November 21, 2024

Announcements

- Exam 4
- 4CH1 ext
- 4CH2 release
- HW9

Goals

- error
- piecewise interp.
- integration

Review

h = interval length

$$|f - p_{n-1}| = O(h^n)$$

Going piecewise: Simplest Case

Construct a piecewise linear interpolant at four points.

x_0, y_0		x_1, y_1		x_2, y_2		x_3, y_3
	$f_1 = a_1 x + b_1$		$f_2 = a_2 x + b_2$		$f_3 = a_3 x + b_3$	
	2 unk.		2 unk.		2 unk.	
	$f_1(x_0) = y_0$		$f_2(x_1) = y_1$		$f_3(x_2) = y_2$	
	$f_1(x_1) = y_1$		$f_2(x_2) = y_2$		$f_3(x_3) = y_3$	
	2 eqn.		2 eqn.		2 eqn.	

Why three intervals?

two end one middle

Piecewise Cubic ('Splines')

x_0, y_0		x_1, y_1		x_2, y_2		x_3, y_3
	f_1		f_2		f_3	
	$a_1x^3 + b_1x^2 + c_1x + d_1$		$a_2x^3 + b_2x^2 + c_2x + d_2$		$a_3x^3 + b_3x^2 + c_3x + d_3$	

4 unk. | 4 unk. | 4 unk. — 12 unk.

$f_1(x_0) = y_0$

$f_1(x_1) = y_1$

. . . . . . — 6

2 eqn.

$f_1'(x_1) = f_2'(x_1)$ $\quad$ $f_2'(x_2) = f_3'(x_2)$ — 8

$f_1''(x_1) = f_2''(x_1)$ $\quad$ $f_2''(x_2) = f_3''(x_2)$ — 10

f''' ... ?

"Natural" spline:

$f_1''(x_0) = 0$ $\qquad$ $f_3''(x_3) = 0$

Piecewise Cubic ('Splines'): Accounting

x_0, y_0		x_1, y_1		x_2, y_2		x_3, y_3
	f_1		f_2		f_3	
	$a_1x^3 + b_1x^2 + c_1x + d_1$		$a_2x^3 + b_2x^2 + c_2x + d_2$		$a_3x^3 + b_3x^2 + c_3x + d_3$	

condition: $2N_{\text{intervals}} + 2N_{\text{middle nodes}} + 2$

$$N_{\text{middle}} = N_{\text{intervals}} - 1$$

$N = N_{\text{intervals}}$

$$2N + 2(N-1) + 2 = 4N$$

Outline

Numerical Integration: About the Problem

What is numerical integration? (Or quadrature?)

What about existence and uniqueness?

Numerical Integration: About the Problem

What is numerical integration? (Or quadrature?)

Given a, b, f, approximate

$$\int_a^b f(x)\mathrm{d}x.$$

What about existence and uniqueness?

- Answer exists e.g. if f is *integrable* in the Riemann or Lebesgue senses.
- Answer is unique if f is e.g. piecewise continuous and bounded. (this also implies existence)

Conditioning

Derive the (absolute) condition number for numerical integration.

$$\tilde{f}(x) = f(x) + e(x)$$

$$\left| \int_a^b \tilde{f}\,dx - \int_a^b f(x)\,dx \right|$$

$$= \left| \int ((f - f) + e)(x)\,dx \right| = \left| \int e(x)\,dx \right| \le \int_a^b |e(x)|\,dx$$

$$\le \|e\|_\infty (b-a)$$

Interpolatory Quadrature: Examples

Fix (x_i). Find p_{n-1} s.t. $p_{n-1}(x_i) = f(x_i)$

$$p_{n-1}(x) = \sum f(x_i)\, \ell_i(x)$$

$$\ell_j(x_i) = \begin{cases} 1 & i=j \\ 0 & \text{otherwise} \end{cases} \quad \text{(Lagrange)}$$

$$\int_a^b f(x)\,dx \approx \int_a^b p_{n-1}(x)\,dx = \int_a^b \sum f(x_i)\, \ell_i(x)\,dx$$

$$= \sum_{i=1}^{n} f(x_i) \underbrace{\int \ell_i(x)\,dx}_{w_i}$$

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

$w_i \leftarrow$ "weight"

Interpolatory Quadrature: Computing Weights

How do the weights in interpolatory quadrature get computed?

$$b-a = \int_a^b x^0\,dx = \int_a^b 1\,dx = 1 w_1 + \cdots + 1 w_n$$

$$\frac{b^2-a^2}{2} = \int x^1\,dx = x_1 w_1 + \cdots + x_n w_n$$

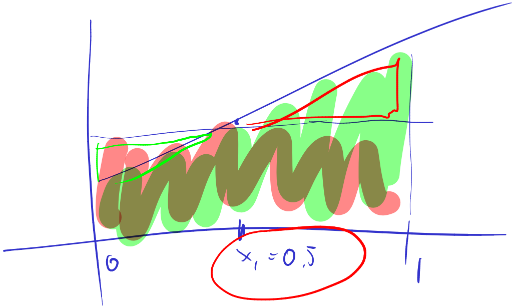
$$\vdots$$

$$\frac{b^{k+1}-a^{k+1}}{k+1} = \int_a^b x^k\,dx = x_1^k w_1 + \cdots + x_n^k w_n$$

⚠ Fixed after class

"method of undet. coeff."

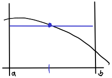
Demo: Newton-Cotes weight finder [cleared]


0
$x_1 = 0.5$
1

Examples and Exactness

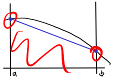
To what polynomial degree are the following rules exact?

Midpoint rule $(b-a)f\left(\frac{a+b}{2}\right)$



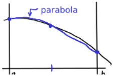
1

Trapezoidal rule $\frac{b-a}{2}(f(a)+f(b))$



1

Simpson's rule $\frac{b-a}{6}\left(f(a)+4f\left(\frac{a+b}{2}\right)+f(b)\right)$



3

Interpolatory Quadrature: Accuracy

Let p_{n-1} be an interpolant of f at nodes $x_1, \ldots, x_n$ (of degree $n-1$)
Recall

$$\sum_i \omega_i f(x_i) = \int_a^b p_{n-1}(x)\mathrm{d}x.$$

What can you say about the accuracy of the method?

$$\left| \int_a^b f(x)\,dx - \int_a^b p_{n-1}(x)\,dx \right|$$

$$\leq \int_a^b |f(x) - p_{n-1}(x)|\,dx \leq \underline{(b-a)} \cdot \|f - p_{n-1}\|_\infty$$

$$\leq \underline{h} \cdot C \cdot h^n \cdot \|f^{(n)}\|_\infty = C\,\|f^{(n)}\|_\infty \cdot h^{n+1}$$

too slow

h for odd

Quadrature: Overview of Rules

	n	Deg.	Ex.Int.Deg. (w/odd)	Intp.Ord.	Quad.Ord. (regular)	Quad.Ord. (w/odd)
		$n-1$	$(n-1)+1_{\text{odd}}$	n	$n+1$	$(n+1)+1_{\text{odd}}$
Midp.	1	0	1	1	2	3
Trapz.	2	1	1	2	3	3
Simps.	3	2	3	3	4	5
S. 3/8	4	3	3	4	5	5

- n: number of points
- "Deg.": Degree of polynomial used in interpolation ($= n-1$)
- "Ex.Int.Deg.": Polynomials of up to (and including) this degree *actually* get integrated exactly. (including the odd-order bump)
- "Intp.Ord.": Order of Accuracy of Interpolation: $O(h^n)$
- "Quad.Ord. (regular)": Order of accuracy for quadrature predicted by the error result above: $O(h^{n+1})$
- "Quad.Ord. (w/odd):" Actual order of accuracy for quadrature given 'bonus' degrees for rules with odd point count

Observation: Quadrature gets (at least) 'one order higher' than interpolation–even more for odd-order rules. (i.e. more accurate)

Interpolatory Quadrature: Stability

Let p_{n-1} be an interpolant of f at nodes $x_1, \ldots, x_n$ (of degree $n-1$)
Recall

$$\sum_i \omega_i f(x_i) = \int_a^b p_{n-1}(x)\mathrm{d}x$$

What can you say about the stability of this method?

$$\tilde{f}(x) = f(x) + e(x)$$

$$\left|\sum_{i=1}^n \tilde{f}(x_i)\,\omega_i - \sum_{i=1}^n f(x_i)\,\omega_i\right| \le \sum_{i=1}^n |e(x_i)|\,|\omega_i| \le \|e(x)\|_\infty \sum_{i=1}^n |\omega_i|$$

So, what quadrature weights make for bad stability bounds?

neg. weights bad

About Newton-Cotes

What's not to like about Newton-Cotes quadrature?

Demo: Newton-Cotes weight finder [cleared] (again, with many nodes)

About Newton-Cotes

What's not to like about Newton-Cotes quadrature?

Demo: Newton-Cotes weight finder [cleared] (again, with many nodes)

In fact, Newton-Cotes must have at least one negative weight as soon as $n \geqslant 11$.

More drawbacks:

- All the fun of high-order interpolation with monomials and equispaced nodes (i.e. convergence not guaranteed)
- Weights possibly non-negative ($\rightarrow$stability issues)
- Coefficients determined by (possibly ill-conditioned) Vandermonde matrix
- Thus hard to extend to arbitrary number of points.

Gaussian Quadrature

So far: nodes chosen from outside.
Can we gain something if we let the quadrature rule choose the nodes, too? Hope: More design freedom → Exact to higher degree.

$2n$ unknowns

→ $2n$ polynomials integrated exactly

Demo: Gaussian quadrature weight finder [cleared]

Composite Quadrature

High-order polynomial interpolation requires a high degree of smoothness of the function.

Idea: Stitch together multiple lower-order quadrature rules to alleviate smoothness requirement.