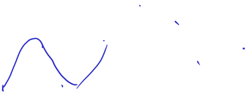
Announcements

- ICES
- 4CH assignments 1 & 2
- Content cutoff for Final: Thu Nov 30

Goals

- Splines
- num. integration

Review



Interpolation basis φ_i

nodes x_i

$$V = (\varphi_j(x_i))_{ij} \qquad V\vec{\alpha} = \vec{y}$$

$$\max |f - p_{n-1}(x)| \leq C \max |f^{(n)}(x)|\, h^n$$

"n-th order convergence"

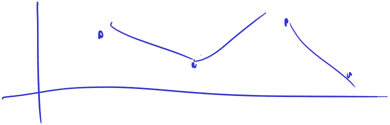
Going piecewise: Simplest Case

Construct a pieceweise linear interpolant at four points.

x_0, y_0		x_1, y_1		x_2, y_2		x_3, y_3
|	$f_1 = a_1x + b_1$	|	$f_2 = a_2x + b_2$	|	$f_3 = a_3x + b_3$	|
|	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?

2 end intervals, 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.

$f_1(x_0) = y_0$ | $f_2(x_1) = y_1$ | $f_3(x_2) = y_2$

$f_1(x_1) = y_1$ | $f_2(x_2) = y_2$ | $f_3(x_3) = y_3$

$f_1'(x_1) = f_2'(x_1)$ | $f_2'(x_2) = f_3'(x_2)$

$f_1''(x_1) = f_2''(x_1)$ | $f_2''(x_2) = f_3''(x_2)$

$f_1''(x_0) = 0$ | $f_3''(x_3) = 0$

"natural" spline

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	
	$a_1x^3 + b_1x^2 + c_1x + d_1$		$a_2x^3 + b_2x^2 + c_2x + d_2$		$a_3x^3 + b_3x^2 + c_3x + d_3$	

\# unknowns: $4 N_{\text{intervals}}$

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

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

$= 2 N_{\text{intervals}} + 2(N_{\text{intervals}} - 1) + 2 = 4 N_{\text{intervals}}$

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, compute (approximately!)

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

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.

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

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

$$= \left| \int_a^b e(x)\,dx \right| \leq \int_a^b |e(x)|\,dx \leq (b-a) \max_{x\in[a,b]} |e(x)|$$

Interpolatory Quadrature: Examples

$$f(x) \approx p_{n-1}(x) = \sum_{i=1}^{n} f(x_i)\, \ell_i(x)$$

Lagrange poly

$$\int_a^b f(x)\,dx \approx \sum_{i=1}^{n} f(x_i) \underbrace{\int_a^b \ell_i(x)\,dx}_{\omega_i}$$

"quadrature rule"

$$\int_a^b f(x)\,dx \approx \sum_{i=1}^{n} f(x_i)\, \omega_i$$

nodes

weights,

By construction: integrates polynomials of up to degree $n-1$ exactly

Interpolatory Quadrature: Computing Weights

How do the weights in interpolatory quadrature get computed?

$$1, x, \dots, x^{n-1}$$

$$b-a = \int_a^b 1\,dx = \omega_1 \cdot 1 + \dots + \omega_n \cdot 1$$

$$\left[\tfrac{1}{2}x^2\right]_a^b = \int_a^b x\,dx = \omega_1 \cdot x_1 + \dots + \omega_n \cdot x_n$$

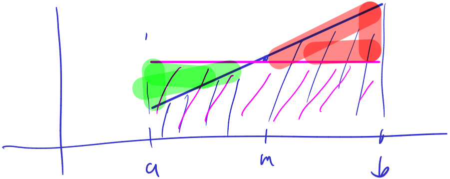
$$\vdots$$

$$\left[\tfrac{1}{n}x^n\right]_a^b = \int_a^b x^{n-1}\,dx = \omega_1 \cdot x_1^{n-1} + \dots + \omega_n \cdot x_n^{n-1}$$

$$\vec{r} = \underbrace{V^T}\ \vec{\omega}$$

"method of undetermined coefficients"

Demo: Newton-Cotes weight finder [cleared]

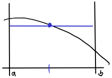


$$f(m)\cdot(b-a) \approx \int_a^b f(x)\,dx$$

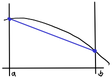
Examples and Exactness

To what polynomial degree are the following rules exact?

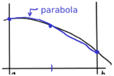
Midpoint rule $(b-a)f\left(\frac{a+b}{2}\right)$



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

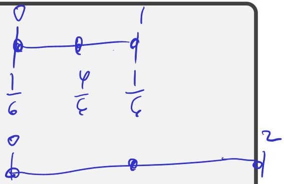


Simpson's rule $\frac{b-a}{6}\left(f(a)+4f\left(\frac{a+b}{2}\right)+f(b)\right)$



$b-a = \int_a^b 1 \approx \sum 1 w_i$

Observed: symmetric nodes & odd node count: extra degree of polynomial integrate d exactly



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?

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

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

$$\leq (b-a) \max_{x\in(a,b)} |f(x) - p_{n-1}(x)|$$

$$\leq Ch \max_{x\in(a,b)} |f^{(n)}(x)|\, h^n$$

$$\leq C \max |f^{(n)}|\, h^{n+1}$$

Quadrature: Overview of Rules

"Newton - Cotes" rules

	n	Deg.	Ex.Int.Deg. (w/odd)	Intp.Ord. h^n	Quad.Ord. (regular) h^{n+1}	Quad.Ord. (w/odd) $h^?$
		$n-1$	$(n-1)+1_{\text{odd}}$	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 be an interpolant of f at nodes $x_1, \dots, x_n$ (of degree $n-1$)

Recall

$$\sum_i \omega_i f(x_i) = \int_a^b p_n(x)\,\mathrm{d}x$$

$\sum w_i = b-a$

What can you say about the stability of this method?

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

when bigger

$$\left|\sum w_i f(x_i) - \sum w_i \hat{f}(x_i)\right| \le \sum |w_i\, e(x_i)| \le \left(\sum |w_i|\right) \max_{x \in [a,b]} |e(x)|$$

So, what quadrature weights make for bad stability bounds?

negative weights

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 $\rightarrow$ Exact to higher degree.

Demo: Gaussian quadrature weight finder [cleared]