3 Quadrature

LECTURE 21: Interpolatory Quadrature Rules

The past two chapters have developed a suite of tools for polynomial interpolation and approximation. We shall now apply these tools toward the approximation of definite integrals.

To compute the least squares approximations discussed in Section 2.4, one needs to compute integrals for the inner products

$$\langle f, \phi_i \rangle = \int_a^b f(x) \phi_i(x) \, dx$$

that form the right-hand side of the Gram matrix equation $Gc = b$. Of course, many other applications require the evaluation of definite integrals; integrals across (many) different variables pose additional challenges.

Many definite integrals are difficult or impossible to evaluate exactly, so our next charge is to develop algorithms that approximate such integrals quickly and accurately. This field is known as quadrature, a name that suggests the approximation of the area under a curve by area of subtending quadrilaterals. (a “Riemann sum”).

3.1 Interpolatory Quadrature

Given $f \in C[a, b]$, we seek approximations to the definite integral

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

All the methods we consider in these notes are variants of interpolatory quadrature rules, meaning that they approximate the integral of $f$ by the exact integral of a polynomial interpolant to $f$:

$$\int_a^b f(x) \, dx \approx \int_a^b p_n(x) \, dx,$$

The term quadrature is used to distinguish the numerical approximation of a definite integral from the numerical solution of an ordinary differential equation, which is often called numerical integration. Approximation of a double integral is sometimes called cubature.
where \( p_n \in \mathcal{P}_n \) interpolates \( f \) at \( n + 1 \) points in \([a, b]\). Will such rules produce a reasonable estimate to the integral? Of course, that depends on properties of \( f \) and the interpolation points.

Our goal in this section is to develop a convenient formula for the approximation

\[
\int_a^b p_n(x) \, dx
\]

that will not require the explicit construction of \( p_n \). As is often the case, the task becomes direct and simple if we express the interpolant in the correct basis. Recall the Lagrange form of the interpolant presented in Section 1.5: Given \( n + 1 \) distinct interpolation points

\[x_0, \ldots, x_n \in [a, b],\]

the interpolant can be written as

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

where the basis functions \( \ell_0, \ldots, \ell_n \) take the familiar form

\[
\ell_j(x) = \prod_{k=0}^n \frac{x - x_k}{x_j - x_k}.
\]

The integral of \( p_n \) can then be computed in terms of the integral of the basis functions:

\[
\int_a^b p_n(x) \, dx = \int_a^b \sum_{j=0}^n f(x_j) \ell_j(x) \, dx = \sum_{j=0}^n f(x_j) \int_a^b \ell_j(x) \, dx.
\]

In the nomenclature of quadrature rules, the integrals of the basis functions are called *weights*, denoted

\[
w_j := \int_a^b \ell_j(x) \, dx.
\]

Why is the Lagrange basis special? Could you not do the same kind of expansion with the monomial or Newton bases? Yes indeed: but then you would need to compute the coefficients \( c_j \) that multiply these basis functions in the expansion \( p_n(x) = \sum c_j \phi_j(x) \), which requires the solution of a (non-trivial) linear system. The beauty of the Lagrange approach is that these coefficients are instantly available by evaluating \( f \) at the quadrature nodes: \( c_j = f(x_j) \).

The degree-\( n \) interpolatory quadrature rule at distinct nodes \( x_0, \ldots, x_n \) takes the form

\[
\int_a^b f(x) \, dx \approx \sum_{j=0}^n w_j f(x_j),
\]

for the weights

\[
w_j = \int_a^b \ell_j(x) \, dx.
\]

It is worth stating an obvious theorem, which we will revisit in future lectures.
Theorem 3.1 (Exactness of Interpolatory Quadrature).
The degree-$n$ interpolatory quadrature rule at distinct points $x_0, \ldots, x_n \in [a, b]$ is exact for any polynomial of degree $n$ or less: if $f \in P_n$, then
\[
\int_a^b f(x) \, dx = \sum_{j=0}^n w_j f(x_j).
\]

The proof is simple: If $f \in P_n$, its polynomial interpolant $p_n \in P_n$ is exactly $f$, and so the exact integral of $p_n$ is the same thing as the exact integral of $f$. However, the result is not inconsequential. There are some circumstances in numerical computations where it is easier to use a quadrature rule to evaluate the integral of a polynomial, rather than computing the integral directly from the polynomial coefficients.

Finite element methods give one such setting, where in some cases $f$ is represented by its values $f(x_j)$ on a computational mesh, rather than by its coefficients.

Definition 3.1. An interpolatory quadrature rule has degree of exactness $m$ if for all $f \in P_m$,
\[
\int_a^b f(x) \, dx = \sum_{j=0}^n w_j f(x_j).
\]

By Theorem 3.1, a degree-$n$ quadrature rule has degree of exactness $m \geq n$. Thus it will be particularly interesting to see circumstances in which this degree of exactness is exceeded.