LECTURE 24: *Gaussian quadrature rules: fundamentals*

3.4 *Gaussian quadrature*

It is clear that the trapezoid rule,

$$\frac{b-a}{2}\big(f(a)+f(b)\big),$$

exactly integrates linear polynomials, but not all quadratics. In fact, one can show that *no* quadrature rule of the form

$$w_a f(a) + w_b f(b)$$

will exactly integrate all quadratics over $[a,b]$, regardless of the choice of constants w_a and w_b. However, notice that a general quadrature rule with two points,

$$w_0 f(x_0) + w_1 f(x_1),$$

has four parameters (w_0, x_0, w_1, x_1). We might then hope that we could pick these four parameters in such a fashion that the quadrature rule is exact for a four-dimensional subspace of functions, $\mathcal{P}_3$. This section explores generalizations of this question.

3.4.1 *A special 2-point rule*

Suppose we consider a more general class of 2-point quadrature rules, where we do not initially fix the points at which the integrand f is evaluated:

$$I(f) = w_0 f(x_0) + w_1 f(x_1)$$

for unknowns *nodes* $x_0, x_1 \in [a,b]$ and *weights* w_0 and w_1. We wish to pick x_0, x_1, w_0, and w_1 so that the quadrature rule exactly integrates all polynomials of the largest degree possible. Since this quadrature rule is linear, it will suffice to check that it is exact on monomials. There are four unknowns; to get four equations, we will require $I(f)$ to exactly integrate 1, x, x^2, x^3.

$$
\begin{aligned}
f(x) = 1: &\quad \int_a^b 1\,\mathrm{d}x = I(1) &\Longrightarrow&\quad b-a \;=\; w_0 + w_1\\
f(x) = x: &\quad \int_a^b x\,\mathrm{d}x = I(x) &\Longrightarrow&\quad \tfrac{1}{2}(b^2-a^2) \;=\; w_0x_0 + w_1x_1\\
f(x) = x^2: &\quad \int_a^b x^2\,\mathrm{d}x = I(x^2) &\Longrightarrow&\quad \tfrac{1}{3}(b^3-a^3) \;=\; w_0x_0^2 + w_1x_1^2\\
f(x) = x^3: &\quad \int_a^b x^3\,\mathrm{d}x = I(x^3) &\Longrightarrow&\quad \tfrac{1}{4}(b^4-a^4) \;=\; w_0x_0^3 + w_1x_1^3
\end{aligned}
$$

Three of these constraints are *nonlinear* equations of the unknowns x_0, x_1, w_0, and w_1: thus questions of existence and uniqueness of solutions becomes a bit more subtle than for the linear equations we so often encounter.

In this case, a solution *does* exist:

$$w_0 = w_1 = \tfrac{1}{2}(b-a),$$

$$x_0 = \tfrac{1}{2}(b+a) - \tfrac{\sqrt{3}}{6}(b-a), \qquad x_1 = \tfrac{1}{2}(b+a) + \tfrac{\sqrt{3}}{6}(b-a).$$

Notice that $x_0, x_1 \in [a,b]$: If this were not the case, we could not use these points as quadrature nodes, since f might not be defined outside $[a,b]$. When $[a,b] = [-1,1]$, the interpolation points are $\pm 1/\sqrt{3}$, giving the quadrature rule

$$I(f) = f(-1/\sqrt{3}) + f(1/\sqrt{3}).$$

3.4.2 Generalization to higher degrees

Emboldened by the success of this humble 2-point rule, we consider generalizations to higher degrees. If some two-point rule ($n+1$ integration nodes, for $n=1$) will exactly integrate all cubics ($3 = 2n+1$), one might anticipate the existence of rules based on $n+1$ points that exactly integrate all polynomials of degree $2n+1$, for general values of n. Toward this end, consider quadrature rules of the form

$$I_n(f) = \sum_{j=0}^{n} w_j f(x_j),$$

for which we will choose the nodes $\{x_j\}$ and weights $\{w_j\}$ (a total of $2n+2$ variables) to maximize the degree of polynomial that is integrated exactly.

The primary challenge is to find satisfactory quadrature nodes. Once these are found, the weights follow easily: in theory, one could obtain them by integrating the polynomial interpolant at the nodes, though better methods are available in practice. In particular, this procedure for assigning weights ensures, at a minimum, that $I_n(f)$ will exactly integrate all polynomials of degree n. This assumption will play a key role in the coming development.

Orthogonal polynomials, introduced in Section 2.5, play a central role in this exposition, and they suggest a generalization of the interpolatory quadrature procedures we have studied up to this point.

Let $\{\phi_j\}_{j=0}^{n+1}$ be a system of orthogonal polynomials with respect to the inner product

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

for some weight function $w \in C(a,b)$ that is non-negative over (a,b) and takes the value of zero only on a set of measure zero.

This weight function plays an essential role in the discussion: it defines the inner product, and so it dictates what it means for two functions to be orthogonal. Change the weight function, and you will change the orthogonal polynomials.

Now we wish to construct an interpolatory quadrature rule for an integral that incorporates the weight function $w(x)$ in the integrand:

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

It is our aim to make $I_n(p)$ exact for all $p \in \mathcal{P}_{2n+1}$. First, we will show that any interpolatory quadrature rule I_n will at least be exact for the weighted integral of degree-n polynomials. Showing this is a simple modification of the argument made in Section 3.1 for unweighted integrals.

In the Section 3.4.4 we shall see some useful examples of weight functions.

Given a set of distinct nodes $x_0, \ldots, x_n$, construct the polynomial interpolant to f at those nodes:

$$p_n(x) = \sum_{j=0}^{n} f(x_j)\ell_j(x),$$

where $\ell_j(x)$ is the usual Lagrange basis function for polynomial interpolation. The interpolatory quadrature rule will exactly integrate the *weighted integral* of the interpolant p_n:

$$\begin{aligned}\int_a^b f(x)w(x)\,\mathrm{d}x \approx \int_a^b p_n(x)w(x)\,\mathrm{d}x &= \int_a^b \Big(\sum_{j=0}^{n} f(x_j)\ell_j(x)\Big)w(x)\,\mathrm{d}x \\ &= \sum_{j=0}^{n} f(x_j)\int_a^b \ell_j(x)w(x)\,\mathrm{d}x.\end{aligned}$$

Thus we define the *quadrature weights* for the *weighted integral* to be

$$w_j := \int_a^b \ell_j(x)w(x)\,\mathrm{d}x,$$

giving the rule

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

Apply this rule to a degree-n polynomial, p. Since $p \in \mathcal{P}_n$, it is its own degree-n polynomial interpolant, so the integral of the interpolant delivers the exact weighted integral of p:

Note that the weight function $w(x)$ can include all sorts of nastiness, all of which is absorbed in the quadrature weights $w_0, \ldots, w_n$.

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

This is the case regardless of how the (distinct) nodes $x_0, \ldots, x_n$ were chosen. Now we seek a way to choose the nodes so that the quadature rule is exactly for a higher degree polynomials.

To begin, consider an arbitrary $p \in \mathcal{P}_{2n+1}$. Using polynomial division, we can always write

$$p(x) = \phi_{n+1}(x)q(x) + r(x)$$

for some $q, r \in \mathcal{P}_n$ that depend on p. Integrating this p, we obtain

$$\begin{aligned}\int_a^b p(x)w(x)\,\mathrm{d}x &= \int_a^b \phi_{n+1}(x)q(x)w(x)\,\mathrm{d}x + \int_a^b r(x)w(x)\,\mathrm{d}x \\ &= \langle \phi_{n+1}, q\rangle + \int_a^b r(x)w(x)\,\mathrm{d}x \\ &= \int_a^b r(x)w(x)\,\mathrm{d}x.\end{aligned}$$

The last step is a consequence that important basic fact, proved in Section 2.5, that the orthogonal polynomial ϕ_{n+1} is orthogonal to all $q \in \mathcal{P}_n$.

Now apply the quadrature rule to p, and attempt to pick the interpolation nodes $\{x_j\}$ to yield the value of the exact integral computed above. In particular,

$$\begin{aligned}I_n(p) = \sum_{j=0}^{n} w_j p(x_j) &= \sum_{j=0}^{n} w_j \phi_{n+1}(x_j)q(x_j) + \sum_{j=0}^{n} w_j r(x_j) \\ &= \sum_{j=0}^{n} w_j \phi_{n+1}(x_j)q(x_j) + \int_a^b r(x)w(x)\,\mathrm{d}x.\end{aligned}$$

This last statement is a consequence of the fact that $I_n(\cdot)$ will exactly integrate all $r \in \mathcal{P}_n$. This will be true regardless of our choice for the distinct nodes $\{x_j\} \subset [a, b]$. (Recall that the quadrature rule is constructed so that it exactly integrates a degree-n polynomial interpolant to the integrand, and in this case the integrand, r, is a degree n polynomial. Hence $I_n(r)$ will be exact.)

Notice that we can force agreement between $I_n(p)$ and $\int_a^b p(x)w(x)\,\mathrm{d}x$ provided

$$\sum_{j=0}^{n} w_j \phi_{n+1}(x_j)q(x_j) = 0.$$

We cannot make assumptions about $q \in \mathcal{P}_n$, as this polynomial will vary with the choice of p, but we can exploit properties of ϕ_{n+1}. Since ϕ_{n+1} has exact degree $n+1$ (recall this property of all orthogonal polynomials), it must have $n+1$ roots. If we choose the interpolation nodes $\{x_j\}$ to be the roots of ϕ_{n+1}, then $\sum_{j=0}^{n} w_j\, \phi_{n+1}(x_j)q(x_j) = 0$ as required, and we have a quadrature rule that is exact for all polynomials of degree $2n+1$.

Before we can declare victory, though, we must exercise some caution. Perhaps ϕ_{n+1} has repeated roots (so that the nodes $\{x_j\}$ are not distinct), or perhaps these roots lie at points in the complex plane where f may not even be defined. Since we are integrating f over the interval $[a, b]$, it is crucial that ϕ_{n+1} has $n+1$ distinct roots in $[a, b]$. Fortunately, this is one of the many beautiful properties enjoyed by orthogonal polynomials.

Theorem 3.6 (Roots of Orthogonal Polynomials)**.**
Let $\{\phi_k\}_{k=0}^{n+1}$ be a system of orthogonal polynomials on $[a, b]$ with respect to the weight function $w(x)$. Then ϕ_k has k distinct real roots, $\{x_j^{(k)}\}_{j=1}^{k}$, with $x_j^{(k)} \in [a, b]$ for $j = 1, \ldots, k$.

Proof. The result is trivial for ϕ_0. Fix any $k \in \{1, \ldots, n+1\}$. Suppose that ϕ_k, a polynomial of exact degree k, changes sign at $j < k$ distinct roots $\{x_\ell^{(k)}\}_{\ell=1}^{j}$, in the interval $[a, b]$. Then define

$$q(x) = (x - x_1^{(k)})(x - x_2^{(k)}) \cdots (x - x_j^{(k)}) \in \mathcal{P}_j.$$

This function changes sign at exactly the same points as ϕ_k does on $[a, b]$. Thus, the product of these two functions, $\phi_k(x)q(x)$, *does not change sign* on $[a, b]$. See the illustration in Figure 3.8.

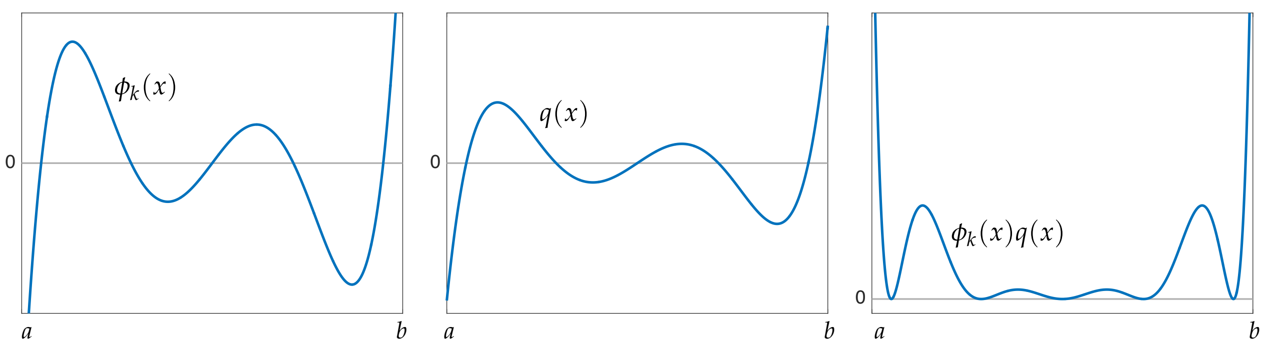

Figure 3.8: The functions ϕ_k, q, and $\phi_k q$ from the proof of Theorem 3.9.

As the weight function $w(x)$ is nonnegative on $[a, b]$, it must also be that $\phi_k q w$ does not change sign on $[a, b]$. However, the fact that $q \in \mathcal{P}_j$ for $j < k$ implies that

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

since ϕ_k is orthogonal to all polynomials of degree $k-1$ or lower (Lemma 2.3). Thus, we conclude that the integral of some continuous nonzero function $\phi_k q w$ that never changes sign on $[a, b]$ must be zero. This is a contradiction, as the integral of such a function must always be positive. Thus, ϕ_k must have at least k distinct zeros in $[a, b]$. As ϕ_k is a polynomial of degree k, it can have no more than k zeros. ∎

We have arrived at *Gaussian quadrature rules*: Integrate the polynomial that interpolates f at the roots of the orthogonal polynomial ϕ_{n+1}. What are the weights $\{w_j\}$? Write the interpolant, p_n, in the Lagrange basis,

$$p_n(x) = \sum_{j=0}^{n} f(x_j)\ell_j(x),$$

where the basis polynomials ℓ_j are defined as usual,

$$\ell_j(x) = \prod_{k=0,k\neq j}^{n} \frac{(x-x_k)}{(x_j-x_k)}. \tag{3.2}$$

Integrating this interpolant gives

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

revealing a formula for the quadrature weights:

$$w_j = \int_a^b \ell_j(x)w(x)\,\mathrm{d}x.$$

This construction proves the following result.

Theorem 3.7. Suppose $I_n(f)$ is the Gaussian quadrature rule

$$I_n(f) = \sum_{j=0}^{n} w_j f(x_j),$$

where the nodes $\{x_j\}_{j=0}^{n}$ are the $n+1$ roots of a degree-$(n+1)$ orthogonal polynomial on $[a,b]$ with weight function w, and $w_j = \int_a^b \ell_j(x)w(x)\,\mathrm{d}x$. Then

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

for all polynomials f of degree $2n+1$.

As a side-effect of this high-degree exactness, we obtain an interesting new formula for the weights in Gaussian quadrature. Since the Lagrange basis polynomial ℓ_k is the product of n linear factors (see (3.2)), $\ell_k \in \mathcal{P}_n$, and

$$(\ell_k)^2 \in \mathcal{P}_{2n} \subseteq \mathcal{P}_{2n+1}.$$

Thus the Gaussian quadrature rule exactly integrates $(\ell_k)^2 w(x)$. We write

$$\begin{aligned}\int_a^b (\ell_k(x))^2 w(x)\mathrm{d}x &= \sum_{j=0}^{n} w_j(\ell_k(x_j))^2 \\ &= w_k(\ell_k(x_k))^2 = w_k,\end{aligned}$$

where we have used the fact that $\ell_k(x_j) = 0$ if $j \neq k$, and $\ell_k(x_k) = 1$. This leads to another formula for the Gaussian quadrature weights:

$$w_k = \int_a^b \ell_k(x) w(x)\,\mathrm{d}x = \int_a^b (\ell_k(x))^2 w(x)\,\mathrm{d}x. \tag{3.3}$$

This latter formula is more computationally appealing than the former, because it is more numerically reliable to integrate positive-valued integrands. This is a neat fact, but, as described in Section , there is a still-better way to compute these weights: by computing eigenvectors of a symmetric tridiagonal matrix.

One avoids floating point errors that can be introduced by adding quantities that are similar in magnitude but opposite in sign, known as *catastrophic cancellation*.

Of course, in many circumstances we are not simply integrating polynomials, but more complicated functions, so we want better insight about the method's performance than Theorem 3.7 provides. One can prove the following error bound.

See, e.g., Süli and Mayers, pp. 282–283.

Theorem 3.8. Suppose $f \in C^{2n+2}[a, b]$ and let $I_n(f)$ be the usual $(n+1)$-point Gaussian quadrature rule on $[a, b]$ with weight function $w(x)$ and nodes $\{x_j\}_{j=0}^n$. Then

$$\int_a^b f(x)w(x)\,\mathrm{d}x - I_n(f) = \frac{f^{(2n+2)}(\xi)}{(2n+2)!}\int_a^b \psi^2(x)w(x)\,\mathrm{d}x$$

for some $\xi \in [a, b]$ and $\psi(x) = \prod_{j=0}^n (x - x_j)$.
