Gaussian Quadrature

Tuesday, October 13, 2020 5:42 PM

Gaussian quadrature is a family of quadrature rules (Gauss-Legendre, Chebyshev-Gauss, Gauss-Laguerre, Gauss-Hermite) that choose n nodes such that the degree of exactness/accuracy/precision is as large as possible $(2n-1)$.

Closely related to Gauss-Lobatto and Clenshaw-Curtis

Gauss-Legendre to compute $\int_a^b f(x)dx$

This will be an open formula (nodes do not include endpoints a, b) so (following Burden and Faires) we're going to enumerate the nodes as $\{x_1, x_2, \ldots, x_n\}$

⚠ Our previous convention was $n+1$ nodes, $\{x_0, x_1, x_2, \ldots, x_n\}$

Just like Newton-Cotes, this will be an interpolatory formula, meaning the high-level idea is
1. polynomial interpolation of f on the nodes
2. integrate the polynomial

The big difference is that now we won't require uniformly spaced nodes, instead we'll pick the nodes to optimize a criterion

we'll use degree of exactness

For Newton-Cotes, recall

⚠ we used to call this "$n=1$" since we were 0-based

Name	# nodes		degree of exactness	error
midpoint rule (open),	1 node	$(n=1)$,	1 (lucky)	$h^3/3\, f''(\xi)$
trapezoidal rule (closed),	2 nodes	$(n=2)$	1	$-h^3/12\, f''(\xi)$
Simpson's rule (closed),	3 nodes	$(n=3)$	3 (lucky)	$-h^5/90\, f^{(4)}(\xi)$
Simpson's 3/8 rule (closed),	4 nodes	$(n=4)$	3	$-3h^5/80\, f^{(4)}(\xi)$

How do we know degree of exactness?

Method 1 (general purpose but only a lower bound)

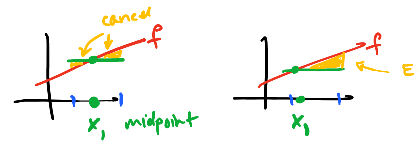
For n nodes, we've seen the $\overbrace{(n-1) \text{ degree interpolating polynomial}}^{p(x)}$ is unique

So if f itself is a $(n-1)$ degree polynomial, since it interpolates itself, it must be p. Hence we integrate it exactly

$\Rightarrow$ For any n nodes, interpolating quadrature has degree of exactness at least $n-1$

But... doesn't explain why midpoint rule has degree of exactness 1

Here, we got lucky. It was because we used a node exactly in the middle



Method 2 (more work to derive)

Derive an error estimate like $h^5 f^{(4)}(\xi)$

hence if $f^{(4)}(x) = 0 \ (\forall x) \Rightarrow$ no error.

If f is a n-degree polynomial, it's $(n+1)$ derivative is 0

ie., $f(x) = ax^2 + bx + c$, $f'''(x) = 0$

So, Newton Cotes w/ n nodes gives order of exactness $n-1$ (if n even) or n (if n odd)

Can we do better? We've been thinking of $\{x_1, \ldots, x_n\} \subseteq [a,b]$ as given, and then find weights w_i s.t. our formula is

$$\sum_{i=1}^{n} w_i f(x_i)$$

What if we choose nodes $\{x_i\}$? This gives n more parameters, so might hope we can get order of exactness $2n-1$.

In fact, we can!

Remember!

Def Gaussian quadrature means picking n nodes such that order of exactness is $2n-1$

Example

$n=2$, find weights and nodes so that $\underbrace{w_1 f(x_1) + w_2 f(x_2)}_{= I_2(f)}$ approximates $\int_{-1}^{1} f(x)\,dx$ with order of exactness $3 \ (= 2n-1)$.

i.e., integrate exactly any $f(x) = a_3 x^3 + a_2 x^2 + a_1 x + a_0$, ie., $\int_{-1}^{1} f(x)\,dx = I_2(f)$

since $\int f(x) + g(x)\,dx = \int f(x)\,dx + \int g(x)\,dx$

and $I_2(f+g) = I_2(f) + I_2(g)$ for generic f, g

and $\int a \cdot f(x)dx = a \cdot \int f(x)dx$
$I_2(a \cdot f) = a \cdot I_2(f)$,

i.e., both $\int dx$ and I_2 are linear operators

it suffices to show:

① $\int_{-1}^{1} x^3 \, dx = I_2(x^3)$

② $\int_{-1}^{1} x^2 \, dx = I_2(x^2)$

③ $\int_{-1}^{1} x \, dx = I_2(x)$

④ $\int_{-1}^{1} 1 \, dx = I_2(1)$

4 equations, 4 unknowns x_1, x_2, w_1, w_2

So... compute

① $\int_{-1}^{1} x^3 \, dx = 0$ (by odd symmetry) $= w_1 \cdot x_1^3 + w_2 x_2^3$

② $\int_{-1}^{1} x^2 dx = \frac{1}{3}x^3 \Big|_{-1}^{1} = \frac{2}{3} = w_1 x_1^2 + w_2 x_2^2$

③ $\int_{-1}^{1} x \, dx = 0$ (by odd symmetry) $= w_1 x_1 + w_2 x_2$

④ $\int_{-1}^{1} 1 \, dx = 2 = w_1 + w_2$

you can solve this (to make it more bearable, note we have a symmetry, so expect $w_1 = w_2$ and $x_1 = -x_2$,

so ④ $2 = w_1 + w_2$ and symmetry $(w_1 = w_2) \Rightarrow w_1 = w_2 = 1$

then ③ $\Rightarrow x_1 = -x_2$ (more symmetry) (and ④ is redundant)

so ② $\Rightarrow 2/3 = 2 \cdot x_1^2 \Rightarrow x_1 = -1/\sqrt{3} = -\sqrt{3}/3$

$x_2 = +1/\sqrt{3} = +\sqrt{3}/3$

Making it more systematic

(*) **Theorem**: The nodes of the n-point Gauss-Legendre quadrature rule are given by the roots of the nth Legendre polynomial P_n

Before we prove this, we better say what P_n is!

Notation: let $\mathbb{P}_n = \{$ all polynomials with degree n or less $\}$ and recall from linear algebra that this is a vector space of dimension $n+1$

Def Legendre polynomials $\{P_0, P_1, P_2, \ldots\}$ are polynomials such that

(*) ① $\forall p \in \mathbb{P}_{n-1}$ $\int_{-1}^{1} p(x) P_n(x) dx = 0$ "orthogonality"

(2a) P_n is a polynomial of degree n

(2b) P_n is monic (= leading coefficient is 1)

(not too important, just a way to ensure uniqueness)

i.e., $\text{Span}(\{P_0, \dots, P_n\}) = \mathbb{P}_n$

$\Rightarrow \{P_0, P_1, \dots, P_n\}$ is a basis for $\mathbb{P}_n$

and in fact it's orthogonal by ①

(not orthonormal though)

"monomial basis"

$\{1, x, x^2, \dots, x^n\}$ is also a basis for $\mathbb{P}_n$, but not orthogonal and not as nice to work with

If we orthonormalize the monomial basis (eg, Gram-Schmidt) then (up to scaling, ②b) we get the Legendre polynomials. So these are pretty fundamental.

Ex of Legendre polynomials

$P_0(x) = 1$

$P_1(x) = x$

$P_2(x) = x^2 - 1/3$

$P_3(x) = x^3 - 3/5\, x$

$P_4(x) = x^4 - \frac{6}{7}x^2 + 3/35$

have many known facts, e.g.,

$$P_n(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n}(x^2-1)^n$$

$$P_n(x) = \frac{2n-1}{n} x \cdot P_{n-1}(x) - \frac{n-1}{n} P_{n-2}(x) \quad \text{3-term recurrence}$$

The roots of $P_n(x)$ can be efficiently found in $O(n^2)$ time by the 1969 Golub and Welsch algorithm (efficiently finds eigenvalues of a tridiagonal matrix) – before then it was hard to use in practice. (but known since 1814 by Gauss) NOT ON TEST

The weights of the Gauss-Legendre formula are easier: form the interpolating polynomial p (Lagrange interpolating polynomial) and integrate that

⚠ Lagrange $\neq$ Legendre

So..

we'll do $\int_a^b$ later

Def. Gauss-Legendre quadrature for $\int_{-1}^{1} f(x)\,dx$:

① pick nodes $\{x_1, \dots, x_n\}$ to be the roots of P_n ← Legendre polynomial

Fact: all roots are simple, real, and lie in $(-1,1)$

Ⓐ Remember

② pick weights $w_i = \int_{-1}^{1} \overbrace{\prod_{\substack{j=1 \\ j\neq i}} \frac{x - x_j}{x_i - x_j}}^{L_i(x)}\, dx$

Since our interpolating polynomial is $\sum_{i=1}^{n} f(x_i) L_i(x)$

and define $I_n^{GL}(f) = \sum_{i=1}^{n} w_i f(x_i)$

now back to theorem

Thm if f is a polynomial of degree $2n-1$ or less ($f \in \mathbb{P}_{2n-1}$),

then $\int_{-1}^{1} f(x)dx = I_n^{GL}(f)$, i.e., Gauss-Legendre quadrature w/ n nodes has degree of exactness $2n-1$

Proof

① first, suppose $f \in \mathbb{P}_{n-1}$. Then since this is an interpolating quadrature on n nodes, f is its own interpolating polynomial (by uniqueness) hence it is integrated exactly

② now, let f have degree between $[n, 2n-1]$.

Divide f by the Legendre polynomial P_n:

$$f(x) = Q(x)P_n(x) + \underbrace{R(x)}_{\text{remainder}}$$

see wikipedia "polynomial long division" or "Euclidean division"

$$P_n(x)\overline{)\,f(x)\,}^{\,Q(x)} \quad \vdots \quad R(x)$$

where degree(R) < degree$(P_n) = n$.

Also, degree$(f) \in [n, 2n-1]$

degree(f) = degree$(Q \cdot P_n)$ = degree(Q) + degree(P_n)

= degree$(Q) + n$

so, degree$(Q) \in [0, n-1]$

$\Rightarrow$ degree$(Q) \leq n-1$, i.e., $Q \in \mathbb{P}_{n-1}$

Thus

$$\int_{-1}^{1} f(x)dx = \underbrace{\int_{-1}^{1} Q(x)P_n(x)}_{=0 \text{ by orthogonality}} + \underbrace{\int_{-1}^{1} R(x)dx}_{\text{equals G-L quadrature by part ①}}$$

$$= 0 + I_n^{GL}(R)$$

$$= I_n^{GL}(f)$$

since $R(x_i) = f(x_i)$. Why?

$$f(x_i) = Q(x_i)\underbrace{P_n(x_i)}_{x_i \text{ is a root of } P_n} + R(x_i)$$

$\square$

Extending from $\int_{-1}^{1} f(x)dx$ to $\int_{a}^{b} f(x)dx$ ⚠ Important detail!

Just do a change-of-variables

$x = a \qquad x = b$

$t = -1 \qquad t = 1$

center it

normalize it

Define $t = \dfrac{x - \frac{a+b}{2}}{2\cdot(b-a)} = \dfrac{2x-a-b}{b-a}$

So $x = \frac{1}{2}\left[(b-a)t + a + b\right]$, so $dx = \frac{b-a}{2}\,dt$

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

i.e., define $g(t) = f\left(\frac{(b-a)t + a+b}{2}\right)$

So the Gauss-Legendre quadrature estimate of $\int_a^b f(x)\,dx$

is $\frac{b-a}{2} I_n^{GL}(g)$.

More info

- Not only is integration more accurate w/ Gauss-Legendre nodes, but so is interpolation. The Runge phenomenon goes away.

- Nodes are precomputed (see Table 4.2 in book) or see scipy.special.roots_legendre or gl int (Driscoll, Braun text)

- A similar method that works almost as well is Clenshaw-Curtis

  see "Is Gaussian Quadrature Better than Clenshaw-Curtis?"
  Lloyd Trefethen, SIAM Review 2008
  (it's faster: $O(n \log n)$ vs $O(n^2)$)

- The Legendre polynomials are orthogonal in the sense

$$\underbrace{\int_{-1}^{1} P_n(x)P_m(x)\,dx}_{\langle P_n, P_m\rangle \text{ an inner-product}} = 0 \quad \text{if } m \neq n$$

(just as $\vec{x}^T\vec{y}$ is an inner product)

If we make a weighted inner product,

$$\int_{-1}^{1} f(x)g(x)w(x)\,dx = \langle f, g\rangle \quad \text{for } w(x) > 0$$

we get different "orthogonal" polynomials, (below is not on any test)

ex: $w(x) = \frac{1}{\sqrt{1-x^2}}$ on $(-1,1)$ gives rise to T_n Chebyshev polynomials of the 1st kind

$T_0(x) = 1 \qquad T_n(x) = 2x\,T_{n-1}(x) - T_{n-2}(x)$

$T_1(x) = x \qquad T_n(x) = \cos(n \cdot \cos^{-1}(x)) \quad (*)$

all roots simple, real, lie in $(-1,1)$.

In fact, $(*) \Rightarrow$ roots x_i are $x_i = \cos\left(\frac{2i-1}{2n}\pi\right)$ $i=1,\dots,n$

"Chebyshev nodes of the 1st kind"

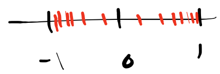
ex $w(x) = \sqrt{1-x^2}$ on $[-1,1]$, gives rise to U_n Chebyshev polynomials of the 2nd kind

w/ roots

$$x_i = -\cos\left(\frac{i\pi}{n}\right), \quad i=0,1,\dots,n$$

"Chebyshev nodes/points of the 2nd kind"

These are x-projections of equispaced pts on unit circle



and beats Runge phenomenon, so good for interpolation

stronger than C^∞

Thm (9.3.1 Driscoll + Braun) "Spectral Convergence"

* real analytic: Taylor series converges everywhere

Let p_n be the unique degree n polynomial interpolant of f using $n+1$ Chebyshev nodes of the 2nd kind

If f is real analytic* then $\exists\, C>0$, $\exists\, K>1$ such that

$$\max_{x\in[-1,1]} |f(x) - p_n(x)| \le C \cdot \underbrace{K^{-n}}_{\text{way better than } n^{-\alpha}}$$

ex. weight $w(x) = e^{-x}$ (or $e^{-x/2}$) on domain $[0,\infty)$

gives Laguerre polynomials

ex weight $w(x) = e^{-x^2}$ (or $e^{-x^2/2}$) on domain $(-\infty,\infty)$

gives Hermite polynomials

For all these choices of weights, we have a corresponding Gaussian quadrature rule

Name	Interval	Weight
Gauss-Legendre	$[-1,1]$	1
Chebyshev-Gauss	$(-1,1)$	$(1-x^2)^{-1/2}$
	$[-1,1]$	$(1-x^2)^{1/2}$
Gauss-Laguerre	$[0,\infty)$	e^{-x}
Gauss-Hermite	$(-\infty,\infty)$	e^{-x^2}