Jim Lambers
MAT 460/560
Fall Semeseter 2009-10
Lecture 31 Notes

These notes correspond to Section 4.7 in the text.

Gaussian Quadrature

In Lecture 26, we learned that a Newton-Cotes quadrature rule with n nodes has degree at most n. Therefore, it is natural to ask whether it is possible to select the nodes and weights of an n-point quadrature rule so that the rule has degree greater than n. *Gaussian quadrature rules* have the surprising property that they can be used to integrate polynomials of degree $2n-1$ exactly using only n nodes.

Gaussian quadrature rules can be constructed using a technique known as *moment matching.* For any nonnegative integer k, the k^{th} moment is defined to be

$$\mu_k = \int_a^b x^k\,dx.$$

For given n, our goal is to select weights and nodes so that the first $2n$ moments are computed exactly; i.e.,

$$\mu_k = \sum_{i=0}^{n-1} w_i x_i^k, \quad k = 0, 1, \ldots, 2n-1.$$

Since we have $2n$ free parameters, it is reasonable to think that appropriate nodes and weights can be found. Unfortunately, this system of equations is nonlinear, so we cannot be certain that a solution exists.

Suppose $g(x)$ is a polynomial of degree $2n-1$. For convenience, we will write $g \in \mathcal{P}_{2n-1}$, where, for any natural number k, $\mathcal{P}_k$ denotes the space of polynomials of degree at most k. We shall show that there exist weights $\{w_i\}_{i=0}^{n-1}$ and nodes $\{x_i\}_{i=0}^{n-1}$ such that

$$\int_a^b g(x)\,dx = \sum_{i=0}^{n-1} w_i g(x_i).$$

Furthermore, for more general functions, $G(x)$,

$$\int_a^b G(x)\,dx = \sum_{i=0}^{n-1} w_i G(x_i) + E[G]$$

where

1. x_i are real, distinct, and $a < x_i < b$ for $i = 0, 1, \dots, n-1$.

2. The weights $\{w_i\}$ satisfy $w_i > 0$ for $i = 0, 1, \dots, n-1$.

3. The error $E[G]$ satisfies $E[G] = \frac{G^{(2n)}(\xi)}{(2n)!} \int_a^b \prod_{i=0}^{n-1} (x - x_i)^2 \, dx$.

Notice that this method is exact for polynomials of degree $2n-1$ since the error functional $E[G]$ depends on the $(2n)^{th}$ derivative of G.

To prove this, we shall construct an *orthonormal family* of polynomials $\{q_i(x)\}_{i=0}^n$ so that

$$\int_a^b q_r(x) q_s(x) \, dx = \begin{cases} 0 & r \neq s, \\ 1 & r = s. \end{cases}$$

This can be accomplished using the fact that such a family of polynomials satisfies a *three-term recurrence relation*

$$\beta_j q_{j+1}(x) = (x - \alpha_j) q_j(x) - \beta_{j-1} q_{j-1}(x), \quad q_0(x) = (b-a)^{-1/2}, \quad q_{-1}(x) = 0,$$

where

$$\alpha_j = \int_a^b x q_j(x)^2 \, dx, \quad \beta_j = \int_a^b x q_{j+1}(x) q_j(x) \, dx.$$

We choose the nodes $\{x_i\}$ to be the roots of the n^{th}-degree polynomial in this family. Next, we construct the interpolant of degree $n-1$, denoted $p_{n-1}(x)$, of $g(x)$ through the nodes:

$$L_{n-1}(x) = \sum_{i=0}^{n-1} g(x_i) \mathcal{L}_{n-1,i}(x),$$

where, for $i = 0, \dots, n-1$, $\mathcal{L}_{n-1,i}(x)$ is the ith Lagrange polynomial for the points $x_0, \dots, x_{n-1}$. We shall now look at the interpolation error function

$$e(x) = g(x) - p_{n-1}(x).$$

Clearly, since $g \in \mathcal{P}_{2n-1}$, $e \in \mathcal{P}_{2n-1}$. Since $e(x)$ has roots at each of the roots of $q_n(x)$, we can factor e so that

$$e(x) = q_n(x) r(x),$$

where $r \in \mathcal{P}_{n-1}$. It follows from the fact that $q_n(x)$ is orthogonal to *any* polynomial in $\mathcal{P}_{n-1}$ that the integral of g can then be written as

$$\begin{aligned} I(g) &= \int_a^b p_{n-1}(x) \, dx + \int_a^b q_n(x) r(x) \, dx \\ &= \int_a^b p_{n-1}(x) \, dx \end{aligned}$$

$$
\begin{aligned}
&= \int_a^b \sum_{i=0}^{n-1} g(x_i)\mathcal{L}_{n-1,i}(x)\,dx \\
&= \sum_{i=0}^{n-1} g(x_i) \int_a^b \mathcal{L}_{n-1,i}(x)\,dx \\
&= \sum_{i=0}^{n-1} g(x_i) w_i
\end{aligned}
$$

where

$$
w_i = \int_a^b \mathcal{L}_{n-1,i}(x)\,dx, \quad i = 0, 1, \ldots, n-1.
$$

For a more general function $G(x)$, the error functional $E[g]$ can be obtained from the expression for the interpolation error presented in Lecture 16.

It is easy to show that the weights are positive. Since the interpolation basis functions $\mathcal{L}_{n-1,i}$ belong to $\mathcal{P}_{n-1}$, it follows that $\mathcal{L}_{n-1,i}^2 \in \mathcal{P}_{2n-2}$, and therefore

$$
0 < \int_a^b \mathcal{L}_{n-1,i}^2(x)\,dx = \sum_{j=0}^{n-1} w_j \mathcal{L}_{n-1,i}^2(x_j) = w_i.
$$

Also, we can easily obtain qualitative bounds on the error. For instance, if we know that the even derivatives of g are positive, then we know that the quadrature rule yields an upper bound for $I(g)$. Similarly, if the even derivatives of g are negative, then the quadrature rule gives a lower bound.

Example We will use *Gaussian quadrature* to approximate the integral

$$
\int_0^1 e^{-x^2}\,dx.
$$

The particular Gaussian quadrature rule that we will use consists of 5 nodes x_1, x_2, x_3, x_4 and x_5, and 5 weights w_1, w_2, w_3, w_4 and w_5. To determine the proper nodes and weights, we use the fact that the nodes and weights of a 5-point Gaussian rule for integrating over the interval $[-1, 1]$ are given by

i	Nodes $r_{5,i}$	Weights $c_{5,i}$
1	0.9061798459	0.2369268850
2	0.5384693101	0.4786286705
3	0.0000000000	0.5688888889
4	−0.5384693101	0.4786286705
5	−0.9061798459	0.2369268850

To obtain the corresponding nodes and weights for integrating over $[-1, 1]$, we can use the fact that in general,

$$\int_a^b f(x)\,dx = \int_{-1}^1 f\left(\frac{b-a}{2}t + \frac{a+b}{2}\right)\frac{b-a}{2}\,dt,$$

as can be shown using the change of variable $x = [(b-a)/2]t + (a+b)/2$ that maps $[a, b]$ into $[-1, 1]$. We then have

$$\begin{aligned}\int_a^b f(x)\,dx &= \int_{-1}^1 f\left(\frac{b-a}{2}t + \frac{a+b}{2}\right)\frac{b-a}{2}\,dt \\ &\approx \sum_{i=1}^5 f\left(\frac{b-a}{2}r_{5,i} + \frac{a+b}{2}\right)\frac{b-a}{2}c_{5,i} \\ &\approx \sum_{i=1}^5 f(x_i)w_i,\end{aligned}$$

where

$$x_i = \frac{b-a}{2}r_{5,i} + \frac{a+b}{2}, \quad w_i = \frac{b-a}{2}c_{5,i}, \quad i = 1, \ldots, 5.$$

In this example, $a = 0$ and $b = 1$, so the nodes and weights for a 5-point Gaussian quadrature rule for integrating over $[0, 1]$ are given by

$$x_i = \frac{1}{2}r_{5,i} + \frac{1}{2}, \quad w_i = \frac{1}{2}c_{5,i}, \quad i = 1, \ldots, 5,$$

which yields

i	Nodes x_i	Weights w_i
1	0.95308992295	0.11846344250
2	0.76923465505	0.23931433525
3	0.50000000000	0.28444444444
4	0.23076534495	0.23931433525
5	0.04691007705	0.11846344250

It follows that

$$\begin{aligned}\int_0^1 e^{-x^2}\,dx &\approx \sum_{i=1}^5 e^{-x_i^2}w_i \\ &\approx 0.11846344250e^{-0.95308992295^2} + 0.23931433525e^{-0.76923465505^2} + \\ &\quad 0.28444444444e^{-0.5^2} + 0.23931433525e^{-0.23076534495^2} + \\ &\quad 0.11846344250e^{-0.04691007705^2} \\ &\approx 0.74682412673352.\end{aligned}$$

Since the exact value is 0.74682413281243, the absolute error is -6.08×10^{-9}, which is remarkably accurate considering that only fives nodes are used. □

The high degree of accuracy of Gaussian quadrature rules make them the most commonly used rules in practice. However, they are not without their drawbacks:

- They are not progressive, so the nodes must be recomputed whenever additional degrees of accuracy are desired. An alternative is to use *Gauss-Kronrod rules*. A $(2n + 1)$-point Gauss-Kronrod rule uses the nodes of the n-point Gaussian rule. For this reason, practical quadrature procedures use both the Gaussian rule and the corresponding Gauss-Kronrod rule to estimate accuracy.

- Because the nodes are the roots of a polynomial, they must be computed using traditional root-finding methods, which are not always accurate. Errors in the computed nodes lead to lost degrees of accuracy in the approximate integral. In practice, however, this does not normally cause significant difficulty.

Often, variations of Gaussian quadrature rules are used in which one or more nodes are prescribed. For example, *Gauss-Radau rules* are rules in which either of the endpoints of the interval $[a, b]$ are chosen to be a node, and n additional nodes are determined by a procedure similar to that used in Gaussian quadrature, resulting in a rule of degree $2n$. In *Gauss-Lobatto rules*, both endpoints of $[a, b]$ are nodes, with n additional nodes chosen in order to obtain a rule of degree $2n + 1$. It should be noted that Gauss-Lobatto rules are closed, whereas Gaussian rules are open.