Numerical Analysis Hilary Term 2025
Lecture 11: Gauss quadrature

Terminology: Quadrature $\equiv$ numerical integration

Goal: given a (continuous) function $f : [a, b] \to \mathbb{R}$, compute its integral $I = \int_a^b f(x)\,\mathrm{d}x$, as accurately as possible.

The simplest idea is to subdivide the interval into n subintervals of equal length and use a rectangle rule:

$$\int_0^1 f(x)\,\mathrm{d}x \approx \frac{1}{n}\sum_{i=0}^{n-1} f\left(\frac{i}{n}\right) \qquad \text{[left sums]},$$

$$\int_0^1 f(x)\,\mathrm{d}x \approx \frac{1}{n}\sum_{i=0}^{n-1} f\left(\frac{i+1}{n}\right) \qquad \text{[right sums]},$$

$$\int_0^1 f(x)\,\mathrm{d}x \approx \frac{1}{n}\sum_{i=0}^{n-1} f\left(\frac{i+1/2}{n}\right) \qquad \text{[midpoint sums]}.$$

In general, the above methods have $\mathcal{O}(1/n)$ accuracy, but we want to do better.

Idea: Approximate and Integrate. Find a polynomial p_n from data $\{(x_k, f(x_k))\}_{k=0}^n$ by Lagrange interpolation (lecture 1), and integrate $\int_{x_0}^{x_n} p_n(x)\,\mathrm{d}x =: I_n$. Ideally, $I_n = I$ or at least $I_n \approx I$. Is this true?

If we choose x_k to be equispaced points in $[a, b]$, the resulting I_n is known as the Newton-Cotes quadrature. This method is actually quite unstable and inaccurate, and a much more accurate and elegant quadrature rule exists: Gauss quadrature. In this lecture we cover this beautiful result involving orthogonal polynomials.

Preparations: Suppose that w is a weight function, defined, positive and integrable on the open interval (a, b) of $\mathbb{R}$.

Lemma. Let $\{\phi_0, \phi_1, \ldots, \phi_n, \ldots\}$ be orthogonal polynomials for the inner product $\langle f, g\rangle = \int_a^b w(x)f(x)g(x)\,\mathrm{d}x$. Then, for each $k = 0, 1, \ldots,$ ϕ_k has k distinct roots in the interval (a, b).

Proof. Since $\phi_0(x) \equiv \text{const.} \neq 0$, the result is trivially true for $k = 0$. Suppose that $k \geq 1$: $\langle \phi_k, \phi_0\rangle = \int_a^b w(x)\phi_k(x)\phi_0(x)\,\mathrm{d}x = 0$ with ϕ_0 constant implies that $\int_a^b w(x)\phi_k(x)\,\mathrm{d}x = 0$. Since $w(x) > 0$ for $x \in (a, b)$, $\phi_k(x)$ must change sign in (a, b), i.e., ϕ_k has at least one root in (a, b).

Suppose that there are ℓ points $a < r_1 < r_2 < \cdots < r_\ell < b$ where ϕ_k changes sign for some $1 \leq \ell \leq k$. That is, $\phi_k(r_l) = 0$ and $\phi_k(r_l - \epsilon)$ has a different sign than $\phi_k(r_l + \epsilon)$ for $\epsilon > 0$ sufficiently small. Then

$$q(x) = \prod_{j=1}^{\ell}(x - r_j) \times \text{ the sign of } \phi_k \text{ on } (r_\ell, b)$$

has the same sign as ϕ_k on (a, b). Hence

$$\langle \phi_k, q\rangle = \int_a^b w(x)\phi_k(x)q(x)\,\mathrm{d}x > 0.$$

From Lecture 10, we know that the above integral must be zero if q is of degree $k-1$, and so q (which is of degree ℓ) must be of degree $\geq k$, i.e., $\ell \geq k$. However, ϕ_k is of exact degree k, and therefore the number of its distinct roots, ℓ, must be $\leq k$. Hence $\ell = k$, and ϕ_k has k distinct roots in (a, b). □

Application to quadrature. The above lemma leads to very efficient quadrature rules. As we shall see, it answers the question: how should we choose the quadrature points $x_0, x_1, \ldots, x_n$ in the quadrature rule

$$\int_a^b w(x)f(x)\,\mathrm{d}x \approx \sum_{j=0}^{n} w_j f(x_j) \tag{1}$$

so that the rule is exact for polynomials of degree as high as possible? (The case $w(x) \equiv 1$ is the most common.)

Recall: Suppose that $f \in \Pi_n$. The Lagrange interpolating polynomial

$$p_n = \sum_{j=0}^{n} f(x_j)L_{n,j} \in \Pi_n$$

is unique, so $f \in \Pi_n \implies p_n \equiv f$ for any choice of interpolation points, and moreover

$$\int_a^b w(x)f(x)\,\mathrm{d}x = \int_a^b w(x)p_n(x)\,\mathrm{d}x = \sum_{j=0}^{n} f(x_j)\int_a^b w(x)L_{n,j}(x)\,\mathrm{d}x = \sum_{j=0}^{n} w_j f(x_j),$$

exactly provided that

$$w_j = \int_a^b w(x)L_{n,j}(x)\,\mathrm{d}x. \tag{2}$$

Theorem. Suppose that $x_0 < x_1 < \cdots < x_n$ are the roots of the degree $n+1$ orthogonal polynomial ϕ_{n+1} with respect to the inner product

$$\langle g, h\rangle = \int_a^b w(x)g(x)h(x)\,\mathrm{d}x.$$

Then, the quadrature formula (1) with weights (2) is exact whenever $f \in \Pi_{2n+1}$.

Proof. Let $p \in \Pi_{2n+1}$. Then by the Division Algorithm $p(x) = q(x)\phi_{n+1}(x) + r(x)$ with $q, r \in \Pi_n$. So

$$\int_a^b w(x)p(x)\,\mathrm{d}x = \int_a^b w(x)q(x)\phi_{n+1}(x)\,\mathrm{d}x + \int_a^b w(x)r(x)\,\mathrm{d}x = \sum_{j=0}^{n} w_j r(x_j) \tag{3}$$

since the integral involving $q \in \Pi_n$ is zero by the lemma above and the other is integrated exactly since $r \in \Pi_n$. Finally $p(x_j) = q(x_j)\phi_{n+1}(x_j) + r(x_j) = r(x_j)$ for $j = 0, 1, \ldots, n$ as the x_j are the roots of ϕ_{n+1}. So (3) gives

$$\int_a^b w(x)p(x)\,\mathrm{d}x = \sum_{j=0}^{n} w_j p(x_j),$$

where w_j is given by (2) whenever $p \in \Pi_{2n+1}$. $\square$

Optimality: For an n-point quadrature rule, there are n quadrature points and n weights, giving a total of $2n$ degrees of freedom ("variables") to choose to maximize the degree of polynomials integrated exactly. Since $2n = \dim \Pi_{2n-1}$, $2n - 1$ is the largest possible degree we can integrate exactly with an n-point quadrature rule. Thus, the above theorem (an $(n+1)$-point quadrature rule) is optimal in this sense.

The quadrature rules in the above theorem are called **Gauss quadratures**.

- $w(x) \equiv 1$, $(a, b) = (-1, 1)$: Gauss–Legendre quadrature.
- $w(x) = (1 - x^2)^{-1/2}$ and $(a, b) = (-1, 1)$: Gauss–Chebyshev quadrature.
- $w(x) = \mathrm{e}^{-x}$ and $(a, b) = (0, \infty)$: Gauss–Laguerre quadrature.
- $w(x) = \mathrm{e}^{-x^2}$ and $(a, b) = (-\infty, \infty)$: Gauss–Hermite quadrature.

They give better accuracy than Newton–Cotes quadrature for the same number of function evaluations.

Other intervals: By the linear change of variable $t = (2x - a - b)/(b - a)$, which maps $[a, b] \to [-1, 1]$, we can evaluate for example

$$\int_a^b f(x)\,\mathrm{d}x = \int_{-1}^{1} f\left(\frac{(b-a)t + b + a}{2}\right)\frac{b-a}{2}\,\mathrm{d}t \simeq \frac{b-a}{2}\sum_{j=0}^{n} w_j f\left(\frac{b-a}{2}t_j + \frac{b+a}{2}\right),$$

where $\simeq$ denotes "quadrature" and the t_j, $j = 0, 1, \ldots, n$, are the roots of the $n+1$-st degree Legendre polynomial.

Unbounded intervals: For unbounded intervals (a, b) ($a = -\infty$ and/or $b = \infty$), there are two possibilities. If we have a weight function so that $\int_a^b w(x)p(x)\,\mathrm{d}x < \infty$ for any polynomial $p \in \Pi_n$, $n = 0, 1, \ldots,$, then there exists orthogonal polynomials and an a corresponding Gauss quadrature rule (e.g. Gauss-Laguerre and Gauss-Hermite). If the weight function does not satisfy this property, then we typically truncate the unbounded interval to some bounded interval (e.g. truncate $(-\infty, \infty)$ to $(-N, N)$ for some large N) and use a quadrature rule on $(-N, N)$, ignoring the contribution from $(-\infty, \infty)\setminus(-N, N)$.

Exercise (positivity of weights): (non-examinable) Show that the weights $\{w_j\}_{j=0}^n$ (2) of Gauss quadrature are positive by choosing

$$f(x) = \prod_{\substack{j=0\\ j\neq i}}^{n} \frac{(x - x_j)^2}{(x_i - x_j)^2} \in \Pi_{2n}, \qquad i = 0, 1, \ldots, n,$$

and using the exactness of the quadrature rule. The positivity of the weights is an important property that helps ensure the numerical stability of the Gauss rules.

Example. 2-point Gauss–Legendre quadrature: $\phi_2(t) = t^2 - \frac{1}{3} \implies t_0 = -\frac{1}{\sqrt{3}}$, $t_1 = \frac{1}{\sqrt{3}}$ and

$$w_0 = \int_{-1}^{1} \frac{t - \frac{1}{\sqrt{3}}}{-\frac{1}{\sqrt{3}} - \frac{1}{\sqrt{3}}} \, \mathrm{d}t = -\int_{-1}^{1} \left(\frac{\sqrt{3}}{2} t - \frac{1}{2} \right) \mathrm{d}t = 1,$$

with $w_1 = 1$, similarly. So e.g., changing variables $x = (t+3)/2$,

$$\int_1^2 \frac{1}{x} \, \mathrm{d}x = \frac{1}{2} \int_{-1}^{1} \frac{2}{t+3} \, \mathrm{d}t \simeq \frac{1}{3 + \frac{1}{\sqrt{3}}} + \frac{1}{3 - \frac{1}{\sqrt{3}}} = 0.6923077\ldots .$$

Note that the trapezium rule (also two evaluations of the integrand) gives

$$\int_1^2 \frac{1}{x} \, \mathrm{d}x \simeq \frac{1}{2} \left[\frac{1}{2} + 1 \right] = 0.75,$$

whereas $\int_1^2 \frac{1}{x} \, \mathrm{d}x = \ln 2 = 0.6931472\ldots$.

Theorem. Error in Gauss quadrature: suppose that $f^{(2n+2)}$ is continuous on (a,b). Then

$$\int_a^b w(x) f(x) \, \mathrm{d}x = \sum_{j=0}^{n} w_j f(x_j) + \frac{f^{(2n+2)}(\eta)}{(2n+2)!} \int_a^b w(x) \prod_{j=0}^{n} (x - x_j)^2 \, \mathrm{d}x,$$

for some $\eta \in (a,b)$.

Proof. The proof is based on the Hermite interpolating polynomial H_{2n+1} to f on $x_0, x_1, \ldots, x_n$. [Recall that $H_{2n+1}(x_j) = f(x_j)$ and $H'_{2n+1}(x_j) = f'(x_j)$ for $j = 0, 1, \ldots, n$.] The error in Hermite interpolation is

$$f(x) - H_{2n+1}(x) = \frac{1}{(2n+2)!} f^{(2n+2)}(\eta(x)) \prod_{j=0}^{n} (x - x_j)^2$$

for some $\eta = \eta(x) \in (a,b)$ which is also continuous. Now $H_{2n+1} \in \Pi_{2n+1}$, so

$$\int_a^b w(x) H_{2n+1}(x) \, \mathrm{d}x = \sum_{j=0}^{n} w_j H_{2n+1}(x_j) = \sum_{j=0}^{n} w_j f(x_j),$$

the first identity because Gauss quadrature is exact for polynomials of this degree and the second by interpolation. Thus

$$\begin{aligned} \int_a^b w(x) f(x) \, \mathrm{d}x - \sum_{j=0}^{n} w_j f(x_j) &= \int_a^b w(x) [f(x) - H_{2n+1}(x)] \, \mathrm{d}x \\ &= \frac{1}{(2n+2)!} \int_a^b f^{(2n+2)}(\eta(x)) w(x) \prod_{j=0}^{n} (x - x_j)^2 \, \mathrm{d}x, \end{aligned}$$

and hence the required result follows from the integral mean value theorem as $w(x)\prod_{j=0}^{n}(x-x_j)^2 \ge 0$. $\square$

Remark: the "direct" approach of finding Gauss quadrature formulae sometimes works for small n, but more sophisticated algorithms are used for large n.[1]

Example. We can also find the quadrature weights and points without explicitly computing the orthogonal polynomials. To find the two-point Gauss–Legendre rule $w_0 f(x_0) + w_1 f(x_1)$ on $(-1, 1)$ with weight function $w(x) \equiv 1$, we need to be able to integrate any cubic polynomial exactly, so

$$2 = \int_{-1}^{1} 1 \,\mathrm{d}x = w_0 + w_1 \tag{4}$$

$$0 = \int_{-1}^{1} x \,\mathrm{d}x = w_0 x_0 + w_1 x_1 \tag{5}$$

$$\frac{2}{3} = \int_{-1}^{1} x^2 \,\mathrm{d}x = w_0 x_0^2 + w_1 x_1^2 \tag{6}$$

$$0 = \int_{-1}^{1} x^3 \,\mathrm{d}x = w_0 x_0^3 + w_1 x_1^3. \tag{7}$$

These are four nonlinear equations in four unknowns w_0, w_1, x_0 and x_1. Equations (5) and (7) give

$$\begin{bmatrix} x_0 & x_1 \\ x_0^3 & x_1^3 \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix},$$

which implies that

$$x_0 x_1^3 - x_1 x_0^3 = 0$$

for w_0, $w_1 \neq 0$, i.e.,

$$x_0 x_1 (x_1 - x_0)(x_1 + x_0) = 0.$$

If $x_0 = 0$, this implies $w_1 = 0$ or $x_1 = 0$ by (5), either of which contradicts (6). Thus $x_0 \neq 0$, and similarly $x_1 \neq 0$. If $x_1 = x_0$, (5) implies $w_1 = -w_0$, which contradicts (4). So $x_1 = -x_0$, and hence (5) implies $w_1 = w_0$. But then (4) implies that $w_0 = w_1 = 1$ and (6) gives

$$x_0 = -\frac{1}{\sqrt{3}} \qquad \text{and} \qquad x_1 = \frac{1}{\sqrt{3}},$$

which are the roots of the Legendre polynomial $x^2 - \frac{1}{3}$.

Convergence: Gauss quadrature converges astonishingly fast. It can be shown that if f is analytic on $[a, b]$, the convergence is geometric (exponential) in the number of quadrature points. This is in contrast to other (more straightforward) quadrature rules:

- Newton-Cotes: Find interpolant in n equispaced points, and integrate interpolant. Convergence: (often) Divergent!

[1] See e.g., the research paper by Hale and Townsend, "Fast and accurate computation of Guass–Legendre and Gauss–Jacobi quadrature nodes and weights" SIAM J. Sci. Comput. 2013.

- (Composite) trapezium rule: Find piecewise-linear interpolant in n equispaced points, and integrate interpolant. Convergence: $O(1/n^2)$ (assumes f'' exists)
- (Composite) Simpson's rule: Find piecewise-quadratic interpolant in n equispaced points (each subinterval containing three points), and integrate interpolant. Convergence: $O(1/n^4)$ (assumes f'''' exists)

The figure below illustrates the performance on integrating the Runge function.

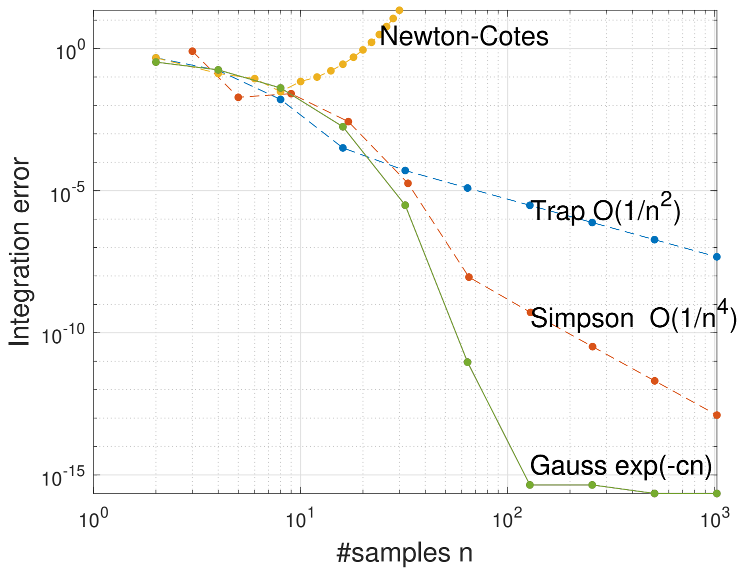
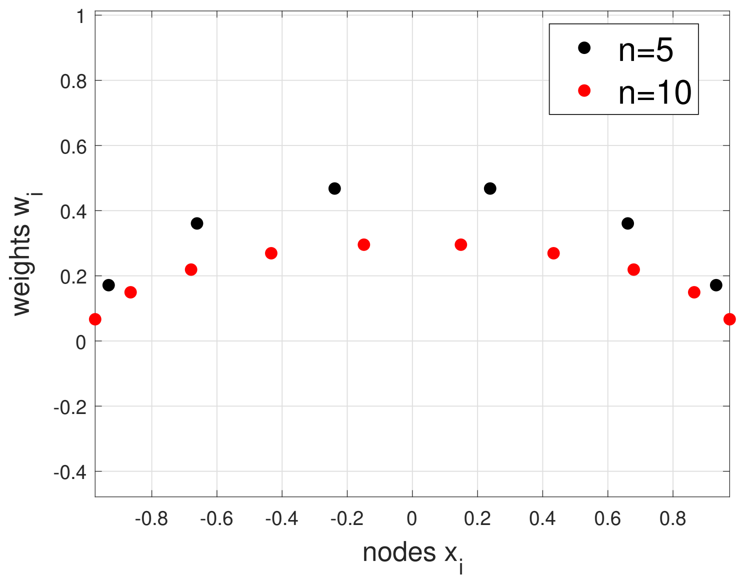


Figure 1: Convergence of quadrature rules with n points (samples) for $\int_{-1}^{1} \frac{1}{25x^2+1} dx$ (Runge function)

Nodes and weights for Gauss(-Legendre) quadrature The figure below shows the nodes (interpolation points) and the corrsponding weights with the standard Gauss-Legendre quadrature rule, i.e., when $w(x) = 1$ and $[a, b] = [-1, 1]$. In Chebfun these are computed conveniently by `[x,w] = legpts(n+1)`



Note that the nodes/interpolation points cluster near endpoints (and sparser in the middle); this is a general phenomenon, and very analogous to the Chebyshev interpolation points mentioned in the least-squares lecture (Gauss and Chebyshev points have asymptotically the same distribution of points). Note also that the weights are all positive and shrink as n grows; they have to because they sum to 2 (why?).

Everything below is non-examinable.

Other quadrature rules

- **Gauss-Lobatto:** $w(x) \equiv 1$ and $(a, b) = (-1, 1)$. We include both endpoints of the interval in the list of quadrature points: An $(n+1)$-point Gauss-Lobatto rule is of the form

$$\int_{-1}^{1} f(x)\,\mathrm{d}x \approx w_0 f(-1) + \sum_{j=1}^{n-1} w_j f(x_j) + w_n f(1).$$

  There are $2n$ remaining degrees of freedom, so weights and points are chosen to integrate polynomials of degree $2n-1$ exactly.

- **Gauss-Radau:** $w(x) \equiv 1$ and $(a, b) = (-1, 1)$. We include only the left endpoint of the interval in the list of quadrature points: An $(n+1)$-point Gauss-Radau rule is of the form

$$\int_{-1}^{1} f(x)\,\mathrm{d}x \approx w_0 f(-1) + \sum_{j=1}^{n} w_j f(x_j).$$

  There are $2n+1$ remaining degrees of freedom, so weights and points are chosen to integrate polynomials of degree $2n$ exactly.

- **Gauss-Kronrod:** Suppose we start with an $(n+1)$-point Gauss rule:

$$\int_{-1}^{1} f(x)\,\mathrm{d}x \approx \sum_{j=0}^{n} w_j f(x_j),$$

  and we want to use a more accurate quadrature rule to estimate the error of our current Gauss rule. One possibility is to use a Gauss rule with a larger number of points. However, this more accurate Gauss rule will not share any of the same quadrature points and computing f may be expensive. A Gauss-Kronrod rule keeps all of the same current quadrature points (so we can reuse the values of f) and adds $n+2$ additional points:

$$\int_{-1}^{1} f(x)\,\mathrm{d}x \approx \sum_{j=0}^{n} \tilde{w}_j f(x_j) + \sum_{k=0}^{n+1} \omega_k f(y_k),$$

  where we modify the old $n+1$ weights $\{\tilde{w}_j\}$ and choose new $n+2$ weights $\{\omega_k\}$ and $n+2$ points $\{y_k\}$ to integrate polynomials of degree $3n+4$ exactly.