Numerical Analysis

Raphael Hauser
with thanks to Endre Süli

Oxford Mathematical Institute

HT 2019

Recap on Gaussian Quadrature

Theorem

Let $x_0 < x_1 < \cdots < x_n$ *be the roots of the* $n+1$*-st degree 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

$$\int_a^b w(x) f(x) \, \mathrm{d}x \approx \sum_{j=0}^{n} w_j f(x_j) \tag{14.1}$$

with weights $w_j = \int_a^b w(x) L_{n,j}(x) \, \mathrm{d}x$ *is exact whenever* $f \in \Pi_{2n+1}$.

Example

$$\int_1^2 \frac{1}{x}\,\mathrm{d}x \overset{\text{2-pt Gauss-Legrendre}}{=} \frac{1}{2}\int_{-1}^1 \frac{2}{t+3}\,\mathrm{d}t$$

$$\simeq \frac{1}{3+\frac{1}{\sqrt{3}}} + \frac{1}{3-\frac{1}{\sqrt{3}}} = 0.6923077\ldots,$$

$$\int_1^2 \frac{1}{x}\,\mathrm{d}x \overset{\text{2-pt Newton-Cotes}}{\simeq} \frac{1}{2}\left[\frac{1}{2}+1\right] = 0.75,$$

$$\int_1^2 \frac{1}{x}\,\mathrm{d} \overset{\text{exact}}{=} 0.6931472\ldots$$

Example

$$\int_1^2 \frac{1}{x}\,\mathrm{d}x \overset{\text{2-pt Gauss-Legrendre}}{=} \frac{1}{2}\int_{-1}^{1} \frac{2}{t+3}\,\mathrm{d}t$$

$$\simeq \frac{1}{3+\frac{1}{\sqrt{3}}} + \frac{1}{3-\frac{1}{\sqrt{3}}} = 0.6923077\ldots,$$

$$\int_1^2 \frac{1}{x}\,\mathrm{d}x \overset{\text{2-pt Newton-Cotes}}{\simeq} \frac{1}{2}\left[\frac{1}{2}+1\right] = 0.75,$$

$$\int_1^2 \frac{1}{x}\,\mathrm{d} \overset{\text{exact}}{=} 0.6931472\ldots$$

Gaussian quadrature seems to be much more accurate than Newton-Cotes quadrature with the same number of nodes, not just for polynomials!

Theorem (Error of Gaussian Quadrature)

Let $f \in C^{2n+2}(a, b)$, and let x_j and w_j be as defined above. 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 $\eta \in (a, b)$.

Recall from Lecture 2: Given data $f_i = f(x_i)$ and $g_i = f'(x_i)$ at $n+1$ distinct points

$$x_0 < x_1 < \cdots < x_n,$$

there exists a unique polynomial $p \in \Pi_{2n+1}$ such that $p(x_i) = f_i$ and $p'(x_i) = g_i$ for $i = 0, 1, \ldots, n$, the *Hermite interpolating polynomial*

$$p_{2n+1}(x) = \sum_{k=0}^{n} [f_k H_{n,k}(x) + g_k K_{n,k}(x)],$$

which can be constructed as follows,

$$\begin{aligned}
L_{n,k}(x) &= \frac{(x-x_0)\cdots(x-x_{k-1})(x-x_{k+1})\cdots(x-x_n)}{(x_k-x_0)\cdots(x_k-x_{k-1})(x_k-x_{k+1})\cdots(x_k-x_n)}, \\
H_{n,k}(x) &= [L_{n,k}(x)]^2(1-2(x-x_k)L'_{n,k}(x_k)) \\
K_{n,k}(x) &= [L_{n,k}(x)]^2(x-x_k).
\end{aligned}$$

Proof of Theorem (Error of Gaussian Quadrature). The proof is based on the Hermite Interpolating Polynomial H_{2n+1} to f at $x_0, x_1, \ldots, x_n$.

Proof of Theorem (Error of Gaussian Quadrature). The proof is based on the Hermite Interpolating Polynomial H_{2n+1} to f at $x_0, x_1, \ldots, x_n$.

The error in Hermite interpolation is (see Lecture 2)

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

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

Proof of Theorem (Error of Gaussian Quadrature). The proof is based on the Hermite Interpolating Polynomial H_{2n+1} to f at $x_0, x_1, \ldots, x_n$.

The error in Hermite interpolation is (see Lecture 2)

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

for some $\eta(x) \in (a, b)$. 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 Gaussian Quadrature is exact for polynomials of this degree and the second by interpolation.

Thus

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

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 $w(x)\prod_{j=0}^{n}(x-x_j)^2 \geq 0$,

Thus

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

$$= \int_a^b w(x)[f(x) - H_{2n+1}(x)]\,\mathrm{d}x$$

$$= \frac{1}{(2n+2)!}\int_a^b f^{(2n+2)}(\eta(x))w(x)\prod_{j=0}^{n}(x-x_j)^2\,\mathrm{d}x,$$

and hence the required result follows from the Integral Mean Value Theorem as $w(x)\prod_{j=0}^{n}(x-x_j)^2 \geq 0$,

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

Remark: the "direct" approach of finding Gaussian Quadrature formulae sometimes works for small n, but is usually hard.

Remark: the "direct" approach of finding Gaussian Quadrature formulae sometimes works for small n, but is usually hard.

Example. 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

$$2 = \int_{-1}^{1} 1 \, \mathrm{d}x = w_0 + w_1 \quad (14.2)$$

$$0 = \int_{-1}^{1} x \, \mathrm{d}x = w_0 x_0 + w_1 x_1 \quad (14.3)$$

$$\tfrac{2}{3} = \int_{-1}^{1} x^2 \, \mathrm{d}x = w_0 x_0^2 + w_1 x_1^2 \quad (14.4)$$

$$0 = \int_{-1}^{1} x^3 \, \mathrm{d}x = w_0 x_0^3 + w_1 x_1^3. \quad (14.5)$$

Solution: These are four nonlinear equations in four unknowns w_0, w_1, x_0 and x_1.

Solution: These are four nonlinear equations in four unknowns w_0, w_1, x_0 and x_1. Equations (14.3) and (14.5) give

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

which implies that

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

for w_1, $w_2 \neq 0$, i.e.,

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

Solution: These are four nonlinear equations in four unknowns w_0, w_1, x_0 and x_1. Equations (14.3) and (14.5) give

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

which implies that

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

for w_1, $w_2 \neq 0$, 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 (14.3), either of which contradicts (14.4).

Solution: These are four nonlinear equations in four unknowns w_0, w_1, x_0 and x_1. Equations (14.3) and (14.5) give

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

which implies that

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

for w_1, $w_2 \neq 0$, 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 (14.3), either of which contradicts (14.4). Thus $x_0 \neq 0$, and similarly $x_1 \neq 0$.

Solution: These are four nonlinear equations in four unknowns w_0, w_1, x_0 and x_1. Equations (14.3) and (14.5) give

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

which implies that

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

for w_1, $w_2 \neq 0$, 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 (14.3), either of which contradicts (14.4). Thus $x_0 \neq 0$, and similarly $x_1 \neq 0$. If $x_1 = x_0$, (14.3) implies $w_1 = -w_0$, which contradicts (14.2).

Solution: These are four nonlinear equations in four unknowns w_0, w_1, x_0 and x_1. Equations (14.3) and (14.5) give

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

which implies that

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

for w_1, $w_2 \neq 0$, 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 (14.3), either of which contradicts (14.4). Thus $x_0 \neq 0$, and similarly $x_1 \neq 0$. If $x_1 = x_0$, (14.3) implies $w_1 = -w_0$, which contradicts (14.2). So $x_1 = -x_0$, and hence (14.3) implies $w_1 = w_0$.

Solution: These are four nonlinear equations in four unknowns w_0, w_1, x_0 and x_1. Equations (14.3) and (14.5) give

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

which implies that

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

for w_1, $w_2 \neq 0$, i.e.,

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

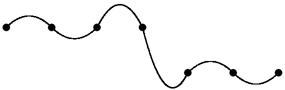
If $x_0 = 0$, this implies $w_1 = 0$ or $x_1 = 0$ by (14.3), either of which contradicts (14.4). Thus $x_0 \neq 0$, and similarly $x_1 \neq 0$. If $x_1 = x_0$, (14.3) implies $w_1 = -w_0$, which contradicts (14.2). So $x_1 = -x_0$, and hence (14.3) implies $w_1 = w_0$. Then (14.2) yields $w_0 = w_1 = 1$ and (14.4) gives

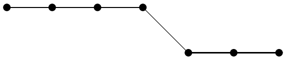
$$x_0 = -\tfrac{1}{\sqrt{3}} \qquad \text{and} \qquad x_1 = \tfrac{1}{\sqrt{3}},$$

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

Piecewise Polynomial Interpolation: Splines

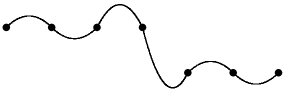
Sometimes a ‘global’ approximation like Lagrange Interpolation is not appropriate, e.g., for ’rough’ data.

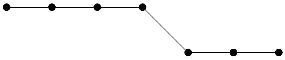




Piecewise Polynomial Interpolation: Splines

Sometimes a 'global' approximation like Lagrange Interpolation is not appropriate, e.g., for 'rough' data.





On the left the Lagrange Interpolant p_7 'wiggles' through the points, while on the right a **piecewise** linear interpolant ('join the dots'), or linear **spline** interpolant, s appears to represent the data better.

Remark: for any given data s clearly exists and is unique.

Remark: for any given data s clearly exists and is unique.

Suppose that $a = x_0 < x_1 < \cdots < x_n = b$. Then, s is linear on each interval $[x_{i-1}, x_i]$ for $i = 1, \ldots, n$ and continuous on $[a, b]$.

Remark: for any given data s clearly exists and is unique.

Suppose that $a = x_0 < x_1 < \cdots < x_n = b$. Then, s is linear on each interval $[x_{i-1}, x_i]$ for $i = 1, \ldots, n$ and continuous on $[a, b]$.

The x_i, $i = 0, 1, \ldots, n$, are called the **knots** of the **linear spline**.

Remark: for any given data s clearly exists and is unique.

Suppose that $a = x_0 < x_1 < \cdots < x_n = b$. Then, s is linear on each interval $[x_{i-1}, x_i]$ for $i = 1, \ldots, n$ and continuous on $[a, b]$.

The x_i, $i = 0, 1, \ldots, n$, are called the **knots** of the **linear spline**.

Notation: $f \in \mathrm{C}^k[a, b]$ if $f, f', \ldots, f^k$ exist and are continuous on $[a, b]$.

Theorem

Let s *be the linear spline interpolation of a function* $f \in \mathrm{C}^2[a,b]$ *at nodes* $x_0 < x_1 < \cdots < x_n$. *Then,*

$$\|f - s\|_\infty \le \frac{1}{8} h^2 \|f''\|_\infty$$

where $h = \max\limits_{1 \le i \le n} (x_i - x_{i-1})$ *and* $\|f''\|_\infty = \max\limits_{x \in [a,b]} |f''(x)|$.

Proof. For $x \in [x_{i-1}, x_i]$, the error from linear interpolation is

$$f(x) - s(x) = \frac{1}{2} f''(\eta(x))(x - x_{i-1})(x - x_i)$$

where $\eta(x) \in (x_{i-1}, x_i)$, by the error formula of the Lagrange interpolating polynomial p_1 that interpolates f at x_{i-1} and x_i (see Lecture 1).

Proof. For $x \in [x_{i-1}, x_i]$, the error from linear interpolation is

$$f(x) - s(x) = \frac{1}{2} f''(\eta(x))(x - x_{i-1})(x - x_i)$$

where $\eta(x) \in (x_{i-1}, x_i)$, by the error formula of the Lagrange interpolating polynomial p_1 that interpolates f at x_{i-1} and x_i (see Lecture 1).

However,

$$|(x - x_{i-1})(x - x_i)| = (x - x_{i-1})(x_i - x) = -x^2 + x(x_{i-1} + x_i) - x_{i-1}x_i,$$

which has its maximum value when $2x = x_i + x_{i-1}$, i.e., when

$$x - x_{i-1} = x_i - x = \tfrac{1}{2}(x_i - x_{i-1}).$$

Proof. For $x \in [x_{i-1}, x_i]$, the error from linear interpolation is

$$f(x) - s(x) = \frac{1}{2} f''(\eta(x))(x - x_{i-1})(x - x_i)$$

where $\eta(x) \in (x_{i-1}, x_i)$, by the error formula of the Lagrange interpolating polynomial p_1 that interpolates f at x_{i-1} and x_i (see Lecture 1).

However,

$$|(x - x_{i-1})(x - x_i)| = (x - x_{i-1})(x_i - x) = -x^2 + x(x_{i-1} + x_i) - x_{i-1}x_i,$$

which has its maximum value when $2x = x_i + x_{i-1}$, i.e., when

$$x - x_{i-1} = x_i - x = \tfrac{1}{2}(x_i - x_{i-1}).$$

Thus for any $x \in [x_{i-1}, x_i]$, $i = 1, 2, \ldots, n$,

$$|f(x) - s(x)| \le \frac{1}{2} \|f''\|_\infty \max_{x \in [x_{i-1}, x_i]} |(x - x_{i-1})(x - x_i)| \le \frac{1}{8} h^2 \|f''\|_\infty. \qquad \square$$

Note that s may have discontinuous derivatives, but it is a locally defined approximation, since changing the value of one data point affects the approximation in only two intervals.

Note that s may have discontinuous derivatives, but it is a locally defined approximation, since changing the value of one data point affects the approximation in only two intervals.

To get greater smoothness but retain some 'locality', we can define **cubic splines** $s \in \mathrm{C}^2[a,b]$. For a given 'partition',

$$a = x_0 < x_1 < \cdots < x_n = b,$$

there are (generally different!) cubic polynomials in each interval (x_{i-1}, x_i), $i = 1, \ldots, n$, which are 'joined' at each knot to have continuity and continuity of s' and s''.

Note that s may have discontinuous derivatives, but it is a locally defined approximation, since changing the value of one data point affects the approximation in only two intervals.

To get greater smoothness but retain some 'locality', we can define **cubic splines** $s \in \mathrm{C}^2[a, b]$. For a given 'partition',

$$a = x_0 < x_1 < \cdots < x_n = b,$$

there are (generally different!) cubic polynomials in each interval (x_{i-1}, x_i), $i = 1, \ldots, n$, which are 'joined' at each knot to have continuity and continuity of s' and s''.

Interpolating cubic splines also satisfy $s(x_i) = f_i$ for given data f_i, $i = 0, 1, \ldots, n$.

Remark: if there are n intervals, there are $4n$ free coefficients (four for each cubic 'piece'), *but* $2n$ interpolation conditions (one each at the ends of each interval), $n - 1$ derivative continuity conditions (at $x_1, \ldots, x_{n-1}$) and $n - 1$ second derivative continuity conditions (at the same points), giving a total of $4n - 2$ conditions (which are linear in the free coefficients).

Remark: if there are n intervals, there are $4n$ free coefficients (four for each cubic 'piece'), *but* $2n$ interpolation conditions (one each at the ends of each interval), $n-1$ derivative continuity conditions (at $x_1, \ldots, x_{n-1}$) and $n-1$ second derivative continuity conditions (at the same points), giving a total of $4n-2$ conditions (which are linear in the free coefficients).

Thus the spline is not unique.

Remark: if there are n intervals, there are $4n$ free coefficients (four for each cubic 'piece'), *but* $2n$ interpolation conditions (one each at the ends of each interval), $n-1$ derivative continuity conditions (at $x_1, \ldots, x_{n-1}$) and $n-1$ second derivative continuity conditions (at the same points), giving a total of $4n-2$ conditions (which are linear in the free coefficients).

Thus the spline is not unique.

So we need to add two extra conditions to generate a spline that might be unique.

There are three common ways of doings this:

There are three common ways of doings this:

(a) specify $s'(x_0) = f'(x_0)$ and $s'(x_n) = f'(x_n)$; or

There are three common ways of doings this:

(a) specify $s'(x_0) = f'(x_0)$ and $s'(x_n) = f'(x_n)$; or

(b) specify $s''(x_0) = 0 = s''(x_n)$ — this gives a **natural** cubic spline; or

There are three common ways of doings this:

(a) specify $s'(x_0) = f'(x_0)$ and $s'(x_n) = f'(x_n)$; or

(b) specify $s''(x_0) = 0 = s''(x_n)$ — this gives a **natural** cubic spline; or

(c) enforce continuity of s''' at x_1 and x_{n-1} (which implies that the first two pieces are the same cubic spline, i.e., on $[x_0, x_2]$, and similarly for the last two pieces, i.e., on $[x_{n-2}, x_n]$, from which it follows that x_1 and x_{n-1} are not knots! — this is usually described as the 'not a knot' end-conditions).