Today:

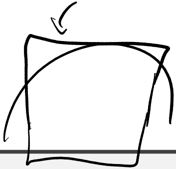
- Calculus

Announcements

- HW11 (today!)
- Examlet 3
- Examlet 4

Numerical Integration: About the Problem

$\sqrt{1-x^2}$



What is numerical integration? (Or quadrature?)

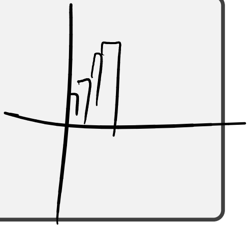
$$\int_a^b f(x)\,dx \qquad \text{given } a, b, f$$

What about existence and uniqueness?

- Riemann
- Lebesgue

} integrability

$\exists$ & uniqueness if pw continuous and bounded



~~Riemann~~ Lebesgue

$$\int \lim_{n\to\infty} f_n = \lim_{n\to\infty} \int f_n$$

Conditioning

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

Derive the (absolute) condition number for numerical integration.

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

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

Interpolatory Quadrature

$\vec{f} = f(\vec{x})$

Design a quadrature method based on interpolation.

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

gen. Vdm.

$$V\vec{\alpha} = \vec{f}$$
$$\Leftrightarrow \vec{\alpha} = V^{-1}\vec{f}$$

$$p_{n-1}(x) = \sum_{i=0}^{n-1} \alpha_i \varphi_i(x_i)$$

$$\vec{t}: t_i = \int_a^b \varphi_i(x)\,dx$$

$$\int_a^b f(x)\,dx \simeq \int_a^b p_{n-1}(x)\,dx = \sum_{i=0}^{n-1} \alpha_i \int_a^b \varphi_i(x_i) = \vec{\alpha}\cdot\vec{t}$$

$$= \vec{t}^{\,T}\vec{\alpha} = \underbrace{\vec{t}\,V^{-1}}_{\vec{\omega}}\,\vec{f} = \vec{\omega}\cdot\vec{f}.$$

Interpolatory Quadrature: Examples

↳ w/ Chebyshev nodes & Chebyshev polys

~~Chebyshev~~ Clenshaw-Curtis quadrature

↳ w/ equispaced nodes and monomials

Newton-Cotes quadratures

Interpolatory Quadrature: Computing Weights

How do the weights in interpolatory quadrature get computed?

$$\int_a^b \varphi_i(x)\,dx = \vec{\omega} \cdot \varphi_i(\vec{x}) \qquad i = 0 \ldots n-1$$

For monomials

$$b-a = \int_a^b 1\,dx = \omega_1 \cdot 1 + \omega_2 \cdot 1 + \omega_3 \cdots + \omega_n \cdot 1$$

$$\frac{1}{2}(b^2 - a^2) = \int_a^b x\,dx = \omega_1 \cdot x_1 + \omega_2 \cdot x_2 + \omega_3 \cdot x_3 \quad + \omega_n \cdot x_n$$

$$\frac{1}{k}(b^{k+1} - a^{k+1}) = \int_a^b x^k = \omega_1 \cdot x_1^k + \cdots$$

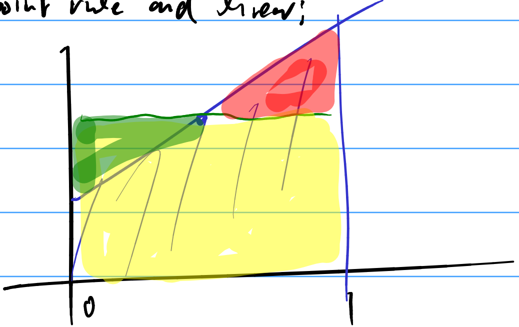
$$= \underbrace{V^T}\, \vec{\omega}$$

"method of undetermined coeff."

Demo: Newton-Cotes weight finder

$$\int_0^1 f(x)\,dx \approx \underbrace{\tfrac{1}{2}}\, f(0) + \underbrace{\tfrac{1}{2}}\, f(1)$$

Midpoint rule and linear:



$$\int_3^4 f(x)\,dx = \int_3^4 2\,dx$$

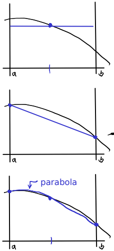
Examples and Exactness

To what polynomial degree are the following rules exact?

Midpoint rule $(b-a)f\left(\frac{a+b}{2}\right)$ $\to 1$ (0)

Trapezoidal rule $\frac{b-a}{2}(f(a)+f(b))$ $\to 1$

Simpson's rule $\frac{b-a}{6}\left(f(a)+4f\left(\frac{a+b}{2}\right)+f(b)\right)$ $\to 3$ (2)



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?

$$|p_{n-1}(x) - f(x)| \le C \cdot f^{(n)}(\xi)\, h^n$$

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

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

$$\le \int C \cdot \max_{\xi \in [a,b]} |f^{(n)}(\xi)| \cdot h^n \,dx$$

$$= C\,\underbrace{(b-a)}_{h} \cdot \max_{\xi \in [a,b]} |f^{(n)}(\xi)| \cdot h^n$$

$$= C \cdots h^{n+1}$$

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	$\leq C \cdot h^3$
Trapz.	2	1	1	2	3	3	
Simps.	3	2	3	3	4	5	$\leq C \cdot h^5$
—	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

$f \to \tilde{f}$

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

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

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

What can you say about the stability of this method?

$$\left|\sum_i w_i f(x_i) - \sum_i w_i \hat{f}(x_i)\right| = \left|\sum_i w_i e(x_i)\right|$$

$$\le \sum_i |w_i| \underbrace{|e(x_i)|}_{\le \max_j |e(x_j)|}$$

$$\leq \left(\sum |w_i|\right) \cdot \max |f - \tilde{f}|$$

bad quadrature rule : oscillating weights

good quad rule : pos. weights

About Newton-Cotes

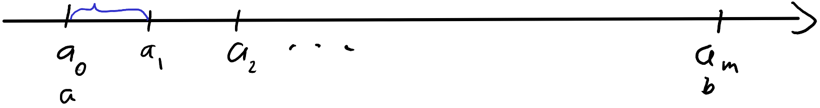
What's not to like about Newton-Cotes quadrature?

Composite Quadrature (not yet discussed)

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.

e.g. trapezoidal



Error in Composite Quadrature (not yet discussed)

What can we say about the error in the case of composite quadrature?

Composite Quadrature: Notes (not yet discussed)

Observation: Composite quadrature loses an order compared to non-composite.

Idea: If we can estimate errors on each subinterval, we can shrink (e.g. by splitting in half) only those contributing the most to the error.
(adaptivity, $\rightarrow$ hw)

Gaussian Quadrature (discussed)

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

In-Class Activity: Quadrature

In-class activity: Quadrature

Taking Derivatives Numerically

Why *shouldn't* you take derivatives numerically?

Demo: Taking Derivatives with Vandermonde Matrices