldsymbol{x})$ is its posterior distribution, and the sample size is large enough for the CLT to apply.

If we want the integral of $g(\boldsymbol{x})f(\boldsymbol{x})$ for smooth g, then we may multiply (7.9) by $g(\boldsymbol{x}_*)$. The following theorem gives the details:

Theorem 7.4. *The asymptotic equivalence*

$$\int_A g(\boldsymbol{x})e^{-\lambda h(\boldsymbol{x})}\,\mathrm{d}\boldsymbol{x} \sim g(\boldsymbol{x}_*)(2\pi)^{d/2}|\lambda H(\boldsymbol{x}_*)|^{-1/2}e^{-\lambda h(\boldsymbol{x}_*)}$$

holds as $\lambda \to \infty$, if

- **i)** *$A \subseteq \mathbb{R}^d$ is open,*
- **ii)** *$\int_A |g(\boldsymbol{x})|e^{-\lambda h(\boldsymbol{x})}\,\mathrm{d}\boldsymbol{x} < \infty$ for all $\lambda \geqslant \lambda_0$ for some $\lambda_0 < \infty$,*
- **iii)** *there exists $\boldsymbol{x}_* \in A$ such that for all $\epsilon > 0$*

$$\inf\{h(\boldsymbol{x}) - h(\boldsymbol{x}_*) \mid \boldsymbol{x} \in A, \|\boldsymbol{x}-\boldsymbol{x}_*\| > \epsilon\} > 0,$$

- **iv)** *g is continuous in a neighborhood of $\boldsymbol{x}_*$ with $g(\boldsymbol{x}_*) \neq 0$, and*
- **v)** *h has two continuous derivatives on A where $H(\boldsymbol{x}_*) \equiv \partial^2 h(\boldsymbol{x})/\partial\boldsymbol{x}\partial\boldsymbol{x}^{\mathsf{T}}$ is positive definite.*

Proof. This is Theorem 4.14 of Evans and Swartz (2000) which is based on a result in Wong (1989). □

The parameter λ is a measure of how strongly spiked f is. In statistical applications, λ is often the number n of observations in a sample. Similarly, the argument $\boldsymbol{x}$ is usually a parameter vector θ in statistical applications, and f is a distribution for θ. For the rest of this discussion we switch to the notation used for Bayesian statistical applications.

Suppose that a parameter $\theta \in \Theta \subseteq \mathbb{R}^p$ has prior distribution $\pi_0(\theta)$, and the data are independent draws from the density or mass function $p(\boldsymbol{x};\theta)$. We will use $\mathcal{X}$ to represent the full collection of observations $\boldsymbol{x}_i \in \mathbb{R}^d$, for $i = 1, \ldots, n$. The log likelihood function is

$$\ell(\theta) = \ell(\theta; \mathcal{X}) = \sum_{i=1}^{n} \log(p(\boldsymbol{x}_i;\theta))$$

and so the posterior distribution of θ given $\mathcal{X}$ is

$$\pi(\theta) = \pi(\theta \,|\, \mathcal{X}) \propto \pi_0(\theta) e^{\ell(\theta)}.$$

The value $\hat\theta = \arg\max_\theta \ell(\theta)$ is the maximum likelihood estimate of θ. When $\hat\theta$ is interior to Θ and ℓ is smooth, then we have the Taylor approximation $\ell(\theta) \doteq \ell(\hat\theta) - \frac{1}{2}(\theta-\hat\theta)^{\mathsf{T}} H(\hat\theta)(\theta-\hat\theta)$ where

$$H(\theta) = -\sum_{i=1}^{n} \frac{\partial^2 \log(p(\boldsymbol{x}_i;\theta))}{\partial\theta\partial\theta^{\mathsf{T}}} \equiv n\hat I(\theta).$$

The quantity $\hat I(\hat\theta)$ is known as the empirical Fisher information for θ. Because H is a sum of n independent and identically distributed terms, we find by the law of large numbers that $H(\theta) \approx -n\,\mathbb{E}\left(\partial^2 \log(p(\boldsymbol{x};\theta))/\partial\theta\partial\theta^{\mathsf{T}}\right) \equiv nI(\theta)$. This $I(\theta)$ is the ordinary (non-empirical) Fisher information for θ. In asymptotic expansions, it is most convenient to work with $nI(\hat\theta)$, but to present the method we work with $H(\hat\theta)$.

The best predictor of $g(\theta)$, with respect to squared error, is

$$\mathbb{E}(g(\theta) \,|\, \mathcal{X}) = \frac{\int g(\theta)\pi_0(\theta) e^{\ell(\theta)}\,\mathrm{d}\theta}{\int \pi_0(\theta) e^{\ell(\theta)}\,\mathrm{d}\theta},$$

which we may write as

$$\frac{\int g_{\mathrm{N}}(\theta) e^{-H_{\mathrm{N}}(\theta)}\,\mathrm{d}\theta}{\int g_{\mathrm{D}}(\theta) e^{-H_{\mathrm{D}}(\theta)}\,\mathrm{d}\theta} \tag{7.10}$$

where, for example, $g_{\mathrm{N}}(\theta)$ is either 1 or $g(\theta)$ or $g(\theta)\pi_0(\theta)$ with $H_{\mathrm{N}}(\theta)$ adjusted accordingly, and similar choices are available for the denominator. We investigate several specific formulations of (7.10) below.

If we take $\hat\theta_{\mathrm{N}}$ and $\hat\theta_{\mathrm{D}}$ to be the minimizers of H_{N} and H_{D} respectively, then a straightforward use of the Laplace approximation yields

$$\widehat{\mathbb{E}}(g(\theta) \,|\, \mathcal{X}) = \frac{g_{\mathrm{N}}(\hat\theta_{\mathrm{N}})|\bar H_{\mathrm{N}}(\hat\theta_{\mathrm{N}})|^{-1/2} e^{-H_{\mathrm{N}}(\hat\theta_{\mathrm{N}})}}{g_{\mathrm{D}}(\hat\theta_{\mathrm{D}})|\bar H_{\mathrm{D}}(\hat\theta_{\mathrm{D}})|^{-1/2} e^{-H_{\mathrm{D}}(\hat\theta_{\mathrm{D}})}} \tag{7.11}$$

where $\bar H_{\mathrm{N}}$ and $\bar H_{\mathrm{D}}$ are the Hessian matrices of H_{N} and H_{D} respectively.

The Laplace approximation is said to be in ***standard form*** when $H_{\mathrm{N}}(\theta) = H_{\mathrm{D}}(\theta)$. In the standard form, the estimate (7.11) simplifies to

$$\widehat{\mathbb{E}}(g(\theta)\,|\,\mathcal{X}) = \frac{g_{\mathrm{N}}(\hat\theta_{\mathrm{N}})}{g_{\mathrm{D}}(\hat\theta_{\mathrm{D}})} = \frac{g_{\mathrm{N}}(\hat\theta_{\mathrm{N}})}{g_{\mathrm{D}}(\hat\theta_{\mathrm{N}})} \tag{7.12}$$

because $\hat\theta_{\mathrm{D}} = \hat\theta_{\mathrm{N}}$ in the standard form. If we take $H_{\mathrm{N}}(\theta) = H_{\mathrm{D}}(\theta) = \ell(\theta)$, so that $g_{\mathrm{N}}(\theta) = g(\theta)\pi_0(\theta)$ and $g_{\mathrm{D}}(\theta) = \pi_0(\theta)$, we obtain $\widehat{\mathbb{E}}(g(\theta)\,|\,\mathcal{X}) = g(\hat\theta)$ where $\hat\theta$ is the maximum likelihood estimator of θ. If instead, we take $H_{\mathrm{N}}(\theta) = \ell(\theta) - \log(\pi_0(\theta))$, with $g_{\mathrm{N}}(\theta) = g(\theta)$ and $g_{\mathrm{D}}(\theta) = 1$, then we obtain $\widehat{\mathbb{E}}(g(\theta)\,|\,\mathcal{X}) = g(\widetilde\theta)$, where $\widetilde\theta$ is the maximum a posteriori (MAP) estimate of θ.

The Laplace approximation is in ***fully exponential form*** when $g_{\mathrm{N}}(\theta) = g_{\mathrm{D}}(\theta)$. For example, we might have $g_{\mathrm{N}}(\theta) = g_{\mathrm{D}}(\theta) = 1$ with $H_{\mathrm{N}}(\theta) = \ell(\theta) - \log(\theta) - \log(g(\theta))$ and $H_{\mathrm{D}}(\theta) = \ell(\theta) - \log(\theta)$. The fully exponential form requires $g(\theta) > 0$. In the fully exponential form, the estimate (7.11) becomes

$$\widehat{\mathbb{E}}(g(\theta)\,|\,\mathcal{X}) = \frac{|\bar H_{\mathrm{N}}(\hat\theta_{\mathrm{N}})|^{-1/2} e^{-H_{\mathrm{N}}(\hat\theta_{\mathrm{N}})}}{|\bar H_{\mathrm{D}}(\hat\theta_{\mathrm{D}})|^{-1/2} e^{-H_{\mathrm{D}}(\hat\theta_{\mathrm{D}})}}. \tag{7.13}$$

It now requires two separate optimizations and the determinants of two different Hessian matrices. In return for this extra work, the method attains higher accuracy. While the standard form has errors of order n^{-1}, the exponential form has errors of order n^{-2} (Tierney and Kadane, 1986). This extra accuracy arises because the numerator and denominator estimate their corresponding quantities with very nearly the same relative error, and so much of that error cancels.

It is possible to attain an error of order n^{-2} from the standard form. To do so, we replace the numerator and denominator in the right hand side of (7.11) by a Taylor expansion taken as far as third order mixed partial derivatives of H_{N} and H_{D}. See Evans and Swartz (2000) for a formula. This process is less convenient than the fully exponential one, especially for a large number p of parameters in θ.

The positivity constraint on $g(\theta)$ from the fully exponential form is a nuisance. Tierney et al. (1989) consider several ways around it. One way is to replace $g(\theta)$ by $g(\theta) + c$ for some $c > 0$, apply the fully exponential approximation, subtract c from the resulting estimate, and use the limit of this process as $c \to \infty$. Another is to work with the moment generating function $M(t) \equiv \mathbb{E}(e^{tg(\theta)}\,|\,\mathcal{X})$. When $M(t)$ exists we can estimate it using the fully exponential form because $e^{tg(\theta)} > 0$. Then $\mathbb{E}(g(\theta)\,|\,\mathcal{X}) = M'(0)$ which can be estimated numerically.

The Laplace approximation is now largely, but not completely, overshadowed by Markov chain Monte Carlo (MCMC). See Chapter 11. One reason is that the Laplace approximation is designed for unimodal functions. When $\pi(\theta\,|\,\mathcal{X})$ has two or more important modes, then the space Θ can perhaps be cut into pieces containing one mode each, and Laplace approximations applied separately and combined, but such a process can be cumbersome. MCMC by

contrast is designed to find and sample from multiple modes, although on some problems it will have difficulty doing so. The Laplace approximation also requires finding the optimum of a d-dimensional function and working with the Hessian at the mode. In some settings that optimization may be difficult, and when d is extremely large, then finding the determinant of the Hessian can be a challenge. Finally, posterior distributions that are discrete or are mixtures of continuous and discrete parts can be handled by MCMC but are not suitable for the Laplace approximation.

The Laplace approximation is not completely superceded by MCMC. In particular, the fully exponential version is very accurate for problems with modest dimension d and large n. When the optimization problem is readily solved then it may provide a much more automatic and fast answer than MCMC does. Furthermore, the Laplace approximation is much faster than MCMC.

The integrated nested Laplace approximation of Rue et al. (2009) is an alternative to MCMC for large scale problems. It uses a weighted sum of Laplace approximations. If the variables θ have components (η, ϕ) then it uses a Laplace approximation to integrate over ϕ given η summed over a weighted set of values η_k for $k = 1, \dots, K$. It is thus a numerical integral over η values of a Laplace approximation over ϕ given η. The statistical context behind η and ϕ can be complex. See Palacios and Minin (2012) for an example in phylodynamics, estimating historical sizes of populations from present day genetic data. See Martino and Riebler (2019) for an introductory account.

7.7 Weighted spaces and tractability

Bahkvalov's Theorem 7.2 establishes a curse of dimension. We cannot succeed well on all f in those large classes $\mathcal{F}$ of high dimensional integrands covered by his result. In many applications, however, one does successfully integrate a high dimensional function. That does not contract Bahkvalov in any way. His curse of dimension remains intact, and we learn that the given problem was not like the worst cases. Weighted space models describe classes of functions in which integration is easier than the classes Bahkvalov studied. Under those additional assumptions, the curse of dimension can be augmented with some tractability results presented here.

The function classes in Bahkvalov's theorems are very broad and are defined only in terms of which partial derivatives exist. While a user's given integrand f might well belong to one of those classes, those classes include lots of functions that in no way resemble f. Then worst case accuracy over $\mathcal{F}$ might not be an accurate description of what happens for f. There are also ways to measure average performance over these infinite classes $\mathcal{F}$, but there again, the average might not be a good description of our problem. If our f is differentiable but $\mathcal{F}$ includes all continuous functions, then the non-differentiable ones might 'out vote' ours in the average error measure. If we take close account of the precise number of derivatives, as Bahkvalov does, then our f might still be in some out-voted subset in some other way as described next.

One well-studied sort of problem where high dimensional integration succeeds has integrands $f(x_1, \dots, x_d)$ that depend in ever weaker ways on x_j as the index j increases. That is, the inputs are not equally important. The integrands may also depend mostly on the inputs x_j one or two at a time, and hardly at all on combinations of many more inputs. Some tractability results presented below, give conditions under which a curse of dimension might not apply, through further assumptions about how f depends on $\boldsymbol{x}$.

To study these properties, write

$$f(\boldsymbol{x}) = \sum_{u \subseteq \{1,2,\dots,d\}} f_u(\boldsymbol{x}) \tag{7.14}$$

where the function $f_u(\boldsymbol{x})$ only depends on those x_j for $j \in u$. The ANOVA decomposition of Appendix A represents f this way, and there are other decompositions.

Using (7.14), we may bound the integration error over $[0,1]^d$ by

$$|\hat{I} - I| \leqslant \sum_u \left| \frac{1}{n} \sum_{i=1}^n f_u(\boldsymbol{x}_i) - \int_{[0,1]^d} f_u(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \right|.$$

If some of the functions f_u are always close to zero or are otherwise easy to integrate, then they might not contribute much to $\hat{I} - I$. In a weighted space analysis, presented below, we introduce weights $\gamma_u > 0$ for each subset u of inputs to f. Larger weights γ_u will allow f_u to be a more complicated and harder to integrate function, while smaller weights γ_u will force f_u to be closer to zero. Other things being equal, γ_u is usually chosen to be a decreasing function of the number of variables included in u (so $\gamma_{\{1,2,3\}} < \gamma_{\{1,2\}}$) and also of the indices of those variables (so $\gamma_{\{1,2,3\}} < \gamma_{\{3,4,5\}}$). All 2^d values γ_u together are represented by $\boldsymbol{\gamma} \in (0, \infty)^{2^d}$.

To shorten some expressions, we introduce notation $1{:}d$ for $\{1, 2, \dots, d\}$. For $\boldsymbol{x} \in \mathbb{R}^d$ and $u \subseteq 1{:}d$ we let $\boldsymbol{x}_u$ be made up of entries x_j with $j \in u$. We also use

$$\partial^u f(\boldsymbol{x}) \equiv \frac{\partial^{|u|}}{\prod_{j \in u} \partial x_j} f(\boldsymbol{x})$$

with $\partial^{\varnothing} f = f$, where $|u|$ is the cardinality of u. In a weighted space model, we suppose that f belongs to the function class

$$\begin{aligned} \mathcal{F}_{\boldsymbol{\gamma}} &= \{f \mid \|f\|_{\boldsymbol{\gamma}} \leqslant 1\}, \quad \text{where} \\ \|f\|_{\boldsymbol{\gamma}}^2 &= \sum_{u \subseteq 1:d} \frac{1}{\gamma_u} \int_{[0,1]^{|u|}} \left(\int_{[0,1]^{d-|u|}} \partial^u f(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x}_{-u} \right)^2 \mathrm{d}\boldsymbol{x}_u. \end{aligned} \tag{7.15}$$

If $\|f\|_{\boldsymbol{\gamma}} = c > 1$, then $f/c \in \mathcal{F}_{\boldsymbol{\gamma}}$. Therefore error bounds for $\mathcal{F}_{\boldsymbol{\gamma}}$ apply to f after rescaling by c. It is usual to include in $\mathcal{F}_{\boldsymbol{\gamma}}$ the limits of any convergent sequence of functions from $\mathcal{F}_{\boldsymbol{\gamma}}$, though we do not pursue that point here.

We can use the ANOVA decomposition to simplify equation (7.15). If $\partial^{1:d} f$ is continuous on $[0,1]^d$ and f_u is from the ANOVA decomposition, then after some simplifications

$$\|f\|_{\boldsymbol{\gamma}}^2 = \sum_{u \subseteq 1:d} \frac{1}{\gamma_u} \int_{[0,1]^{|u|}} |\partial^u f_u(\boldsymbol{x})|^2 \, \mathrm{d}\boldsymbol{x}_u. \tag{7.16}$$

See Exercise 7.6. For example, the contributions of $u = \{j\}$ and $u = \{j,k\}$ are

$$\frac{1}{\gamma_{\{j\}}} \int_0^1 \left| \frac{\partial}{\partial x_j} f_{\{j\}}(\boldsymbol{x}) \right|^2 \mathrm{d}x_j \quad \text{and} \quad \frac{1}{\gamma_{\{j,k\}}} \int_0^1 \int_0^1 \left| \frac{\partial^2}{\partial x_j \partial x_k} f_{\{j,k\}}(\boldsymbol{x}) \right|^2 \mathrm{d}x_j \, \mathrm{d}x_k,$$

respectively. The first one takes a mean squared slope of the main effect $f_{\{j\}}$ for variable j and penalizes it by $1/\gamma_{\{j\}}$. The larger $\gamma_{\{j\}}$ is, the less that term contributes to $\|f\|_{\boldsymbol{\gamma}}$ and by that measure, the more important x_j can be, without putting f over the complexity budget $\|f\|_{\boldsymbol{\gamma}} \leqslant 1$. For $u = \{j,k\}$, it is the mean squared second order mixed partial that is penalized by $1/\gamma_{\{j,k\}}$.

Recall that Bahkvalov's theorems 7.2 and 7.3 measure difficulty through mixed partial derivatives. It is not just that f changes with x_j that complicates a problem, but also that the nature of this change varies with x_k for $k \neq j$ and so on for other indices. The criterion in (7.16) sharply constrains how complicated f_u can be when γ_u is small. The smaller γ_u is, the smaller $|\partial^u f_u|^2$ must be on average, for f to fit into $\mathcal{F}_{\boldsymbol{\gamma}}$.

It is very common to study product weights with $\gamma_u = \prod_{j \in u} \gamma_j$ where $1 \geqslant \gamma_1 \geqslant \gamma_2 \geqslant \cdots \geqslant \gamma_d > 0$. Then sets u with larger cardinality are modeled as no more important, and so are sets u containing larger indices j as described above. Popular choices include $\gamma_j = j^{-\eta}$ for a parameter $\eta > 0$. Some additional weight choices are described in the end notes.

Tractability is a way to describe the integration problem not getting too much harder as $d \to \infty$. For product weights, we can consider $d \to \infty$ by using the first d values of γ_j. Letting $\mathcal{X}_{1:n}$ denote all the points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in [0,1]^d$, the worst case error in $\mathcal{F}_{\boldsymbol{\gamma}}$ is

$$e_{n,d}(\mathcal{X}_{1:n}, \mathcal{F}_{\boldsymbol{\gamma}}) = \sup_{f \in \mathcal{F}_{\boldsymbol{\gamma}}} \left| \frac{1}{n} \sum_{i=1}^n f(\boldsymbol{x}_i) - \int_{[0,1]^d} f(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \right|.$$

If we had no points at all to use, then our best estimate is $\hat{I} = 0$ and we would incur the ***initial error***

$$e_{0,d}(\mathcal{F}_{\boldsymbol{\gamma}}) = \sup_{f \in \mathcal{F}_{\boldsymbol{\gamma}}} \left| \int_{[0,1]^d} f(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \right| = \sup_{f \in \mathcal{F}_{\boldsymbol{\gamma}}} \int_{[0,1]^d} |f(\boldsymbol{x})| \, \mathrm{d}\boldsymbol{x}.$$

Sloan and Woźniakowski (1998) let $n = n_{\boldsymbol{\gamma}}(\varepsilon, d)$ be the minimal number of function values necessary to reduce the initial error by a factor of $\varepsilon > 0$, when we approximate $\int_{[0,1]^d} f(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x}$ by $(1/n) \sum_{i=1}^n f(\boldsymbol{x}_i)$. At this n, there exist some collection $\mathcal{X}_{1:n}$ of points $\boldsymbol{x}_i$ for which

$$e_{n,d}(\mathcal{X}_{1:n}, \mathcal{F}_{\boldsymbol{\gamma}}) \leqslant \varepsilon \times e_{0,d}(\mathcal{F}_{\boldsymbol{\gamma}}).$$

For some weighted spaces this minimal number grows rapidly with dimension. Let $\Gamma_d = \sum_{j=1}^{d} \gamma_j$. They show that

$$n_{\boldsymbol{\gamma}}(\varepsilon, d) \geqslant (1-\varepsilon^2)1.055^{\Gamma_d}.$$

Then, in an equally weighted space with all $\gamma_j = 1$, $n_{\boldsymbol{\gamma}}(\varepsilon, d)$ grows at least exponentially fast with d. Some settings with decreasing γ_j avoid this exponential cost and are considered tractable by the definitions below.

Definition 7.1. The problem of integrating $f \in \mathcal{F}_{\boldsymbol{\gamma}}$ is ***tractable*** if

$$n_{\boldsymbol{\gamma}}(\varepsilon, d) \leqslant C d^q \varepsilon^{-p}$$

for positive constants C, p and q, independent of $d \geqslant 1$ and $\varepsilon > 0$.

Definition 7.2. The problem of integrating $f \in \mathcal{F}_{\boldsymbol{\gamma}}$ is ***strongly tractable*** if

$$n_{\boldsymbol{\gamma}}(\varepsilon, d) \leqslant C \varepsilon^{-p}$$

for positive constants C and p, independent of $d \geqslant 1$ and $\varepsilon > 0$.

Both definitions 7.1 and 7.2 describe problems where the initial error can be reduced without incurring exponential cost. Tractability can be established by proving that the necessary points $\boldsymbol{x}_1, \dots, \boldsymbol{x}_n \in [0,1]^d$ exist. Sometimes those results do not indicate how to find any such points. Other proofs are constructive, meaning that they can be used to produce points $\boldsymbol{x}_i$ with which to compute $\hat{I}$.

Sloan and Woźniakowski (1998) show that integration is strongly tractable if $\sum_{j=1}^{\infty} \gamma_j < \infty$ but their proof is non-constructive and since it has $p = 2$, it is the rate we might expect from plain Monte Carlo. Taking $\gamma_j = j^{-\eta}$ for $\eta > 1$ suffices. Hickernell and Woźniakowski (2000) show that when $\sum_{j=1}^{\infty} \gamma_j^{1/2} < \infty$, then for any $\delta \in (0,1)$, one can attain strong tractability with $p = 1/(1-\delta)$, so that the initial error can be reduced almost as fast as n^{-1}, though their result is also non-constructive. Taking $\gamma_j = j^{-\eta}$ for $\eta > 2$ suffices. Nuyens and Cools (2006a,b) give constructions for weighted spaces. Their algorithm costs $O(dn\log(n))$ to construct the points, a marked improvement over earlier methods. Their constructions are lattice rules, a kind of quasi-Monte Carlo. Chapters 15, 16, and 17 are on quasi-Monte Carlo.

In favorable weighted spaces, the error drops rapidly with n for all d. The rates from Bahkvalov's theorems have an $n^{-r/d}$ factor in them. The most favorable weighted space rates are almost $O(n^{-1})$, comparable to the Bahkvalov bounds for $r = d$ (or $r = d/2$ for random inputs).

When our function f is not really dominated by low dimensional terms, then the rapid decay of initial error might not be enough to ensure an accurate high dimensional integration. It is not enough for f to have continuous mixed partial derivatives of all orders. Consider

$$f_d(\boldsymbol{x}) = \prod_{j=1}^{d} \sqrt{12}(x_j - 1/2). \tag{7.17}$$

This function has integral 0 over $[0,1]^d$. It has variance 1 for $\boldsymbol{x} \sim \mathbf{U}[0,1]^d$, and in that sense it is equally hard to integrate by plain Monte Carlo for any d. From (7.15),

$$\|f\|_{\boldsymbol{\gamma}} = \frac{12^d}{\gamma_{1:d}} \tag{7.18}$$

See Exercise 7.8. With $\gamma_j = j^{-\eta}$, we get $\|f\|_{\boldsymbol{\gamma}} = (d!)^\eta 12^d$. Now $f/c \in \mathcal{F}_{\boldsymbol{\gamma}}$ for $c = (d!)^\eta 12^d$, and so the error in integrating f/c can be brought below ε at a cost of $n = C\varepsilon^{-p}$ function evaluations. The error in integrating f is then below

$$c \times \varepsilon = 12^d (d!)^\eta \varepsilon$$

at that n. Now suppose that we want to ensure that $|\hat{I} - I| = |\hat{I}| \leqslant \epsilon$. Then we require

$$\varepsilon \leqslant \frac{\epsilon}{12^d (d!)^\eta} \quad \text{and} \quad n \geqslant C\epsilon^{-p} 12^{dp} (d!)^{\eta p}.$$

In this example, larger d requires greater rescaling of f_d to fit into $\mathcal{F}_{\boldsymbol{\gamma}}$, leading to a larger initial error, a greater reduction ε in the initial error to attain error ϵ, and finally a required sample size that grows rapidly with d. For functions like f_d that are not really dominated by low dimensional terms, the curse of dimension can reappear within the initial error.

7.8 Sparse grids

Sparse grids are another way to integrate multidimensional functions at much less than the exponential cost of using a full d-dimensional grid of evaluation points. They are also known as Smolyak rules because they were first proposed by Smolyak (1963).

With sparse grids, we evaluate the integrand only at a tiny subset of the points in a full grid. See Figure 7.1 for examples of some evaluation points for the case $d = 2$. In this section, we present some of the main ideas behind sparse grids but defer most of the details to the cited references.

We begin with an infinite sequence of one-dimensional quadrature rules

$$Q_k(f) = \sum_{i=1}^{m_k} w_{i,k} f(x_{i,k}), \quad k \geqslant 1.$$

For instance, for integration over $[0,1]$, we might take $Q_1(f) = f(1/2)$ and then for $k \geqslant 2$, use a trapezoid rule

$$Q_k(f) = \frac{1}{2^k}\Big(\frac{1}{2}f(0) + \sum_{i=1}^{2^{k-1}-1} f\Big(\frac{i}{2^{k-1}}\Big) + \frac{1}{2}f(1)\Big),$$

with $m_k = 2^{k-1} + 1$. There are many other choices. Choosing $Q_1(f) = f(1/2)$ and, for $k \geqslant 2$, rescaling the Clenshaw-Curtis rule from $[-1,1]$ to $[0,1]$ is effective. That rescaling involves replacing nodes $x \in [-1,1]$ by $(x+1)/2$ and

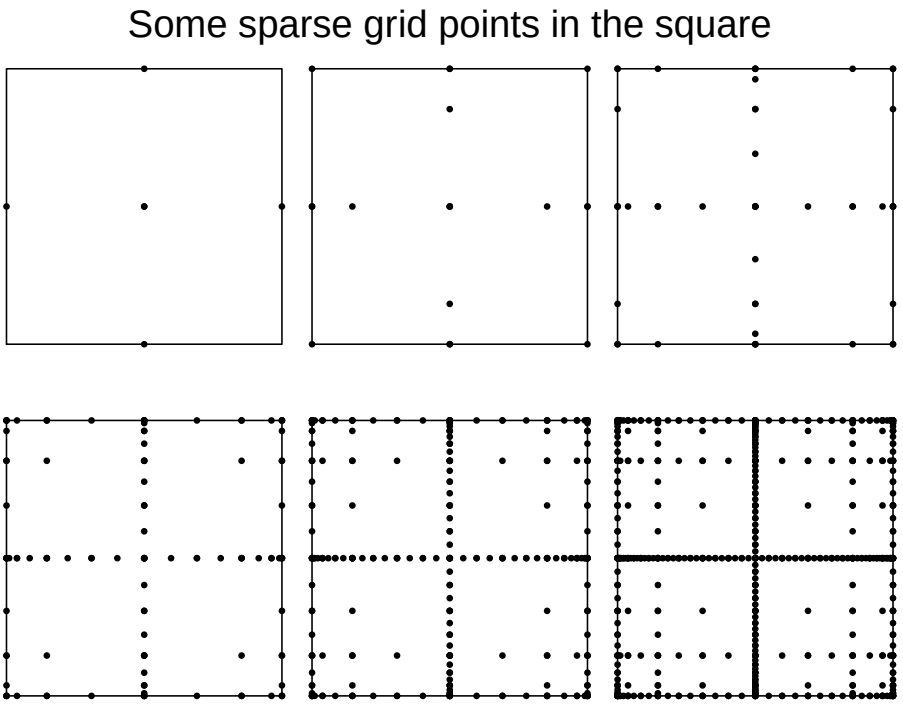

Figure 7.1: Each panel shows a sparse grid in $[0,1]^2$, based on univariate Clenshaw-Curtis rules. The corresponding quadrature rules are unequally weighted sums of function values over these sparse grids of points.

replacing weights w by $w/2$. It will prove important to have $m_1 = 1$ and it is best to have nested rules, in which any point $x_{i,k}$ appears again as a point $x_{i',k+1}$ for some i'.

If f has the smoothness needed by our quadrature method, then

$$\lim_{k\to\infty} Q_k(f) = \int_0^1 f(x)\,\mathrm{d}x.$$

Now, for $k \geqslant 1$, let $\Delta_k(f) = Q_k(f) - Q_{k-1}(f)$, defining $Q_0(f) = 0$. Then taking a telescoping sum

$$\sum_{k=1}^{\infty} \Delta_k(f) = \int_0^1 f(x)\,\mathrm{d}x = \lim_{\ell\to\infty} \sum_{k=1}^{\ell} \Delta_k(f). \tag{7.19}$$

For $f(\boldsymbol{x})$ defined on $[0,1]^d$, we define

$$Q_{k_1} \otimes Q_{k_2} \otimes \cdots \otimes Q_{k_d}(f) = \sum_{i_1=1}^{m_{k_1}} \cdots \sum_{i_d=1}^{m_{k_d}} \prod_{j=1}^{d} w_{i_j,k_j} f(x_{i_1,k_1}, \ldots, x_{i_d,k_d}). \tag{7.20}$$

Equation (7.20) approximates $\int_{[0,1]^d} f(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$ using a weighted sum of function evaluations on a grid of $\prod_{j=1}^d m_{k_j}$ points. We similarly define $\Delta_{k_1}\otimes\cdots\otimes\Delta_{k_d}$. For example, with $d=2$,

$$\begin{aligned}\Delta_{k_1}\otimes\Delta_{k_2} &= (Q_{k_1}-Q_{k_1-1})\otimes(Q_{k_2}-Q_{k_2-1})\\ &= Q_{k_1}\otimes Q_{k_2}-Q_{k_1-1}\otimes Q_{k_2}-Q_{k_1}\otimes Q_{k_2-1}+Q_{k_1-1}\otimes Q_{k_2-1}.\end{aligned}$$

In the multidimensional case, the telescoping sum becomes

$$\sum_{k_1=1}^{\infty}\cdots\sum_{k_d=1}^{\infty}\Delta_{k_1}\otimes\cdots\otimes\Delta_{k_d}(f)=\int_{[0,1]^d}f(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}. \tag{7.21}$$

Let $\boldsymbol{k}=(k_1,\dots,k_d)$ and $\Delta_{\boldsymbol{k}}=\Delta_{k_1}\otimes\cdots\otimes\Delta_{k_d}$, and define

$$\|\boldsymbol{k}\|_\infty=\max_{1\leqslant j\leqslant d}k_j\quad\text{and}\quad\|\boldsymbol{k}\|_1=\sum_{j=1}^d k_j.$$

Sampling on a regular m_ℓ^d grid, we can approximate I by

$$\sum_{\boldsymbol{k}:\,\|\boldsymbol{k}\|_\infty\leqslant\ell}\Delta_{\boldsymbol{k}}(f), \tag{7.22}$$

which we know to be expensive and ineffective for large d.

For sparse grids we use

$$Q_{\ell,d}(f)=\sum_{\boldsymbol{k}:\,\|\boldsymbol{k}\|_1\leqslant\ell+d-1}\Delta_{\boldsymbol{k}}(f),\quad\ell\geqslant1 \tag{7.23}$$

instead. Then, the first rule is

$$Q_{1,d}(f)=\Delta_{(1,\dots,1)}(f)=Q_1\otimes Q_1\otimes\cdots\otimes Q_1(f)$$

which requires m_1^d evaluations of f. There is thus a strong advantage to having $m_1=1$. If we also have $x_{1,1}=1/2$, then

$$Q_{1,d}(f)=f\Big(\frac12,\dots,\frac12\Big),$$

which seems to be the most reasonable one point rule. In (7.23), we compute $Q_{\ell,d}(f)$ via $\Delta_{\boldsymbol{k}}$ which is built up from differences $\Delta_k=Q_k-Q_{k-1}$. When the points of Q_{k-1} also appear in Q_k, then we only have to evaluate f at the m_k-m_{k-1} new points of Q_k in order to apply Δ_k.

Let $N_{\ell,d}$ be the number of points evaluation points $\boldsymbol{x}\in[0,1]^d$ used in the sparse grid estimate (7.23). Then

$$N_{\ell,d}\leqslant\sum_{\boldsymbol{k}:\,\|\boldsymbol{k}\|_1\leqslant\ell+d-1}\prod_{j=1}^d m_{k_j}$$

ℓ	m_ℓ	$N_{\ell,8}$	m_ℓ^8	$m_\ell \log_2(m_\ell)^7/N_{\ell,8}$
1	1	1	1	0.0
2	3	17	6.6×10^3	4.4
3	5	145	3.9×10^5	13.0
4	9	849	4.3×10^7	34.0
5	17	3,937	7.0×10^9	82.0
6	33	15,713	1.4×10^{12}	170.0

Table 7.3: Example sparse grid sizes for $d = 8$. The third column is from Table 3 of Gerstner and Griebel (1998). The fourth column is the approximate size of a non-sparse grid. The final column divides an asymptotic estimate for $N_{\ell,8}$ by its value.

and is generally less than that because the formula above counts some of the input points multiple times. The widely used sparse grid points, such as those based on Clenshaw-Curtis rules, have $m_k = O(2^k)$. Then

$$N_{\ell,d} = O(2^\ell \ell^{d-1}) = O(m_\ell \log_2(m_\ell)^{d-1})$$

compared to m_ℓ^d for the non-sparse grid (7.22) (Gerstner and Griebel, 1998). Table 7.3 shows an example for $d = 8$. Lemma 1 of Müller-Gronbach (1998) gives a sharper result, that

$$\lim_{\ell\to\infty} \frac{N_{\ell,d}}{2^{\ell-d}(\ell-1)^{d-1}/(d-1)!} = 1,$$

for fixed d, though this asymptotic equivalence may be slow to take hold.

The sparse grid estimate can be written

$$Q_{\ell,d}(f) = \sum_{i=1}^{N_{\ell,d}} W_i f(\boldsymbol{x}_i)$$

where the sum is over all $N_{\ell,d}$ points $\boldsymbol{x}_i$ (ordered somehow) in the sparse grid and for each of them W_i is the total of all the weights applied to $\boldsymbol{x}_i$ from all of the $\Delta_{\boldsymbol{k}}$ that include it. The bookkeeping underlying W_i is somewhat complicated. See Gerstner and Griebel (1998). Some of these W_i can be negative, even when all of the Q_k have only non-negative $w_{i,k}$, owing to the differences $Q_k - Q_{k-1}$ appearing in the formula for $Q_{\ell,d}$. While $\sum_i W_i = 1$, the appearance of negative weights can potentially make $\sum_i |W_i|$ much larger than one, affecting the numerical stability of the sparse grid estimate. Novak and Ritter (1997, Lemma 2) show that $\sum_i |W_i| = O(\log(N_{\ell,d})^{d-1})$, even for non-nested sparse grid rules. The implied coefficient of $\log(N_{\ell,d})^{d-1}$ depends on d. Gerstner and Griebel (1998) describe a recursive computation of $Q_{\ell,d}$ that reduces roundoff error.

To describe the accuracy of sparse grids with the Clenshaw-Curtis quadrature rules, we present results from Novak and Ritter (1997). For $\alpha = (\alpha_1, \ldots, \alpha_d)$

with integer $\alpha_j \geqslant 0$, let

$$f^{(\alpha)}(\boldsymbol{x}) = \frac{\partial^{\|\alpha\|_1} f(\boldsymbol{x})}{\partial x_1^{\alpha_1} \cdots \partial x_d^{\alpha_d}}.$$

Now, consider these two sets of functions, for $r \geqslant 1$ and $d \geqslant 1$,

$$\mathcal{C}_d^r = \{f : [0,1]^d \to \mathbb{R} \mid \max_{\|\alpha\|_1 \leqslant r} \|f^{(\alpha)}\|_\infty \leqslant 1\} \quad \text{and} \tag{7.24}$$

$$\mathcal{F}_d^r = \{f : [0,1]^d \to \mathbb{R} \mid \max_{\|\alpha\|_\infty \leqslant r} \|f^{(\alpha)}\|_\infty \leqslant 1\}. \tag{7.25}$$

Novak and Ritter (1997) defined functions on $[-1,1]^d$ but that difference will not affect the results we see here. The sets above are the unit balls in the function classes that they define. For $d > 1$, the unit ball $\mathcal{F}_d^r$ constrains more partial derivatives of f than $\mathcal{C}_d^r$ does, and so $\mathcal{F}_d^r \subsetneq \mathcal{C}_d^r$.

Let $Q_{\ell,d}$ be a sparse grid quadrature using the Clenshaw-Curtis rules and let $N = N_{\ell,d}$ be the number of points in it. Novak and Ritter (1997) show that there are finite constants $c_{d,r}$ such that

$$\sup_{f \in \mathcal{F}_d^r} |Q_{\ell,d}(f) - I(f)| \leqslant c_{r,d} N^{-r} (\log N)^{(d-1)(r+1)}, \quad \text{and} \tag{7.26}$$

$$\sup_{f \in \mathcal{C}_d^r} |Q_{\ell,d}(f) - I(f)| \leqslant c_{r,d} N^{-r/d} (\log N)^{(d-1)(r+1)}. \tag{7.27}$$

The results for $\mathcal{C}_d^r$ are within logarithmic factors of the bounds in Bahkvalov's theorem. Under the stricter conditions described by $\mathcal{F}_d^r$, we see that a much better rate can be attained. When using sparse grids with Clenshaw-Curtis, we do not have to know the largest r for which f can be scaled to belong to $\mathcal{C}_d^r$ or $\mathcal{F}_d^r$. The error satisfies (7.26) for all of those r even though the method is defined without regard to r.

The rate (7.26) does not set in until N is large enough to make use of r'th derivative information in the sample. As N increases, ever larger values of r become relevant. For fixed r the bound in (7.26) decreases linearly versus N on a log-log plot. With the effective value of r increasing with N, we can get an ever steepening concave plot of error versus N in a log-log plot. Holtz (2010, Chapter 4.2) shows some examples of this phenomenon for very smooth f with $d < 10$.

Gerstner and Griebel (1998) include several numerical examples of sparse grid quadrature. In some of their examples, using the Clenshaw-Curtis rule is much better than the trapezoid rule, while in others the results are close. Similarly, in some of their examples a great improvement arises from using a Gauss rule instead of Clenshaw-Curtis, despite the larger sample sizes required from a non-nested base rule. In other examples the method based on Clenshaw-Curtis is almost equally effective as using a Gauss rule.

The cost of sparse grids eventually grows rapidly with dimension. Suppose that we want to include points that differ from 1/2 in s or more of their d components. Then we must have $\ell \geqslant s+1$. Estimating $N_{\ell,d}$ by $2^\ell \ell^{d-1}$ we see

that it will become very expensive to have large d with even modest ℓ. The estimate $2^{\ell-d}(\ell-1)^{d-1}/(d-1)!$ from Müller-Gronbach (1998) is more favorable but we must recall that it applies to $\ell \to \infty$ for fixed d, not $d \to \infty$ for fixed ℓ.

Delayed basis sequences, introduced by Petras (2003) are a method to cope with the high cost of sparse grids in high dimensions. Given a sequence of sample sizes $m_1, m_2, m_3, \ldots$, a delayed sequence might use sizes $\tilde{m}_k$ from a sequence like

$$m_1,\ m_2, m_2,\ m_3, m_3, m_3,\ m_4 \cdots .$$

The resulting sizes $\tilde{m}_k$ increase slowly with k. When two consecutive rules Q_{k-1} and Q_k are identical, then $\Delta_k = 0$ and need not be computed. Holtz (2010) has several examples.

Chapter end notes

Davis and Rabinowitz (1984, Chapters 2–4) have a very comprehensive discussion of one-dimensional quadrature rules. The emphasis is on problems where one can obtain highly accurate answers. They give special attention to integration over unbounded intervals and to integrands that oscillate. Their Chapter 2.12 covers unbounded integrands over bounded intervals. Evans and Swartz (2000) describe many multivariate quadrature methods.

Clenshaw-Curtis rules and their history are described in Trefethen (2008). An almost identical proposal was made earlier by Fejér (1933). They were presented by Clenshaw and Curtis (1960). For $k \geqslant 2$, the points x_i of a Clenshaw-Curtis rule on $m = 2^{k-1} + 1$ points are the points at which Chebychev polynomials of the first kind, $T_{m-1}(x) = \cos((m-1)\arccos(x))$, attain their extrema on $[-1, 1]$.

Adaptive integration

Press et al. (2007) present some adaptive quadrature methods. The idea is to take a rule like the midpoint rule and use it with a higher density of points in some subintervals than others. For example, subintervals where the function varies more, such as having a larger $|f''|$, get more points while other intervals get fewer points. The information on where the function varies most is gathered along with the function values. They also present some multivariate adaptive quadrature methods.

There is some mild controversy about the use of adaptive methods. There are theoretical results showing that adaptive methods cannot improve significantly over non-adaptive ones. There are also theoretical and empirical results showing that adaptive methods may do much better than non-adaptive ones. These results are not contradictory. Instead, they make different assumptions about the problem. For a survey of conditions when adaptation helps, see Novak (1996).

Hickernell et al. (2013) present a two stage adaptive Monte Carlo method to estimate a mean to within a guaranteed error size with a guaranteed confidence

level. The guarantees depend on an assumption that

$$\int_{\mathbb{R}^d} (f(\boldsymbol{x}) - I(f))^4 p(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x} \leqslant B\sigma^4(f) \tag{7.28}$$

where $\sigma^2(f) = \int_{\mathbb{R}^d}(f(\boldsymbol{x}) - I(f))^2 p(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}$, for some reasonable upper bound B. Here $p(\boldsymbol{x})$ is a probability density function from which they sample. Condition (7.28) defines a 'cone' of functions in the sense that if f satisfies it then so does cf for $c > 0$. It is not necessarily convex because $(f(\boldsymbol{x}) + g(\boldsymbol{x}))/2$ might not satisfy (7.28) even if both f and g do. Convexity of this type is a common assumption in theorems that show adaptive methods cannot improve on non-adaptive ones. As a result, those theorems do not apply to that two stage algorithm.

Hickernell and Rugama (2016) present an adaptive quasi-Monte Carlo algorithm with guaranteed accuracy under an assumption that the integrand's coefficients in a Walsh basis expansion decay in a reasonably, though not necessarily perfectly, monotone way.

Weighted spaces

Hickernell (1998) used weighted spaces to describe problems where multivariable interactions have decreasing importance as the number of variables involved increases. His weights took the form $\gamma_u = \gamma^{|u|}$ for $0 < \gamma < 1$, and they serve to make higher order dependence in f less important. Sloan and Woźniakowski (1998) added a dependence on the index of the variable and introduced the product weights described in §7.7. If $f \in \mathcal{F}_{\boldsymbol{\gamma}}$ for product weights with $\gamma_j = j^{-\eta}$ for $\eta > 0$, then f from (7.16), we find that $\int(\partial^u f_u(\boldsymbol{x}))^2\,\mathrm{d}\boldsymbol{x}$ cannot be very large for large $|u|$. It follows that $\int f_u(\boldsymbol{x})^2\,\mathrm{d}\boldsymbol{x}$ cannot be very large either (Owen, 2019), providing a sense in which those spaces have small effective dimension.

Product and order weights (POD weights) take the form $\gamma_u = G_{|u|}\prod_{j\in u}\gamma_j$, for some positive constants $G_{|u|}$. These POD weights were developed to solve integration problems arising from partial differential equations with random coefficients. See Kuo et al. (2012). Some additional choices for weights are included in the survey of quasi-Monte Carlo methods given by Dick et al. (2013).

There have been a few papers about choosing the weighted space for a given application. Wang and Sloan (2006) say that usually tuned lattice rules consider higher order interactions to be more important than lower ones and don't necessarily do better than digital nets (of Chapter 15) on finance problems. Their idea to tune weights is to choose them so that typical functions have similar sensitivity indices to the given function. L'Ecuyer and Munger (2012) estimate weights via $\gamma_u = \sigma_u^2$ from an ANOVA decomposition.

Wang and Sloan (2007) note two difficulties with weighted spaces: it can be hard to choose the weights, and, sometimes the given function is not smooth enough to be in the weighted space. They propose a method of picking weights proportional to $\gamma_u = 6^{|u|/2} w_u(f)$ where

$$w_u(f) = \int_{[0,1]^{|u|}} \Bigl(\int_{[0,1]^{d-|u|}} \partial^u f(\boldsymbol{x})\,\mathrm{d}\boldsymbol{x}_{-u}\Bigr)^2 \mathrm{d}\boldsymbol{x}_u$$

is the quantity appearing for subset u in $\|f\|^2_{\boldsymbol{\gamma}}$.

Equation (7.15) is a squared 'unanchored norm'. There is a corresponding anchored norm version. Let $\boldsymbol{x}_u{:}\mathbf{1}_{-u} \in [0,1]^d$ be the point $\boldsymbol{y}$ with $y_j = x_j$ for $j \in u$ and $y_j = 1$ for $j \notin u$. Then the anchored version has

$$\|f\|^2_{\boldsymbol{\gamma}} = \sum_{u \subseteq 1:d} \frac{1}{\gamma_u} \int_{[0,1]^{|u|}} |\partial^u f(\boldsymbol{x}_u{:}\mathbf{1}_{-u})|^2 \,\mathrm{d}\boldsymbol{x}_u. \tag{7.29}$$

The analysis of integration in weighted spaces is very dependent on reproducing kernel Hilbert space methods (Berlinet and Thomas-Agnan, 2011) which are beyond the prerequisites assumed for this book. Dick and Pillichshammer (2010, Chapter 2) provide a concise introduction geared to integration. More results are in Ritter (2007) and for a comprehensive treatment see Novak and Woźniakowski (2008, 2010, 2012), especially volume II. Those methods allow one to construct the worst case integrand f for given inputs $\boldsymbol{x}_1, \dots, \boldsymbol{x}_n \in [0,1]^d$ and given weights $\boldsymbol{\gamma}$.

Sparse grids

Sparse grids were introduced by Smolyak (1963). They also go by the names 'hyperbolic cross points', 'Boolean method' and 'discrete blending method'. Bungartz and Griebel (2004) is a comprehensive reference on sparse grid methods. Gerstner and Griebel (1998) focus on their use in numerical integration and Holtz (2010) considers applications to finance.

The sparse grids presented in §7.8 retained terms $\Delta_{\boldsymbol{k}}(f)$ for $\sum_{j=1}^d k_j \leqslant \ell + d - 1$. This treats all input dimensions equally. One can instead work adaptively using $\boldsymbol{k}$ with $\sum_{j=1}^d v_j k_j$ below some bound where $v_j > 0$ is larger for more important variables j and is smaller for less important ones. It is also possible to use different quadrature rules on each variable as one might when estimating $\int_0^1 \int_0^\infty f(x_1, x_2) \exp(-x_2) \,\mathrm{d}x_1 \,\mathrm{d}x_2$. See Gerstner and Griebel (1998).

Holtz (2010) finds that sparse grids are extremely effective on integrands that are very smooth and are dominated by low dimensional terms in an ANOVA. He remarks that they are not very robust to non-smoothness of the integrands.

It is possible to use the sparse grid construction on s variables at a time. If we want to integrate $f : [0,1]^{ds} \to \mathbb{R}$ then we can replace the basic integration rules $\sum_{i=1}^{m_k} f(x_{i,k})$ for $x_{i,k} \in [0,1]$ by rules $\sum_{i=1}^{m_k} f(\boldsymbol{x}_{i,k})$ with $\boldsymbol{x}_{i,k} \in [0,1]^s$. See Dick et al. (2007) who base a sparse grid rule on quasi-Monte Carlo and randomized quasi-Monte Carlo rules.

Exercises

7.1. A test for Churg-Strauss syndrome (Masi et al., 1990) correctly detects it in 99.7% of affected patients. The test falsely reports Churg-Strauss in 15% of patients without the disease. Suppose that we sample m people at a clinic and x of them test positive for Churg-Strauss. We are interested in the fraction

$p \in (0,1)$ of visitors to the clinic that really have Churg-Strauss. For a uniform prior distribution on p the posterior distribution of p is proportional to

$$\pi_u(p \,|\, x) = (p \times 0.997 + (1-p) \times 0.15)^x ((1-p) \times 0.85 + p \times 0.003)^{m-x}.$$

The clinic finds that $x = 10$ out of $m = 30$ patients test positive.

a) Use the midpoint rule with $n = 1000$ to estimate the posterior mean of p given the data:

$$\mathbb{E}(p \,|\, x) = \frac{\int_0^1 \pi_u(p \,|\, x) p \,\mathrm{d}p}{\int_0^1 \pi_u(p \,|\, x) \,\mathrm{d}p}.$$

b) Use the midpoint rule with $n = 1000$ to estimate the posterior variance

$$\mathrm{Var}(p \,|\, x) = \frac{\int_0^1 \pi_u(p \,|\, x)(p - \mathbb{E}(p \,|\, x))^2 \,\mathrm{d}p}{\int_0^1 \pi_u(p \,|\, x) \,\mathrm{d}p}.$$

c) One third of the patients tested positive. Use the midpoint rule with $n = 1000$ to estimate the posterior probability that $p \leqslant 1/3$,

$$\mathbb{P}(p \leqslant 1/3 \,|\, x) = \frac{\int_0^{1/3} \pi_u(p \,|\, x) \,\mathrm{d}p}{\int_0^1 \pi_u(p \,|\, x) \,\mathrm{d}p}.$$

7.2. Our theoretical understanding of the midpoint rule suggests that the error in the first two parts of Exercise 7.1 should decrease as $O(1/n^2)$. The third part should not attain this rate because $f(p) = \mathbb{1}_{p \leqslant 1/3}$ is not twice differentiable. Use the midpoint rule with $n = 10^6$ as if it were the exact answer. Then plot the absolute error versus n of the midpoint rule for $n = 10^j$ for $j = 1, 2, 3, 4, 5$, for all three of the parts of Exercise 7.1. Does the predicted n^{-2} rate hold for the first two parts? What rate appears to hold in the third part?

7.3. Solve the system of equations

$$\int_0^n x^p \,\mathrm{d}x = \sum_{i=0}^n w_{in} i^n, \quad p = 0, \ldots, n$$

for w_{in}, for $n = 1, 2, 3, 4, 5, 6$. Use your results to give the next symmetric rule after Bode's rule.

7.4. Mike uses the midpoint rule with n points to approximate $\int_0^1 f(x) \,\mathrm{d}x$ by $\hat{I}_n$, and Trish uses the trapezoid rule with the same intervals on the same problem to get $\widetilde{I}_n$.

a) How many function values does Trish use?

b) How many distinct function values did the two people need?

c) Describe how they could combined their data to fit a larger midpoint rule.

d) If f is very smooth, do you expect differences in accuracy among choices $(\hat{I}+\widetilde{I})/2$, $(2\hat{I}-\widetilde{I})$ and $(2\widetilde{I}-\hat{I})$?

7.5. Verify that equation (7.5) is correct for $f(x) = x^3$. Show that it fails for $f(x) = x^4$.

7.6. Prove equation (7.16).

7.7. The function f_d in (7.17) has mean 0 and variance 1, for any dimension $d \geqslant 1$. It does however cost $O(nd)$ computation to evaluate it n times, and in that sense it becomes harder to integrate by Monte Carlo as d increases.

a) Construct a sequence of functions $g_d(\boldsymbol{x}) = \prod_{j=1}^{d} h_j(x_j)$ on $[0,1]$ so that $\mathbb{E}(g_d(\boldsymbol{x})) = 0$ and $\mathbb{E}(g_d(\boldsymbol{x})^2) = 1$ for $\boldsymbol{x} \sim \mathbf{U}[0,1]^d$.

b) Find $\|f\|_{\boldsymbol{\gamma}}$ for your functions g_d, when $\gamma_j = j^{-\eta}$ for $\eta > 0$.

7.8. Prove equation (7.18).

Bibliography

Berlinet, A. and Thomas-Agnan, C. (2011). *Reproducing kernel Hilbert spaces in probability and statistics*. Springer Science & Business Media, New York.

Bungartz, H.-J. and Griebel, M. (2004). Sparse grids. *Acta numerica*, 13:147–269.

Clenshaw, C. W. and Curtis, A. R. (1960). A method for numerical integration on an automatic computer. *Numerische Mathematik*, 2(1):197–205.

Davis, P. J. and Rabinowitz, P. (1984). *Methods of Numerical Integration*. Academic Press, San Diego, 2nd edition.

Dick, J., Kuo, F. Y., and Sloan, I. H. (2013). High-dimensional integration: the quasi-Monte Carlo way. *Acta Numerica*, 22:133–288.

Dick, J., Leobacher, G., and Pillichshammer, F. (2007). Randomized Smolyak algorithms based on digital sequences for multivariate integration. *IMA journal of numerical analysis*, 27(4):655–674.

Dick, J. and Pillichshammer, F. (2010). *Digital sequences, discrepancy and quasi-Monte Carlo integration*. Cambridge University Press, Cambridge.

Dimov, I. T. (2008). *Monte Carlo methods for applied scientists*. World Scientific, Singapore.

Evans, M. J. and Swartz, T. (2000). *Approximating integrals by Monte Carlo and deterministic methods*. Oxford University Press, Oxford.

Fejér, L. (1933). Mechanische quadraturen mit positiven cotesschen zahlen. *Mathematische Zeitschrift*, 37(1):287–309.

Gerstner, T. and Griebel, M. (1998). Numerical integration using sparse grids. *Numerical algorithms*, 18(3–4):209–232.

Hickernell, F. J. (1998). A generalized discrepancy and quadrature error bound. *Mathematics of Computation*, 67:299–322.

Hickernell, F. J., Jiang, L., Liu, Y., and Owen, A. B. (2013). Guaranteed conservative fixed width confidence intervals via Monte Carlo sampling. In Dick, J., Kuo, F. Y., Peters, G., and Sloan, I. H., editors, *Monte Carlo and Quasi-Monte Carlo Methods 2012*, pages 105–128. Springer, Berlin.

Hickernell, F. J. and Rugama, L. A. J. (2016). Reliable adaptive cubature using digital sequences. In Cools, R. and Nuyens, D., editors, *Monte Carlo and Quasi-Monte Carlo Methods 2014*, pages 367–383. Springer, Cham, Switzerland.

Hickernell, F. J. and Woźniakowski, H. (2000). Integration and approximation in arbitrary dimensions. *Advances in Computational Mathematics*, 12:25–58.

Holtz, M. (2010). *Sparse grid quadrature in high dimensions with applications in finance and insurance*. Springer-Verlag, Berlin.

Kuo, F. Y., Schwab, C., and Sloan, I. H. (2012). Quasi-Monte Carlo finite element methods for a class of elliptic partial differential equations with random coefficients. *SIAM Journal on Numerical Analysis*, 50(6):3351–3374.

Lubinsky, D. S. and Rabinowitz, P. (1984). Rates of convergence of Gaussian quadrature for singular integrands. *Mathematics of Computation*, 43(167):219–242.

L'Ecuyer, P. and Munger, D. (2012). On figures of merit for randomly-shifted lattice rules. In Plaskota, L. and Woźniakowski, H., editors, *Monte Carlo and Quasi-Monte Carlo Methods 2010*, pages 133–159. Springer.

Martino, S. and Riebler, A. (2019). Integrated nested Laplace approximations (INLA). Technical Report arXiv:1907.01248, Norwegian University of Science and Technology.

Masi, A. T., Hunder, G. G., Lie, J. T., Michel, B. A., Bloch, D. A., Arend, W. P., Calabrese, L. H., Edworthy, S. M., Fauci, A. S., Leavitt, R. Y., Lightfoot Jr., R. W., McShane, D. J., Mills, J. A., Stevens, M. B., Wallace, S. L., and Zvaifler, N. J. (1990). The American College of Rheumatology 1990 criteria for the classification of Churg-Strauss syndrome (allergic granulomatosis and angiitis). *Arthritis & Rheumatism*, 33(8):1094–1100.

Müller-Gronbach, T. (1998). Hyperbolic cross designs for approximation of random fields. *Journal of statistical planning and inference*, 66(2):321–344.

Novak, E. (1996). On the power of adaption. *Journal of Complexity*, 12(3):199–238.

Novak, E. and Ritter, K. (1997). The curse of dimension and a universal method for numerical integration. In Nürnberger, G., Schmidt, J. W., and Walz, G., editors, *Multivariate approximation and splines*, pages 177–187. Springer.

Novak, E. and Woźniakowski, H. (2008). *Tractability of Multivariate Problems: Linear Information*, volume I. European Mathematical Society, Zurich.

Novak, E. and Woźniakowski, H. (2010). *Tractability of Multivariate Problems: Standard Information for Functionals*, volume II. European Mathematical Society, Zurich.

Novak, E. and Woźniakowski, H. (2012). *Tractability of Multivariate Problems: Standard Information for Linear Operators*, volume III. European Mathematical Society, Zurich.

Nuyens, D. and Cools, R. (2006a). Fast algorithms for component-by-component construction of rank-1 lattice rules in shift-invariant reproducing kernel hilbert spaces. *Mathematics of Computation*, 75(254):903–920.

Nuyens, D. and Cools, R. (2006b). Fast component-by-component construction of rank-1 lattice rules with a non-prime number of points. *Journal of Complexity*, 22:4–28.

Owen, A. B. (2019). Effective dimension of some weighted pre-Sobolev spaces with dominating mixed partial derivatives. *SIAM Journal on Numerical Analysis*, 57(2):547–562.

Palacios, J. A. and Minin, V. N. (2012). Integrated nested Laplace approximation for Bayesian nonparametric phylodynamics. In de Freitas, N. and Murphy, K., editors, *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence*, pages 726–735.

Petras, K. (2003). Smolyak cubature of given polynomial degree with few nodes for increasing dimension. *Numerische Mathematik*, 93(4):729—753.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (2007). *Numerical recipes: the art of scientific computing*. Cambridge University Press, Cambridge, 3rd edition.

Ritter, K. (2007). *Average-case analysis of numerical problems*. Springer-Verlag, Berlin.

Rue, H., Martino, S., and Chopin, N. (2009). Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *Journal of the Royal Statistical Society, Series B*, 71(2):319–392.

Sloan, I. H. and Woźniakowski, H. (1998). When are quasi-Monte Carlo algorithms efficient for high dimensional integration? *Journal of Complexity*, 14:1–33.

Smolyak, S. A. (1963). Quadrature and interpolation formulas for tensor products of certain classes of functions. *Soviet Mathematics Doklady*, 4:240–243.

Tierney, L. and Kadane, J. B. (1986). Accurate approximateions for posterior moments and marginal densities. *Journal of the American Statistical Association*, 81(393):82–86.

Tierney, L., Kass, R. E., and Kadane, J. B. (1989). Fully exponential Laplace approximations to expectations and variances of nonpositive functions. *Journal of the American Statistical Association*, 84(407):710–716.

Trefethen, L. N. (2008). Is Gauss quadrature better than Clenshaw–Curtis? *SIAM Review*, 50(1):67–87.

Wang, X. and Sloan, I. H. (2006). Efficient weighted lattice rules with applications to finance. *SIAM Journal on Scientific Computing*, 28(2):728–750.

Wang, X. and Sloan, I. H. (2007). Brownian bridge and principal component analysis: towards removing the curse of dimensionality. *IMA Journal of Numerical Analysis*, 27(4):631–654.

Wong, R. (1989). *Asymptotic approximation of integrals*. Academic Press, San Diego.