An Introduction to Numerical Integration

Micole De Vera[*]

27 October 2024

1 Introduction

Sometimes, we want to compute a definite integral that might not have closed form. There are two applications (somewhat related) in econometrics of particular interest:

1. We might be interested in computing a particular expectation. For example, consider a random variable X whose distribution can be described by pdf ϕ. We are interested in the expected value of some function $f(X)$. The expectation is an integral

$$\mathbb{E}[g(X)] = \int_{\Omega} f(x)\phi(x)\ dx.$$

2. We might be interested in maximum likelihood estimation of a model with latent variables that requires us to compute the integrated likelihood. That is,

$$\widehat{\theta} = \underset{\theta}{\operatorname{argmax}} \int_{\Omega} p(x \mid \eta; \theta) p(\eta \mid \theta)\ d\eta.$$

   This is key to estimating random coefficient logit models, as in Berry et al. (1995), for instance.

In this note, I discuss deterministic and stochastic integration rules that may be useful in computational (approximate) evaluation of integrals, as in the above situations.

2 Deterministic integration rule: Gaussian quadrature

To introduce Gaussian quadrature, we first discuss another deterministic integration rule: the trapezoidal rule with an equi-spaced grid. Consider the problem of evaluating the following integral

$$\int_a^b f(x)\ dx.$$

[*] This is an extended version of my personal study notes. Comments and suggestions are appreciated. The views expressed in this note are those of the author and do not necessarily reflect the position of the Banco de España or the Eurosystem. All errors are mine. Personal website: https://micoledevera.github.io/.

The domain $[a, b]$ is partitioned into n subintervals of equal length so that the length of each subinterval is $h = (b - a)/n$. The corresponding partition points are therefore $x_i = a + ih$ for $i = 0, ..., n$. The approximation to the integral is then given by

$$\int_a^b f(x)\ dx \approx \sum_{i=0}^{n-1} \frac{h}{2} \left(f(x_i) + f(x_{i+1}) \right).$$

The idea behind this is that the function f is approximated by linear interpolating splines on the grid $x_0, ..., x_n$.[1] The area within each subinterval, then, corresponds to an area of a trapezoid where the bases are $f(x_i)$ and $f(x_{i+1})$ with height h.

In general, the integration rule can be written as

$$\int_a^b f(x)\ dx \approx \sum_{i=0}^{n} \omega_i f(x_i),$$

that is, as a weighted sum of function values on a grid. Integration rules that can be written as a weighted sum, such as the one above, are called *quadratures.*

In the trapezoidal rule discussed above, there are two "user-specified" objects: (i) the choice of interpolating polynomial, and (ii) the number of subintervals. The choice of the number of subintervals n fixes the subinterval length and grid (which corresponds to where we will evaluate the function). The choice of interpolating polynomial, along with the grid points, determine the weights. In general, increasing n increases the precision of the integration rule.

Can we do better by choosing the weights and grid points jointly? Intuitively, this effectively increases our degrees of freedom. By taking advantage of these additional degrees of freedom, we are able to improve on the trapezoidal rule along some dimensions.

Gaussian quadrature. One way to select the weights and grid points is the so-called Gaussian quadrature. We select the weights and grid points such that, for some fixed n,

$$\int_a^b f(x)\ dx = \omega_0 f(x_0) + \omega_1 f(x_1) + ... + \omega_n f(x_n),$$

for all polynomials f of order m (made as large as possible). That is, the weights and grid are chosen such that the rule is *exact* for polynomials up to order m. While we are still fixing the number of grid points, unlike the trapezoidal rule, we are not imposing that they be equidistant points in the domain.

For such a rule to be exact for any arbitrary polynomial of order m, it is sufficient to show that the rule is exact for the corresponding basis functions $\{1, x, ..., x^{m-1}, x^m\}$. This

[1] We can create an alternative integration rule by approximating the function f with higher-order polynomial spline interpolation. This class of rules is more generally known as the "Newton-Cotes formulas".

provides $m+1$ restrictions for $2(n+1)$ unknown weights and grid points. Thus, for a fixed n, the rule would be exact for polynomials up to $2n+1$.

Let us consider a concrete example. We will find the Gaussian quadrature with $n=1$ on the domain $[-1,1]$ so that our integration rule will be

$$\int_{-1}^{1} f(x)\ dx \approx \omega_0 f(x_0) + \omega_1 f(x_1).$$

With $n=1$, our quadrature will be exact for polynomials up to order 3 which implies the following restrictions:

$$\begin{array}{ll} f(x)=1: & \int_{-1}^{1} 1\ dx = 2 = \omega_0+\omega_1 \\ f(x)=x: & \int_{-1}^{1} x\ dx = 0 = \omega_0 x_0 + \omega_1 x_1 \\ f(x)=x^2: & \int_{-1}^{1} x^2\ dx = \frac{2}{3} = \omega_0 x_0^2 + \omega_1 x_1^2 \\ f(x)=x^3: & \int_{-1}^{1} x^3\ dx = 0 = \omega_0 x_0^3 + \omega_1 x_1^3 \end{array}$$

From the second restriction, we know $\omega_0 x_0 = -\omega_1 x_1$. From the fourth, we know $\omega_0 x_0^3 = -\omega_1 x_1^3$. Therefore, $x_0^2 = x_1^2$. Thus, $x_0 = -x_1$, ignoring the degenerate solution where $x_0 = x_1$. Going back to the second, we find $\omega_0 = \omega_1$. From the first restriction, $\omega_0 = \omega_1 = 1$. From the third restriction, $x_0 = -1/\sqrt{3}$ and $x_1 = 1/\sqrt{3}$. The resulting integration rule is therefore

$$\int_{-1}^{1} f(x)\ dx \approx f\left(-\frac{1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right).$$

The corresponding quadratures with more points could be obtained similarly.

A change of variable allows us to generalize the quadrature on $[-1,1]$ to definite integrals over the arbitrary domain $[a,b]$. Consider a quadrature rule for the domain $[-1,1]$ with grid $\{x_0, x_1, ..., x_n\}$ and corresponding weights $\{\omega_0, ..., \omega_n\}$. Then,

$$\begin{aligned} \int_a^b f(u)\ du &= \int_{-1}^{1} f\left(\frac{b-a}{2}x + \frac{a+b}{2}\right)\frac{b-a}{2}\ dx \\ &= \frac{b-a}{2}\int_{-1}^{1} f\left(\frac{b-a}{2}x + \frac{a+b}{2}\right)\ dx \\ &\approx \frac{b-a}{2}\sum_{i=0}^{n} \omega_i f\left(\frac{b-a}{2}x_i + \frac{a+b}{2}\right). \end{aligned}$$

Other Gaussian quadrature rules. The quadrature rules generated in the way above perform best when the integrand is close to a polynomial or can be approximated closely by a series of polynomials.[2] We can generalize the quadrature rules to deal with integrals of the

[2] Gaussian quadrature is consistent for Riemann integrable functions in which case we can get the approximation arbitrarily precise by increasing the number of grid points.

form

$$\int_D w(x)f(x)\ dx,$$

where $w(x)$ is a weight function. The rules will still be exact for polynomials f up to certain order. Thus, the integrand need not be well-approximated as a polynomial directly, but is close to a polynomial after dividing by $w(x)$. In the case above, $w(x) = 1$. However, we may also have rules corresponding to $w(x) = \exp(-x)$ or $w(x) = \exp(-x^2)$ which may be useful in computing integrals relating to the exponential or normal distributions, respectively. The resulting quadrature rules are such that

$$\int_D w(x)f(x)\ dx \approx \sum_{i=0}^{n} \omega_i f(x_i).$$

Table 1: Gaussian Quadrature Rules

Domain, D	Weighting function, $w(x)$	Orthogonal polynomials
$[-1,1]$	1	Legendre
$[0,\infty)$	$\exp(-x)$	Laguerre
$(-\infty,\infty)$	$\exp(-x^2)$	Hermite

Orthogonal polynomials. It can be shown that the grid points for a Gaussian quadrature of order n is given by the roots of the orthogonal polynomial $p_n(x)$ corresponding to the inner product with weights $w(x)$, i.e.,

$$\langle f, g\rangle_w = \int_D w(x)f(x)g(x)\ dx.$$

For $w(x) = 1$ on the domain $[-1, 1]$, the relevant orthogonal polynomials are the Legendre polynomials. Thus, the above quadrature is also sometimes recognized as the Gauss-Legendre quadrature. For $w(x) = \exp(-x^2)$ on the domain $(-\infty, \infty)$, we have the Gauss-Hermite quadrature corresponding to roots of the Hermite polynomials. Once the roots are known, then the weights can be obtained by solving a linear system just as above (the restrictions are linear in the weights).

Application: Expectations related to the normal distribution. Consider a random variable $X \sim \mathcal{N}(\mu, \sigma^2)$. Suppose that we are interested in computing the following expectation

$$\mathbb{E}[f(X)] = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) f(x)\ dx.$$

Consider the change of variable $u = (x-\mu)/\sqrt{2\sigma^2}$. Then,

$$\mathbb{E}[f(X)] = \int_{-\infty}^{\infty} \frac{1}{\sqrt{\pi}} \exp(-u^2) f(\mu + \sqrt{2\sigma^2}u)\ dx \approx \frac{1}{\sqrt{\pi}} \sum_{i=0}^{n} \omega_i f(\mu + \sqrt{2\sigma^2}x_i),$$

where $\{x_0, ..., x_n\}$ and $\{\omega_0, ..., \omega_n\}$ are the grid points and weights corresponding to a Gauss-Hermite quadrature rule.

Extension to multiple integrals. The quadratures we have thus far considered are for unidimensional definite integrals. How do we construct quadratures that work for integrals over multiple variables? To illustrate possible solutions, let us consider again the problem of finding expectations but now of a function of two random variables, say,

$$\iint f(x,y)\phi(x,y)\ dx\ dy,$$

where $\phi(x,y)$ is the appropriate pdf.

It is intuitive that a quadrature rule that takes a tensor product of unidimensional quadrature rules may work. To illustrate, in the two-dimensional case,

$$\begin{aligned}
\iint f(x,y)\phi(x,y)\ dx\ dy &= \iint f(x,y)\phi(x)\phi(y \mid x)\ dx\ dy \quad \text{(by def of conditional expectations)} \\
&= \int \underbrace{\int f(x,y)\phi(x)\ dx}\ \phi(y \mid x)\ dy \\
&\approx \int \left[\sum_{i=0}^{n} \omega_i f(x_i, y)\phi(x_i)\right] \phi(y \mid x_i)\ dy \quad \text{(quadrature for inner integral)} \\
&\approx \sum_{j=0}^{n} \omega_j \sum_{i=0}^{n} \omega_i f(x_i, y_j)\phi(x_i)\phi(y_j \mid x_i) \quad \text{(quadrature for outer integral)} \\
&= \sum_{j=0}^{n}\sum_{i=0}^{n} \omega_i\omega_j f(x_i, y_j)\phi(x_i)\phi(y_j \mid x_i),
\end{aligned}$$

which shows that it is similar to using a quadrature where the grid points are the Cartesian product of the individual unidimensional grid points with corresponding weights that are multiplied. This also requires us to be able to write down clearly the conditional distribution of one variable conditional on the other.

Note that the number of function evaluations increase exponentially with the number of dimensions. If we start with individual n-point unidimensional quadratures which require n function evaluations each, then using that in a 2-dimensional integration will require n^2 function evaluations. However, some of these function evaluations will have very small effective contributions to the overall approximation, especially if the effective weights $(\omega_i\omega_j)$ are small.

As such, one can do *pruning* which involves dropping grid points where the effective weights are smaller than some chosen threshold. In cases where the function is computationally costly to evaluate or when one needs to repeatedly do variations of the same integration multiple times, this may save substantial computing time. Jäckel (2005) provides a discussion of other considerations in the context of multivariate Gauss-Hermite quadrature.

3 Stochastic integration rules: Monte Carlo integration

In the specific examples I mentioned in the introduction, there are natural stochastic rules that give us approximations of the integrals. In particular, Monte Carlo Integration. To illustrate, suppose we are interested in the expectation example above. For some large N, we can draw $(x_1, ..., x_N) \sim \phi$ then

$$\widehat{\mathbb{E}}[f(X)] = \frac{1}{N}\sum_{i=1}^{N} f(x_i).$$

This method of taking expectations is underpinned by some law of large numbers. It is important to note that this method can easily be extended into multiple dimensions by drawing samples from multivariate distributions.

Generic integrals. The same idea can be used to get approximations for integrals that are not necessarily expectations. Consider the integral

$$\int_0^1 f(x)\ dx$$

over the interval $[0, 1]$. This integral can be interpreted as an expectation of the function $f(X)$ where X is uniformly distributed on $[0, 1]$. Then, we can draw independent draws $\{x_i\}_{i=1}^N$ from $U[0, 1]$ and

$$\int_0^1 f(x)\ dx \approx \frac{1}{N}\sum_{i=1}^{N} f(x_i).$$

The extension to multidimensional integrals and integrals with arbitrary bounds is straightforward.

As mentioned, a LLN would assure that as $N \to \infty$, the average converges to the integral of interest. A central limit theorem tells us something about the large sample distribution of the approximation of the integral. Regardless of the number of dimensions of the integral, the error of the integral will be of the order $N^{-1/2}$ proportional to the variance of the integrand. Intuitively, there are then two natural ways to decrease the variance of the approximation: (1) increase the number of samples drawn N, or (2) make the integrand smoother.

Variance reduction: Importance sampling. We start with the following insight:

$$\int f(x)\ dx = \int \frac{f(x)}{p(x)} p(x)\ dx.$$

That is, the expectation of $f(X)$ with X distributed uniform is the same as the expectation of $\frac{f(x)}{p(x)}$ where X is distributed with pdf p. Then,

$$\int f(x)\ dx \approx \frac{1}{N} \sum_{i=1}^{N} \frac{f(x)}{p(x)},$$

where $(x_1, ..., x_N)$ are independent draws from a distribution with pdf p. The performance of this approximation depends on how $\frac{f(x)}{p(x)}$ looks like. We choose p to be such that (i) it is a distribution we can draw from, and (ii) $\frac{f(x)}{p(x)}$ is close to constant.

Variance reduction: Control variates. Consider a function $h(x)$ such that the integral

$$\int h(x)p(x)\ dx = \mu$$

is known. Then,

$$\int f(x)p(x)\ dx = \int \left[f(x) + \alpha(h(x) - \mu)\right] p(x)\ dx,$$

for any arbitrary constant α. Moreover, for a random sample $\{x_1, ..., x_N\} \sim p$,

$$\int f(x)p(x)\ dx \approx \frac{1}{N} \sum_{i=1}^{N} \left[f(x_i) + \alpha(h(x_i) - \mu)\right]$$

is a consistent estimator of the integral. The variance of this estimator is proportional to

$$\text{Var}(f(x_i) + \alpha(h(x_i) - \mu)) = \text{Var}(f(x_i)) + \alpha^2 \text{Var}(h(x_i)) + 2\alpha \text{Cov}(f(x_i), h(x_i)),$$

so it may be possible to get lower variance if $f(x)$ and $h(x)$ are correlated and we choose α strategically. The α that minimizes this variance is

$$\alpha = -\frac{\text{Cov}(f(x_i), h(x_i))}{\text{Var}(h(x_i))}$$

so the resulting variance is $\text{Var}(f(x_i))(1 - \rho^2)$ where $\rho = \text{Corr}(f(x_i), h(x_i))$.

We note, however, that the estimator with the optimal α is not feasible. In practice, we consider the feasible estimator

$$\int f(x)p(x)\ dx \approx \frac{1}{N}\sum_{i=1}^{N}\left[f(x_i)+\hat{\alpha}(h(x_i)-\mu)\right],$$

where $\hat{\alpha}$ is an estimate of α obtained from replacing $\mathrm{Cov}(f(x_i),h(x_i))$ and $\mathrm{Var}(h(x_i))$ by their finite sample analogs.

Variance reduction: Antithetic variates. This method of variance reduction is based on the idea that an average over a random sample is less efficient than an average over a sample with negative correlation (but correct marginal distribution). For simplicity, consider even N and consider two samples $\{x_1,...,x_{N/2}\}\sim p$ and $\{y_1,...,y_{N/2}\}\sim p$. Then,

$$\int f(x)p(x)\ dx \approx \frac{1}{N}\sum_{i=1}^{N/2}\left[h(x_i)+h(y_i)\right]$$

will be more efficient than an estimator based on a random sample of size N if $h(x_i)$ and $h(y_i)$ are negatively correlated.

The correlation of $h(x_i)$ and $h(y_i)$ depends on the correlation of x_i and y_i, and the shape of h. It is not easy, in general, to obtain a strategy to draw (x_i,y_i) with the right correlation and correct marginals (pdf p) so that $h(x_i)$ and $h(y_i)$ are negatively correlated. A case where we can generate a generic rule is the following: consider uniform draws $\{u_1,...,u_{N/2}\}$. Then $\{1-u_1,...,1-u_{N/2}\}$ is also distributed uniformly on $[0,1]$. Moreover, if f is monotone, then $f(u_i)$ and $f(1-u_i)$ are negatively correlated (Rubinstein, 1981). Then,

$$\int_0^1 f(x)\ dx \approx \frac{1}{N}\sum_{i=1}^{N/2}\left[f(u_i)+f(1-u_i)\right].$$

This can be made slightly more general to take expectations with general distributions. In particular, we take advantage of the inverse transform methods of drawing random samples. Let F be the cdf corresponding to the density p. Then $\{F^{-1}(u_1),...,F^{-1}(u_{N/2})\}$ and $\{F^{-1}(1-u_1),...,F^{-1}(1-u_{N/2})\}$ are distributed according to F. Consider the estimator

$$\int h(x)p(x)\ dx \approx \frac{1}{N}\sum_{i=1}^{N/2}\left[h(F^{-1}(u_i))+h(F^{-1}(1-u_i))\right].$$

Since F is monotone, then so is F^{-1}. As such, if h is monotone, $h(F^{-1}(u_i))$ and $h(F^{-1}(1-u_i))$ are negatively correlated and we obtain efficiency gains.

Variance reduction: Quasi Monte Carlo. Another way to get more "efficient" Monte Carlo integral approximations is to change the way draws are "sampled". The assurances we have from randomly drawing from a PDF is about the expected densities after an arbitrary large number of drawns. In actuality, draws may tend to be clumped. This clumping tends to be wasteful because points close together do not give much additional information about the function we are integrating. We go back to the ideas in Gaussian quadrature: can we do better with a predetermined set of evaluation points (that mimic a random draw in some sense)?

The answer is given by the theory of discrepancies. We want to somehow measure how good a set of points represents the uniform distribution. One of the discrepancies we can use is the so-called star discrepancy. More precisely, the star discrepancy between a set of points $\{x_1, ..., x_N\}$ and the uniform distribution is

$$D_N^* = \sup_{B \in \mathcal{B}} \left| \sum_{i=1}^{N} \frac{\mathbb{1}(x_i \in B)}{N} - \mu(B) \right|,$$

where $\mathcal{B}$ is the set of anchored boxed, that is, a box with vertices in the origin and $x \in [0,1]^d$. And $\mu(B)$ is the Lebesgue measure of the set B which in this case is just the volume of B. This also corresponds to the Kullback-Leibler divergence. It can be shown that there are deterministic $\{x_i\}$ that have lower discrepancy than a random uniform draw. Moreover, the Koksma-Hlawka inequality tells us that the precision or the error of an integral approximation using a set of points is bounded above by the product of the discrepancy of the sequence and a measure of smoothness of the function (Hardy-Krause total variation). Unfortunately, the improved accuracy of QMC methods are lost in settings with high dimensions or when the integrand is not smooth (Morokoff and Caflisch, 1995).

How do we construct such "low-discrepancy" sequences? A class of generators is called "digital nets". The simplest is the radical inverse sequence or the so-called van der Corput sequence, defined in one-dimension. Let $b \geq 2$ be an integer base. Then any non-negative integer n can be written in this base such that $n = \sum_{k=1}^{\infty} n_k b^{k-1}$ for $n_k \in \{0, 1, ..., b-1\}$. If $b = 2$ then this is just the binary representation of non-negative integers. The corresponding inverse radical function in base b is then $\phi_b(n) = \sum_{k=1}^{\infty} n_k b^{-k}$ which lies in $[0,1)$. To illustrate: since $13 = 1101_2$, then

$$\phi_2(13) = 1 \times \frac{1}{2} + 0 \times \frac{1}{2^2} + 1 \times \frac{1}{2^3} + 1 \times \frac{1}{2^4} = \frac{11}{16}.$$

Then, a sequence $\{x_1, ..., x_N\}$ can be obtained from the radical inverse transformation on the sequence $\{1, ..., N\}$. This can be extended to multiple dimensions where we choose a different prime integer base for each dimension. This extension is the so-called Halton Sequence. In general, sequences generated this way have star discrepancy $D_N^* = O((\log n)^d/n)$.[3] The so-called Sobol and Faure sequences are alternative extensions of this idea in multiple dimensions.

Since the sequences are deterministic, though we can get assurances on precision for a fixed sequence, we cannot get finite sample error estimates. One way to “randomize” the QMC is by rotating the sequences. For a random U drawn from a uniform distribution on $[0,1]^d$, we can get the “rotated” sequence

$$x_i' = x_i + U \pmod 1,$$

where the addition and remainder operations are taken componentwise. With a number of these sequences, we may obtain different integral estimates $I_1, ..., I_M$. We can also get a combined estimate $\hat{I} = \frac{1}{M}\sum_{m=1}^{M} I_m$ and this may be more precise for certain f.

[3] For large d, the $(\log n)^d$ term may dominate and therefore the performance of QMC may deteriorate in high dimensions.

References

Berry, Steven, James Levinsohn, and Ariel Pakes (1995) "Automobile Prices in Market Equilibrium," *Econometrica*, 63 (4), 841–890, http://www.jstor.org/stable/2171802. (cited in page 1)

Jäckel, Peter (2005) "A note on multivariate Gauss-Hermite quadrature," *London: ABN-Amro. Re*, http://www.jaeckel.org/ANoteOnMultivariateGaussHermiteQuadrature.pdf. (cited in page 6)

Morokoff, William J. and Russel E. Caflisch (1995) "Quasi-Monte Carlo Integration," *Journal of Computational Physics*, 122 (2), 218–230, https://doi.org/10.1006/jcph.1995.1209. (cited in page 9)

Rubinstein, Reuven Y. (1981) *Monte Carlo Integration and Variance Reduction Techniques*, Chap. 4, 114–157: John Wiley Sons, Ltd, https://doi.org/10.1002/9780470316511.ch4. (cited in page 8)